UNIVERSIDAD DE SEVILLA

DOCTORAL THESIS

Deep Learning-based Computer-Aided Diagnosis systems: a contribution to prostate cancer detection in histopathological images

Author:

Lourdes Durán López

Supervisors:

Dr. Alejandro Linares Barranco
Dr. Saturnino Vicente Díaz

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Robotics and Technology of Computers Lab.
Departamento de Arquitectura y Tecnología de Computadores

June 30, 2021

Declaration of Authorship

I, Lourdes Durán López, declare that this thesis, titled "Deep Learning-based Computer-Aided Diagnosis systems: a contribution to prostate cancer detection in histopathological images", and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

UNIVERSIDAD DE SEVILLA

Abstract

Departamento de Arquitectura y Tecnología de Computadores

Doctor of Philosophy

Deep Learning-based Computer-Aided Diagnosis systems: a contribution to prostate cancer detection in histopathological images

by Lourdes Durán López

In this work, novel computer-aided diagnosis systems for medical image analysis focusing on prostate cancer are proposed and implemented. First, the histopathology of prostate cancer was studied, along with the Gleason Grading System, which measures the aggressiveness of a tumor through different patterns with the purpose of driving therapies dealing with this disease. Furthermore, a study of Deep Learning techniques, particularly focusing on neural networks applied to medical image analysis, was conducted.

Based on these studies, a Deep Learning-based system to detect malignant regions in gigapixel-size whole-slide prostate cancer tissue images was proposed and developed, which is able to report spatial information of the malignant areas. This solution was evaluated in terms of performance and execution time, obtaining promising results when compared to other state-of-the-art methods. Since the implemented system locates malignant regions within the image without providing a global class, a custom Wide & Deep network was developed to report a slide-level label per image. The proposed system provides a fast screening method for analyzing histopathological images. Next, a neural network was proposed to assign a specific Gleason pattern to the malignant areas of the tissue. Finally, with the purpose of developing a global computer-aided diagnosis system for prostate cancer detection and classification, the three aforementioned subsystems were combined, allowing a complete analysis of histopathological images by reporting whether the sample is normal or malignant, and, in the last case, a heatmap of the malignant areas with their corresponding Gleason pattern.

The studied algorithms were also used for other medical image analysis tasks. The performance of these systems were evaluated, discussing the obtained results, presenting conclusions and proposing improvements for future works.

Acknowledgements

The work presented in this thesis could not have been carried out without the help and support of many people to whom I would like to dedicate some words to express my sincere gratitude.

First of all, I would like to thank my family for their unconditional support even in the worst moments. Particularly, to my parents, thank you for your support and advice that have helped me moving forward. Thanks to you I have become the person I am today, which I am very proud of.

To my thesis supervisors, Alejandro and Sátur, thank you for your guidance and all your help during this process. Thank you for looking after me since I started my research career in the department. I would never have been in the position I am today without your advice and support.

To all the members and colleagues of the Department of Computer Architecture and Technology and the Robotics and Technology of Computers Lab., with whom I have learned a lot and enjoyed in many good times. Special thanks to those who gave me their help and advice at some point during these years.

In these lines of gratitude, I would like to also thank Félix and Rafael from the Pathological Anatomy Unit of Virgen de Valme Hospital, whose work have been essential in this Thesis. Thank you for all your assistance and dedication and for kindly welcoming me in your Unit as if it were my home.

To Vitro S.A., in particular, to Fernando and Javier. Thank you for your help and collaboration with the project, and for allowing the work presented in this thesis to become a reality.

And, finally, in particular, I am extremely grateful to Juanpe for all his help and support from the moment I joined the department. Thank you for giving me all your dedication and always look for myself. This work is dedicated to him.

Contents

List of Figures

List of Tables

List of Equations

List of Abbreviations

AI	**A**rtificial **I**ntelligence
ANN	**A**rtificial **N**eural **N**etwork
AUC	**A**rea **U**nder **C**urve
CAD	**C**omputer-**A**ided **D**iagnosis
CAM	**C**lass **A**ctivation **M**ap
CNN	**C**onvolutional **N**eural **N**etwork
CPU	**C**entral **P**rocessing **U**nit
DL	**D**eep **L**earning
DRE	**D**igital **R**ectal **E**xamination
FC	**F**ully **C**onnected
FN	**F**alse **N**egatives
FP	**F**alse **P**ositives
GAP	**G**lobal **A**verage **P**ooling
GB	**G**iga**b**yte
GGS	**G**leason **G**rading **S**ystem
GPU	**G**raphics **P**rocessing **U**nit
H&E	**H**ematoxylin and **E**osin
KNN	**K**-**N**earest **N**eighbors
MCC	**M**alignant **C**onnected **C**omponents
LSRL	**L**east **S**quares **R**egression **L**ine
ML	**M**achine **L**earning
MPH	**M**alignant **P**robability **H**istogram
MTR	**M**alignant **T**issue **R**atio
PC	**P**ersonal **C**omputer
PCa	**P**rostate **Ca**ncer
PSA	**P**rostate-**S**pecific **A**ntigen
ReLU	**Re**ctified **L**inear **U**nit
RF	**R**andom **F**orest
RNN	**R**ecurrent **N**eural **N**etwork
ROC	**R**eceiver **O**perating **C**haracteristic
ROI	**R**egion **o**f **I**nterest
SVM	**S**upport **V**ector **M**achine
TN	**T**rue **N**egatives
TP	**T**rue **P**ositives
TPU	**T**ensor **P**rocessing **U**nit
W&D	**W**ide & **D**eep

WHO **W**orld **H**ealth **O**rganization
WPF **W**indows **P**resentation **F**oundation
WSI **W**hole-**S**lide **I**mage

Part I

Thesis

Chapter 1

Introduction

Life expectancy has increased significantly since the 20th century. According to the World Health Organization (WHO), it has been increased worldwide by five years on average since 2000, reaching 72 years old. This has become the fastest increase on life expectancy since 1960 (Gulland, 2016). The causes of mortality differ depending on the country and gender. On average, women tend to live longer than men in all countries, even in the most remote regions where access to healthcare is more difficult, as in the case of poor countries (Verbrugge et al., 1987).

This increase in life expectancy can be due to certain factors (Roser et al., 2013). One of the most important reasons in many cases is the evolution of medical treatments that have revolutionized the world of medicine. On the other hand, discovering new previously-unknown diseases and the way in which they can be diagnosed have also been key aspects in this regard. Finally, the application of therapies that prevent the most deadly diseases or delay age-related disorders have also contributed to welfare and longevity (Stearns and Koella, 2008). Therefore, modern medicine has been crucial to improve both life expectancy and quality of life, and, consequently, people live longer nowadays. However, the fact of living longer has caused other age-related disease to increase significantly in recent years (Brown, 2015). This is the case of cancer, the second most frequent cause of death with more than 9 million deaths worldwide in 2017, followed by cardiovascular diseases, with around 18 million cases (Ritchie and Roser, 2018; You and Henneberg, 2018). It is estimated that a 20% of men and a 17% of women will develop a tumor in their lifetime, while one out of eight men and one out of eleven women will die from this disease, although this is not applicable for all regions (Bray et al., 2018).

The International Agency for Research on Cancer determined that in 2018 there were a total of 18.1 million cases of cancer around the world (World Health Organization, 2018). Europe accounts for 23% of the total worldwide cancer cases and 20% of the deaths, with only 9% of the world's population. America accounts for 21% of the incidence and 14% of the deaths, with 13% of the world's population. Africa, with almost 500 million more inhabitants than Europe,

accounts for less than 6% of the cases. According to the Catalan Institute of Oncology (Barcelona, Spain), in more developed societies, life expectancy tends to be higher and, consequently, there is a higher incidence of cancer. On the other hand, in less developed countries, life expectancy is lower, mainly due to their scarce medical resources. This leads to infections, such as HIV/AIDS and malaria, being the most common cause of death (Lozano et al., 2012), and, therefore, less cancer cases are reported.

Detecting cancer at an advanced stage is a negative factor in the prognosis. Early diagnosis leads to a higher survival rate in cases of cancer such as breast, cervix, lung, stomach, prostate, liver and bladder (Ott et al., 2009). Early detection is possible in the majority of cancers. If a cancer is diagnosed early and treated appropriately, the chance of survival beyond 5 years is greater than when it is detected at a later stage. According to Cancer Research UK, cancer stages are based on the size of the tumor and how far has it spread through other parts of the body. The earlier the stage of the tumor is, the more treatment alternatives exist and the more effective they are (American Society of Clinical Oncology, 2019).

Advances in medical imaging techniques have significantly improved cancer detection and diagnosis (Wagner Jr and Conti, 1991). In order to identify and diagnose diseases, imaging technology are commonly used, which aid specialists to make a medical decision and monitor the response to therapy. In the traditional approach, physicians examine and inspect medical images in order to search for abnormalities and then make a decision. The diagnosis obtained in this time-consuming laborious process could be biased by factors including fatigue, the experience of the specialist and the intra-observer variation (Reiner and Krupinski, 2012). These, together with other many factors, lead to an inherent inter-observer variability among specialists that could be refined by means of other alternative approaches (Gomes et al., 2014; Mesquita et al., 2010; Krieger et al., 1994; Baldin et al., 2015).

Thanks to the advances in computer science and the emergence of new computer technologies, the field of medical image processing has experienced an exponential growth in the last decades (Shung et al., 2012). With the development and implementation of algorithms for medical image analysis, Computer-Aided Diagnosis (CAD) systems emerged as a new alternative to the traditional approach (Kim et al., 2011). The purpose of CAD systems is to assist doctors in the interpretation of medical images, providing a second opinion to support the diagnosis. The development of CAD systems has become one of the main research topics in different hospitals and research centers (Doi, 2007), as it has allowed to improve the accuracy and robustness of the diagnosis reported by specialists, as well as to reduce the response time of the diagnosis.

Artificial Intelligence (AI), and, in particular, Deep Learning (DL) algorithms, have grown in popularity inside the biomedical image analysis field (Altaf et al., 2019; Razzak et al., 2018; Santos et al., 2019). These algorithms

are able to automatically extract relevant features from input images, using that information to identify certain patterns and perform a computer-based diagnosis. Convolutional Neural Networks (CNNs), which are one of the most popular neural networks in DL, are widely used for image analysis in several fields (Li et al., 2020b), including biomedical image analysis (Anwar et al., 2018). This type of neural network is specialized in extracting complex features from images by applying a set of processing layers after training them in a supervised manner. These features are then combined to perform a prediction and report a final decision.

In this work, a novel computer-aided diagnosis system for cancer detection in digitized histopathological images is proposed and implemented. Particularly, this study is focused on prostate cancer (PCa) classification in image samples obtained from prostate biopsies. The developed global CAD system is divided into different subsystems: first, the input image is analyzed in search for malignant tissue regions, which are then used to perform a global classification of the image. After this, deeper features are obtained from each malignant area in order to categorize them into the different PCa patterns based on the Gleason Grading System. The proposed CAD system could play an important role to aid pathologists, providing a fast and accurate second opinion when analyzing a histopathological image.

1.1 Motivation

Although life expectancy has increased over the years thanks to advances in medicine, certain diseases, such as cancer, are still a problem nowadays. According to WHO, it is one of the leading causes of death worldwide, and its rate of cases is expected to increase by 70% in 20 years (Zarocostas, 2010). Detecting cancer as early as possible is a key factor, since time directly affects the development of this disease. It is demonstrated that the later the cancer is detected, the lower the survival rate of the patient will be (Hiom, 2015; Hawkes, 2019). Therefore, an early diagnosis is crucial to increase the probability of beating cancer. To this regard, CAD systems could be of great importance in order to speed up the diagnosis, and also to serve as a second opinion to aid physicians when making a decision. In this context, this work aims to contribute to the fight against cancer, particularly focusing on PCa, which is one of the most common diagnosed cancers among men (Ferlay et al., 2019).

Currently, there are many researchers studying the application of the aforementioned systems to automatically diagnose PCa. However, this field is still in an early stage of development and, to the best of the author's knowledge, it has not been applied yet to real-case scenarios, including hospitals and other medical centers. In this regard, this Thesis aligns with different tasks of a regional research project called "Prototipo de dispositivo médico de apoyo al diagnóstico de cáncer de próstata mediante teorías de clasificación de imagen

con Deep Learning (PROMETEO)" (AT17_5410_USE). This project is carried out by the Robotics and Technology of Computers Lab. (RTC, TEP-108), to which the author belongs, and Vitro S.A., one of the main private companies in Spain in the distribution and production of In Vitro Diagnostic reagents, platforms and related services, including pathology. The aim of this project goes beyond developing a CAD system for PCa detection, also looking for implementing this idea in medical centers in order to contribute and serve as a support to pathologists.

1.2 Prostate cancer

Cancer is a disease caused when cells divide uncontrollably and spread into surrounding tissues (Ruddon, 2007). Cancer cells have the ability to infiltrate and destroy normal body tissue, and it is due to this reason that cancer is one of the most deadly diseases. It is originated by mutations (modifications of the genetic sequence) in key genes that control normal cell growth and division (Stratton et al., 2009). These DNA mutations may be caused by different reasons including age as one of the most important factors (Aunan et al., 2017; De Magalhães, 2013). Over time, a number of mutations may occur in a single cell, allowing it to divide and grow in a way that becomes a cancer.

PCa occurs when some prostate cells mutate and begin to multiply uncontrollably. These may also spread from the prostate to other parts of the body causing metastasis. PCa commonly spreads to the bones and lymph nodes, although it can also spread to the lungs, bladder, and liver (Saitoh et al., 1984).

1.2.1 A biological introduction to the prostate

The prostate is the largest accessory reproductive gland of the male genital system (Lee et al., 2011). Classically described as “walnut-shaped", it measures about $2 \times 3 \times 4$ cm in thickness, length and width, respectively, and weighs around 30 grams, although its size is affected by age (Zhang et al., 2013). This gland resides inside the pelvis cavity.

The prostate’s main function is to produce seminal fluid, which, together with sperm cells from the testicles, fluid from the seminal vesicles and the secretions released by other glands, constitutes semen (Dixon et al., 1999). This seminal fluid protects, maintains and helps transport sperm, which is crucial for fertility in men. In addition, the prostate’s muscles ensure that semen is pressed into the urethra and released outside during ejaculation.

1.2.1.1 Prostate anatomy

This gland is located behind the base of the penis, in front of the rectum and below the bladder. Figure 1.1 shows the location of the prostate inside the male reproductive system and its surrounding organs. The excretory ducts in

the prostate gland flow into the urethra, which is surrounded by the prostate and carries urine and semen through the penis. The prostate is enveloped by a capsule of connective tissue which contains several smooth muscle fibers and elastic connective tissue. During ejaculation, these muscle cells contract, forcing the fluid that has been deposited in the prostate out into the urethra.

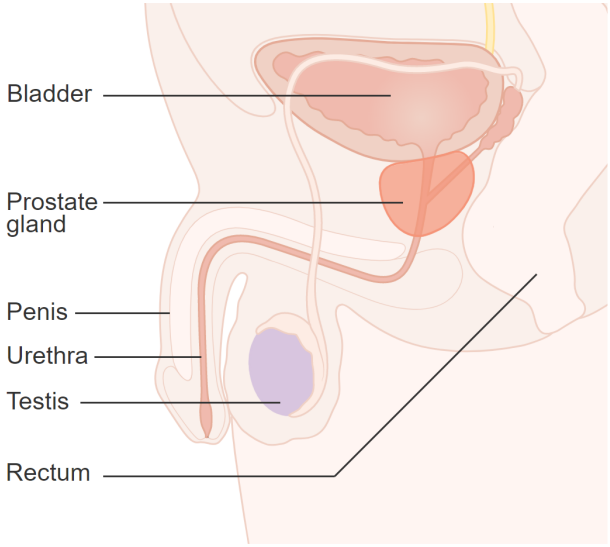

FIGURE 1.1: Position of the prostate within the male reproductive system.

The prostate is divided into three anatomic zones: central, transitional and peripheral zone (Selman, 2011).

- The central zone surrounds the ejaculatory ducts and corresponds to approximately the 25% of the prostate volume. The ducts of the glands from the central zone are obliquely emptying in the prostatic urethra, thus being rather immune to urine reflux.
- The transitional zone surrounds the urethra, comprising approximately 5-10% of the prostate volume. It is the most central area of the gland, circumscribing the distal end of the preprostatic urethra (proximal to the seminal colliculus; where the ejaculatory and prostatic ducts pierce the posterior wall of the prostatic urethra) to a point just proximal to the ejaculatory ducts and the central zone's apex.
- The peripheral zone is the outermost region of the prostate gland and makes up the main body of it (approximately 65%). It encircles the central zone posteroanteriorly and most of the transitional zone. With the exception of the anterior portion of the prostatic urethra, the peripheral zone contains most of the tube.

Together with the three aforementioned zones, some specialists also consider a fourth zone, called the fibromuscular stroma, which is situated anteriorly in the gland and contains the part of the prostatic urethra that is not enclosed within the peripheral zone. Figure 1.2 depicts the four zones of the prostate and their location within the gland.

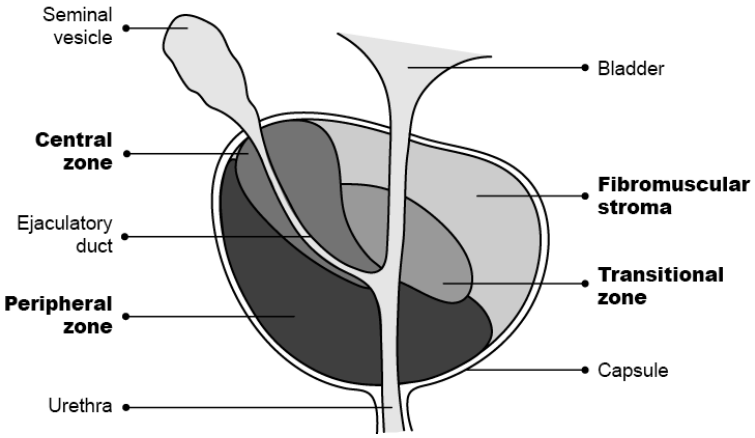

FIGURE 1.2: Close-up look at the prostate gland, highlighting its four main zones.

1.2.1.2 Prostate histology

The prostate consists of around 30 to 50 tubulo-alveolar glands which release the seminal fluid to the urethra through the excretory ducts (Wetter and Vogl, 2016; Hassan et al., 2013). These glands present a convoluted morphology, therefore, their secretory alveoli appear irregular and vary in size.

Normal glands have lumina inside (space of a tubular structure) lined by a variable-height epithelium (tissue made up of one or more layers of cells joined together). The main epithelial cell type is a tall cylindrical secretory cell with prominent, basal, rounded nucleus and clear cytoplasm. The activity of these secretory cells influences the cellular height. More activity induces higher cells, whereas, less activity makes cells cuboidal or nearly flat. At the base of the epithelium of the glands, basal cells are located, corresponding to, approximately, the 10% of a gland. These are small, round, with scant cytoplasm and large irregular nuclei, and they are estimated to be the stem cells of the prostate (Schalken and Leenders, 2003). Glands are enclosed in stroma (connective tissue that normally separates individual glands), which contains smooth muscle, blood vessels and ducts, among others.

Lumen may contain spherical prostatic concretions called corpora amylacea, which are formed by solidification of prostatic secretions. The number of

concretions increases with age, although this increment varies between persons (Hassan et al., 2013). Figure 1.3 presents a section of a tissue image which contains normal tissue together with a zoomed region of a prostatic gland to highlight its main parts.

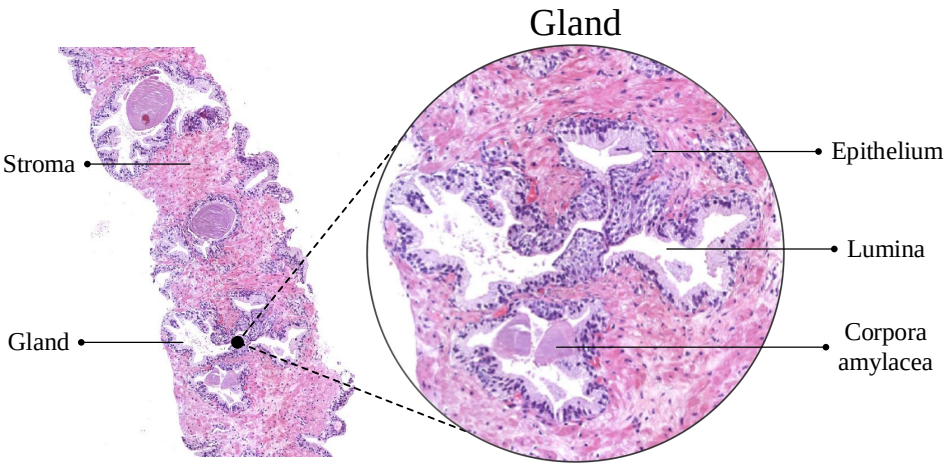

FIGURE 1.3: Histological image of an extracted prostatic tissue section, highlighting its main parts.

The peripheral and central zones contain the majority of glandular tissue (peripheral zone: 70% of glandular tissue; central zone: 25% of glandular tissue), while the transitional zone contains the minor portion (5% of glandular tissue) (Bhavsar and Verma, 2014). On the other hand, glands are not present in the fibromuscular stroma zone (Weinreb et al., 2016).

1.2.1.3 Prostate physiology

The main function of the prostate is the secretion of prostatic fluid, which, together with secretions from the seminal vesicles and sperm, constitutes approximately the 30% of semen. The prostatic secretion is an alkaline liquid that neutralizes vaginal acid content, provides nutrients and transports the sperm. The most prominent protein components contained in prostatic secretion are Prostatic Acid Phosphatase (PAP), Prostate Binding Protein (PBP) and Prostate-Specific Antigen (PSA) (Dixon et al., 1999). In addition, a high concentration of zinc is also found, which is densely presented in spermatozoon's head, and may contribute to chromatin stability (Björndahl and Kvist, 2011).

PSA is a serine protease secreted from the prostatic epithelium into the secretory ducts, whose physiological function is to liquefy semen in ejaculate. It is a relevant biomarker for the diagnosis of some prostatic pathologies, such as PCa, since PSA is predominantly released in prostatic secretion and, thus,

only a very small amount (approximately 4 ng/mL) circulates in the blood under normal conditions (Velonas et al., 2013).

The slightly alkaline character of prostatic fluid may be important for ovum fertilization since sperm is relatively acidic due to the presence of citric acid among other important components, which, consequently, may inhibit sperm fertility. Additionally, alkaline prostatic fluid helps neutralize female vaginal secretions, which are acidic (pH of 3.5 to 4), and may prevent sperm from achieving optimal motility (pH in the range of 6 to 6.5) (Guyton and Hall, 2006).

1.2.2 Epidemiology

According to GLOBOCAN, PCa is the second most frequently diagnosed cancer among men, with more than 1.2 million cases, and the fifth leading cause of cancer death in men with around 350000 deaths in 2018 (Ferlay et al., 2019). It is the most common cancer among men in more than half of the world's countries (105 of 185), particularly in the Americas, Northern and Western Europe, Australia/New Zealand, and most of Sub-Saharan Africa (Bray et al., 2018). Moreover, this cancer is the leading cause of cancer death among men in 46 countries, notably in Sub-Saharan Africa and the Caribbean. Figure 1.4 presents the global statistics of both worldwide incidence and mortality rates for 36 type of cancers, including PCa, considering 185 countries.

PCa death rates have been declining in several countries such as those in Northern America, Oceania, Northern and Western Europe, developed Asian countries and the United States. This fact has been attributed to earlier diagnosis and improved treatment, which has resulted in a genuine postponement of death for some men with metastatic cancer (Bray et al., 2018).

According to WHO, there will be an increase of PCa cases worldwide, with 1017712 new cases being estimated for 2040. Most of these cases will be registered in Africa, Latin America, the Caribbean and Asia, and appear to be related to an increased life expectancy (Rawla, 2019).

1.2.3 Causes

Based on epidemiological observations, five main risk factors of PCa have been identified. A risk factor is anything that increases a person's likelihood of developing cancer. While risk factors directly influence the probability of producing cancer, they do not imply its development[1]. Thus, people with several known risk factors could may never develop any cancer, whereas, in other cases, other people that do not have any associated risk factor could contract the disease. The main risk factors of developing PCa are age, race, genetic, hormonal, environmental and infectious factors.

[1] https://www.cancer.net/navigating-cancer-care/prevention-and-healthy-living/understanding-cancer-risk (accessed on June 30, 2021)

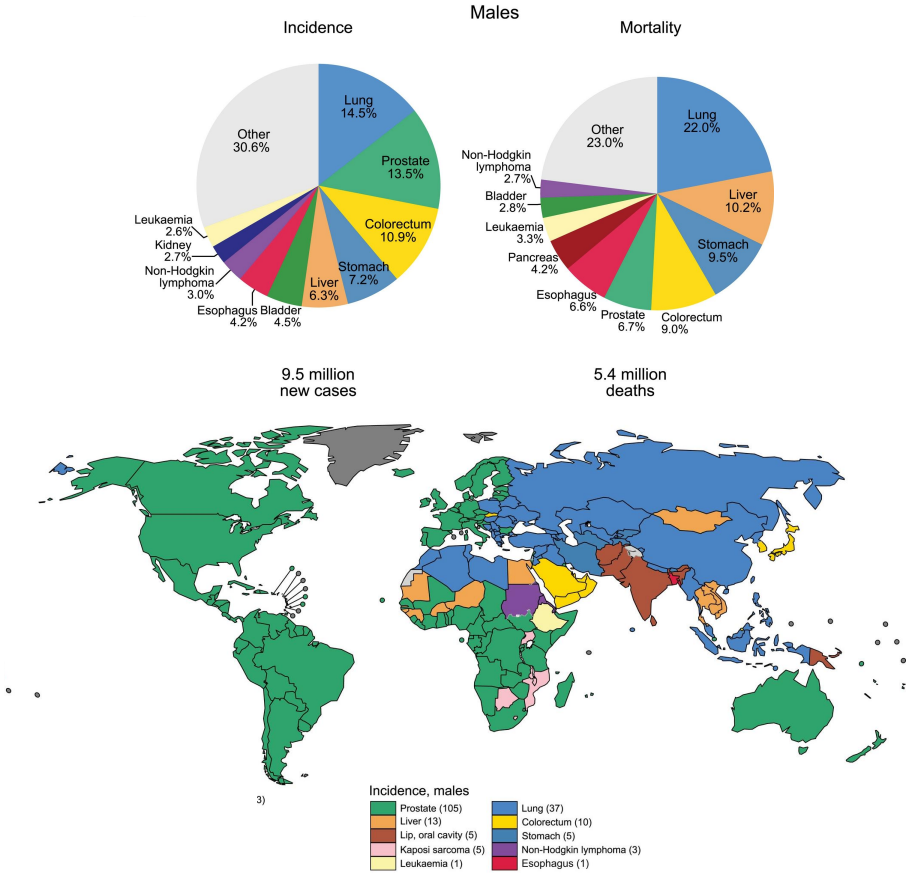

FIGURE 1.4: Diagram showing cancer incidence and mortality rates in males in 2018. Top: pie charts for incidence and mortality rates for the ten most common cancers among men. Bottom: world map representing the most frequent cancers among men for each country. Image taken from GLOBOCAN 2018 (Bray et al., 2018)

- **Age**

  As it was mentioned, age is the major risk factor for the development of any kind of cancer. Therefore, the risk of PCa increases with age, particularly in people with more than 50 years old. More than 64% of PCa cases are diagnosed in 65-years-old men or older, and 23% in men older than age 75 years (Bechis et al., 2011).

- **Race**

  There has been wide variation in the reported incidence of clinical PCa

among different ethnic groups (Pienta and Esper, 1993). The incidence of clinical PCa is low in Asian men and higher in Scandinavian men, according to both incidence and mortality statistics. However, it is unclear if these differences are based on life expectancy, diet, socioeconomic status, genetic or environmental factors (Pienta and Esper, 1993).

- **Hormonal factors**

  Several researchers have studied how hormonal factors may be considered as a relevant factor in the development of PCa (Bostwick et al., 2004). These suggest that PCa growth rates are greatly influenced by androgens (Bostwick et al., 2004). This theory is based on different reasons, including that this disease has not been observed in eunuchs (Wu and Gu, 1991), where, due to the castration, a total androgen suppression is made. In addition, it has been proved that PCa can be induced in rats by chronic administration of estrogens and androgens (Ozten et al., 2019). Moreover, some studies suggest that elevated concentrations of testosterone may increase PCa risk, although results have been inconsistent (Bostwick et al., 2004).

- **Environmental factors**

  Several environmental factors, such as diets exceeding in regular amounts of animal fat (Fleshner et al., 2004), the exposure to vehicle exhaust fumes and air pollution (Parent et al., 2013), have also been identified to be promoters of PCa. Furthermore, endocrine disrupting chemicals (EDCs), which can be defined as environmental agents that positively or negatively alters hormone activity, has raised awareness. It has been observed that EDCs elicit effects on estrogen, androgen, and/or thyroid activities (Bostwick et al., 2004).

- **Genetic factors**

  One of the significant risk factors for the development of PCa is the presence of this disease in the family health history. Some cases of hereditary PCa are caused by inherited mutations in particular genes, such as BRCA1, BRCA2, and HOXB13, which have been associated with more aggressive disease and poor clinical outcomes (Castro and Eeles, 2012; Ewing et al., 2012). Men with mutations in these genes have a high risk of developing PCa during their lifetimes.

- **Infectious agents**

  It has been suggested that sexually transmitted infectious agents, and the subsequent inflammation, may be an important risk factor in the pathogenesis of PCa (Sutcliffe, 2010; Caini et al., 2014). Among the sexually transmitted infections, gonorrhea, syphilis and human papillomavirus

infection (HPV) are considered to be highly related to the development of this disease (Taylor et al., 2005).

1.2.4 Diagnosis procedure

Digital Rectal Examination (DRE) is the primary test for the initial clinical assessment of the prostate (Borley and Feneley, 2009). A DRE of the prostate allows physicians to know, in the majority of cases, if the prostate is normal or if it presents an abnormality. These abnormalities could be caused by the growth of a tumor, such as when the gland is larger, presents a nodule, or if there is a loss of definition of the anatomical shape.

Then, a PSA analysis is performed as a screening method for the investigation of an abnormal result on DRE (Borley and Feneley, 2009). This test measures the concentrations of PSA in the blood. As it was previously mentioned, this biomarker is a substance produced by the prostate that can be found in increased amounts on men who have PCa. However, it is a test that is not conclusive, since there are two situations to take into account. Firstly, in more than 20% of cases of PCa, the PSA does not increase, and, secondly, in a very high percentage of men, an elevated PSA is detected, particularly as they get older, without the presence of PCa (Velonas et al., 2013). For this reason, this test is usually complemented with an ultrasound, a biopsy of the prostate, or both (National Collaborating Centre for Cancer (UK), 2008).

The biopsy consists in extracting small tissue samples of the prostate (Borley and Feneley, 2009). It is the most reliable test to confirm or exclude the presence of cancer. Transrectal ultrasound is used to guide and insert a thin, hollow needle through the wall of the rectum into certain areas of the prostate gland. Then, the needle is used to remove a cylinder of tissue (core), usually one centimeter long and 2 millimeters wide, which is sent to a pathologist for examination. This technique is called core needle biopsy and it is the main method used to diagnose PCa (Kwast et al., 2003; Renshaw, 1997). Several biopsy samples are extracted from different areas of the prostate. Different studies suggest that 10 to 12 cores are optimal to have a representative sample of the gland where the cancer's involvement could be seen (Presti Jr, 2003).

After the tissue is extracted during a biopsy, pathologists inspect and examine the obtained samples. However, direct observation under the microscope does not allow the pathologist to observe the morphological characteristics of the cells within the tissue. Therefore, these tissue samples are processed in a laboratory by using specific stains to enhance the contrast of biological structures.

One of the most popular staining methods used in histology and diagnostic medicine is Hematoxylin and Eosin (H&E) stain (Chan, 2014). H&E is the combination of two histological stains: hematoxylin and eosin. Hematoxylin

stains acidic (basophilic) structures in blue and purple colors, such as cell nuclei, due to its cationic or basic behaviour, while eosin stains basic (acidophilic) components in pink color due to its anionic or acidic nature, such as cells' cytoplasm. The H&E stain provides a general overview of a tissue sample's structure, aiding pathologists to discriminate easier between the nuclear and cytoplasmic parts of a cell, and, consequently, allows them to analyze the overall patterns and cell distribution.

Traditionally, a pathologist would analyze and inspect these slides under a microscope. However, with the advent of new technologies and computational advances, digital pathology emerges as a new alternative to traditional approaches (Pantanowitz, 2010). Digital pathology employs virtual microscopy with the use of computer-based technology. This way, slides are converted into digital images that can be viewed, managed, shared and analyzed in a computer. These gigapixel-resolution digital slides are called Whole-Slide Images (WSIs) (Figure 1.5).

FIGURE 1.5: Thumbnail of a WSI. Each region corresponds to two different tissue slices from the same sample. The gray section of the WSI is an unwanted area, which does not contain tissue, ignored when scanning the slide. Tissue stained with bluish-purple colors are due to the hematoxylin, which stains acidic structures, while pink colors are due to the eosin, which stains basic components.

1.2.5 Histopathology

Around 95% PCas begin when secreting gland cells mutate into cancer cells (Bast Jr et al., 2010; Humphrey, 2017), which are called adenocarcinomas (from Greek: "malignant tumor originated from cells of the glandular epithelium"). As the peripheral zone of the prostate gland concentrates the majority of glandular tissue, it is the most common region for developing an adenocarcinoma (around 70% of the tumors) (Swallow et al., 2012). Adenocarcinomas can also be located

in the central zone (25%) or in the transitional zone (5%). Since no glands are found in the fibromuscular stroma, no tumors can be originated there.

In general, prostatic adenocarcinoma is composed of small to intermediate sized glands with a tendency to form irregular clusters, growing between large benign glands (Lavery et al., 2016). As cell differentiation is lost, i.e. cancer cells resemble less to normal cells and tend to grow and spread faster, the size of the glands decreases, and they may attach together and form clumps. The characteristic cells of a prostatic adenocarcinoma present enlarged nuclei with prominent nucleoli and abundant cytoplasm. The basal cell layer is present in normal glands, but they are absent in prostate tumors (Lavery et al., 2016). They also tend to accumulate proteinaceous secretory material called crystalloid (Bennett and Gardner, 1988).

1.2.5.1 Gleason Grading System

Serum levels of PSA and tumor's clinical staging (size of the tumor and whether if it has spread) are some of the most important elements that allow pathologists to determine the tumor status and to predict the biological behavior of the tumor, and, thus, to decide the most appropriate therapy for each patient. However, together with these factors, evaluating and accurately measuring the tumor's aggressiveness is essential in order to decide the best therapeutic option. Even though there are numerous grading systems for the evaluation of prostatic adenocarcinoma, the Gleason Grading System (GGS) is the most widely accepted (Gordetsky and Epstein, 2016).

The GGS is focused on determining the cellular differentiation degree of a tumor, considering 5 different patterns (1 to 5) (Amin and Tickoo, 2016). In order to assign a specific pattern or grade, pathologists observe the sample at low magnifications ($5\times$ or $10\times$), which provides a general overview of the tumor's structure, and, then, higher magnification objectives ($20\times$ or $40\times$) are used for visualizing cellular detail in order to confirm the diagnosis. Gleason pattern 1 is assigned to areas of the tissue containing cells that resemble to normal prostate cells, whereas, in pattern 5, cancer cells greatly differ to normal prostate cells. This way, the higher the pattern, the higher the aggressiveness of the cancer and the lower the differentiation between cancer cells. Figure 1.6 presents the differences between the different patterns of GGS. Pathologists observe the structure of the cells in WSIs and assign a lower or higher pattern depending on whether the appearance is that of healthy or abnormal tissue, respectively.

- **Pattern 1**

  The most well-differentiated tumor pattern is Gleason grade 1. It is a well-defined small cluster of cells or nodules with single, separate, closely and densely packed gland pattern that does not invade healthy prostatic tissue (Pierorazio et al., 2013; Epstein et al., 2005). This pattern is extremely rare to find, if not non-existent (Chen and Zhou, 2016).

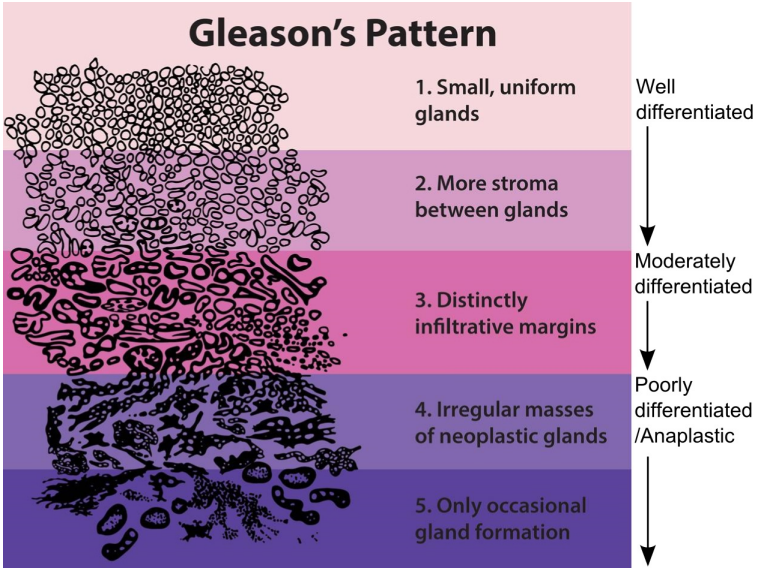

FIGURE 1.6: GGS diagram describing 1-5 Gleason patterns. Well-differentiated cancer cells resemble to normal cells, and they tend to form and spread more slowly than poorly differentiated or undifferentiated cancer cells.

- **Pattern 2**

  Gleason pattern 2 corresponds to fairly well circumscribed nodules of single, separate glands. However, the glands are looser in arrangement and less uniform than pattern 1. Thus, the main difference between Gleason 1 and 2 is the density of gland packing observed, and while invasion is not possible in Gleason 1 by definition, in pattern 2 it can occur (Pierorazio et al., 2013; Epstein et al., 2005). This grade is also considered to be very unusual (Chen and Zhou, 2016).

- **Pattern 3**

  Gleason pattern 3 is an infiltrative tumor that has spread to nearby healthy prostate tissue (Figure 1.7). The glands vary in size and shape (they are often long and angular) and tend to infiltrate into the stroma in between the benign glands. In contrast to Gleason 1 and 2 grades, they are typically small/microglandular although some of them may be medium to large in size. The small glands of Gleason 3 are distinct glandular units (Pierorazio et al., 2013; Epstein et al., 2005). Gleason pattern 3 has been seen in many series as the most common pattern (Chen and Zhou, 2016).

- **Pattern 4**

Gleason pattern 4 glands are no longer single/separated glands in comparison with those seen in patterns 1-3 (Figure 1.7). They appear fused together, difficult to discern, and have rare lumen formation compared to Gleason 1-3, which normally have open lumen within the glands. Fused glands are chains, nests, or groups of glands that are no longer fully isolated by stroma. Also, Gleason grade 4 glands may be presented with a cribiform pattern in which the tumor appears to have open spaces or small holes in it, similar to a sieve (Pierorazio et al., 2013; Epstein et al., 2005).

- **Pattern 5**

  In Gleason pattern 5, there is no glandular distinction in the tumor, thus, not resembling to normal prostate tissue at all (Figure 1.7). It is composed of sheets (groups of cells that tend to be almost planar), clumps, or individual cells. Round glands with lumenal spaces, which resemble more to the normal prostate gland appearance, should no longer be seen (Pierorazio et al., 2013; Epstein et al., 2005).

The two most predominant Gleason patterns in a WSI are summed up to determine the corresponding Gleason score, which ranges from 2 to 10. However, pathologists almost never use scores 2 to 5, since, as mentioned above, patterns 1-2 are very unusual to find, being 6 the lowest Gleason score (Chen and Zhou, 2016). A Gleason score of 7 corresponds to a mid-grade cancer, and a score of 8-10 correspond to a high-grade cancer. A lower-grade cancer grows more slowly and it has a lower risk of spreading than a high-grade cancer.

As mentioned before, GGS is currently the most widely used grading system for PCa. However, many studies (Lessells et al., 1997; McLean et al., 1997) have reported inter-observer variability among pathologists when diagnosing PCa with this system (more than 30% degree of discrepancy in the score, as reported in Arvaniti et al., 2018b and Salmo, 2015). In Berg et al., 2011, the authors investigated the consequences when re-evaluating 350 patients, comparing the results with the primary pathology reports. There was full agreement between primary reports and the re-evaluations only in around 76.9% of the cases, where re-evaluations were scored higher. According to the authors, this fact may lead to changing the clinical assessment and surgical strategy, and, therefore, it justifies the re-evaluation of PCa patients with a primary low Gleason score diagnosed. This involves double effort, requiring several pathologists to analyze the same images, which is not a very efficient practice.

1.3 Computer-Aided Diagnosis systems

Although the percentage of agreement among professionals in the interpretation of prostate WSIs is acceptable (around 70%) (Berg et al., 2011), this inter-observer variability based on the subjectivity of each pathologist's own skills and experience should be contemplated (Gurcan et al., 2009).

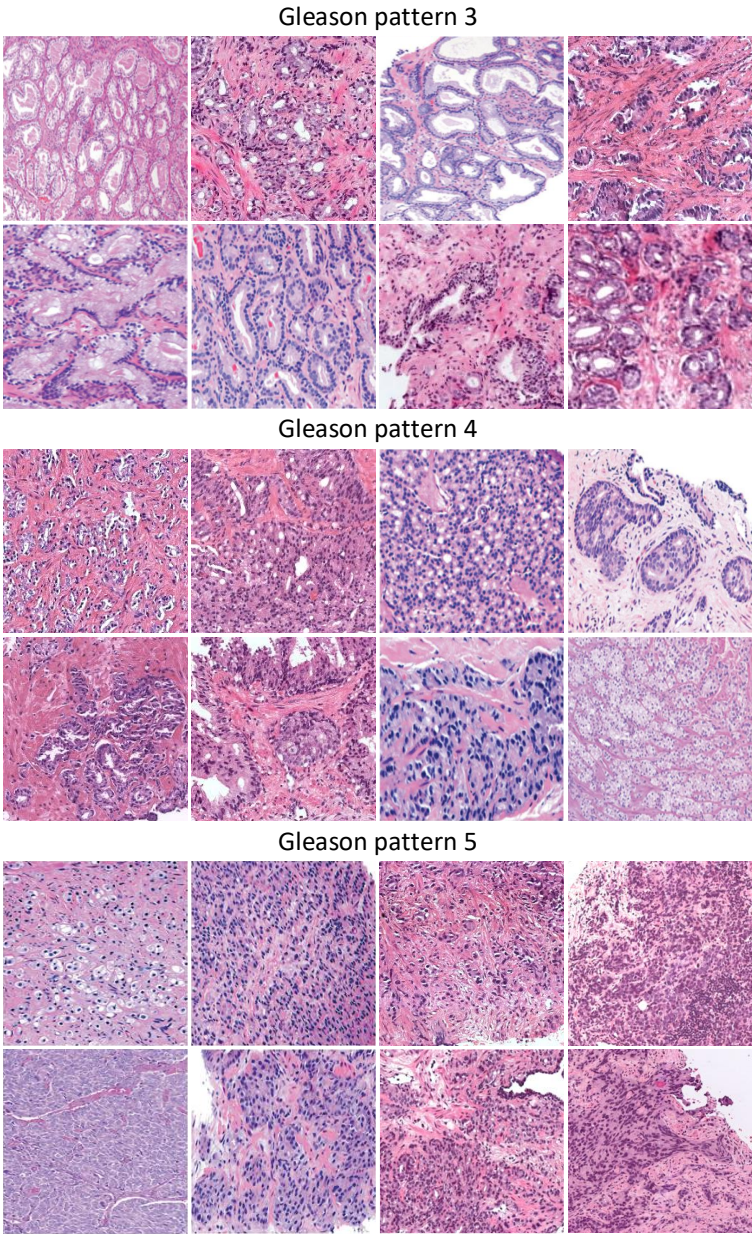

FIGURE 1.7: Examples of GGS patterns 3-5.

On the other hand, during their routine work, pathologists have to analyze a large number of WSIs, which, as introduced, are huge high-resolution gigapixel

images, and, therefore, take much time for the specialist to completely analyze, making the whole process tedious. In addition, PCa is not always present in all cases, in fact, it represents just a small percentage among all patients. Inspecting all samples in order to search for abnormalities (in the case these exist) is time-consuming and demands a high level of concentration, in addition of being laborious to pathologists. Moreover, due to the large number of cases to be routinely analyzed and the considerable amount of normal cases, pathologists may miss some subtle abnormalities when inspecting the slides. Therefore, the drawbacks of the traditional diagnosis lead medical professionals and scientists to explore alternative approaches.

To this respect, Computer-Aided Diagnosis (CAD) systems, which combine elements from both medicine and computer science fields, emerged as a new interdisciplinary technology with a potential future in digital pathology. CAD systems are automatic or semi-automatic algorithms whose main goal is to assist specialists by reporting a second opinion when making an interpretation of medical images.

1.3.1 History

The first CAD systems were developed in the late 1950s, soon after the computer age began (Yanase and Triantaphyllou, 2019). Biomedical researchers started to study the possibility of solving biological and medicine problems by means of computer technology. These systems called "expert systems in medicine" were based in flow-charts, statistical pattern matching, probability theory or knowledge bases as the main decision drivers.

However, by the early 1970s, this kind of systems had some significant limitations for providing accurate diagnosis. The recognition and acceptance of these limitations led researchers to begin developing new kinds of CAD systems by using advanced approaches. Therefore, by the late 1980s and early 1990s, data mining algorithms became the new research topic for the purpose of building more advanced and flexible CAD systems. After that, researchers have focused on artificial intelligence and specialized computer algorithms, such as pattern recognition and classification algorithms, as the main pillar to develop new CAD systems.

In 1998, the first commercial CAD system called the ImageChecker system was developed and approved by the US Food and Drug Administration (FDA). The ImageChecker system was intended for use as an aid for radiologists when reading routine screening mammograms. To date, several commercial CAD systems, involved in breast, lung, colon, and heart imaging, have also been approved by the FDA (Yanase and Triantaphyllou, 2019).

Currently, researchers have investigated and analyzed the use of CAD systems to automatically diagnose different diseases, including some types

of cancer, such as breast (Jalalian et al., 2013), lung (El-Baz et al., 2013) and liver (Chang et al., 2017) cancers; cardiovascular diseases (Faust et al., 2017); Alzheimer's disease (Ramírez et al., 2010); among others. Moreover, these algorithms have proved to be effective in a wide range of image modalities (Doi, 2005), from radiological imaging, including conventional X-ray radiology, computed tomography, ultrasound, magnetic resonance imaging and fluoroscopy; to histological imaging. In the particular case of histology, researchers found drawbacks when processing WSIs, due to their huge size. A solution for this problem, which has become a common approach when processing histological images, is to divide them into small subimages called patches. This procedure has been widely used in order to develop CAD systems in this field (Roy et al., 2019; Litjens et al., 2016; Hou et al., 2016).

Nowadays, this kind of systems are considered as a potential future component of a diagnostic process to provide physicians for better medical decision-making (Doi, 2007), which also, unquestionably, involves human experts participation.

1.4 Deep Learning

Artificial Intelligence (AI) emerged in the mid-20th century as a new complex and misunderstood discipline (Haenlein and Kaplan, 2019). However, over the last years, the higher computing power and the enormous amount of digitally-stored data led to the massively popularization of AI applications, which have spread to the everyday scene (Poola, 2017). The success of these algorithms resides on their own capability to learn rules from data, in contrast to the conventional and traditional analysis techniques, which are based on the execution of pre-programmed rules. Figure 1.8 shows some of the most important events in the history of AI.

In this context, one of its most prominent branches is Machine Learning (ML), which proposes an analytical and automatic modeling of data (Alpaydin, 2016). To this end, analysis is approached as a learning process, where the programmer provides a series of starting rules that the learning algorithm has to adapt and create new ones, thus, trying to improve the accuracy rate of the generated model.

At the same time, within ML, there is a subset of algorithms known as Deep Learning (DL). This approach is inspired by the structure and functioning of the human brain, which is the reason why these methods are commonly referred to as neural networks. In the last decade, DL techniques have become one of the most popular branches of AI, providing results far superior to those obtained with other ML methods (Arel et al., 2010). DL has already proved its success in several applications (Ahmad et al., 2019). Even some major technology companies,

such as Google, Facebook and Microsoft, have already incorporated them as development tools for their products (Parloff, 2016).

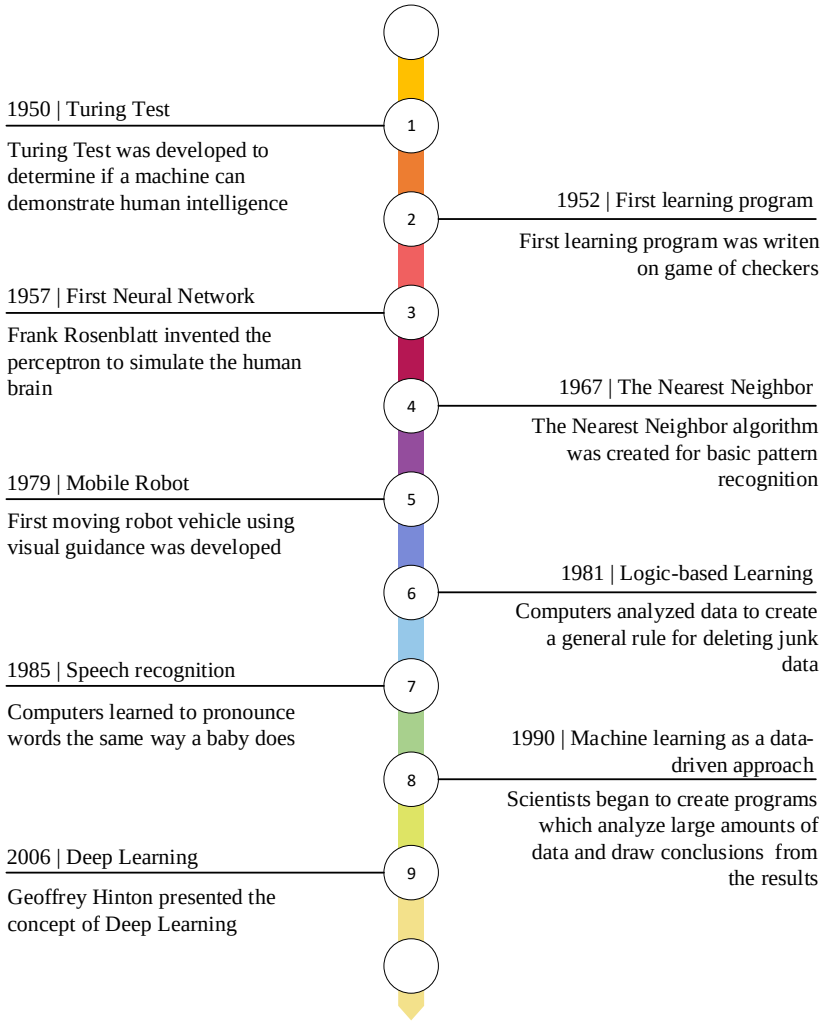

FIGURE 1.8: Some of the most relevant events in the history of AI.

1.4.1 An introduction to Artificial Neural Networks

The brain is the most complex information processing system that we know (Bassett and Gazzaniga, 2011). It is able to perform certain operations such as spatial and temporal pattern recognition, and processing sensory information in real time. The human brain consists of, approximately, 10 billion neurons

with massive structural and functional interconnections called synapses, which mediate interactions between them. It presents a property called plasticity, which promotes the development of the nervous system to adapt to its environment (Huttenlocher, 2009). Plasticity allows the creation of new synaptic connections between neurons and also the modification of existing ones.

An Artificial Neural Network (ANN) is inspired by the human brain's operation. It could be defined as a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use (Simon, 1999). An ANN resembles the brain in two approaches: knowledge is acquired by the network from its environment through a learning process, also called training process, and interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge. During the learning process, network's synaptic weights are adapted in order to achieve a desired design goal.

1.4.1.1 Artificial neuron models

The basic information processing unit of a neural network are artificial neurons, which are inspired in their biological counterpart. Although the human brain presents different types of these cells, artificial neuron is based on the most common type of biological neuron. In a simplified form, a biological neuron consists of different parts:

- A set of terminal branches from which external signals are received, called dendrites.
- The central body, called soma, which contains the cell nucleus and processes the combination of stimuli received through the dendrites.
- An extension of the soma, called the axon, which allows communication of the signal resulting from the processing carried out by the nucleus.

The first mathematical model which imitates the functionality of a biological neuron, called McCulloch-Pitts Neuron Model, was presented in 1943 with the purpose of performing simple tasks (Hayman, 1999). This artificial neuron received binary inputs and produced a binary output based on a certain threshold value which could be adjusted. This model supposed a relevant event in the AI history. However, the McCulloch-Pitts neuron presented some limitations, such as only accepting boolean inputs and not allowing weights, which made the model less flexible (Minsky and Papert, 2017).

In 1958, Frank Rosenblatt created the perceptron model based on the McCulloch-Pitts neuron (Rosenblatt, 1958). In contrast to the first model, the perceptron presents two advantages: first, it can process any real input value, and, second, projections between neurons are weighted. This neuron model consists of five basic elements:

1. A set of data inputs x_1, ...x_n.
2. A set of synaptic connections, each characterized by a weight w_1, ... w_n, corresponding to each input.
3. An aggregation function, $\sum$, to sum the input signals, weighted by the respective synapses of the neuron.
4. An activation function, φ, to limit the amplitude of the neuron's output.
5. An output, Y.

The inputs are the stimulus that the artificial neuron receives from the surrounding environment, and the output is the response to that stimulus. As in a human brain, the neuron can adapt to the surrounding environment and learn from it by modifying the value of its synaptic weights. This way, the output Y of an artificial neuron is defined as:

$$Y = \varphi(\sum_{i=1}^{n} w_i x_i) \tag{1.1}$$

where i refers to each data input. Figure 1.9 presents a representation of the perceptron model highlighting its main components, together with a simple biological neuron. The activation function φ returns an output from an input value (Sharma, 2017). The most common activation functions are:

- Sigmoid

  The sigmoid function is a continuous, monotonically increasing function with a characteristic 'S'-like curve (Figure 1.10). It transforms the values in the range 0 to 1. In addition, the sigmoid function is not symmetric around zero with respect to the Y-axis, which implies that the signs of all output values will be positive. It was the first activation function implementation and the basis of most neural networks for many decades, although in recent years it has lost popularity. The main problem of the sigmoid function is that it has a saturation zone limited within the range 0 and 1. This zone means that at the output of the neuron, and during certain phases of the network training process, the values do not change significantly, which may cause a possible "stagnation". The sigmoid function is defined in Equation 1.2 (Sharma, 2017).

$$\varphi(x) = \frac{1}{1 - e^{-x}} \tag{1.2}$$

- Rectified Linear Unit (ReLu)

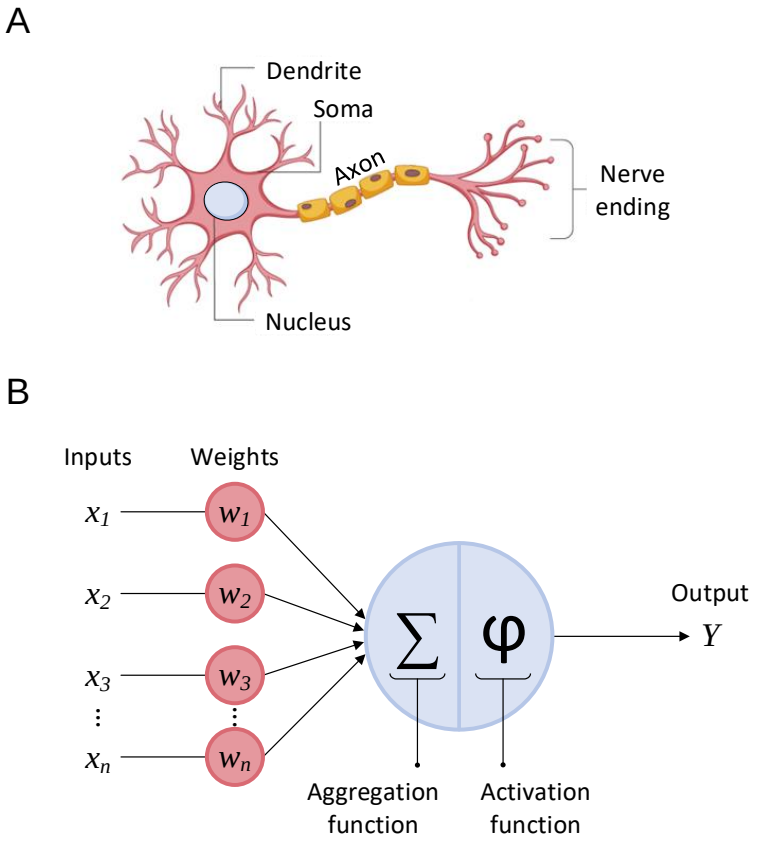

FIGURE 1.9: Biological neuron representation (A) and perceptron model (B).

The ReLu is a piecewise linear function which returns 0 if it receives any negative input, while for any positive value x it returns that value back (Figure 1.11). It has become the default activation function for many types of ANNs, since it makes the training phase easier to the network and also often achieves better performance than other activation functions (Agarap, 2018; Shang et al., 2016). The advantage of using the ReLU function over the sigmoid is that it does not have any saturation region, as it has a linear behavior for positive inputs. The ReLu function is mathematically described in Equation 1.3 (Sharma, 2017).

$$\varphi(x) = max(0, x) = \begin{cases} 0 & for \quad x < 0 \\ x & for \quad x \geq 0 \end{cases} \tag{1.3}$$

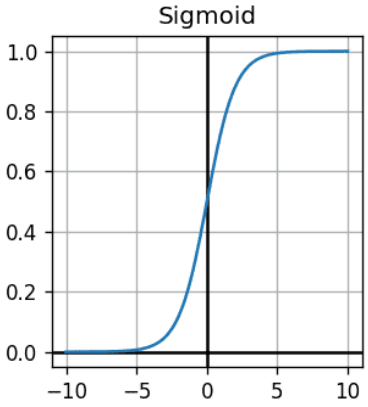

FIGURE 1.10: Representation of the sigmoid activation function.

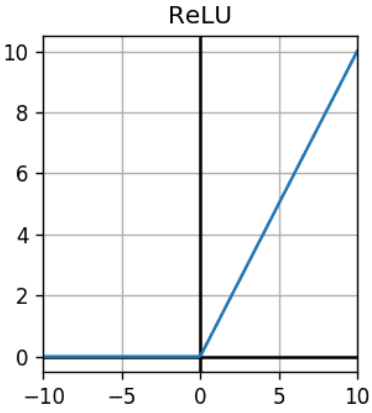

FIGURE 1.11: Representation of the ReLu activation function.

- Softmax

  The Softmax activation function is a generalization of the logistic function used for multi-class cases. This function compresses a K-dimensional vector, x, of arbitrary real values into a vector of real values in the range 0 to 1. The sum of the probabilities of each class must sum 1.0. It is commonly used in the last layer of a multi-class ANN, as it normalizes the output values establishing a probabilistic distribution. This additional constraint allows the training process to converge faster. The softmax function is defined in Equation 1.4 (Sharma, 2017).

$$\varphi(x)_j = \frac{e^{x_j}}{\sum_{K}^{k=1} e^{x_k}}, j = 1, ..., K \tag{1.4}$$

1.4.1.2 Neural network architecture

As it was mentioned, neurons are connected by synapses in a neural network, being the behaviour of the network determined by the structure of the synaptic connections. This structure, topology or connection pattern of a neural network is called architecture.

In general, neurons are usually grouped into structural units called layers (Abraham, 2005). A set of one or more layers constitutes the neural network. Three types of layers are distinguished: input, output and hidden layers (Figure 1.12). The input layer is composed of neurons that receive data from the environment, the output layer is composed of neurons that provide the response of the neural network, while the hidden layers reside between the input and output layers, not being visible to external systems. The larger the number of hidden layers in a neural network is, the longer it will take for the neural network to produce the output and the more complex problems the neural network will be able to solve.

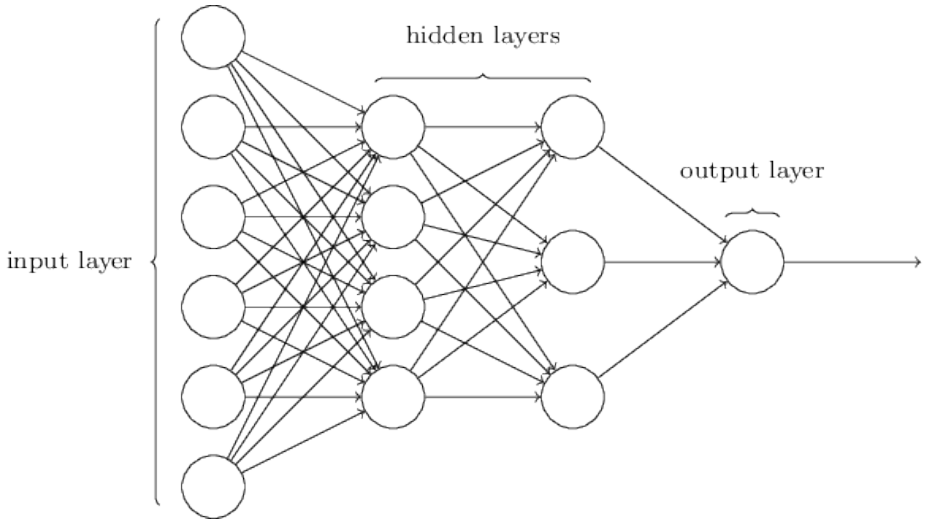

FIGURE 1.12: An example of an ANN architecture with an input layer, an output layer and two hidden layers.

Different types of neural architectures can be established depending on the concept we focus on. Thus, considering their structure, there are monolayer networks, composed of a single layer of neurons, or multilayer networks, in

which neurons are organized in several layers. Considering the flow of the data, neural networks can be distinguished between unidirectional networks, named as feedforward networks, and recurrent networks or feedback networks (Abraham, 2005). In feedforward networks, information circulates in a single direction, whereas in recurrent networks information can circulate between the different layers of neurons in any direction, even in the output-input direction (Figure 1.13).

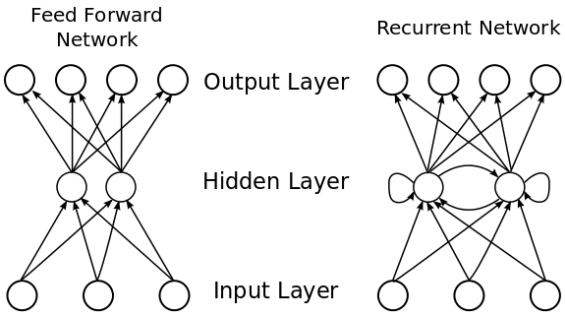

FIGURE 1.13: Comparison between a feedforward network and a recurrent network.

1.4.1.3 Learning process in a neural network

The learning phase of a neural network depends on which task it will try to solve. Most tasks are classified into two main groups: classification (also known as pattern recognition) (Zhang, 2000) and regression (Specht et al., 1991). Classification networks associate a given input to an output class. The second type of networks are used for estimating the relationship between a dependent variable and one or more independent variables.

The development of a neural network consists mainly in two stages. The first includes designing, training and validating the network. Training is the procedure for carrying out the learning process, in which the synaptic weights of the network are modified, allowing the network to determine the appropriate response for a given input stimuli; while the validation is used to confirm the generalization of the designed model and to report a set of metrics to measure how well the model learnt. The second stage is commonly referred to as the production phase, during which the neural network is already operational, and both its structure and the values of the weights are not modified. During this phase, the neural network is used effectively to solve the problems for which it has been designed.

Regarding the training phase, two types of training processes can be found: unsupervised and supervised learning (Sathya and Abraham, 2013). The unsupervised learning bases its training process on a dataset without previously

defined labels or classes (ground truth). Therefore, no class value, either categorical or numerical, is known. Unsupervised learning is dedicated to grouping tasks, also called clustering or segmentation, where its purpose is to find similar groups or clusters of data in the dataset. In contrast, in supervised learning, the network learns through known input and output patterns, so they fit into a set of examples with an associated label or target for which it is known the relationship between the input and the desired output.

The most used algorithm for training an ANN in a supervised manner is gradient descent (Kiefer, Wolfowitz, et al., 1952). This is a first-order iterative optimization algorithm whose purpose is to minimize any differentiable function by finding a local minimum. In the training phase the function to be minimized is the loss function (also called cost function), which quantifies the error between the prediction and the ground truth. The main goal of training a neural network is to set the best values for the synaptic weights (parameters) for which the network minimizes the loss function. The training starts with random weights and, during the learning phase, these are adjusted so that the error between the output obtained and the desired output is minimal.

The following set of steps are performed for each of the iterations of the gradient descent algorithm:

1. First, a batch of N random samples from the training set is used as input.
2. Then, this batch is forward-propagated through the set of layers of the network, performing all the corresponding operations in between and obtaining the predictions at the output.
3. After the previous step, the loss function is evaluated for the input batch. As mentioned above, this function evaluates the difference between the obtained predictions and the ground truth labels. Through the different iterations, the gradient descent algorithm tries to minimize the value of the loss function.
4. Then, partial derivatives of the lost function with respect to each of the network parameters is calculated and the results are stored in a gradient. Since neural networks contain a massive amount of parameters, calculating the gradient is not trivial. Consequently, the well-known backpropagation algorithm is used (Leung and Haykin, 1991). This algorithm consists in start calculating the partial derivatives of the loss function at the output only with respect to the parameters of the last layer, which is simplified thanks to the use of the chain rule (Rojas, 1996). Once obtained, the same is computed but for the previous layer, and so on, until reaching the first layer.
5. Once the gradient is obtained, the parameters of the neural network are updated by subtracting the corresponding gradient value multiplied by the learning rate (which allows adjusting the steepness of each training step)

to their current value. In this process, the gradient is subtracted instead of added to the parameter, since the goal is to decrease the cost function and, thus, to move in the opposite direction of the gradient. Theoretically, the closer the solution gets to the global minimum, the smaller the steps will be, since the slope of the cost function will be smaller.

This five-step process is repeated for a configurable number of iterations (epochs), or can also be stopped when the value of the loss function and the output metrics stop improving, or even start to worsen. This way, the learning process is repeated epoch after epoch until the parameters are stabilized and the network performance converges to the best value.

When training a neural network, choosing a good learning rate value is very important, since, as explained in the fifth step, it affects how the weights of the network are modified (Li et al., 2019). Small learning rates could lead to needing a very high number of epochs before reaching the minimum point, whereas selecting a very high learning rate could cause drastic updates which lead to divergent behaviors, as shown in Figure 1.14.

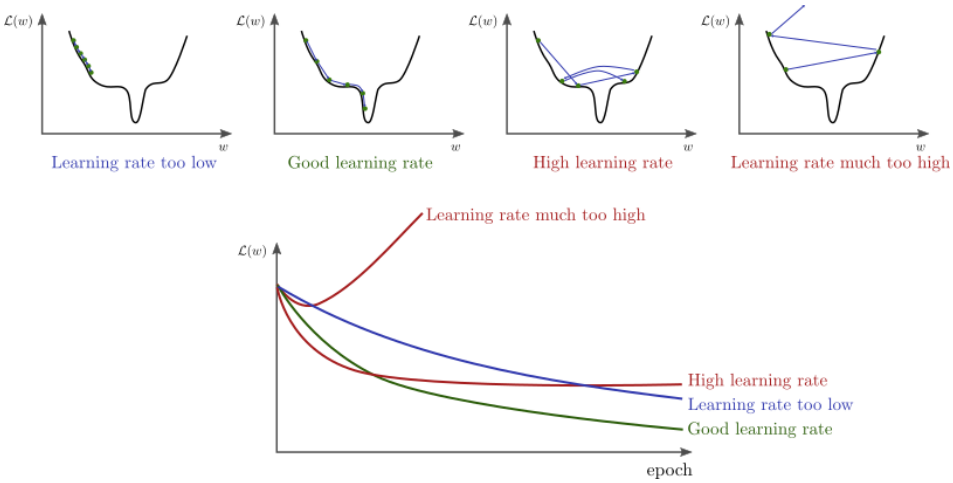

FIGURE 1.14: Effect of the learning rate value when training a neural network. $L(w)$ corresponds to the value of the cost function and w corresponds to the weight.

1.4.2 Convolutional Neural Networks

A Convolutional Neural Network (CNN) is a type of feedforward ANN where neurons correspond to receptive fields which resembles the neurons in the primary visual cortex of a biological brain. Since its application is performed on two-dimensional arrays, CNNs are very effective for computer vision tasks, such as image classification and segmentation, among other applications (Khan

et al., 2018). They are designed to automatically and adaptively learn spatial hierarchies of features.

The operation of a CNN consists in two main stages (Figure 1.15). Firstly, CNNs extract features from input images. This step is performed by the application of convolutional operations, which are the main difference between CNNs and other types of networks. In the feature extraction step there are also other kind of layers that improve and accelerate both the learning and the execution processes. Then, the final classification is performed on the extracted features by layers of perceptron neurons, which are found in the last part of the CNN architecture.

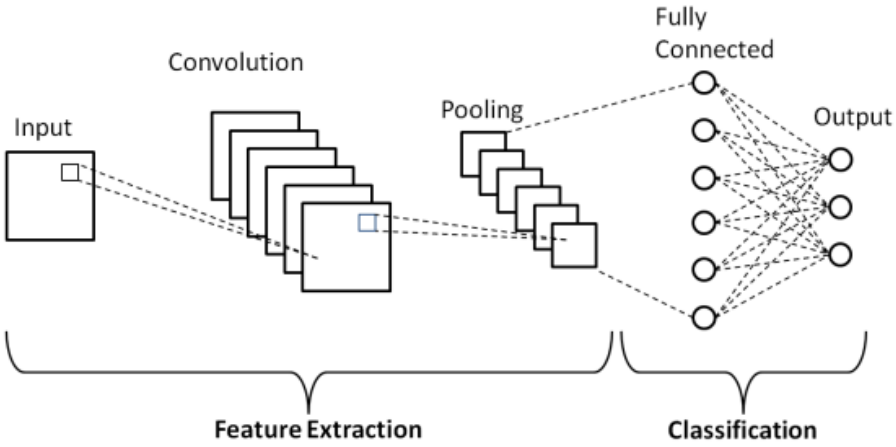

FIGURE 1.15: Example of a CNN architecture, consisting of a feature extraction phase and a classification phase.

1.4.2.1 Convolution layer

The main purpose of the convolution layer is to extract features from the input it receives. This layer is composed of a series of filters or kernels that aim to extract local features. These kernels (also called filters) are matrices defined by values known as weights. The weights of the convolution layer are calculated in the learning phase so that the network minimizes the classification error made based in the training data. Each kernel is used to extract or compute different characteristics obtained from the previous layer. These are called feature maps, which serve as input data to the next layer in the architecture.

The convolution operation allows the network to extract features more efficiently, reducing the number of parameters to learn and, thus, optimizing the training process. Although ANNs could also be used to extract features and learn

patterns from images, it is not feasible due to the vast number of neurons that would be necessary to process them.

The convolution process consists in sliding a kernel K of size $m \times n$ over all the elements of an input image I of size $i \times j$ where for each displacement the scalar product of the filter elements and the input elements is calculated to finally obtain the feature map S (Equation 1.5). The amount by which the filter slides is the stride, which controls how the filter convolves around the input image. Figure 1.16 presents an example of a convolution operation applying a kernel of 3×3 to an input image (5×5), obtaining an output image of size 3×3.

$$S(i,j) = (K \star I)(i,j) = \sum_m \sum_n I(i-m, j-n) K(m,n) \tag{1.5}$$

The first convolution layer extracts significant low-level features such as edges, corners, textures and lines, while the higher level features are extracted in the last convolution layer.

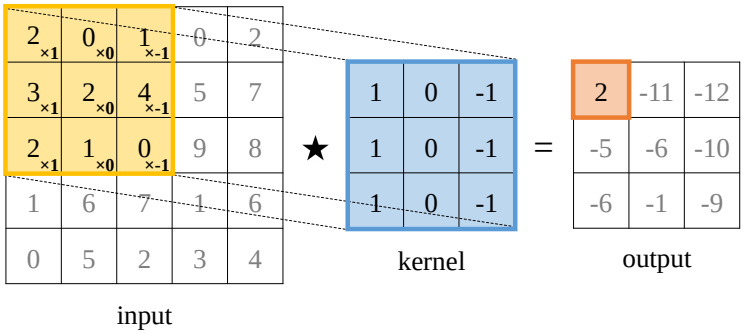

FIGURE 1.16: Example of a convolution operation with a 3×3 kernel and a stride of 1×1.

1.4.2.2 Pooling layer

The pooling layer is a non-linear down-sampling filter which reduces the resolution of the previous feature maps by compacting them, and, thus, reducing the computational complexity of the network. Pooling ensures that the network learns the most relevant patterns to perform the classification. In contrast to the convolution layer, since the pooling layer does not include any configurable parameter that need to be adjusted, this layer is not affected by the training process. In a CNN architecture, it is usual to add a pooling layer between each convolution layer, each one followed by an activation function such as ReLU.

The pooling layer consists in splitting the input feature maps of size $W \times W$ into regions of size $R \times R$ (kernel) to generate one output from each region. The output size P is given by Equation 1.6.

$$P = \left[\frac{W}{R} \right] \tag{1.6}$$

Depending on the operation performed to generate the output, there are different types of pooling layers. The most popular are max-pooling and average-pooling. There are other types of pooling layers that are not so widely used, and, thus, these are not covered in this section.

- **Max pooling**

  The max pooling extracts the most representative value (the maximum) from each split region when applying a kernel, thus reducing the feature map size. It is the most common pooling type used, and usually configured with a kernel size of 2×2. Figure 1.17 shows an example of max pooling, where each of the elements of the output feature map is calculated as the maximum value for each split region.

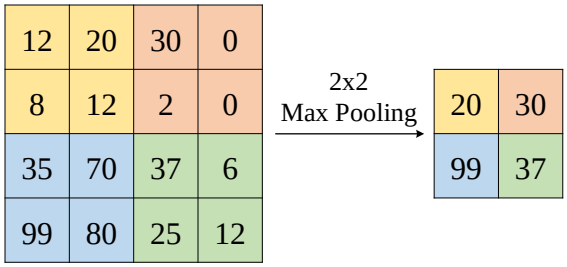

FIGURE 1.17: Example of a max pooling operation with a 2×2 kernel and a stride of 1×1.

- **Average pooling**

  The average pooling, also known as avg pooling, calculates the arithmetic mean from each split region when applying a kernel. This means that each $R \times R$ region of the feature map is down-sampled to the average value in the region. An example of the average pooling function is shown in Figure 1.18.

1.4.2.3 Fully-connected layer

Fully Connected (FC) layers are used as classifiers in the CNN architecture. The operations resulted after applying a set of convolution, pooling and activation function layers are intended for extracting relevant characteristics as feature maps

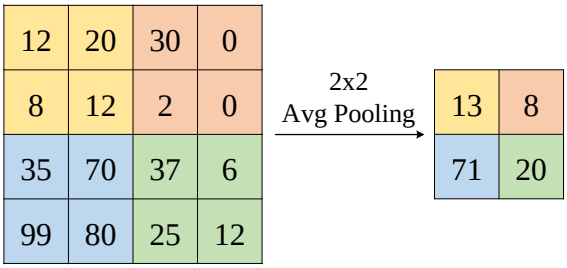

FIGURE 1.18: Example of an average pooling operation with a 2×2 kernel and a stride of 1×1.

from the original image. This learned feature representation is finally mapped by the FC layer to perform a prediction. The number of outputs in the last FC layer correspond to the number of different classes to be classified. Figure 1.19 presents an example of a FC layer.

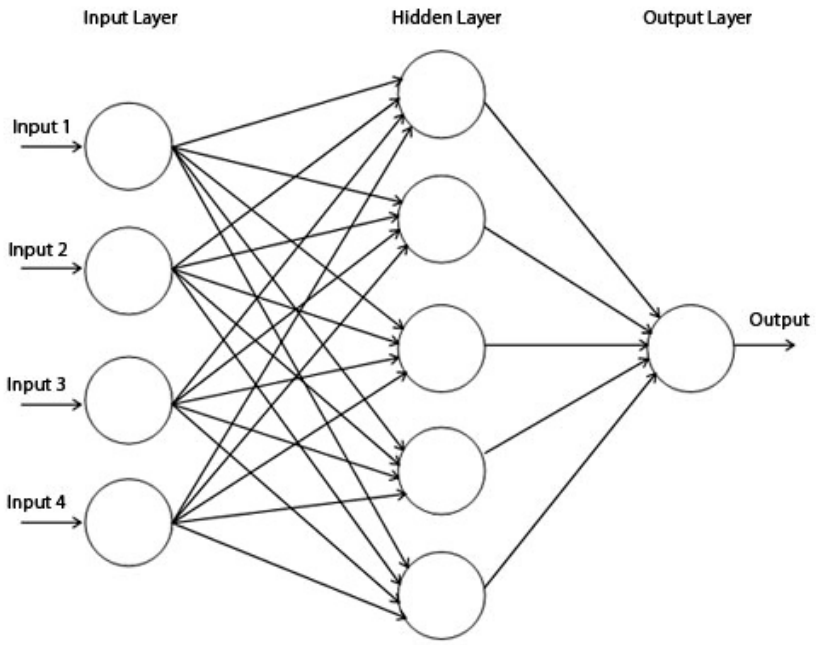

FIGURE 1.19: Example of three consecutive FC layers, with an input layer, one hidden layer and an output layer.

1.4.2.4 State-of-the-art Convolutional Neural Network architectures

In the recent years, many custom CNN architectures have been developed to solve real-world problems and, due to their success, they have become very popular. Some of these include LeNet-5, AlexNet, VGG, Inception, Xception, ResNet, DenseNet and MobileNet.

- **LeNet-5**

  LeNet was developed by Yann LeCun in 1998 for hand-written digit recognition with the MNIST dataset (LeCun et al., 1998). It achieved 99.2% accuracy on isolated character recognition. This model is the most widely-known CNN architecture as it was the first application of CNNs. LeNet-5 CNN architecture is made up of 7 layers. The layer composition consists of 3 convolution layers, 2 pooling layers and 2 FC layers (see Figure 1.20).

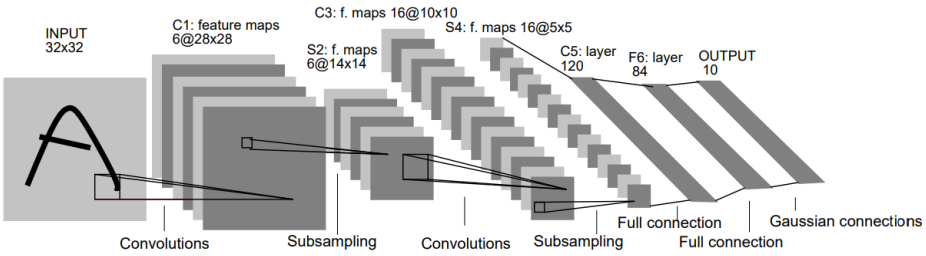

FIGURE 1.20: LeNet-5 architecture. Image taken from LeCun et al., 1998.

- **AlexNet**

  AlexNet is a popular CNN architecture designed in 2012 by Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton, which won the ImageNet Large Scale Visual Recognition Challenge with a test accuracy of 84.6% (Krizhevsky et al., 2012). This challenge consisted in evaluating algorithms for object detection and image classification at large scale in order to classify more than 20000 categories, such as "dog" or "plane". The network consists of 5 convolution layers and 3 FC layers. Figure 1.21 shows the architecture of the AlexNet model.

- **VGG**

  In 2014, researchers at the Visual Geometry Group (VGG) invented a CNN model, called VGG-16 (Simonyan and Zisserman, 2014), that stacks more layers onto AlexNet in order to make it deeper and to improve its performance. This model has 13 convolution layers and 3 FC layers, and use smaller size filters than AlexNet. A deeper variant of this model, called VGG-19, was also developed by the same group.

- **Inception**

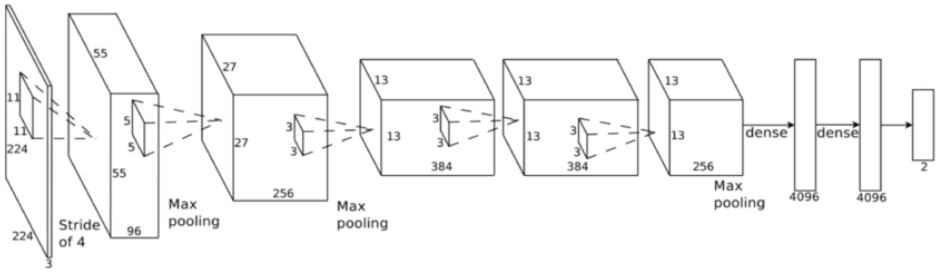

FIGURE 1.21: AlexNet architecture. Image taken from Krizhevsky et al., 2012.

The first version of the Inception model was created by Google researchers in 2014. The novel idea of this architecture consisted in concatenating the results obtained from applying convolution and pooling filters of different size to the same input. This allows the model to benefit from multilevel feature extraction and, therefore, capturing different features at 1×1, 3×3 and 5×5, thereby "clustering" them. Different Inception versions have been developed over time in order to improve previous architectures: InceptionV1 (Szegedy et al., 2015), InceptionV2 and InceptionV3 (Szegedy et al., 2016) and InceptionV4 (Szegedy et al., 2017).

- **Xception**

  This architecture is an extension of the Inception model also developed by Google researchers (Chollet, 2017). The main difference with its predecessor is that it introduces the concept of depthwise separable convolutions. Therefore, instead of approaching convolution as a single step in which a single feature map is obtained as a result of convolving a filter with the input, in the Xception model, a feature map is obtained for each channel. The general meaning of this convolution is to perform convolution and fusion on each depth map separately, which greatly reduces the amount of parameters.

- **ResNet**

  ResNet (Residual Neural Network) was designed by Microsoft and it became popular for wining the ImageNet competition (2015) with an accuracy of 96.4% (He et al., 2016). ResNet model uses skip connections or shortcuts (residual blocks) to skip training from a set of layers and connect them directly to the output. The purpose of these residual blocks is to avoid the problem of vanishing gradients or to mitigate the accuracy saturation problem. The advantage of adding this type of skip connection is that, in case any layer hurts the performance of the architecture, it will be skipped by regularization.

The ResNet-34 model is inspired by VGG-19 and consists of 34 layers in which residual blocks are added. Other deeper architectures based on this same principle are ResNet-50 and ResNet-101.

- **DenseNet**

  DenseNet (Densely Connected Convolutional Networks) was developed by Gao Huang, Zhuang Liu, and their team in 2017 (Huang et al., 2017). In this architecture each layer is directly connected to every other layer in a feed-forward fashion. For each layer, the feature maps of all preceding layers are treated as separate inputs, whereas its own feature maps are passed on as inputs to all subsequent layers. Although there are different DenseNet architectures, DenseNet-121 is the most widely used.

- **MobileNet**

  MobileNet is an architecture proposed by Google, designed particularly for mobile vision applications (Howard et al., 2017). Therefore, it focuses on reducing as much as possible the computational power required for the algorithm. It seeks a very simple architecture even though this makes it sacrifice some accuracy and performance. This is largely achieved through the use of depthwise separable convolutions.

1.4.3 Tools for the implementation of Deep Learning algorithms

Some years ago, when DL was not so widespread, the development of CNN algorithms was a laborious task and was not available to everyone. Nowadays, the situation has changed thanks to the large number of open source software frameworks that have been developed, which greatly facilitate the design and training of these kind of models. These also allow to abstract researchers from the peculiarities of the neural network's development in order to speed up the designing and training processes.

There are several software platforms, libraries and frameworks to develop CNNs. Currently, the most popular, free and open-source software libraries are TensorFlow (Abadi et al., 2016) and Keras (Chollet et al., 2015), followed by the also very well-known software platform called PyTorch (Paszke et al., 2019). Other noteworthy platforms are Theano (Bergstra et al., 2010), Caffe (Jia et al., 2014) and Caffe2[2]. Figure 1.22 shows the interest evolution of the most popular DL frameworks in the last five years based on data obtained from Google Trends.

All these platforms allow researchers to perform all the steps involved in the development of DL algorithms, which includes the design of the architecture, the training and the validation processes. These two last procedures require a high computational cost, as even the simplest neural network requires a large number of mathematical operations to be performed. Conventional computers would not

[2]https://caffe2.ai (accessed on June 30, 2021)

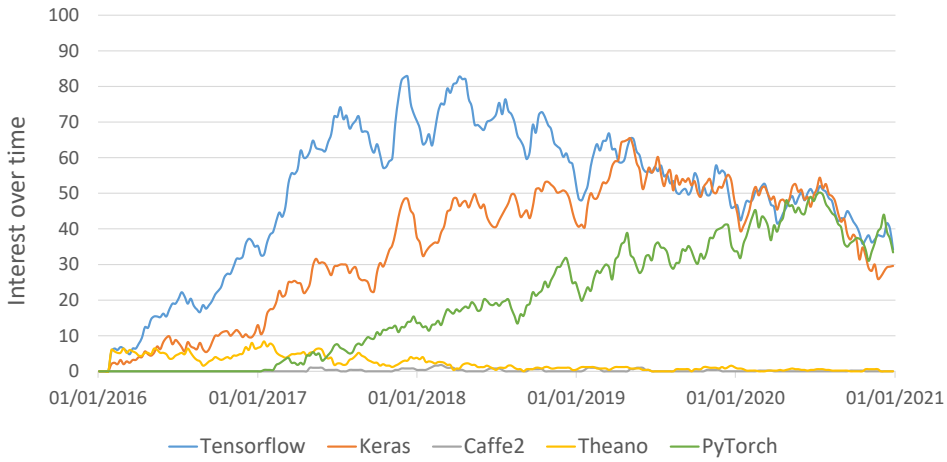

FIGURE 1.22: Worldwide interest evolution of the most popular DL frameworks over time. Information obtained from Google Trends.

be able to handle so much information or would take too long to complete these tasks without additional help. In order to speed up the process, it is important the use of GPUs when working with this type of algorithms (Schlegel, 2015). This acceleration is possible thanks to NVIDIA's CUDA library, which uses parallelism to run a high number of simultaneous threads through GPU's multiple cores, making the whole process achieve better performance. In particular, the library that allows working with DL algorithms is cuDNN, which is based on CUDA. All of the above mentioned frameworks have cuDNN support.

In this Thesis, the implemented DL-based algorithms have been developed using both TensorFlow and Keras, which are detailed next.

- TensorFlow[3] was developed by Google and released as open source software on 2015. This tool was designed for creating multiple ML algorithms, including neural networks. The name TensorFlow derives from the operations such neural networks perform on multidimensional arrays of data. These multidimensional arrays are referred to as "tensors". Tensorflow runs on all the platforms from mobiles to embedded devices and also distributed servers.
- Keras[4] is a high-level Python library commonly used to create neural networks to solve complex challenges, which works as a wrapper to TensorFlow or Theano. It is designed to be modular, fast and easy to use, thus, facilitating the creation of DL algorithms when working with TensorFlow.

[3]https://www.tensorflow.org (accessed on June 30, 2021)
[4]https://keras.io (accessed on June 30, 2021)

Chapter 2

Objectives

At the beginning of this Thesis, two types of objectives were proposed: general objectives, with the aim of analyzing and studying the viability of neural networks and CAD systems for PCa detection in histopathological images; and more specific objectives, focused on solving particular problems related to the topic introduced previously and the design of the corresponding systems and neural networks.

General objectives: search for and study of CAD systems for medical image processing, focusing on PCa histopathological images.

1. Study of the histopathology of PCa, focusing on GGS.
2. Study of different neural network architectures and learning algorithms for training systems to perform a specific task, and how these could be applied to medical image analysis.
3. Development of new systems for processing histological images from prostate biopsies.

To achieve this general objective, which is broad and ambitious, the following set of specific objectives were proposed:

Specific objectives: design, implementation and validation of different mechanisms to perform PCa detection and classification in histopathological images based on DL algorithms.

1. Dataset generation:
   (a) Digitization of prostate biopsy samples to obtain a collection of WSIs.
   (b) Study and analysis of state-of-the-art software tools for visualizing and annotating WSIs.
   (c) Development of a new desktop software application in order to allow pathologists to load and visualize the scanned images with the purpose of annotating them and generate a report.

   (d) Study and development of new algorithms for generating datasets from the digitized images and the annotations obtained from pathologists.

2. Prostate cancer detection in WSIs using Convolutional Neural Networks:
   (a) Study of CNNs and how they work.
   (b) Evaluation and study of different DL frameworks for designing, training and testing different neural network architectures.
   (c) Generation of a dataset with patches extracted from the labeled WSIs with the aim of performing a classification between malignant and normal patches.
   (d) Development of a set of filters to pre-process patches.
   (e) Design of a custom CNN model to perform the classification.
   (f) Validation and quantification of the obtained results.
   (g) Generation of a heatmap-like plot showing the malignant tissue areas within the WSI.
3. Performance evaluation of DL-based PCa screening methods:
   (a) Design of a benchmark algorithm to evaluate the performance of DL-based PCa detection CAD systems.
   (b) Analysis of the performance of the proposed CAD system and its processing steps in different hardware platforms.
   (c) Study and evaluation of state-of-the-art prostate cancer detection CAD systems.
   (d) Comparison of the improvements of the proposed CAD system over other works.
4. Patch aggregation in DL-based PCa detection systems:
   (a) Study of different mechanisms to perform patch aggregation.
   (b) Development of a set of algorithms to obtain relevant spatial and statistical features from patch-level classification results obtained after analyzing WSIs with the proposed CAD system.
   (c) Generation of a training and testing dataset.
   (d) Design of a custom ANN model to perform the classification.
   (e) Validation and quantification of the obtained results.
5. Development of a global CAD system for Gleason pattern classification in WSIs:

(a) Study the GGS system and the difference between Gleason patterns.

(b) Generation of a dataset with patches extracted from the labeled WSIs with the aim of performing a classification between GGS patterns 3, 4 and 5.

(c) Design of a custom CNN model to perform the classification.

(d) Validation and quantification of the obtained results.

(e) Generation of a heatmap highlighting the malignant tissue regions with their corresponding Gleason pattern.

(f) Integration and connection of the prostate cancer detection CNN, the patch aggregation system and the GGS classification CNN into a global CAD system.

Chapter 3

Prostate cancer detection in WSIs using Convolutional Neural Networks

3.1 Introduction

Recently, many researchers have investigated the application of CAD systems to the diagnosis of PCa based on different methodologies. Some of these studies use ML techniques, such as neural networks, Support Vector Machines (SVMs), or some complex algorithms to carry out the classification (Kwak and Hewitt, 2017; Toro et al., 2017; Litjens et al., 2016; Li et al., 2018; Doyle et al., 2007; Ren et al., 2017; Campanella et al., 2019; Ström et al., 2020; Bulten et al., 2020; Arvaniti et al., 2018b), while others are based on algebraic tools, such as Homology Profile algorithms, which extracts features from a structure of a topological space (Yan et al., 2020). Many of them have performed a binary classification (Kwak and Hewitt, 2017; Toro et al., 2017; Litjens et al., 2016; Doyle et al., 2007; Ren et al., 2017; Campanella et al., 2019), distinguishing between cancerous and normal tissue or between different GGS scores, whereas others have performed a multi-class detection (Li et al., 2018; Arvaniti et al., 2018b; Ström et al., 2020; Bulten et al., 2020).

For this kind of systems, preprocessing the information could be a key factor to make it easier for the classifier to extract the most relevant features from the input images. Background and noise removal are key processes to consider when working with histopathological images. Otsu's thresholding (Otsu, 1979) is one of the most well-known and used methods for extracting background and tissue from WSIs (Kwak and Hewitt, 2017; Arvaniti et al., 2018b). In Toro et al., 2017, the Blue Ratio method, which detects nuclei from cells in stained images, is used to obtain tissue regions. Other simpler mechanisms to remove background are based on thresholding procedures on the optical density of the RGB channels (Litjens et al., 2016).

Stain normalization has also proved to be useful for histopathological images, since it reduces color variations that could have been produced in the staining process of the tissue sample (Ciompi et al., 2017). This has been used

in different cancer studies based on histopathological images (Vesal et al., 2018). In Roy et al., 2018, the authors compared the effect of applying different stain normalization methods in histopathological images for liver, breast, kidney and colorectal cancer.

Table 3.1 presents a comparison of some of these studies, summarizing the characteristics of the dataset, the preprocessing step applied to the data, the main classification method procedure of the CAD system, the number of classes taken into account and the results obtained with their corresponding performance metrics. These works have used many different techniques for the preprocessing step, although apart from Doyle et al., 2007, which uses SVMs, the rest have performed either the classification or part of the preprocessing by using CNNs. As introduced in Section 1.4, these complex architectures have increased in popularity in the recent years thanks to the rise in the computation capabilities of current general purpose computers, reducing the gap of achieving a robust and accurate CAD system.

In this experiment, a novel DL-based CAD system for PCa detection in WSI images is presented to support pathologists in this task. A CNN was trained and tested over a new dataset that was built and labeled with the supervision of expert pathologists after processing the images with novel algorithms to improve cancer detection and robustness across WSIs from different hospitals and scanners.

3.2 Materials and methods

3.2.1 Dataset

Training a CNN requires a large amount of data to make the classifier learn and converge to the wanted solution. The lack of free and open datasets with the sufficient amount of samples, and with reliable labels associating the pixels in every image with a specific class, is always a restriction when trying to develop a CAD system for medical image analysis.

For this experiment, a novel dataset that was analyzed and labeled by expert pathologists was created. In this dataset, malignant regions of the WSIs considered by the pathologist for such diagnosis were specified. This kind of labels could provide the necessary information to train a learning system in order to extract relevant features from the cell structures contained in them and, thus, detect specific patterns. Figure 3.1 depicts the whole process applied for obtaining our dataset.

3.2.1.1 Data acquisition and labeling

To obtain a reliable dataset, a collaboration with the Pathological Anatomy Unit of Virgen de Valme Hospital in Seville (Spain) was established. They provided a

TABLE 3.1: Comparative study between state-of-the-art research about PCa detection.

Ref.	Dataset	Preprocessing step	Classifier	Classes	Performance measure
Kwak and Hewitt, 2017	4 TMAs[1]: - Train: 73 (cancer) + 89 (normal) cores - Test: 217 (cancer) + 274 (normal) cores	Otsu's thresholding, Euclidean distance and Watershed algorithm to perform nuclear seed detection. Nuclear seed maps are used as input to the classifier.	CNN[2] (custom)	2: Cancer and normal	AUC[3] at core level: 0.974
Litjens et al., 2016	225 WSIs[4]: - Train: 48 (cancer) + 52 (normal) WSIs - Val: 31 (cancer) + 19 (normal) WSIs - Test: 45 (cancer) + 30 (normal) WSIs	Binary tissue mask by applying thresholding procedure based on optical density of RGB channels to remove background.	CNN (custom)	2: Cancer and normal	AUC at slide level: 0.99
Campanella et al., 2019	24859 WSIs: - Train: 70% - Val: 15% - Test: 15%	Otsu's thresholding to remove background.	CNN (ResNet34) + RNN[5]	2: Tumor and normal	AUC at slide level: 0.986
Ström et al., 2020	8914 WSIs: - Train: 6953 WSIs - Val: 1631 WSIs - Test: 330 WSIs	Segmentation algorithm based on Laplacian filtering.	CNN (60 Inception V3)	2: Normal and malignant 3: GGS[6] 3, GGS 4, GGS 5	AUC for normal and malignant*: 0.997 on validation 0.986 on test Mean pairwise kappa for GGS*: 0.62 *at slide level

[1]: Tissue Microarray. [2]: Convolutional Neural Network. [3]: Area Under Curve. [4]: Whole Slide Image. [5]: Recurrent Neural Network. [6]: Gleason Grading System.

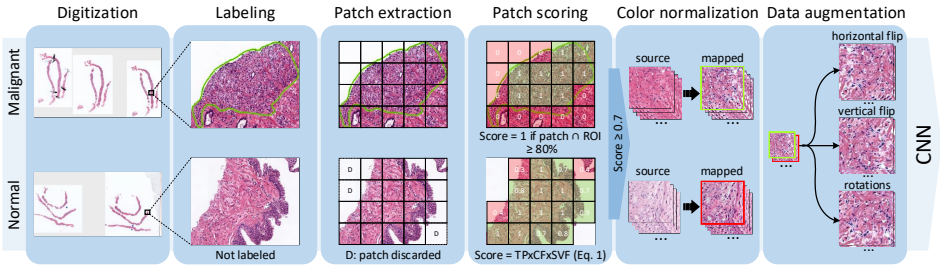

FIGURE 3.1: Flow chart of the whole dataset acquisition and the different preprocessing steps applied.

large set of PCa cases obtained from different patients. These cases consisted in different H&E stained slides (diagnosed as normal or malignant) obtained from needle core biopsy. Then, they were digitized with a VENTANA iScan HT[1] scanner from Roche Diagnostics.

[1]https://diagnostics.roche.com/global/en/products/instruments/ventana-iscan-ht.html (accessed on June 30, 2021)

Once the biopsies were scanned and digitized, the following step consisted in labeling the WSIs. To this end, a desktop software application was designed and developed in C# and Windows Presentation Foundation (WPF) with Microsoft ® .NET Framework with the purpose of allowing pathologists to categorize specific regions of the tissue as malignant. Using this application, experienced pathologists examined WSIs in order to find malignant areas, considered as Regions of Interest (ROIs), indicating the GGS pattern that they belong to, and thus, labeling each of the WSI images. For a more precise and comfortable labeling process, pathologists used computer drawing pads from Wacom® to mark the ROIs inside WSIs. The essential attributes for the dataset creation are summarized in Table 3.2.

TABLE 3.2: Dataset summary.

Attributes	Details
Staining method	Hematoxylin and Eosin stain
Scanner	VENTANA iScan HT from Roche Diagnostics
Scanner resolution	0.25 μm per pixel
Total number of WSIs	97
Optical magnification	10$\times$

3.2.1.2 Patch sampling

As mentioned in Section 1.3, due to the large size of the WSIs obtained from the process presented before (100k$\times$100k pixels, approximately), using them as a direct input for the CNN is not doable. To this end, these images were divided into small patches (100$\times$100 pixels at 10$\times$ optical magnification) in order to obtain a dataset that the neural network could work with for the training, validation and testing steps, ensuring that all patches of a patient are only in one of these subsets. This division would also have some other effects. First of all, it would speed up the computation time for processing a complete WSI, since unwanted areas such as noisy regions of the image or background would not be taken into account. Then, this would also increase the overall accuracy, robustness and reliability of the system, since more images would be considered for training the network. Finally, the CAD system would also be more precise in locating malignant areas of the tissue, which could better help pathologists, rather than just predicting if a whole WSI is malignant or not, as an unique and global diagnosis.

The quality of the dataset is crucial in the training step as well as when testing the network. The lesser number of noisy patches the dataset contains, the more robust and better fitted the training step of the network would be, leading to achieving better results. For these reasons, it is important to discard all unwanted regions (background and noisy regions) from the dataset and only consider areas which contain prostate tissue. Figure 3.2 shows some common noisy agents that could be present in WSIs.

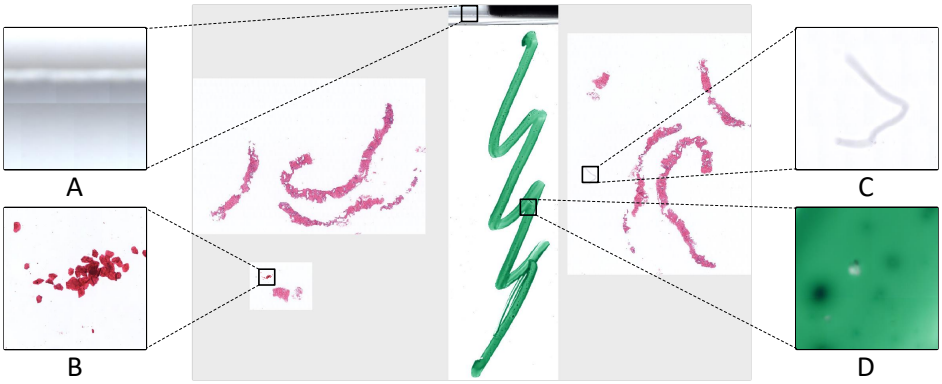

FIGURE 3.2: WSI with unwanted areas: regions which correspond to the edge of the slide cover (A), cells from external tissue not related to the prostate (B), external agents such as dirt (C) and zones highlighted with pen (D).

To obtain the dataset, different patch-extraction algorithms were applied for WSIs labeled as normal and malignant. It is important to mention that patches labeled as normal were only obtained from WSIs diagnosed as normal, and patches labeled as malignant were obtained from ROIs of WSIs diagnosed with cancer, avoiding possible malignant tissue regions that pathologists could have missed when labeling a malignant WSI. For malignant WSIs, the ROIs selected by the pathologists were framed with a polygon, which was then scanned by overlapping patches (with 50% overlap between them) as in Li et al., 2018 and Campanella et al., 2019, due to the smaller amount of malignant patches in comparison with the normal ones. Overlapping was only applied to malignant WSIs in the cross-validation set, and not in the test set (see Section 3.2.2.2). Those patches which had at least 80% of its area within the ROI were considered, and the rest of them were discarded. For normal WSIs, all patches which contained tissue were extracted, following two consecutive processes: first, background patches were discarded based on an RGB value threshold, where patches with a mean color value close to either white or black were removed (below 30 and above 230, using a 8-bit color depth); then, patches corresponding to unwanted areas (noise) were discarded by applying a novel filter process based on Deron

Eriksson's patch scoring formula[2].

This filter applies a score (in a scale that ranges from 0 to 1) to each extracted patch depending on three subfilters (see Equation 3.1). Following this score, if the patch exceeded a threshold (established at 0.7), then the patch was considered for the dataset, if not, it was discarded.

$$\text{Score} = \text{TP} \times \text{CF} \times \text{SVF} \tag{3.1}$$

Where TP stands for Tissue Percentage; CF, Color Factor; and SVF, Saturation and Value Factor.

TP measures the amount of tissue that the patch contains, scoring it from 0 to 1, by counting the number of pixels that do not correspond to background. The more tissue the patch contains, the higher the score it will be given.

CF measures (from 0 to 1) the area of the patch that is inside H&E's color range (which is between pink and blue, including purple, depending on whether the region is acidic or basic). For this, each of the patches were first converted from RGB to HSV scale, which consists of three channels: hue (H), saturation (S) and brightness/value (V). Then, the score assigned to CF depends on the percentage of pixels whose hue lie within H&E's color range.

SVF measures the dispersion (standard deviation) of the saturation and brightness channels of the patch after being converted to HSV scale. As patches which contain tissue have a medium-high dispersion due to their low uniformity, those that do not have tissue or that have a small amount of tissue score lower SVF.

The number of patches obtained from normal and malignant WSIs after applying the mentioned steps is shown in Table 3.3, where the GGS distribution is also reported. Around 50% of the total amount of patches correspond to normal, and the rest to malignant.

3.2.1.3 Preprocessing step

Histology images could present unwanted color variations caused by different factors such as the staining procedure that was performed, the equipment that was used for doing it and the color responses of digital scanners in the digitization process, among others. When comparing WSIs, their color could be very different even if the images are obtained from the same scanner. Therefore, color normalization methods, which reduce the variability of H&E stain appearance, could be useful to improve the classifier. This could also make the system more robust and stable when predicting or inferring over new unseen

[2]https://github.com/deroneriksson/python-wsi-preprocessing (accessed on June 30, 2021)

TABLE 3.3: Dataset classes distribution.

Categories	No. of WSI	No. of patches
Malignant	70	19905, where: 6404 (32.17%) GGS 3 9791 (49.19%) GGS 4 3710 (18.64%) GGS 5
Normal	27	19772
Total	**97**	**39677**

samples from different hospitals and scanners with which the network has not been trained with.

To this end, a color normalization processing, called Reinhard stain-normalization (Reinhard et al., 2001; Magee et al., 2009), was applied. With this color normalization method, the mean and standard deviation of each channel of a source image are matched to that of a target image by applying a linear transformation in a perceptual colourspace (the $l\alpha\beta$ colourspace of Ruderman et al., 1998), obtaining the resulting mapped image. This process is defined by Equations 3.2, 3.3 and 3.4.

$$l_{\text{mapped}} = \frac{l_{\text{source}} - \bar{l}_{\text{source}}}{\hat{l}_{\text{source}}} \hat{l}_{\text{target}} + \bar{l}_{\text{target}} \tag{3.2}$$

$$\alpha_{\text{mapped}} = \frac{\alpha_{\text{source}} - \bar{\alpha}_{\text{source}}}{\hat{\alpha}_{\text{source}}} \hat{\alpha}_{\text{target}} + \bar{\alpha}_{\text{target}} \tag{3.3}$$

$$\beta_{\text{mapped}} = \frac{\beta_{\text{source}} - \bar{\beta}_{\text{source}}}{\hat{\beta}_{\text{source}}} \hat{\beta}_{\text{target}} + \bar{\beta}_{\text{target}} \tag{3.4}$$

Where $\bar{l}$, $\bar{\alpha}$, and $\bar{\beta}$ are the channel means; $\hat{l}$, $\hat{\alpha}$, and $\hat{\beta}$ are the channel standard deviations (calculated over all the pixels in the image). This process was applied to every patch (source) in the dataset, considering target as the mean over all the patches in the training set (dashed purple in Figure 3.5). An example of the application of this process can be seen in Figure 3.3.

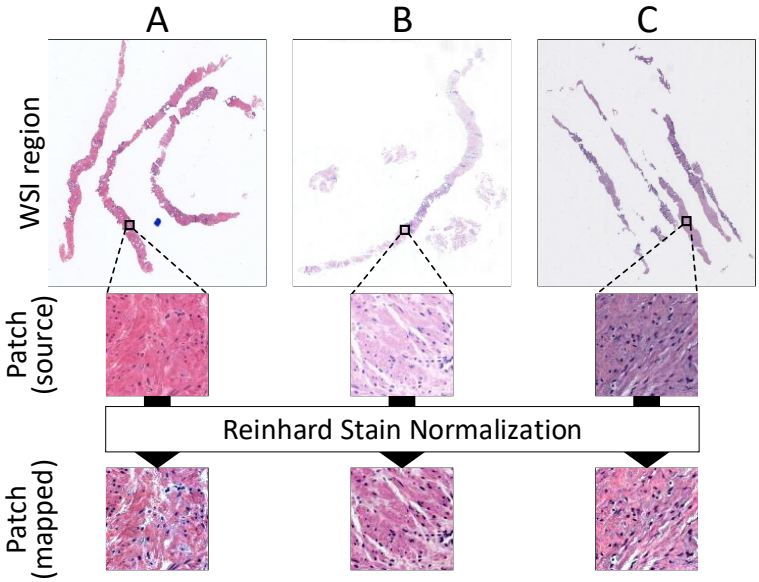

FIGURE 3.3: Examples of the application of Reinhard stain-normalization on three patches (source) from three WSIs (A, B and C) from different scanners, obtaining normalized patches (mapped).

3.2.1.4 Data augmentation

In DL algorithms, the more images the dataset has, the more robust and stable the system will be. Also, having a larger dataset helps to avoid overfitting, since the network has more different data to train with. However, this is not always the case (e.g., adding more noisy samples will not help), and this is why having a clean dataset with region-specific labels is so important.

For this reason, data augmentation techniques were applied to our dataset in order to increase the number of images for the training step, and thus, to contemplate many other cases. Different transformations were performed to the original patches, thus, for each training patch, a horizontal flip and a vertical flip were applied, along with rotations in the whole 360° range with steps of 1°, where the missing information in the corners after rotating the patch was filled by mirroring. Therefore, $2 \times 2 \times 360$ new patches were obtained from each original patch.

3.2.2 Deep learning framework

3.2.2.1 Convolutional Neural Network architecture

A custom CNN, called PROMETEO, was developed to perform the PCa detection task. It is a supervised neural network whose architecture is shown in Figure 3.4. This network consists of five convolution stages. A convolution stage consists of the following layers: convolution, batch normalization, rectified linear unit and 2×2 pooling. These 5 layers have 64 5×5, 64 3×3, 128 3×3, 128 3×3 and 256 3×3 filters, respectively, connected to three consecutive FC layers with 256, 128 and 128 units, respectively. Finally, a Softmax decision layer with two units gets the output from the last FC layer and generates the result of the classification, identifying between normal and malignant patches. Different architectures were tested, including the VGG16 (Simonyan and Zisserman, 2014), VGG19 (Simonyan and Zisserman, 2014), MobileNet (Howard et al., 2017) and DenseNet121 (Huang et al., 2017) architectures (see Section 1.4.2.4), although this custom CNN was selected based on the fact that achieved the best results.

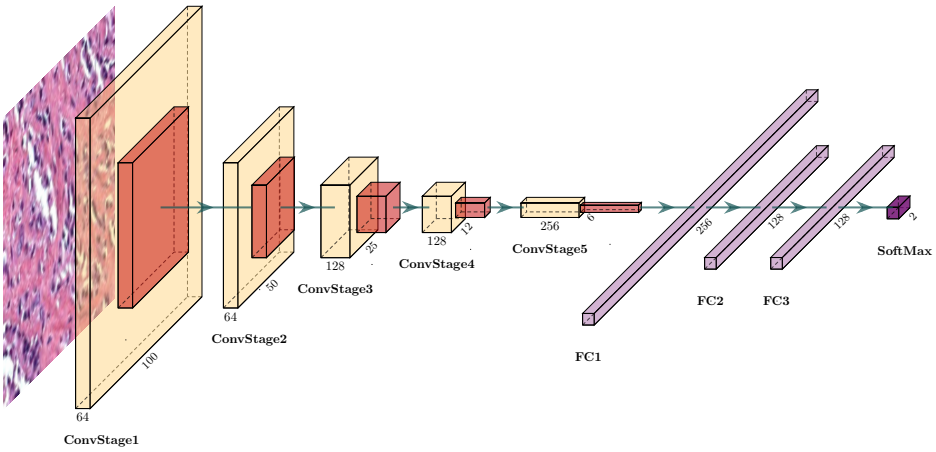

FIGURE 3.4: Diagram of the architecture of the CNN. Each convolution stage (ConvStageX) consists of convolution, batch normalization, ReLU and 2×2 max pooling layers. Each fully connected stage (FCX) consists of dense, batch normalization, ReLU and dropout (0.5) layers. Convolution kernels are: 5×5, 3×3, 3×3, 3×3, 3×3, respectively.

3.2.2.2 Training, validating and testing the system

As mentioned in previous sections, CNNs and other DL algorithms need a large amount of samples for the training phase. When using these architectures, the dataset is commonly divided into three different sets for training, validating and

testing the model, respectively, where the training set is by far the one with more samples.

At the same time, to measure the generalization ability of the model, cross-validation is usually performed. There are different types of cross-validation; the one used in this study was the K-fold stratified cross-validation (where $K = 3$). First, the dataset was split in two sets: 75% was used to perform the 3-fold cross-validation and the remaining 25% for performing a final test of the system.

For performing cross-validation, the 3-fold cross-validation set was divided again into three different subsets, where patches in each of these subsets were also divided following a patient-level split. Each subset consisted of, approximately, 50% cancer and 50% normal cases. Then, the network was trained for 200 epochs with a batch size of 32 using Adadelta optimizer (Zeiler, 2012) and validated three times (once per fold), using two of the subsets for training and the remaining one for validating the system. The results for cancer detection were evaluated as an average of the 3-fold cross-validation results.

After obtaining these results, a final test was performed, using the whole 3-fold cross-validation set (75% of the dataset) for training and then testing with the 25% set that was left apart. Figure 3.5 shows a diagram about the dataset division for the 3-fold-cross-validation and the final test.

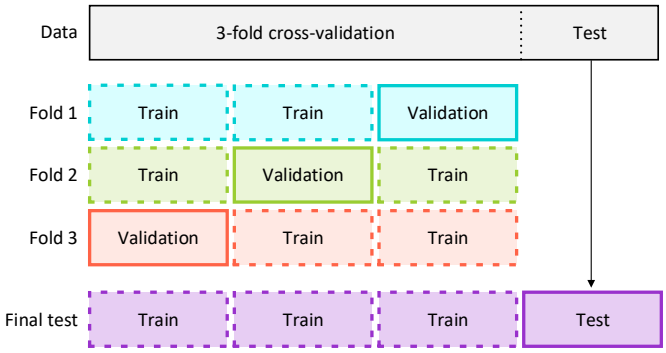

FIGURE 3.5: 3-fold cross-validation and final test diagram. The dataset was divided into four subsets. Two of them were used for training each fold and one for validation. After that evaluation, those three subsets were used to train a final model and the remaining one was used to test the performance of the system.

3.2.2.3 Evaluation metrics

In order to present the capabilities of this implemented CAD system, different evaluation metrics were used. These are accuracy (Equation 3.5), precision (Equation 3.6), sensitivity (Equation 3.7), specificity (Equation 3.8), F1-score

(Equation 3.9), and Area Under Curve (AUC) of the Receiver Operating Characteristic (ROC) curve. All of them were measured at patch level.

$$\text{Accuracy} = 100 \times \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3.5}$$

$$\text{Precision} = 100 \times \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.6}$$

$$\text{Sensitivity} = 100 \times \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.7}$$

$$\text{Specificity} = 100 \times \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{3.8}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \tag{3.9}$$

Where TP and FP denote true positive cases (when the system diagnoses a malignant patch correctly) and false positive cases (the system detects a malignant patch in a region where the tissue does not correspond to a tumor), respectively. TN and FN denote true negative cases (the system classifies a normal patch as normal) and false negative cases (the system classifies a malignant patch as normal), respectively.

The ROC curve shows the diagnostic ability of a binary classifier system as its discrimination threshold is varied. The AUC is a commonly used metric that measures the area that is under the ROC curve, where an area of 1 represents a perfect test.

3.3 Results

3.3.1 Quantitative evaluation

The evolution of the loss and accuracy over 200 epochs for each fold, both for the stain-normalized dataset and for the original one that was not normalized, is shown in Figure 3.6 and Figure 3.7, respectively.

The ROC curve was calculated for the same cases that were taken into account in the loss and accuracy plots (Figures 3.6 and 3.7), along with their corresponding AUC value, which are shown in Figure 3.8 and Figure 3.9.

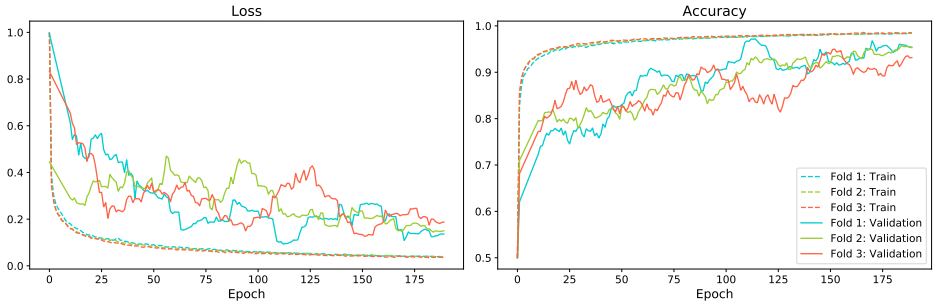

FIGURE 3.6: Loss and accuracy evolution when training with the three cross-validation sets using the stain-normalized dataset.

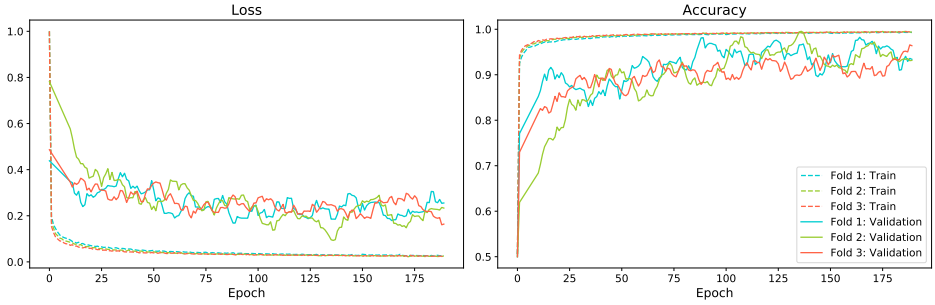

FIGURE 3.7: Loss and accuracy evolution when training with the three cross-validation sets using the dataset that was not stain-normalized.

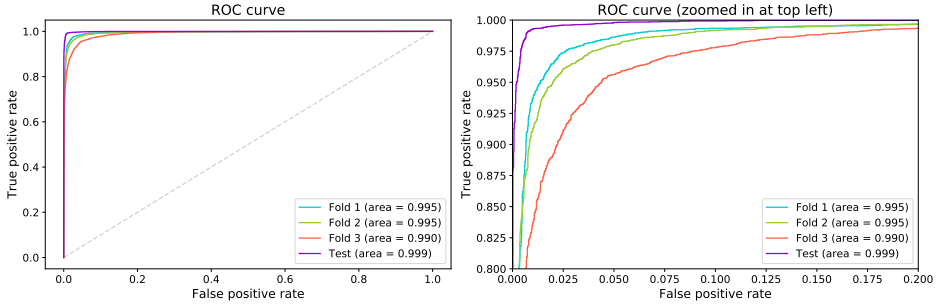

FIGURE 3.8: Left: ROC curve for each cross-validation set and the test set when using the stain-normalized dataset. Right: zoomed in at top left.

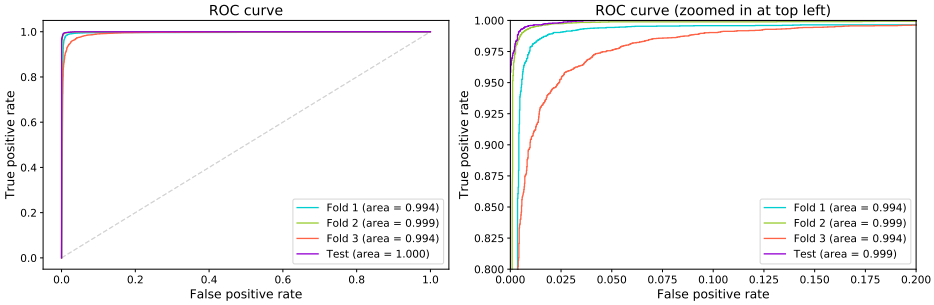

FIGURE 3.9: Left: ROC curve for each cross-validation set and the test set when using the dataset that was not stain-normalized. Right: zoomed in at top left.

TABLE 3.4: Results obtained from each cross-validation fold and the final test.

	Set	Dataset	Accuracy (%)	Specificity (%)	Sensitivity (%)	Precision (%)	F1 score (%)	AUC
Cross-validation	1st fold	Stain-normalized	97.43	97.42	97.44	97.47	97.45	0.995
		Not normalized	98.54	98.99	98.11	99.00	98.55	0.994
	2nd fold	Stain-normalized	96.7	97.85	95.63	97.85	96.73	0.995
		Not normalized	99.24	100.00	98.50	100.00	99.24	0.999
	3rd fold	Stain-normalized	95.35	96.25	94.49	96.28	95.37	0.990
		Not normalized	96.43	95.10	97.72	95.34	96.52	0.994
	Average	**Stain-normalized**	**96.49**	**97.17**	**95.83**	**97.2**	**96.51**	**0.993**
		Not normalized	**98.07**	**98.03**	**98.11**	**98.11**	**98.10**	**0.996**
Final test	**Test**	**Stain-normalized**	**99.14**	**99.18**	**99.10**	**99.27**	**99.19**	**0.999**
		Not normalized	**99.98**	**100**	**99.97**	**100**	**99.98**	**0.999**

Table 3.4 presents the results obtained from each of the cross-validation sets that were trained and validated, considering the stain-normalized dataset and the one that was not normalized. These results consists of the evaluation metrics that were introduced in Section 3.2.2.3, comparing both approaches by calculating the average over the validation sets.

After the cross-validation was performed, and as was explained in Section 3.2.2.2, the three subsets were used for training and the remaining 25%

of the dataset was used to test the network (see Figure 3.5). With this, the stain-normalized approach achieved 99.14% accuracy, while the not-preprocessed achieved 99.98% (see Table 3.4). Figures 3.8 and 3.9 present the ROC curves for these two tests.

As can be seen from these results, both approaches achieved very high scores in all the metrics that were studied for this classification task, with the dataset that was not normalized performing slightly better (less than 1.5% increase in accuracy). However, as was mentioned in previous sections, these results were obtained with WSIs from the same hospital (Virgen de Valme). Therefore, to measure the performance of both approaches with WSIs obtained from different hospitals and scanners, a new test was carried out, which is presented in Section 3.3.4.

3.3.2 Comparison with other methods

The results obtained in the previous section were compared with different state-of-the-art architectures and classifiers using the same dataset. The following well-known CNN models were used to extract features from the dataset: MobileNet, DenseNet121, VGG16 and VGG19. Instead of training these networks from scratch, whose architectures are more complex than the one that was developed for this study, their weights were obtained by using the transfer learning technique from the ImageNet dataset (Deng et al., 2009). This consists in taking a pre-trained neural network and adapting it to a new different dataset. Along with these four models, two different classifiers were tested: Support Vector Machine (SVM) and SoftMax. Moreover, each of the architectures was also fine-tuned, meaning that the weights from ImageNet were adjusted using backpropagation to increase the recognition rate over our dataset. The accuracy results for each of the possible combinations are presented in Table 3.5.

As it is shown in Table 3.5, the accuracy obtained from the different tested methods are very similar compared to PROMETEO. However, as it is presented in Chapter 4, their more complex architectures lead to a higher execution time, which is an important factor to reduce and optimize when developing a CAD system.

3.3.3 Expert pathologists' verification

In addition to the numerical results that were obtained in the previous section, a validation was also performed by expert pathologists. To this end, the network trained for the final test was used. With that model, a prediction was performed over the WSIs from the test subset. To perform a prediction, all patches from WSIs were read and only those which passed the patch filters mentioned in Section 3.2.1.2 were stain normalized and predicted by the CNN. These predictions were represented in a heatmap graph over the original WSI image.

TABLE 3.5: Results comparison for different state-of-the-art methods. Best accuracies for each architecture model are highlighted in bold.

Model	Classifier	Fine-tuning	Accuracy (%)
VGG16	SoftMax	No	86.36
		Yes	**94.76**
	SVM	No	85.37
		Yes	93.85
VGG19	SoftMax	No	83.87
		Yes	**93.54**
	SVM	No	85.11
		Yes	91.22
MobileNet	SoftMax	No	81.48
		Yes	98.96
	SVM	No	80.58
		Yes	**99.08**
DenseNet121	SoftMax	No	78.47
		Yes	96.82
	SVM	No	78.00
		Yes	**97.77**

An example can be seen in Figure 3.10, where the ground truth annotations from the pathologist are also shown. These heatmaps were given to different pathologists together with their corresponding WSIs in order to validate the predictions obtained from the network. The results of the CNN presented by the heatmap mark the same regions that pathologists labeled in the original WSI, with the exception of some isolated false positives, which are indicated.

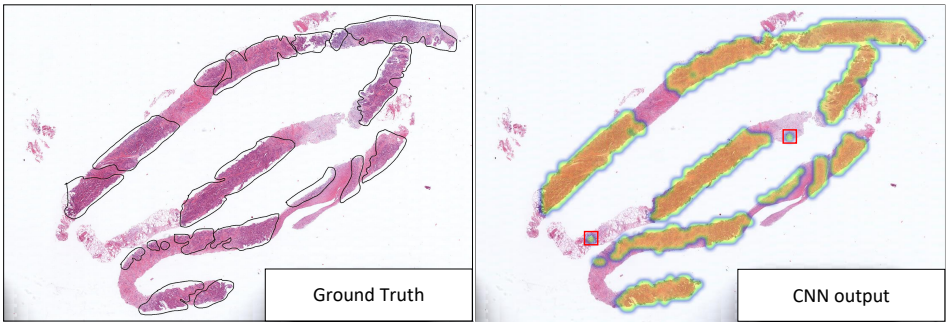

FIGURE 3.10: Left: WSI taken from the test subset with ground truth labels from pathologists. Right: output of the CNN represented with a heatmap. Isolated false positives marked with red squares.

3.3.4 Testing with WSIs from different hospitals

As was mentioned in previous sections, for both training and testing the network, only images from a single hospital were taken into account, which also means images from one laboratory and a specific scanner. A new experiment was carried out in order to measure the performance of the network when using new images obtained from other hospitals. This also allowed determining whether the stain-normalization step was better or not compared to the same images without applying any kind of color normalization.

To perform this experiment, new WSIs were obtained from two different hospitals: Puerta del Mar Hospital (Cádiz, Spain) and Clínic Barcelona Hospital (Barcelona, Spain). It is important to mention that this new images were not labeled the same way as the ones that were used to perform the previous experiments. These WSIs were only diagnosed as normal or malignant, without indicating which specific areas of the tissue were relevant for the pathologists to make that decision. Therefore, this new experiment consisted in measuring the number of false positives against true negatives (specificity) detected by the network in total for all WSIs diagnosed as normal for each hospital. WSIs diagnosed as malignant were not taken into consideration for a sensitivity study due to the fact that there was no ground truth that could be used to evaluate the network when testing it with the patches obtained from them. Instead, a statistical study based on Student's t-test is later presented to compare the predicted patches' distribution between normal and malignant WSIs.

From Clínic Barcelona Hospital, 100 new WSIs diagnosed as normal were used, whereas a total of 79 were considered from Puerta del Mar Hospital: 33 of them were obtained from needle core biopsy (the same procedure as Virgen de Valme Hospital and Clínic Hospital) and the remaining 46 WSIs were obtained from incisional biopsy.

Figure 3.11 shows the mean specificity and standard deviation for each of the three sets from different hospitals, comparing the stain-normalization algorithm (96.08 ± 2.85, 94.82 ± 3.52 and 96.26 ± 2.20, respectively) to the original images (93.31 ± 6.43, 95.87 ± 8.57 and 95.94 ± 3.42, respectively).

Since malignant WSIs only provided a global diagnosis, the sensitivity at patch level could not be calculated. Then, an evaluation relying on the slide-level label was performed, comparing the probability distributions estimated by the CNN for normal and malignant WSIs for each external hospital. To carry out this evaluation, 129 new WSIs diagnosed as malignant from Clínic Barcelona Hospital and 65 new malignant WSIs from Puerta del Mar Hospital (26 obtained from needle core biopsy and 39 from incisional biopsy) were considered, along with the ones diagnosed as normal that were used in the previous experiment. Patches from both normal and malignant WSIs were predicted following the same procedure explained in Section 3.3.3 with the model that was trained with stain-normalized patches and also with the one that was not (in this case, patches

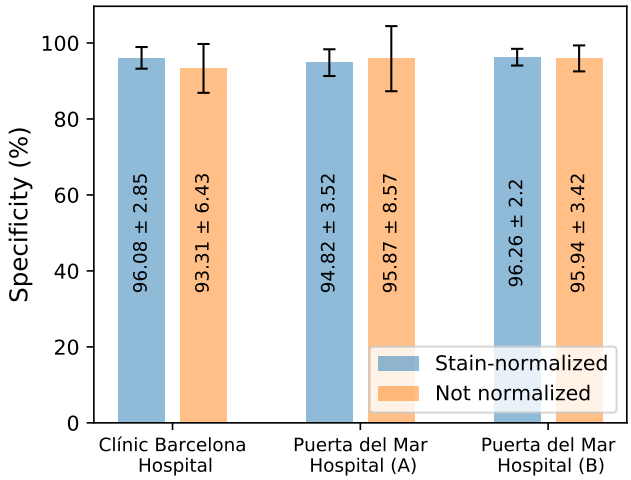

FIGURE 3.11: Mean specificity and standard deviation achieved by the CNN with WSIs obtained from Clínic Barcelona Hospital (Barcelona. Spain), and Puerta del Mar Hospital (Cádiz, Spain). A and B were extracted with incisional biopsy and needle core biopsy, respectively.

extracted from the WSIs from external hospitals were not stain-normalized in the preprocessing step). The average and standard deviation of the percentage of malignant patches in relation to the total amount of tissue patches (those that passed the patch filters mentioned in Section 3.2.1.2) that the models predicted, were calculated for each hospital. Statistical Student's t-test was performed to measure how significant the difference between the results obtained for normal and malignant WSIs were. For the t-test, two values were generated: the t-statistic and the critical t-value. If the first is greater than the second, the test concludes that there is a statistically significant difference between the results obtained for normal and malignant WSIs. The results of this evaluation are presented in Table 3.6, where the impact of using stain-normalization is also shown.

As can be seen from the results obtained when performing the predictions, there is a statistically significant difference between the results obtained for normal and malignant WSIs when using stain-normalization as part of the preprocessing step. On the other hand, significant differences cannot be achieved when predicting without having applied the normalization process to the input patches before, except for the WSIs obtained from Clínic Barcelona Hospital.

Figure 3.12 presents three extreme cases from Puerta del Mar Hospital obtained with needle core biopsy. The first case (A) shows a malignant WSI in which the system detected a high quantity of malignant patches (∼45% of the tissue). On the other hand, the second one (B), corresponds to a malignant

TABLE 3.6: Results of the statistical evaluation performed with malignant and normal WSIs from external hospitals, where Avg%ppm stands for the average of the percentage of patches predicted as malignant, and Std for its standard deviation. Cases where t-static > critical t-value (statistically significant difference found between normal and malignant distributions) are highlighted in bold. A and B were extracted with incisional biopsy and needle core biopsy, respectively.

			Malignant	**Normal**
Clínic Barcelona	Stain-normalized	Avg%ppm	12.22%	3.92%
		Std	9.29%	2.85%
		t-statistic	**8.24**	
		(critical t-value)	**(1.98)**	
	Not normalized	Avg%ppm	15.46%	6.69%
		Std	10.17%	6.43%
		t-statistic	**7.29**	
		(critical t-value)	**(1.98)**	
Puerta del Mar (A)	Stain-normalized	Avg%ppm	11.47%	5.18%
		Std	9.43%	3.52%
		t-statistic	**3.93**	
		(critical t-value)	**(2.01)**	
	Not normalized	Avg%ppm	7.35%	4.13%
		Std	9.97%	8.57%
		t-statistic	1.58	
		(critical t-value)	(1.99)	
Puerta del Mar (B)	Stain-normalized	Avg%ppm	14.00%	3.74%
		Std	11.87%	2.20%
		t-statistic	**4.35**	
		(critical t-value)	**(2.05)**	
	Not normalized	Avg%ppm	3.35%	4.06%
		Std	4.08%	3.42%
		t-statistic	-0.71	
		(critical t-value)	(2.01)	

WSI with around a 6% of the tissue predicted as malignant. Finally, in the third case (C), a normal WSI is shown, in which the system mistakenly detected 5% of the patches that correspond to tissue as malignant. These malignant WSIs present pen marks drawn by the pathologist that globally diagnosed the slide before being scanned, which roughly delimit malignant areas of the tissue. As can be seen in Figure 3.12, C has a relatively high quantity of patches detected as malignant. However, these patches are scattered across the tissue and, hence, not focusing on a specific region, which clearly represents the error of the system. On the other hand, in B, the small quantity of patches detected as malignant are mostly focused inside the area delimited by the pen marks. After being revised

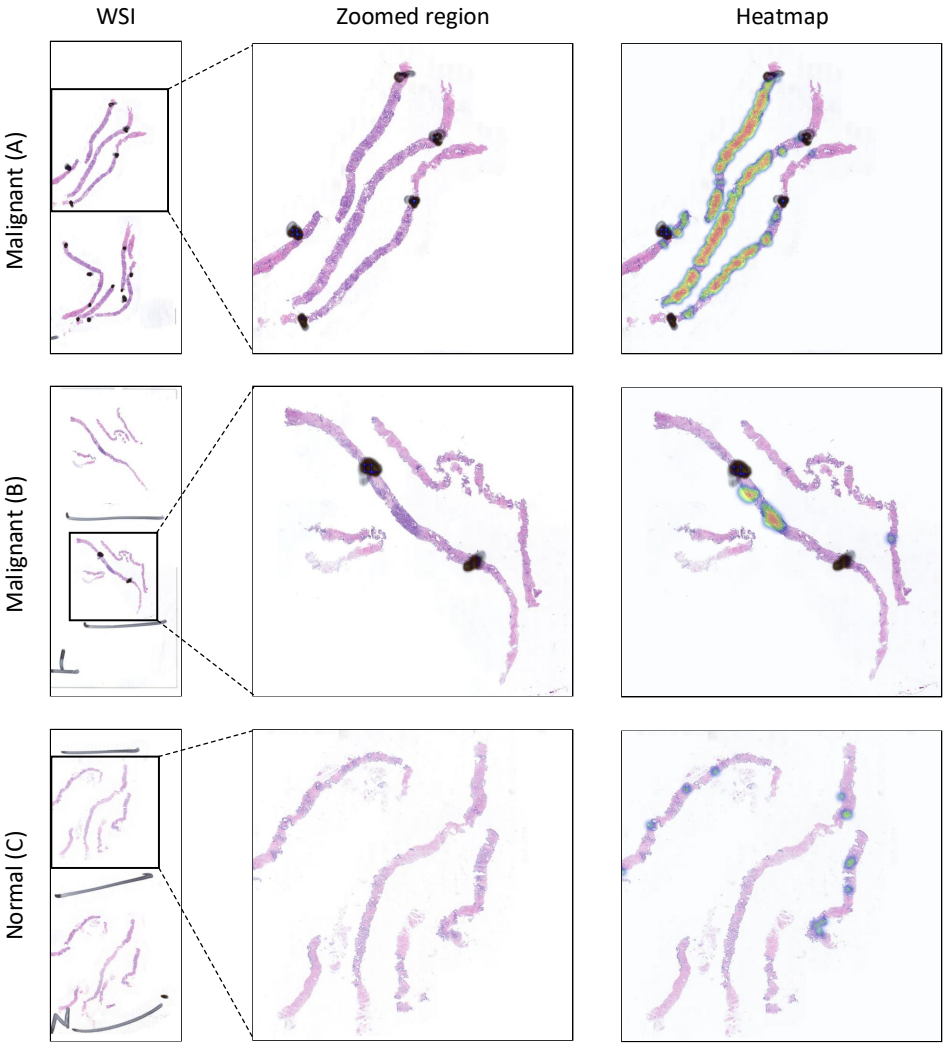

FIGURE 3.12: Heatmaps generated by the system for three different WSIs from Puerta del Mar Hospital. A and B correspond to WSIs globally diagnosed as malignant with high and low quantity of malignant patches detected by the system, respectively, while C represents a normal WSI with a high error rate in the prediction. Zoomed regions are presented for better visualization.

by a pathologist, it was confirmed that the malignant area matches the heatmap, while the rest corresponds to normal tissue, except for a small area that is partially overlapped by the bottom pen mark. Finally, the heatmap presented for A shows that the system detects most of the malignant tissue correctly based on the pen

marks.

The results obtained in this chapter have been published in IEEE Access journal as "PROMETEO: A CNN-based computer-aided diagnosis system for WSI prostate cancer detection" (Duran-Lopez et al., 2020a). More details of this publication can be found in Appendix A.

Chapter 4

Performance evaluation of DL-based prostate cancer screening methods

4.1 Introduction

In the previous chapter, a CAD system for PCa detection in WSIs, called PROMETEO, was presented. As mentioned in Section 3.1, this task has become one of the main topics for many researchers. Ström et al. (Ström et al., 2020) developed a DL-based CAD system to perform a binary classification, distinguishing between malignant and normal tissue. The classification was performed using an ensemble of 30 widely-used InceptionV3 models (Szegedy et al., 2016) pretrained on ImageNet. They achieved an AUC of 0.997 and 0.986 on the validation and test subsets, respectively. For areas detected as malignant, the authors trained another ensemble of 30 InceptionV3 CNNs in order to discriminate between different PCa patterns from GGS, achieving a mean pairwise kappa of 0.62 at slide level. Campanella et al. (Campanella et al., 2019) presented a CAD system to detect malignant areas in WSIs. The classification was performed with the well-known ResNet34 model (He et al., 2016) together with a Recurrent Neural Network (RNN) for tumor/normal classification, achieving an AUC of 0.986 at slide level.

These previous works achieve competitive results in terms of accuracy, precision and other commonly-used evaluation metrics. However, to the best of the author's knowledge, most state-of-the-art works do not focus on prioritizing the speed of the CAD system as an important factor. Many of them used very complex well-known networks to train and test, without taking into account the computational cost and the time required to perform the whole process. Since these algorithms are not intended to replace pathologists but to assist them in their task, in some cases it is better to prioritize the speed of the analysis, sacrificing some precision so that the expert has a faster and more dynamic response from the system.

In this study, a novel benchmark was designed in order to measure the processing and prediction time of a CNN architecture for a PCa screening

task. First, the proposed benchmark was run for the PROMETEO architecture on different computing platforms in order to measure the impact that their hardware components have on the WSI processing time. Then, using the Personal Computer (PC) configuration that achieved the best performance, the benchmark was run with different state-of-the-art CNN models, comparing them in terms of average prediction time both at patch and at slide level, and also reporting the slowdown when compared to PROMETEO.

4.2 Materials and Methods

4.2.1 Dataset

In this experiment, a dataset with the same WSIs obtained from Chapter 3 was used. These cases consisted in different H&E-stained slides globally diagnosed as either normal or malignant from Virgen de Valme, Clínic Barcelona and Puerta del mar hospitals. Table 4.1 summarizes the WSIs considered in the dataset differentiating between normal and malignant cases.

TABLE 4.1: Dataset summary.

Hospital	No. of WSIs		
	Normal	Malignant	Total
Virgen de Valme Hospital	27	70	97
Clínic Hospital	100	129	229
Puerta del Mar Hospital	79	65	144

4.2.2 Convolutional Neural Network models

Different CNN models were considered in order to compare their performance by using the benchmark proposed in section 4.2.3. Three different architectures from state-of-the-art DL-based PCa detection works were compared. The first one is PROMETEO, presented in Chapter 3, where the authors also demonstrated that applying stain-normalization algorithms to the patches in order to reduce color variability could improve the generalization of the model when predicting new unseen images from different hospitals and scanners. The second CNN architecture that was considered is the well-known ResNet34 model (He et al., 2016), which was used in Campanella et al., 2019. The third one is InceptionV3, introduced in Szegedy et al., 2016, which was used by Ström et al., 2020.

Apart from these three CNN models, other widely-known architectures were evaluated with the same benchmark, comparing their performance in terms of execution time with the rest of the networks for the same task. These were VGG16 and VGG19, MobileNet, DenseNet121, Xception and ResNet101 (see Section 1.4.2.4).

4.2.3 Benchmark

A novel benchmark was designed in order to measure and compare the performance of different CNN models and platforms on a PCa screening task. In order to make the benchmark feasible to be shared with other researchers so that it could be run in different computers, a reduced set of WSIs were chosen from the dataset presented in section 4.2.1. Since the total amount of WSIs of the dataset represent more than 300 Gigabytes (GBs) of hard drive space, only 40 of them were considered, building up a benchmark of around 50 GBs, which is much more shareable. These 40 WSIs were randomly selected, considering all the three different hospitals and scanners, and thus representing well the diversity of the dataset in this benchmark.

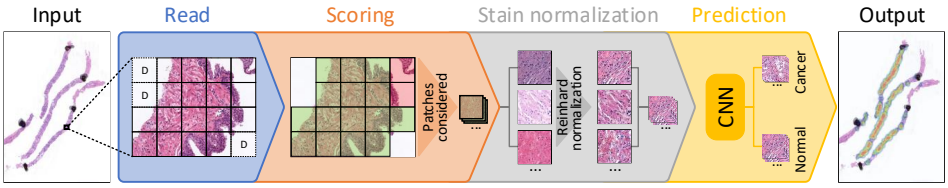

FIGURE 4.1: Block diagram detailing each of the steps considered for processing a WSI in the proposed benchmark (read, scoring, stain normalization and prediction). In *Scoring*, discarded patches are highlighted in red, while those that pass the filter are highlighted in green.

The benchmark performs the set of processing steps presented in Section 3.2.1. Figure 4.1 summarizes the whole process. As it was explained, first, WSIs are divided into patches (100×100 pixels at 10× magnification in this case). This process is called *Read* and, apart from extracting the patches from the input WSI, those corresponding to background are discarded (identified as D in the figure) (see Section 3.2.1.2 for more details). Then, in the *Scoring* step, a score is given to each patch depending on three factors: the amount of tissue that it contains, the percentage of pixels that are within H&E's hue range, and the dispersion of the saturation and brightness channels. This score allows discarding patches corresponding to unwanted areas, such as pen marks, external agents and patches with a small amount of tissue, among others (see Section 3.2.1.2). The third step, *Stain normalization*, performs a color normalization of the patch based on Reinhard's stain-normalization algorithm in order to reduce color variability between samples (see Section 3.2.1.3). In *Prediction*, which is the last step of the process, each of the patches are used as input to a trained CNN, which classifies them as either malignant or normal tissue. When the execution of the benchmark finishes, it reports both the hardware and system information of the computer used to run the benchmark, and the results of the execution. These results consist of the mean execution time and standard deviation for each of the four processes (*Read*, *Scoring*, *Stain normalization* and *Prediction*), both at patch level and at WSI level.

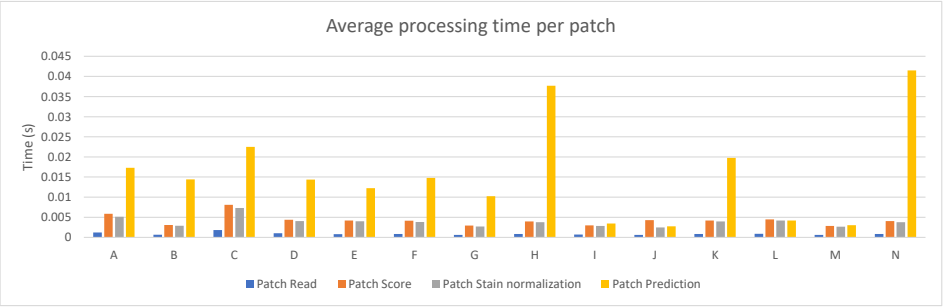

FIGURE 4.2: PROMETEO average patch processing time (in seconds) per step for each of the hardware configurations detailed in Table 4.2.

4.3 Results

The DL-based PROMETEO architecture, described in section 3.2.2.1, was proposed and evaluated in terms of accuracy and many other evaluation metrics in Chapter 3. In this experiment, the authors evaluated this model in terms of performance and execution time per patch and WSI.

First, the same architecture was tested in different platforms using the benchmark proposed in section 4.2.3. These results allowed us to measure and quantify the impact of different components in the whole processing and prediction process, which could be useful for designing an edge-computing PCa detection system. Then, the benchmark was used to evaluate the performance of different state-of-the-art CNN architectures on the computing platform that achieved the best results on the first experiment.

4.3.1 PROMETEO evaluation

Fourteen different PC configurations were used to evaluate the performance of the PROMETEO architecture. The hardware specifications (CPU and GPU) of these computers are listed in Table 4.2. In Figure 4.2, the average patch processing time for each of the fourteen configurations is shown, where the mean time for the steps performed when processing a patch (see Section 4.2.3) is reported. As it can be seen, the step that requires more time is the prediction in most of the cases, but it is highly reduced in configurations consisting of a GPU.

Figure 4.3 depicts the average and standard deviation of the execution time needed per WSI for each of the steps considered in the whole process when running the benchmark on the fourteen different PC configurations. As it can be seen, reading the whole WSI patch by patch is the step that involves the longest amount of time in most of the devices (mainly in those configurations with no GPU). This might seem contradictory considering Figure 4.2, but it is important to mention that, in that step, all patches from a WSI are read and analyzed, but

TABLE 4.2: Hardware specifications (CPU and GPU) of the different computers used in the PROMETEO evaluation.

Device	CPU	GPU
A	Intel® Core™ i7-8850U @ 1.80GHz 4 cores, 8 threads	-
B	Intel® Core™ i9-7900X @ 3.30GHz 10 cores, 20 threads	-
C	Intel® Core™ i7-6700HQ @ 1.20GHz 4 cores, 8 threads	-
D	Intel® Core™ i7-6700HQ @ 2.60GHz 4 cores, 8 threads	-
E	Intel® Core™ i5-6500 @ 3.20GHz 4 cores, 4 threads	-
F	Intel® Core™ i7-4770K @ 3.50GHz 4 cores, 8 threads	-
G	Intel® Core™ i7-8700K @ 3.70GHz 6 cores, 12 threads	-
H	Intel® Core™ i7-4970 @ 3.60GHz 4 cores, 8 threads	-
I	Intel® Core™ i9-7900X @ 3.30GHz 10 cores, 20 threads	NVIDIA® GeForce™ GTX 1080 Ti 11GB GDDR5X
J	AMD® Ryzen™ 9 3900X @ 4.20GHz 12 cores, 24 threads	NVIDIA® GeForce™ GTX 1080 Ti 11GB GDDR5X
K	Intel® Core™ i5-6500 @ 3.20GHz 4 cores, 4 threads	NVIDIA® GeForce™ GT 730 2GB GDDR5
L	Intel® Core™ i7-4770K @ 3.50GHz 4 cores, 8 threads	NVIDIA® GeForce™ GTX 1080 Ti 11GB GDDR5X
M	Intel® Core™ i7-8700K @ 3.70GHz 6 cores, 12 threads	NVIDIA® GeForce™ GTX 1080 Ti 11GB GDDR5X
N	Intel® Core™ i7-4970 @ 3.60GHz 4 cores, 8 threads	NVIDIA® GeForce™ RTX 2060 6GB GDDR6

not all of them are processed in the following steps. Unwanted areas, such as background regions with no tissue, are discarded before being scored. Then, only those which are not background and pass the scoring step are stain normalized and predicted by the CNN.

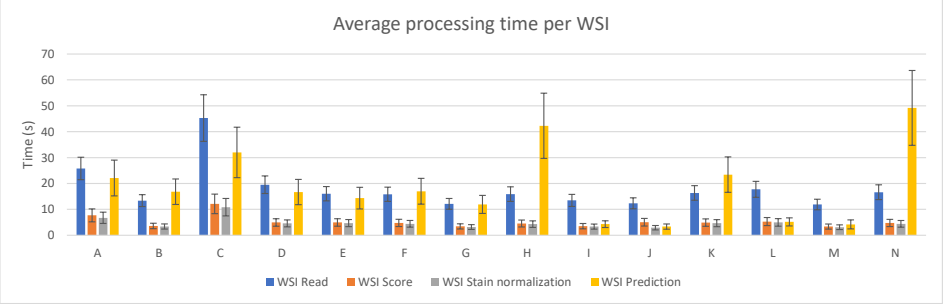

FIGURE 4.3: PROMETEO average WSI processing time (in seconds) and standard deviation per step for each of the hardware configurations detailed in Table 4.2.

Two specific cases of Figure 4.3 are highlighted in Figures 4.4 (C and D configurations) and 4.5 (G and M configurations). Figure 4.4 shows the impact that the frequency of the CPU has in the whole process when using the same computer. As it can be seen, the four processing steps clearly benefit when a faster CPU is used. On the other hand, Figure 4.5 compares two cases where the same configuration is used, except for the GPU, which was removed in one of them. As expected, the GPU highly accelerated the prediction time (by around 3 times in this case). Therefore, in order to build a low-cost edge-computing platform for PCa diagnosis, this analysis could be useful and should be taken into account in order to prioritize in which component the funds should be invested. As it was explained, all patches from a WSI have to be read, but not all of them have to be predicted, since the majority of them correspond to background and are discarded first. Therefore, the CPU has a higher impact than the GPU in the whole process.

The sum of the average execution time of the four preprocessing steps for each WSI was computed and it can be seen in Figure 4.6. The best case (device M) takes 22.56$\pm$5.67 s on average to perform the whole process per WSI, where the prediction step only represents 4.20$\pm$1.73 s.

The execution times obtained and used for generating the plots presented in this subsection are detailed in Table 4.3.

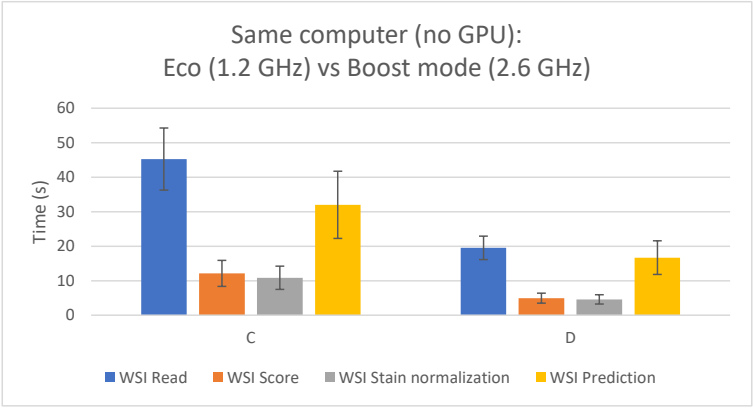

FIGURE 4.4: Impact of the CPU in the different WSI processing steps. Same PC, different CPU frequency. Left: 1.2 GHz; right: 2.6 GHz.

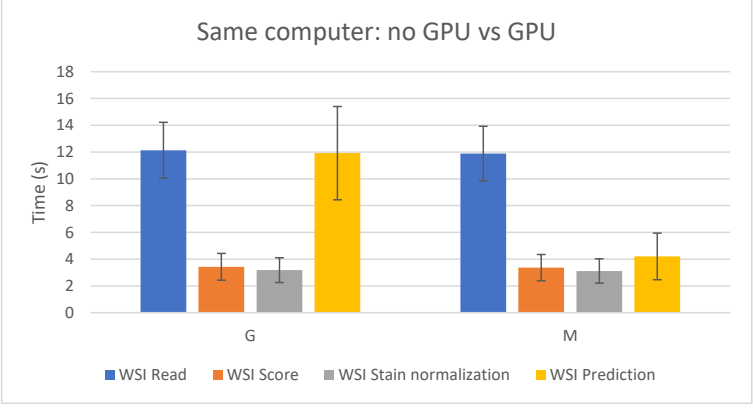

FIGURE 4.5: Impact of the GPU in the different WSI processing steps. Same PC. Left: without using GPU; right: using GPU.

TABLE 4.3: PROMETEO evaluation results. The average and standard deviation of the execution times (in seconds) are shown for each of the four processes presented in section 4.2.3 (Figure 4.1), both at patch level and at slide (WSI) level.

	Patch								WSI							
	Read		Score		Stain normalization		Prediction		Read		Score		Stain normalization		Prediction	
Device	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std
A	0.00120757	0.00311363	0.00585298	0.00502059	0.00512905	0.00418173	0.01730045	0.00850617	25.8035576	4.33829335	7.68691321	2.50054828	6.74114185	2.1823579	22.1192591	6.90573801
B	0.00068973	0.00150109	0.00306787	0,.0037015	0.00288733	0.00062438	0.01441587	0.00175146	13.3892811	2.29629633	3.59321811	1.04956324	3.38756321	0.98944353	16.8213335	4.91918301
C	0.00182337	0.00503892	0.00807697	0.00628436	0.00729532	0.00634226	0.02249318	0.0100416	45.2693313	8.99897452	12.1239614	3.77223369	10.8517522	3.36864663	31.9982855	9.74059672
D	0.00103901	0.00228084	0.00437608	0.00080403	0.00404258	0.00098854	0.01435914	0.00211298	19.5256697	3.39723828	4.94245166	1.44889791	4.59097219	1.34596044	16.6959336	4.87815493
E	0.00082695	0.00167953	0.00421429	0.00068222	0.00400594	0.00090984	0.01223286	0.00216942	16.0484505	2.76278471	4.94652829	1.4439207	4.70010431	1.37218657	14.3399619	4.1861481
F	0.00083281	0.00172759	0.00413688	0.00075105	0.0038354	0.00094666	0.01479934	0.00293383	15.8376234	2.74964356	4.7714866	1.3976735	4.42655776	1.29581397	16.9997911	4.98414625
G	0.00062777	0.00133961	0.00292148	0.00038463	0.00270451	0.00067159	0.0102172	0.00150291	12.1361434	2.0832663	3.42516114	1.00071198	3.17680098	0.92726679	11.9159923	3.48494303
H	0.00084291	0.00175864	0.00398322	0.00065638	0.0037566	0.00093803	0.03768491	0.00818776	15.8986914	2.81638821	4.5458395	1.34199731	4.29495389	1.26743003	42.2879135	12.595224
I	0.00069517	0.0015282	0.00299673	0.0003936	0.00285997	0.00066557	0.00345098	0.01023957	13.4663605	2.32710305	3.51854302	1.02719857	3.35297541	0.97965636	4.29145549	1.29811287
J	0.00062976	0.00137508	0.00428461	0.00027188	0.00246943	0.00060632	0.00275394	0.00906872	12.3392324	2.12166155	5.039479	1.47022453	2.91945068	0.85082711	3.35472003	1.00963885
K	0.00084153	0.00173514	0.00417708	0.00059019	0.00397734	0.00091936	0.01973523	0.01713999	16.462836	2.81694585	4.89897847	1.430121	4.67706547	1.36512005	23.4278429	6.84671918
L	0.00091354	0.00194007	0.00449805	0.00163909	0.00421455	0.00178876	0.0420623	0.01316229	17.743488	3.08314193	5.29275112	1.55927849	4.97026951	1.46232287	5.16441015	1.56004368
M	0.0006116	0.00129119	0.00286777	0.00039014	0.00265452	0.00068521	0.0030545	0.03559015	11.8833887	2.0393166	3.36354507	0.98195641	3.11514971	0.90982144	4.20128196	1.7395392
N	0.0008502	0.00498445	0.00405446	0.0007176	0.00375849	0.00097299	0.04150535	0.01602984	16.6332854	2.85486699	4.75020676	1.39576506	4.41236681	1.2946882	49.1934053	14.4530511

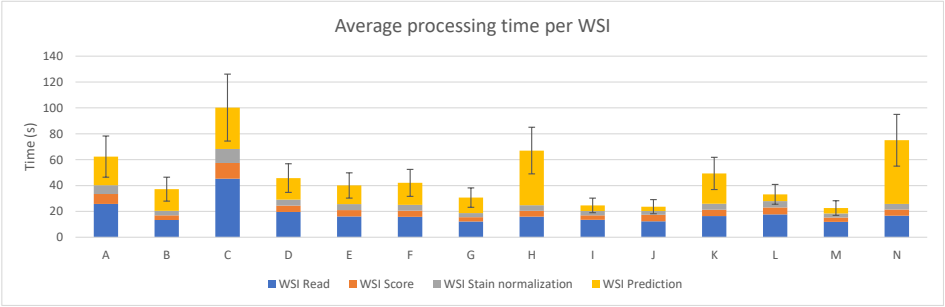

FIGURE 4.6: PROMETEO average WSI processing time (in seconds) and standard deviation of the hardware configurations detailed in Table 4.2.

4.3.2 Performance comparison for different state-of-the-art models

After evaluating the PROMETEO architecture using the benchmark designed for this experiment with different PCs, the same network was compared to other widely-known architectures. For this purpose, the same computer (device M) was used in order to perform a fair comparison. The same benchmark that was used in the previous evaluation (see section 4.3.1) was executed in computer M (see Table 4.2) for each of the CNN architectures mentioned in section 4.2.2. The CNNs considered are PROMETEO, ResNet34 and ResNet101, InceptionV3, VGG16 and VGG19, MobileNet, DenseNet121 and Xception.

The average patch processing time per preprocessing step can be seen in Figure 4.7 for each of the architectures mentioned. Since the architecture does not have an effect on the first three steps (reading the patch from the WSI, scoring it in order to discard unwanted patches, and normalizing it), the time needed to process them is similar across all the different cases reported in the figure. This does not happen with the prediction time, which directly depends on the complexity of the network.

Figure 4.8 reports the combined processing time that device M takes to compute a WSI on average, together with its corresponding standard deviation. The same case explained in section 4.3.1, where the WSI reading step takes much longer than the patch reading step in relation to the rest of the subprocesses, can also be observed in this figure. It is important to mention that the model proposed by the authors is faster than the rest in terms of prediction time, with a total of 22.56 ± 5.67 s per WSI on average.

Table 4.4 presents a summary of the results obtained for each architecture, focusing on the prediction process, which is the only one affected when changing the CNN architecture. Moreover, the number of trainable parameters and the slowdown are also reported. The latter is calculated by dividing the average

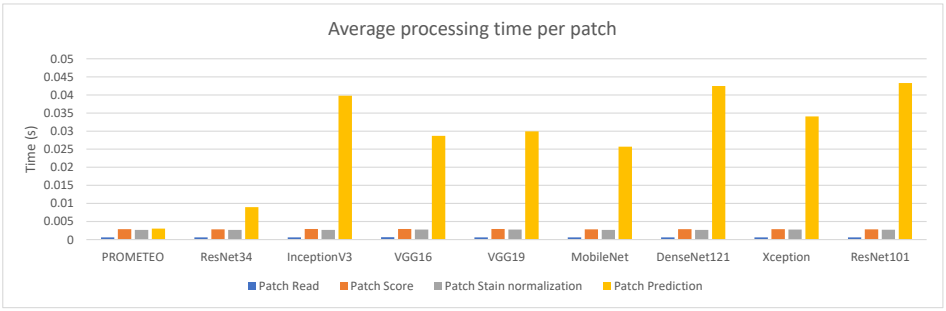

FIGURE 4.7: Average patch processing time (in seconds) per step for each of the CNN architectures using computer M (see Table 4.2).

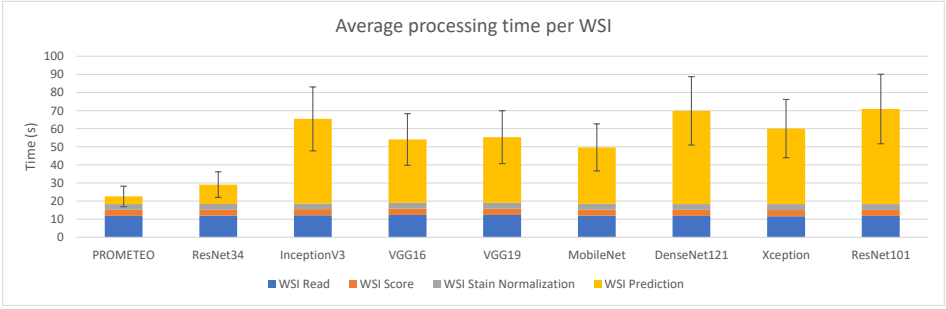

FIGURE 4.8: Average WSI processing time (in seconds) and standard deviation for each of the CNN architectures using computer M (see Table 4.2).

prediction time per WSI of the corresponding CNN by that obtained with PROMETEO. This way, the improvement in terms of prediction time between PROMETEO and the rest of the architectures considered can be clearly seen. The proposed model predicts 2.55× faster than the CNN used in Campanella et al., 2019 and 11.68× faster than the one used in Ström et al., 2020. It is also important to mention that, in the latter, the authors did not use only an InceptionV3 model, but an ensemble of 30 of them. In this case, the figures and tables only report the execution time for a single network. When compared to other different widely-known architectures, PROMETEO is between 7.41× and 12.50× faster.

The execution times obtained and used for generating the plots presented in this subsection are detailed in Table 4.5.

The results obtained in this chapter have been published in Sensors journal as "Performance evaluation of Deep Learning-based prostate cancer screening methods in histopathological images: measuring the impact of the model's complexity on its processing speed" (Duran-Lopez et al., 2021). More details of

TABLE 4.4: Average patch and WSI prediction time, slowdown and number of trainable parameters for each of the CNN architectures considered.

Model	Avg. prediction time (patch)	Avg. prediction time (WSI)	Slowdown*	Trainable parameters
PROMETEO	3.054 ± 4.845 ms	4.201 ± 1.739 s	1×	1,107,010
ResNet34	8.982 ± 10.086 ms	10.712 ± 3.134 s	2.55×	21,800,107
InceptionV3	41.301 ± 44.282 ms	49.076 ± 14.353 s	11.68×	23,851,784
VGG16	28.664 ± 9.241 ms	34.921 ± 10.160 s	8.31×	138,357,544
VGG19	29.931 ± 9.305 ms	36.250 ± 10.536 s	8.63×	143,667,240
MobileNet	25.689 ± 10.986 ms	31.110 ± 9.030 s	7.41×	4,253,864
DenseNet121	42.489 ± 16.859 ms	51.483 ± 14.945 s	12.25×	8,062,504
Xception	34.050 ± 11.789 ms	41.764 ± 12.175 s	9.94×	22,910,480
ResNet101	43.287 ± 14.679 ms	52.517 ± 15.266 s	12.50×	44,707,176

* Calculated by using the average prediction time per WSI and taking the PROMETEO architecture as reference. A slowdown of A× means that model B is A times slower than PROMETEO.

this publication can be found in Appendix B.

TABLE 4.5: Execution time comparison between different architectures. The average and standard deviation of the execution times (in seconds) are shown for each of the four processes presented in section 4.2.3 (Figure 4.1), both at patch level and at slide (WSI) level.

	Patch								WSI							
	Read		Score		Stain normalization		Prediction		Read		Score		Stain normalization		Prediction	
Architecture	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std
PROMETEO	0.000612	0.001291	0.002868	0.00039	0.002655	0.000685	0.003054	0.004845	11.88339	2.039317	3.363545	0.981956	3.11515	0.909821	4.201282	1.739539
ResNet34	0.000612	0.001279	0.002844	0.000405	0.002676	0.000686	0.008982	0.010086	11.92095	2.045551	3.333316	0.973793	3.148634	0.918751	10.71205	3.134596
InceptionV3	0.000621	0.001317	0.002915	0.00041	0.002691	0.001136	0.039772	0.013828	12.10135	2.076446	3.415152	0.997178	3.168544	0.925316	46.72138	13.64997
VGG16	0.000635	0.001341	0.002931	0.000427	0.002785	0.000691	0.028664	0.009241	12.37371	2.140448	3.45475	1.007557	3.280768	0.957531	34.92197	10.16074
VGG19	0.000628	0.001313	0.002931	0.000425	0.00278	0.000682	0.029931	0.009305	12.34846	2.116793	3.44631	1.005834	3.266729	0.953449	36.25006	10.5361
MobileNet	0.000612	0.001278	0.00285	0.00042	0.002688	0.001111	0.025689	0.010986	11.96025	2.044115	3.383497	0.986745	3.208441	0.936362	31.11017	9.030854
DenseNet121	0.000611	0.001284	0.002879	0.000389	0.002687	0.000683	0.042489	0.01686	11.82392	2.035413	3.373127	0.985149	3.148681	0.919557	51.48291	14.94588
Xception	0.0006	0.001261	0.00288	0.000374	0.002758	0.000655	0.03405	0.011789	11.68313	1.987459	3.375536	0.986863	3.235289	0.945187	41.76486	12.17527
ResNet101	0.000607	0.001265	0.002839	0.000398	0.002701	0.00067	0.043287	0.014679	11.84637	2.035472	3.327598	0.971357	3.171104	0.925785	52.51713	15.26661

Chapter 5

Wide & Deep neural network for patch aggregation in DL-based prostate cancer detection systems

5.1 Introduction

As introduced in previous chapters, WSIs are gigapixel-resolution digital slides which pathologists examine to find abnormalities and make a diagnosis. Since it is not possible for a CNN to work with a WSI as input due to its large size, the most common approach is to divide this image into small subimages called patches, which are later used to train and evaluate the ML model (see Section 3.2.1.2 for more details). Previously mentioned works, such as Ström et al., 2020; Campanella et al., 2019; Litjens et al., 2016; Li et al., 2018 and Bulten et al., 2020, have followed this patch-level classification strategy in order to develop DL-based CAD systems for PCa detection in digitized histopathological images, reporting accurate results with different metrics and datasets. Among them, to the best of the author's knowledge, PROMETEO achieved the fastest and least complex model (see Chapter 4) while also obtaining state-of-the-art results, leading to the most-plausible edge-computing solution for PCa detection. As introduced in Section 3.2, this was achieved by means of a 9-layer custom CNN trained and validated with a set of patches after applying different processing steps, including patch filtering, stain normalization and data augmentation. This allowed achieving 99.98% accuracy, 99.98% F1 score and 0.999 AUC on a separate test set (see Section 3.3).

Since when analyzing WSIs by means of CNNs, the results are reported at patch level, different techniques have been proposed in the literature in order to combine them and generate a slide-level classification result, which could be of great importance for developing a fast PCa screening system. This technique is known as patch aggregation. Among the different studies that can be found in the literature, some performed different patch aggregation techniques based on RNNs, Random Forests (RFs) (Campanella et al., 2019) an other ML or statistical

alternatives (Ström et al., 2020; Bulten et al., 2020), achieving accurate solutions and leading to precise screening methods.

In this study, a custom novel Wide & Deep (W&D) model for aggregating the patch-level classification results obtained from PROMETEO into a global slide-level class is presented. This approach allows providing a fast screening method for PCa detection at WSI level, while also benefiting from the spatial resolution obtained at patch level. The promising results obtained, which have also been compared to other state-of-the-art ML-based approaches, show that the proposed solution could aid pathologists when analyzing histopathological images, discriminating between positive and negative PCa samples while fastening up the whole process.

5.2 Materials and Methods

5.2.1 Dataset

A set of H&E-stained slides from Pathological Anatomy Unit of Virgen de Valme Hospital were used (158 normal WSIs and 174 malignant WSIs). These images were preprocessed using the same steps as explained in Section 3.2.1. First, patches (100×100 pixels at 10× magnification), were extracted from WSIs. Next, background patches and patches corresponding to unwanted areas were discarded with a filter that discriminates them based on the amount of tissue that they contain, the percentage of pixels that are within H&E's hue range, and the dispersion of the saturation and brightness channels. Then, Reinhard stain-normalization was applied to patches in order to reduce stain variability between samples. Finally, color-normalized patches were used as input to PROMETEO, which classifies them as either malignant or normal tissue with a certain probability.

Different features were obtained from PROMETEO's output in order to create the dataset. The first feature considered to discriminate between malignant or normal WSIs was the percentage of malignant tissue area, also called malignant tissue ratio (MTR), expressed between 0 and 1. This was calculated by dividing the number of patches classified as malignant by the total amount of tissue patches extracted from the WSI. This is the most representative data to perform a slide-level classification, since the more malignant patches the network detects on the WSI, the more likely it is of being malignant. However, based on the error of the CNN when performing the patch classification, the percentage of malignant tissue of the WSI should not be the only input to be considered for the patch aggregation task, since there are some exceptions that do not meet the aforementioned rule (e.g. a malignant WSI with a small tumor in a specific region or a normal WSI with a relatively high percentage of incorrectly-classified malignant tissue area). Therefore, other features were considered to perform the patch aggregation step.

Another feature taken into account to distinguish between malignant and normal WSIs was the distribution of the prediction probability for malignant patches. When the CNN predicts a patch, it reports the probability of the patch for being either malignant or normal. If we only focus on the malignant probability, the network should have a higher confidence for patches corresponding to malignant tissue than for those corresponding to normal tissue that have been incorrectly predicted as malignant. Thus, a 10-bin histogram with the prediction probabilities of the patches classified as malignant for each WSI was calculated. These probabilities were distributed from 50% to 100%, with 5% range for each bin. The histogram was normalized with respect to the total number of tissue patches. Along with the malignant probability histogram (MPH), the least squares regression line (LSRL) of the histogram, defined as $y = mx + b$, was also calculated, where m and b, which refer to the slope and the Y-intercept, are described in Equations 5.1 and 5.2, respectively. This line represents the best approximation of the set of probabilities for all malignant patches of the corresponding WSI. The mean histogram for both malignant and normal WSIs are shown in Figure 5.1 together with their corresponding LSRLs, which are highlighted in red.

$$m = \frac{N\sum x_i y_i - \sum x_i \sum y_i}{N\sum x_i^2 - (\sum x_i)^2} \tag{5.1}$$

$$b = \frac{N\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{N\sum x_i^2 - (\sum x_i)^2} \tag{5.2}$$

Where x and y represent the coordinates of the different values of the histogram.

As it was previously mentioned, the error of the ML algorithm (a CNN in this case) leads to errors in the classification, which in a WSI is presented as sparse normal tissue patches being classified as malignant. Therefore, in a WSI diagnosed as normal, patches classified as malignant by the CNN are sparsely distributed through the tissue. On the other hand, in a cancerous WSI, malignant-classified patches tend to be focused around the tumor areas. Due to this reason, the dispersion factor of malignant-classified patches was also considered as another relevant input for the slide-level classification between normal and malignant WSIs. This factor was obtained by calculating the number of malignant connected components (MCC), which counts the isolated components (sets of malignant patches) in the classification result according to a specific distance D. Algorithm 1 details the method used to calculate the number of connected components based on the center coordinates of malignant patches and D. In this experiment, five different values were considered for D (142, 283, 425, 566 and 708 pixels), which correspond to the Euclidean distances (i.e., radii) from a patch

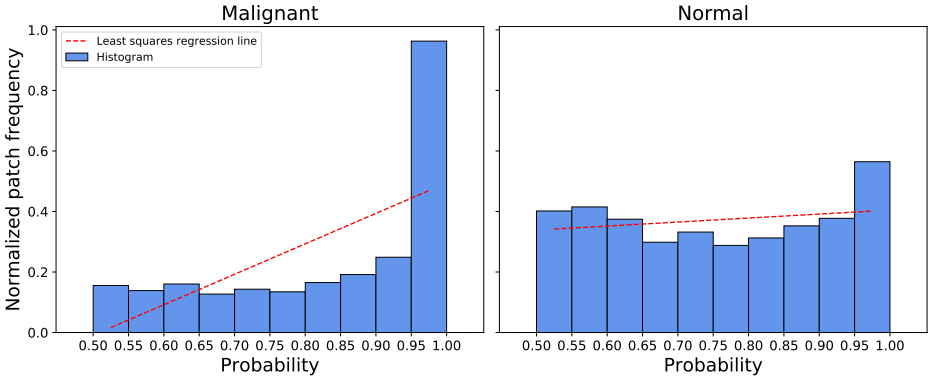

FIGURE 5.1: Mean probability histogram of the normalized patch frequency across all the WSIs, distinguishing between malignant (left) and normal (right) samples. The least squares regression line is shown with a red dashed line. As can be seen, for malignant WSIs, the system tends to classify patches as malignant with a higher confidence. This produces a least squares regression line with a steeper slope. On the other hand, for the normal WSIs, the classification for malignant patches is not that accurate, which leads to a less steep regression line.

to a range of 1 up to 5 patches-distance, taking into account that the distance between two patches is 100 pixels (patches are 100×100 pixels size). The number of connected components was normalized with respect to the total number of malignant patches for each WSI. This way, normal samples with a low quantity of sparse misclassifications are penalized when compared to malignant samples with sparse tumoral tissue regions.

5.2.2 Wide & Deep network model

The dataset described in Section 5.2.1 was used as input to a neural network model called Wide & Deep (W&D) (Cheng et al., 2016) to provide a slide-level classification between normal and malignant WSIs. The W&D model combines both wide and deep components. The wide component memorizes sparse interactions between features effectively, which can be defined as learning how the output responds to combinations of sparse input values. On the other hand, the deep component corresponds to the feed-forward neural network which represents the generalization, this is, the ability to handle unseen data. Therefore, the benefits from both memorization (wide) and generalization (deep) are combined and achieved in a single model (Cheng et al., 2016).

The malignant tissue ratio was used as the wide element while the malignant probability histogram, the slope and Y-intercept of the LSRL and the number of malignant connected components were used as the deep elements. Each of

Algorithm 1: Connected components algorithm

```
ConnectedPatches (centers, D)
    inputs: A list of center points from malignant patches (centers); a
            distance (D).
    output: A set of lists of connected patch centers with relative
            distance D.
    connected_components = [];
    current_component = [];
    while count(centers) > 0 do
        current_component = [];
        current_component.append(centers[0]);
        centers.remove(centers[0]);
        foreach center in current_component do
            foreach point in centers do
                if distance from center to point <= D then
                    current_component.append(point);
                    centers.remove(point);
        connected_components.append(current_component);
```

the deep data were separately connected to two hidden layers of 300 neurons. Then, these layers were concatenated together with the wide element to a hidden block of two hidden layers with 300 neurons each. Finally, this hidden block was connected to the output layer, a SoftMax function which performs the classification of the WSI as either malignant or normal. This way, complex features are extracted from combinations of sparse inputs and then concatenated together in order to perform the final decision.

Figure 5.2 depicts the custom W&D model used, where the different inputs and layers can be seen. Figure 5.3 represents the whole processing step for the prostate screening task, highlighting both the patch-level and the slide-level processes.

5.2.3 Training and validating the system

K-fold stratified cross-validation was performed to measure the generalization ability of the model. This technique consisted in dividing the dataset in 5 sets ($K = 5$). For each fold, the network was trained using four of the five sets (80% of the dataset) for 10000 epochs and validated using the remaining one (20% of the dataset). This way, for each experiment, the network was trained and validated a total of five times with different data. The final results are presented as the mean accuracy calculated over the five cross-validation folds.

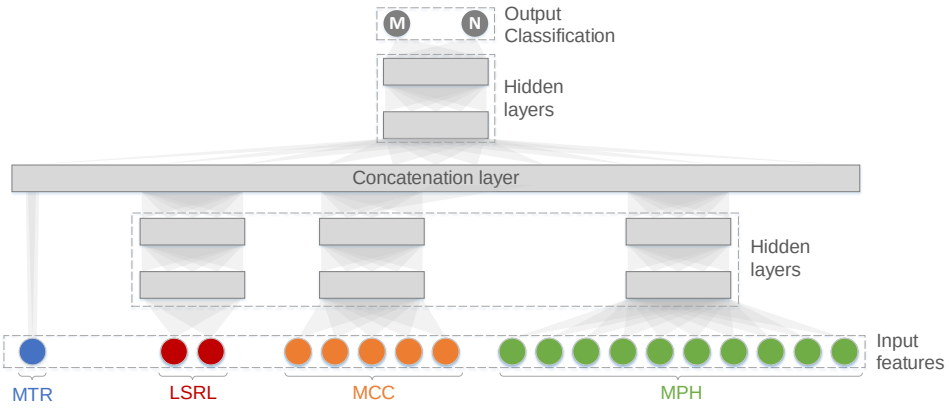

FIGURE 5.2: Diagram of the W&D network model proposed in this study. Each hidden layer consists of 300 neurons. The input features, which are detailed in Section 5.2.1, are: the malignant tissue ratio (MTR) of the WSI, the slope and Y-intercept of the least squares regression line (LSRL) of the histogram, the number of malignant connected components (MCC) with 5 different radii (from 1 to 5 malignant patch distance), and the 10-bin malignant probability histogram (MPH) between 50% and 100% with 5% ticks. These input features are used to classify the WSI as either malignant (M) or normal (N).

To validate the network, different evaluation metrics were used. These were the accuracy (Equation 3.5), precision (Equation 3.6), sensitivity (Equation 3.7), F1 score (Equation 3.9) and AUC of the ROC curve.

5.3 Results

After training the custom W&D model (Section 5.2.2) with the dataset presented in this chapter, all the different metrics were calculated and obtained in order to evaluate the proposed system. Table 5.1 summarizes the results for each cross-validation fold together with the average for all the evaluation metrics. With these, the average results were calculated, achieving an accuracy of 94.24%, a sensitivity of 98.87%, a precision of 90.23%, a F1 score of 94.33% and an AUC of 0.94.

As can be seen, the results obtained across the different folds are consistent. The proposed model achieves very high scores in all the different metrics studied for this classification task, particularly in terms of sensitivity. The sensitivity, which in this field is defined as the ability of the system to identify PCa, is of utmost importance for reporting and assessing the performance of the screening test (Hakama et al., 2007). The proposed system is able to achieve an average sensitivity of around 99%, where three of the folds achieved perfect

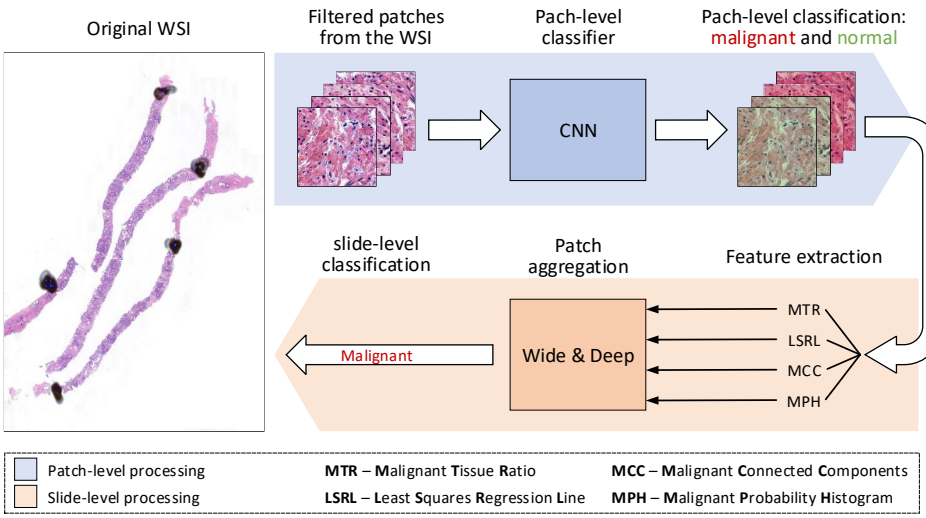

FIGURE 5.3: Diagram of the whole processing step for the PCa screening task. First, the WSI is processed at patch level, following the same procedure presented in Figure 4.1. Then, the output classification for each of the filtered patches from the original WSI is used to perform a slide-level prediction using the W&D model presented in Figure 5.2, where the extracted features are used to classify the WSI as either malignant or normal.

sensitivity (100%). This means that the proposed custom model makes almost no mistakes when predicting a malignant sample as such, making it a reliable patch aggregation method, together with PROMETEO, for PCa detection in WSIs. Figure 5.4 shows some examples of correctly classified WSIs together with their corresponding heatmaps generated by PROMETEO.

The results obtained in this study were compared with different ML-based methods and classifiers using the same dataset. The following well-known machine learning algorithms were used to classify the WSIs: an ANN (Yegnanarayana, 2009), a SVM (Wang, 2005), a RF (Breiman, 2001) and a k-Nearest Neighbors (KNN) (Jiang et al., 2007). Table 5.2 summarizes the results obtained for each method, which are represented as the average of the evaluation metrics (see Section 5.2.3) obtained for each cross-validation fold.

As it can be seen, the best results for accuracy, sensitivity, F1 score and AUC are obtained with the proposed W&D model, with the exception of precision, for which SVM achieves the highest value. As it was previously mentioned, sensitivity is the most relevant metric for measuring the performance of a classifier when performing a screening test. In this case, the proposed architecture is the one achieving the highest sensitivity score among the different algorithms evaluated, with a difference of more than 6% with the second highest, which is the

TABLE 5.1: Validation results obtained with the proposed W&D model. The accuracy, sensitivity, precision, F1 score and AUC) are shown for each of the different cross-validation folds. The average of the obtained metrics across the five folds is also presented.

Fold	Accuracy (%)	Sensitivity (%)	Precision (%)	F1 score (%)	AUC
1	93.93	100	89.74	94.59	0.93
2	93.93	97.29	92.30	94.73	0.93
3	95.45	100	90.32	94.91	0.96
4	93.93	100	87.09	93.10	0.94
5	93.93	97.05	91.66	94.28	0.93
Average	**94.24**	**98.87**	**90.23**	**94.33**	**0.94**

ANN. On the other hand, SVM achieves around 99% precision, which could be very relevant for other binary or multi-class classification tasks, but not as much as the sensitivity when developing a medical screening method to differentiate between positives and negatives samples.

TABLE 5.2: Validation results calculated from the average of the evaluation metrics (accuracy, sensitivity, precision, F1 score and AUC) for the 5 different cross-validation sets. The results obtained with the proposed W&D model are compared to other state-of-the-art ML-based algorithms, namely, ANN, SVM, RF and KNN. The best result for each specific evaluation metric is highlighted in bold.

Model	Accuracy (%)	Sensitivity (%)	Precision (%)	F1 score (%)	AUC
W&D (proposed)	**94.24**	**98.87**	90.23	**94.33**	**0.94**
ANN	89.69	92.47	87.29	89.54	0.89
SVM	88.18	80.78	**98.76**	88.79	0.89
RF	88.84	84.89	92.23	88.22	0.88
KNN	88.48	83.29	94.31	88.31	0.88

More details of the results obtained in this chapter can be found in Appendix C.

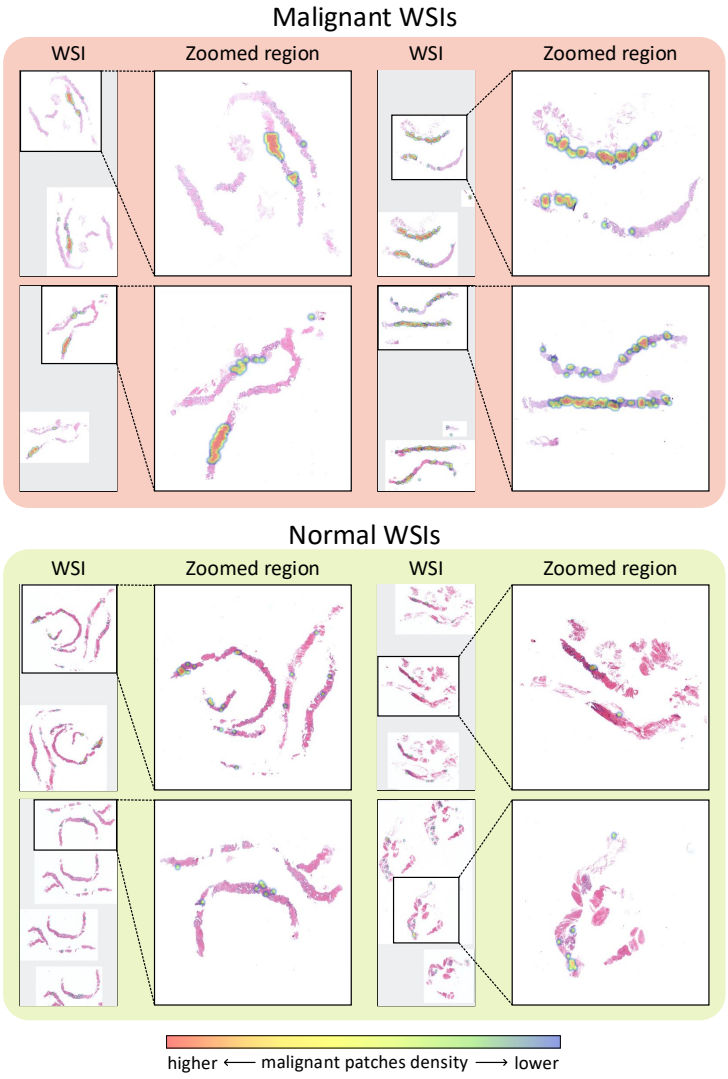

FIGURE 5.4: Eight different WSI samples extracted from the dataset presented in Section 5.2.1. A heatmap of the malignant patches predicted by PROMETEO is drawn on top of the WSI, and zoomed regions are presented for better visualization. Red regions represent higher concentrations of malignant patches, while blue represent the opposite. The examples presented were correctly classified by the proposed W&D model.

Chapter 6

Development of a global CAD system for Gleason pattern classification in WSIs

6.1 Introduction

Once the tumor is detected, it is crucial to measure how aggressive it is. As mentioned in Section 1.2.5.1, Gleason Grading System (GGS) is the most important prognostic marker of PCa, which is critical to patient management since it drives therapies across all disciplines dealing with PCa. The GGS is focused on determining the cellular differentiation degree of a tumor, considering 5 different patterns (1 to 5). Grade 1 is assigned to areas of the tissue containing cells that resemble to normal prostate cells, whereas, in grade 5, cancer cells greatly differ to normal prostate cells.

Several researchers have investigated the application of CAD systems to differentiate GGS patterns. Some of these studies distinguish between low and high grades (Toro et al., 2017; Li et al., 2018), while others have performed multi-class detection, discriminating between the different GGS patterns (Doyle et al., 2007; Ren et al., 2017; Ström et al., 2020; Bulten et al., 2020). Many of these CAD systems are based on ML techniques, such as neural networks or SVMs. Table 6.1 compares some of these studies, summarizing the main characteristics of the dataset, the preprocessing step, the classifier that the authors used, the classes considered and the results obtained with their corresponding performance metrics.

In this chapter, a DL-based CAD system was developed in order to classify Gleason patterns, considering grades 3, 4 and 5; 1-2 patterns were not taken into account, since these are very unusual to find when analyzing malignant WSIs (Chen and Zhou, 2016). Class Activation Maps (CAMs) were applied to the output of the CNN together with the input image in order to generate interactive heatmaps, showing in detail which regions or patterns the network has focused on to perform the classification.

TABLE 6.1: Comparative study between state-of-the-art research about Gleason patterns classification in prostate cancer histological images.

Ref.	Dataset	Preprocessing step	Classifier	Classes	Performance measure
Toro et al., 2017	235 WSIs[1]: - Train: 282k patches - Val: 94k patches - Test: 92k patches	Binary tissue mask obtained by Blue Ratio image to remove background.	CNN[2] (GoogLeNet)	2: High and low GGS[3]	ACC[4] at patch level: 78.2% on test 73.52% on validation
Li et al., 2018	513 WSIs: - Train: not specified - Test: not specified	Normalize procedure to eliminate stain variability.	R-CNN[5] (custom)	4: Stroma, benign glands, low GGS, high GGS	IOU[6] *: 79.56% OPA[7] *: 89.40% SMA[8] *: 88.78% *at tile level (set of patches)
Doyle et al., 2007	54 patches: - Train: not specified - Test: not specified	Extraction of architectural features. Extraction of 1st-order statistical features with the average, median, standard deviation and range of the pixel values. Extraction of 2nd-order statistical features (Haralick features) from a co-occurrence matrix.	SVM[9]	2, but with different classes: Epithelium vs stroma GGS 3 vs GGS 4 GGS 3 vs epithelium GGS 3 vs stroma GGS 4 vs epithelium GGS 4 vs stroma	ACC at patch level: Epithelium vs stroma: 76.9% GGS 3 vs GGS 4: 76.9% GGS 3 vs epithelium: 85.4% GGS 3 vs stroma: 92.8% GGS 4 vs epithelium: 88.9% GGS 4 vs stroma: 89.7%
Ren et al., 2017	22 WSIs - Train: 17 WSIs - Test: 5 WSIs	Segmentation procedure with CNN and superpixel segmentation. Feature extraction with Bag-of-Word to remove background.	RFC[10]	2: GGS 3 and GGS 4	F1-score*: 0.8460 Sensitivity*: 0.70±0.15 Specificity*: 0.89±0.04 ACC*: 0.83±0.03 *at patch level
Ström et al., 2020	8914 WSIs: - Train: 6953 WSIs - Val: 1631 WSIs - Test: 330 WSIs	Segmentation algorithm based on Laplacian filtering.	CNN (60 Inception V3)	2: Normal and malignant 3: GGS 3, GGS 4 and GGS 5	AUC[11] for normal and malignant*: 0.997 on validation 0.986 on test Mean pairwise kappa for GGS*: 0.62 *at slide level
Bulten et al., 2020	1243 WSIs: - Train: 933 WSIs - Val: 100 WSIs - Test: 210 WSIs	Tissue segmentation network for extracting tissue from background. Tumor detection system to define the tumor and epithelial tissue detection system to label the images.	CNN (U-Net)	6: Benign, GGG[12] 1-5.	AUC at slide level: Benign vs malignant: 0.990 Benign and GGG 1 vs GGG≥2: 0.978 Benign and GGG 1-2 vs GGG≥3: 0.974

[1]: Whole Slide Image. [2]: Convolutional Neural Network. [3]: Gleason Grading System. [4]: Accuracy. [5]: Region-based Convolutional Neural Network. [6]: Intersection Over Union. [7]: Overall Pixel Accuracy. [8]: Standard Mean Accuracy. [9]: Support Vector Machine. [10]: Random Forest Classifier. [11]: Area Under Curve. [12]: Gleason Grade Group.

6.2 Materials and methods

6.2.1 Dataset

The dataset used for GGS classification was obtained from the 70 malignant classified H&E stained slides from Virgen de Valme Hospital (see Table 3.3). Each slide was analyzed and labeled by pathologists so that malignant regions were delimited with a specific Gleason pattern (see Section 3.2.1.1).

Due to the small amount of images for the dataset and taking into account that the network has to classify between three different classes with similar patterns, other public datasets were used in order to increase the number of images and their variability. These new images were obtained from:

- Five PCa Tissue Microarrays (TMAs) of H&E stained images, each of these containing 200–300 tissue spots (Arvaniti et al., 2018a).
- SICAPv2 database, which includes 155 biopsies from 95 different patients (Silva-Rodríguez et al., 2020). These database consists of several prostate WSIs with Gleason grades annotations.
- The Gleason Challenge from MICCAI 2019 (Nir et al., 2018), which consists of seven TMA images, each of them containing 100-200 tissue spots, annotated in detail by several expert pathologists.

6.2.1.1 Patch sampling and preprocessing steps

To create the dataset, these WSIs and the images obtained by the public datasets, were preprocessed by applying different steps. The first step consisted in dividing each WSI into patches (256×256 at $10\times$). Only those patches which overlaps with the ROIs selected by the pathologists (at least an 80%) were considered for the dataset. An overlap between patches was applied in order to obtain more images for training. After that, patches corresponding to background or unwanted areas (noise, pen marks, among others) were removed by applying the filter explained in Section 3.2.1.2. The number of patches obtained for each Gleason pattern after applying the aforementioned steps is shown in Table 6.2. Finally, the patches which pass the filter were stain-normalized applying Reinhard method in order to reduce color variability between slides. More details of the whole preprocessing step can be seen in Section 3.2.1.3.

Around 85% of the dataset was used for the training phase, while the remaining 15% was considered for validating the network. This division was made ensuring that patches from the same patient were not present in the two sets at the same time. Patches obtained from the public datasets were only used for the training phase. This way, the network has a higher variability of images, leading to a more robust learning.

TABLE 6.2: GGS dataset classes distribution.

Pattern	No. of patches	% of the total	No. of patches after data augmentation
3	5558	32.43%	8 millions
4	8253	48.15%	12 millions
5	3328	19.42%	5 millions
Total	**17139**	**100%**	**25 millions**

6.2.1.2 Data augmentation

In order to increase the number of images for the training step, data augmentation was applied to the dataset. Different transformations were performed to the original patches. Horizontal flips and vertical flips were applied for each training patch, along with rotations in the whole 360° range with steps of 1°, where the missing information in the corners after rotating the patch was filled by mirroring. This way, for each patch, new 2×2×360 patches were obtained.

6.2.2 Computer-Aided Diagnosis system design

6.2.2.1 Convolutional Neural Network architecture

A custom CNN was developed to perform the GGS classification. This network consists of four convolution stages (convolution layer + ReLu + pooling layer). After that, a final convolution layer + ReLu are included. The convolution layers have 64 3×3, 64 3×3, 128 3×3, 128 3×3 and 256 3×3 filters, respectively. The last layer is connected to a Global Average Pooling (GAP), which generates feature maps that are then used to perform the classification. This classification is made with a FC layer (SoftMax activation function) which distinguishes between the three Gleason patterns. Figure 6.1 shows the architecture of the CNN.

GAP is used to reduce the spatial dimensions of a three-dimensional tensor $h \times w \times d$ to $1 \times 1 \times d$ (Figure 6.2). This layer reduces each $h \times w$ feature map to a single number by calculating the average. GAP is needed as the last layer for the feature extraction phase of a CNN to perform a Class Activation Map (CAM) (see Section 6.2.2.3).

6.2.2.2 Training and validating the system

Due to the imbalance between classes (32.43%, 48.15% and 19.42% for patterns 3, 4 and 5, respectively), a class weights function was used. The whole purpose of this function is to penalize the misclassification made by the minority class by

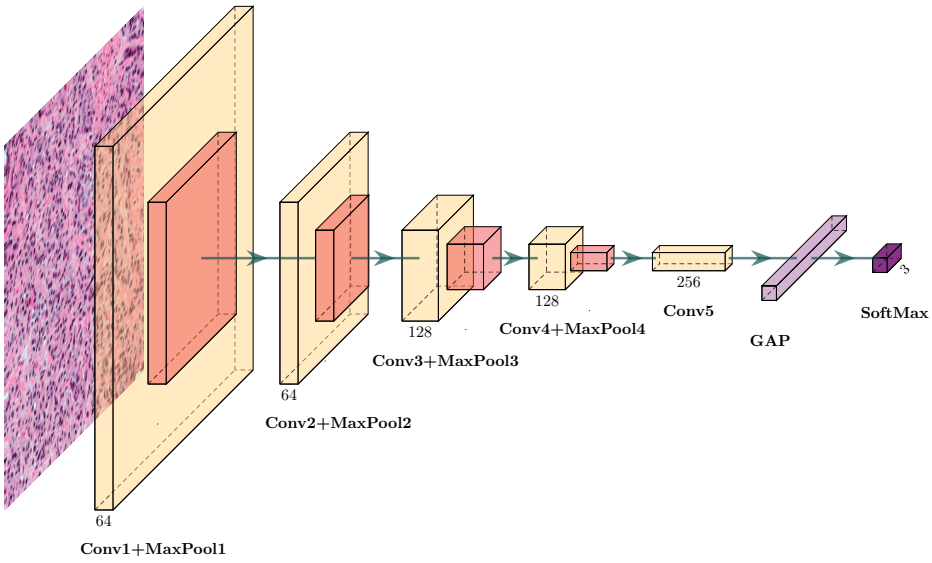

FIGURE 6.1: Diagram of the architecture of the CNN used for the Gleason pattern classification task. Convolution filters are always followed by a ReLU unit, and each of them are of size 3×3. All the pooling layers are 2×2 max pooling. GAP stands for Global Average Pooling layer, which reduces each feature map to a single value by calculating the average. This layer is connected to a SoftMax, which performs the decision between GGS 3, 4 and 5.

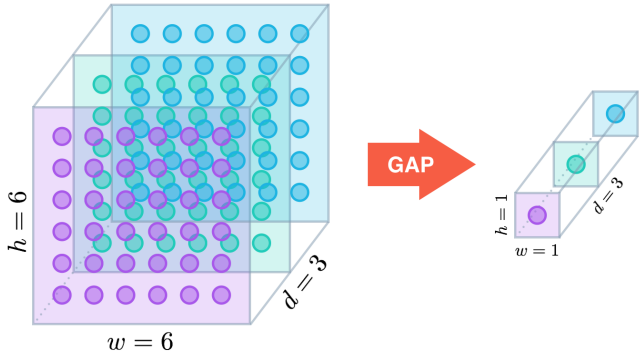

FIGURE 6.2: Example of a GAP operation for an input feature of 6×6.

setting a higher class weight, while, at the same time, reducing the weight for the majority class. Therefore, the unbalancing of the classes does not affect the results

of the network.

Different evaluation metrics were used in order to validate the trained model: sensitivity (Equation 3.7), precision (Equation 3.6), F1 score (Equation 3.9), and AUC of the ROC curve. Each metric was calculated per class, obtaining the mean by averaging them taking into account the imbalance between the three classes in the validation set. Finally, the balanced accuracy was also calculated as the overall measure of the system. All of them were measured at patch level.

6.2.2.3 Class Activation Map

Zhou et al. demonstrated that even without providing any information of the object location inside an input image, convolutional units of CNNs work as unsupervised object detectors (Zhou et al., 2014; Zhou et al., 2016). With this idea in mind, CAMs can be generated. A CAM for a particular class highlights the regions of the input image that the CNN considered relevant to perform the prediction. As mentioned before, when working with CAMs, the use of a GAP layer before the classification step is recommended. GAP encourages the network to identify the extent of the object as compared to a max pooling layer which encourages it to identify just one discriminative part.

To generate a CAM for a class c, firstly, the k weights connected between the GAP layer and the softmax layer of c are obtained. These weights are multiplied with each feature map f used as input to the GAP layer and then added. A weight represents the relevance of every individual channel in the entire feature map. This way, the final weighted sum results in a heatmap of a particular class with values ranging from 0 to 1. The heatmap size corresponds to the size of the feature map f, thus, heatmaps have to be scaled to the size of the input image. This way, each spatial element (x, y) of the CAM for c (M_c) can be defined by:

$$M_c(x, y) = \sum_k w_c^k f_k(x, y) \tag{6.1}$$

Figure 6.3 presents an example from Zhou et al., 2016, showing CAMs generated from the top 5 predicted categories for a given image with "dome" as ground-truth. CAMs are represented as heatmaps, where red highlights the regions which the network considered more relevant when performing the classification.

6.3 Results

After training the custom CNN architecture presented in this chapter with a batch size of 16 and the Adadelta optimizer (Zeiler, 2012), the trained model was evaluated on the dataset. Table 6.3 presents the results obtained from

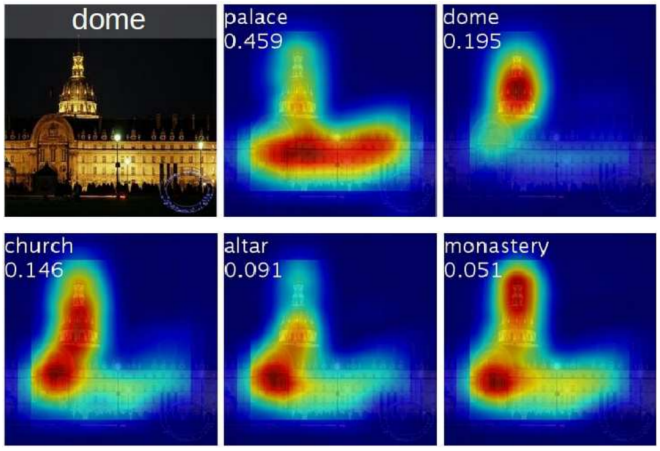

FIGURE 6.3: Example of the CAMs generated from five categories for an input image labeled as "dome". The relevant regions where the network focused on to perform the classification are highlighted in heatmaps. The predicted class and its score are shown in the top part of each CAM. Image taken from Zhou et al., 2016.

the validation images. The proposed CNN model achieved 84.60% balanced accuracy, 83.77% sensitivity, 83.70% precision, 83.66% F1 score and 0.87 AUC.

Figure 6.4 presents the different ROC curves for each class with their corresponding AUC value. The idea for the ROC curve is to carry out a pairwise comparison between two classes, and, since this experiment consists of 3 different classes, a different approach was followed for calculating the ROC curve and the AUC for each of them. A common solution for this is to use a one-versus-all strategy, where the ROC curve for a specific class is calculated against the rest of the classes. This analysis was followed for plotting the curves in this figure, where the blue curve represents class 3 against 4 and 5, and so on.

TABLE 6.3: Validation results obtained for the 3 GGS classes with the proposed CNN model. The average was calculated taking into consideration the imbalance of the validation dataset.

Classes	Sensitivity (%)	Precision (%)	F1 score (%)	AUC	Accuracy (%)
3	77.62	83.29	80.36	0.84	-
4	81.31	75.89	78.51	0.83	-
5	94.87	95.36	95.11	0.97	-
Average	**83.77**	**83.70**	**83.66**	**0.87**	**84.60**

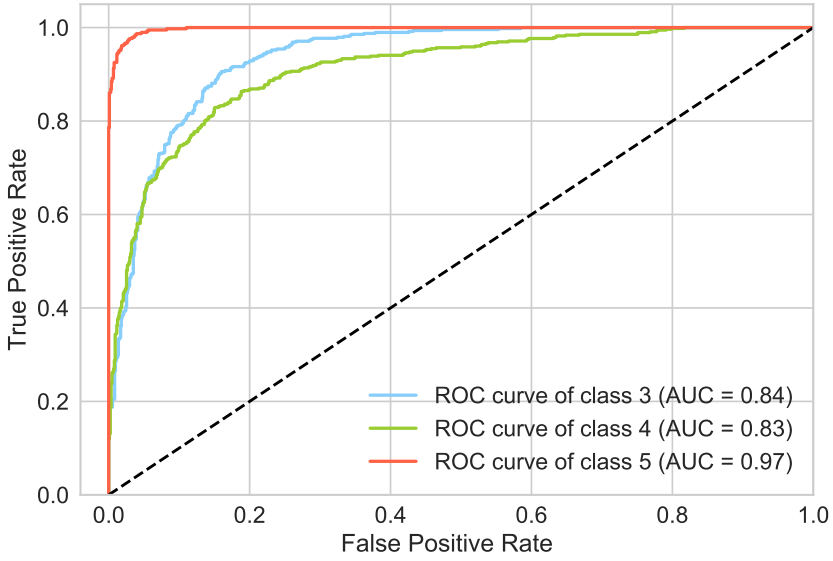

FIGURE 6.4: ROC curves and AUC values for each of the three classes considered: GGS pattern 3 (blue), GGS pattern 4 (green) and GGS pattern 5 (red). A one-versus-all strategy was followed for calculating the ROC curves, where the curve for a specific class was obtained performing a pairwise comparison between that class and the remaining two.

Figure 6.5 presents the confusion matrix obtained over the validation set for the three classes considered in this experiment. As it can be seen, the system correctly learnt the differences between the features extracted and it is able to perform the classification accordingly, achieving a high accuracy for each of the three classes. The confusion matrix shows that Gleason pattern 5 is the class with less error rate. On the other hand, patterns 3 and 4 have a higher error rate, since the proposed system often confuses between them, due to the fact that patterns 3 and 4 are more similar to each other than with respect to 5.

After classifying the input patches as GGS patterns 3, 4 or 5, the system is able to generate the CAM of the winner class for each of them. The output of the system reports an image obtained by applying a threshold to the CAM that corresponds to the input patch, considering only the regions with CAM's values greater than this threshold. These values are stained to blue, yellow or red, depending of the output class predicted by the network (Gleason pattern 3, 4 or 5, respectively). Examples of the output of the CAD system for Gleason pattern 3, 4 and 5 are shown in Figures 6.6, 6.7 and 6.8, respectively.

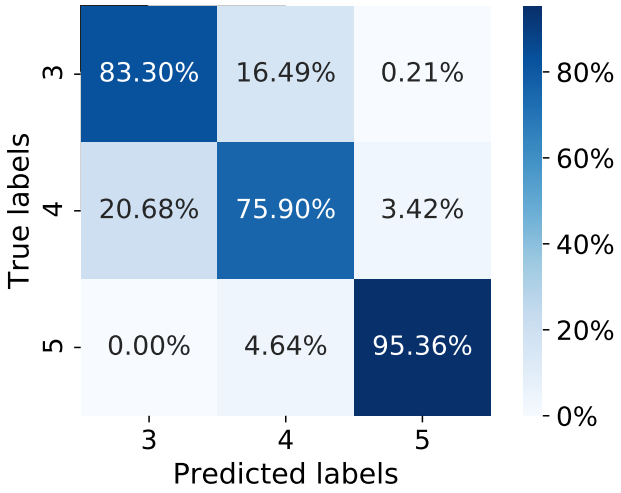

FIGURE 6.5: Confusion matrix for the three classes considered: GGS pattern 3, GGS pattern 4 and GGS pattern 5.

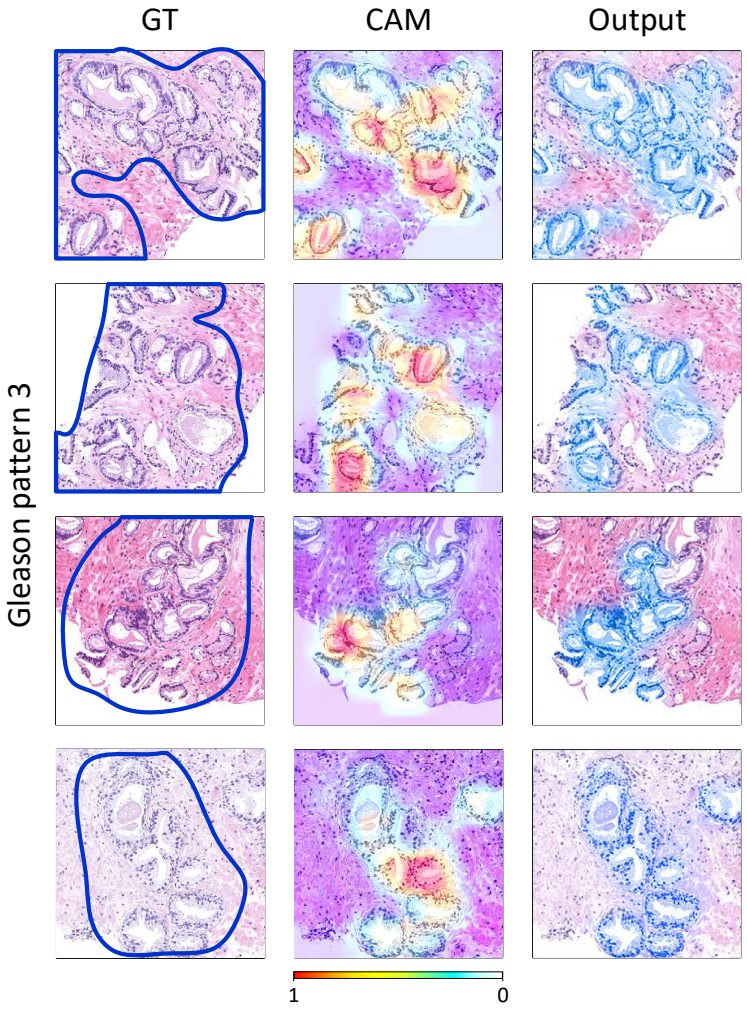

FIGURE 6.6: Examples of the output of the CAD system for Gleason pattern 3. The left column shows the ground truth (GT) with labels obtained from pathologists, locating the malignant areas. At the center column, the CAM for each input image is shown with a heatmap, which ranges from 0 (white) to 1 (red) values representing the relevance that the network considered when performing the classification. At the right column, the output of the system for pattern 3 is presented in blue.

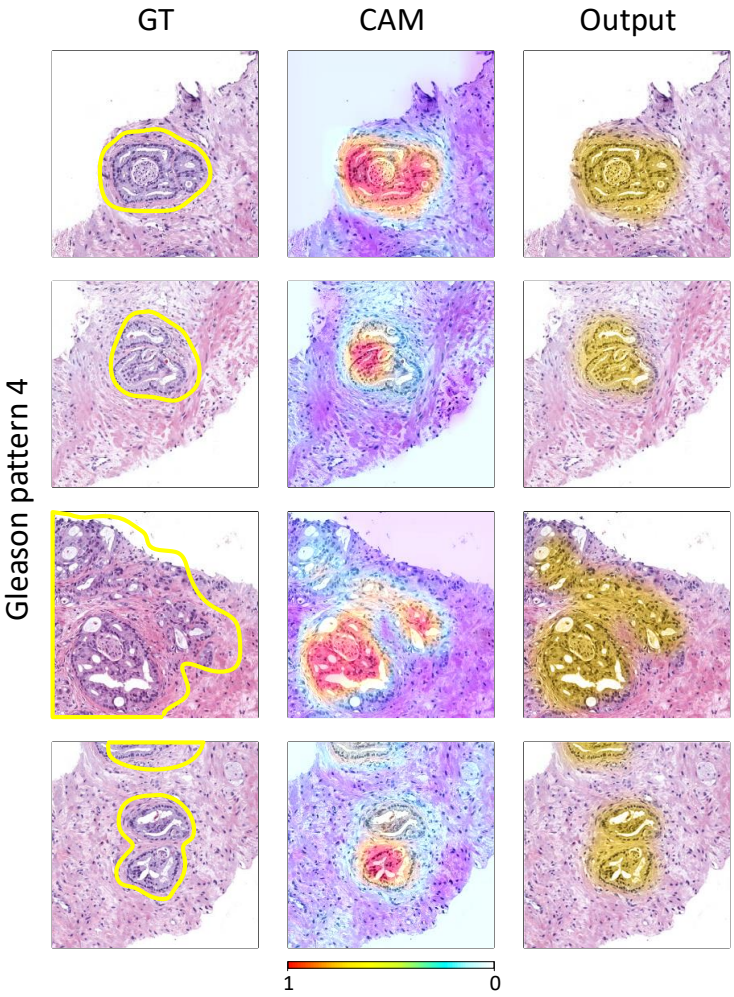

FIGURE 6.7: Examples of the output of the CAD system for Gleason pattern 4. The left column shows the ground truth (GT) with labels obtained from pathologists, locating the malignant areas. At the center column, the CAM for each input image is shown with a heatmap, which ranges from 0 (white) to 1 (red) values representing the relevance that the network considered when performing the classification. At the right column, the output of the system for pattern 4 is presented in yellow.

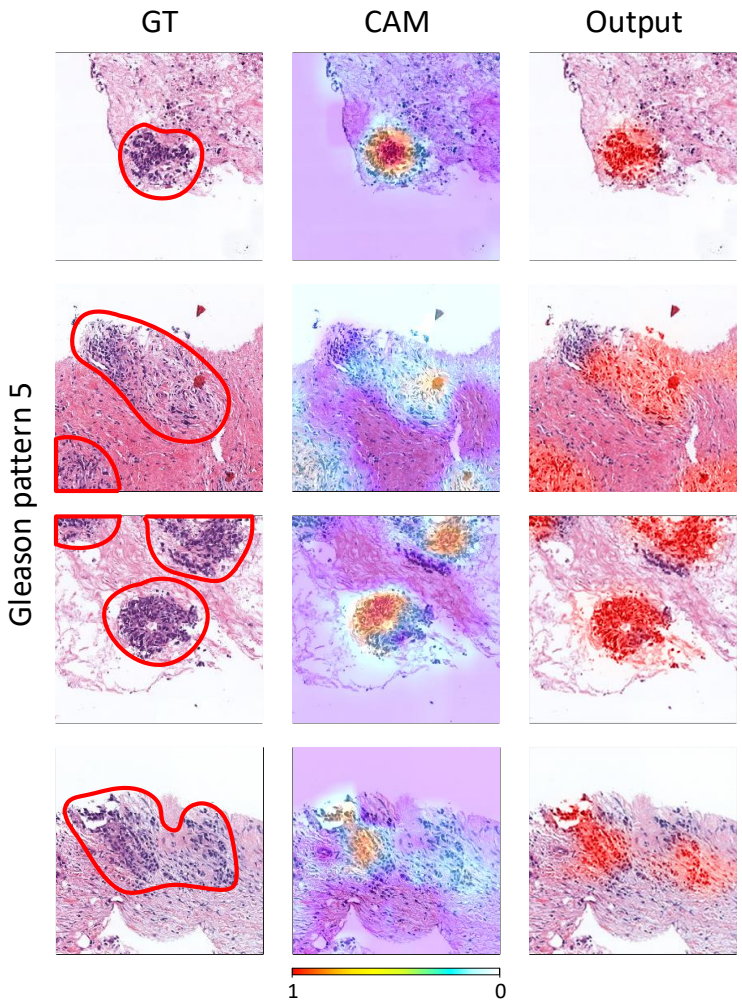

FIGURE 6.8: Examples of the output of the CAD system for Gleason pattern 5. The left column shows the ground truth (GT) with labels obtained from pathologists, locating the malignant areas. At the center column, the CAM for each input image is shown with a heatmap, which ranges from 0 (white) to 1 (red) values representing the relevance that the network considered when performing the classification. At the right column, the output of the system for pattern 5 is presented in red.

6.4 Global CAD system for prostate cancer detection and Gleason pattern recognition

The system presented in this chapter could be used as the last processing step of a global CAD system for PCa detection and Gleason pattern classification when combined with all the different parts presented in this Thesis. Figure 6.9 presents a block diagram of the whole system, which can be summarized in three main steps:

- First, patches are extracted from the original WSI. These are filtered, removing those corresponding to background or unwanted areas, and, then, stain-normalized. After this pre-processing step, patches are predicted using the custom CNN model (PROMETEO) presented and evaluated in Chapter 3 and Chapter 4. This step is shown in Figure 6.9 as patch-level processing.
- After this, different features are extracted from the predictions obtained in the previous step for the patches contained in a single WSI. These are the malignant tissue ratio (MTR), the least squares regression line (LSRL), the number of malignant connected components (MCC) and the malignant probability histogram (MPH). The aforementioned features are used as input to the custom W&D model presented in Chapter 5, which outputs a slide-level label for the original WSI. This allows aggregating the patch-level information into a single slide-level class (malignant or normal WSI). This step is shown in Figure 6.9 as slide-level processing. In the case that this process identifies the WSI as normal, the system would report it and start analyzing the next WSI. In the opposite case, if the system detects that the input image corresponds to a malignant sample, the process continues to the next step.
- Finally, if the WSI is predicted as malignant, 256×256 pixels-size patches centered at the ones predicted as malignant by PROMETEO are read and stained using Reinhard method. After this, the Gleason pattern classification system presented in this chapter assigns a pattern between 3 and 5 to each of them. With the patch-level Gleason pattern report obtained, a heatmap is generated from the CAMs, highlighting the malignant areas of the tissue with their corresponding associated pattern predicted by the proposed CAD system. This step is shown in Figure 6.9 as Gleason pattern classification.

The performance evaluation study described in Chapter 4 allowed analyzing the impact of different hardware components on the execution time of PROMETEO. This study was used to design and configure a custom small-size high-performance barebone to be set as the main processing unit for the global CAD system. The aforementioned barebone consists of an AMD® Ryzen™ 5 5600X 6-Core Processor running at 3.70 GHz, 16 GB DDR4 RAM memory at

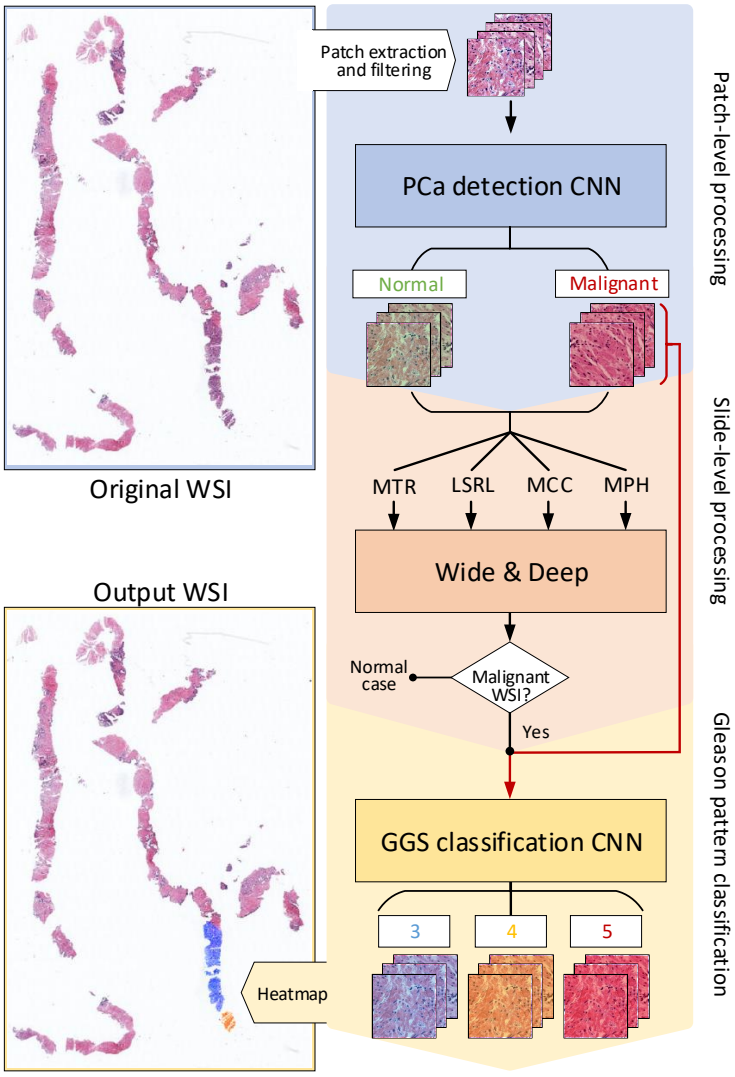

FIGURE 6.9: Block diagram of the global CAD system for PCa detection and Gleason pattern recognition. The original WSI is divided into patches, which are classified as either malignant or normal following the procedure presented in Chapter 3. Then, a global label is assigned following the processing steps presented in Chapter 5. Finally, if the original WSI is classified as malignant, the Gleason pattern report is obtained. MTR stands for malignant tissue ratio; LSRL for least squares regression line; MCC for malignant connected components, and MPH for malignant probability histogram.

3.2 GHz, 1 TB M.2 NVMe PCIe Gen 4 SSD with 7000 MB/s sequential reading speed, and an NVIDIA GeForce GTX® 1660 6GB DDR5 GPU. With this hardware configuration, the same benchmark presented in Chapter 4 was run, while also including three new steps related to the Gleason pattern classification CNN presented in this chapter. Therefore, all the main steps of the global CAD system shown in Figure 6.9 were taken into consideration in order to measure the average time needed by this system to process both a single patch and a single WSI.

Figure 6.10 presents the average time per step when using the custom barebone. *Patch Read*, *Patch Stain Normalization*, *Patch Score* and *Patch Prediction* are related to the PCa detection CNN (PROMETEO) described in Chapters 3 and 4. On the other hand, *Patch Read GGS*, *Patch Stain Normalization GGS* and *Patch Prediction GGS* refer to the processes of reading a 256×256 pixels-size patch centered at the one predicted as malignant by PROMETEO, staining it using Reinhard stain normalization, and predicting it with the Gleason pattern classification CNN, respectively.

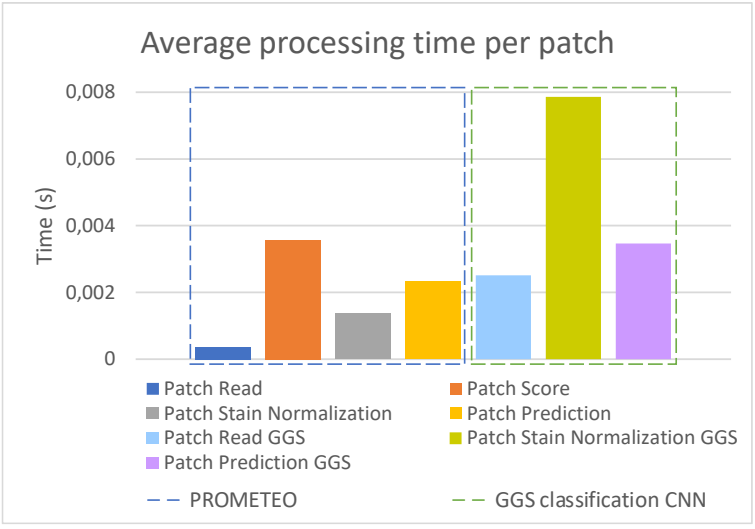

FIGURE 6.10: Average patch processing time (in seconds) for each of the steps performed by the global CAD system. The first four steps correspond to the PCa detection CNN, and the remaining three to the Gleason pattern recognition system.

Figure 6.11 presents the same idea that in Figure 6.10, but, instead of calculating the average execution time per patch, the average time was calculated per WSI. This way, a more realistic scenario can be analyzed, and the execution time that the Gleason pattern recognition system adds to PROMETEO can be studied. As it can be seen in the figure, the GGS classification CNN adds a negligible amount of time to the PCa detection system, even when taking

into consideration that a bigger patch has to be read and stain-normalized before being predicted by the CNN. This is mainly caused by the fact that the Gleason pattern classification is only performed on patches that are predicted as malignant by PROMETEO, and only if the W&D model has considered the WSI as malignant. For normal patches and for WSIs not passing the W&D filter, the Gleason pattern classification process is not executed, reducing execution time and resources, and allowing a faster response to the expert pathologist. A clearer comparison between the original PROMETEO system and the global CAD system involving both PROMETEO and the GGS classification CNN is presented in Figure 6.12. It can be seen from this figure that the time added by the last to the global CAD system is only 0.109 seconds on average, which represents a 0.69% of the total amount of time (15.733 seconds).

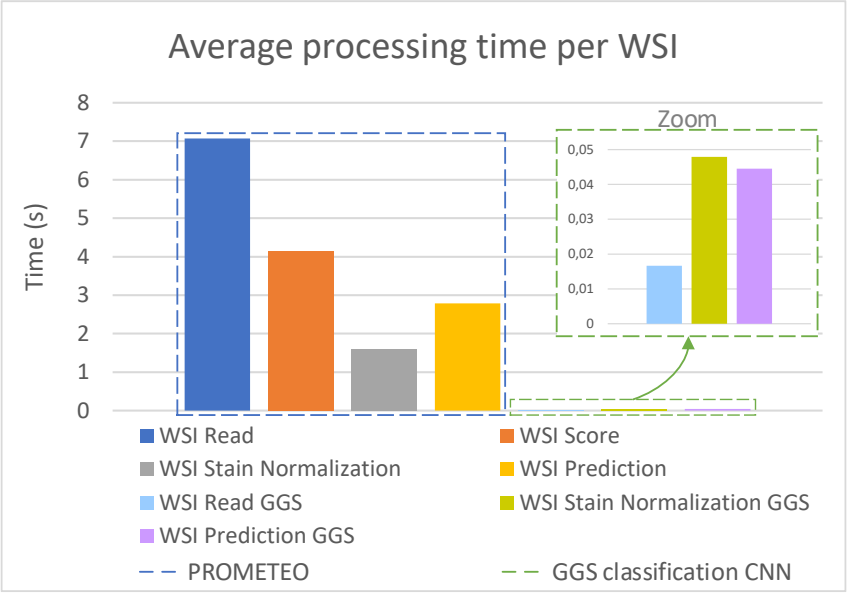

FIGURE 6.11: Average WSI processing time (in seconds) for each of the steps performed by the global CAD system. The first four steps correspond to the PCa detection CNN, and the remaining three to the Gleason pattern recognition system, which is zoomed in for better visualization.

All the information and results regarding the average execution time and standard deviation for each of the steps both per patch and per WSI are detailed in Table 6.4, following the same color scheme used in Figures 6.10, 6.11 and 6.12.

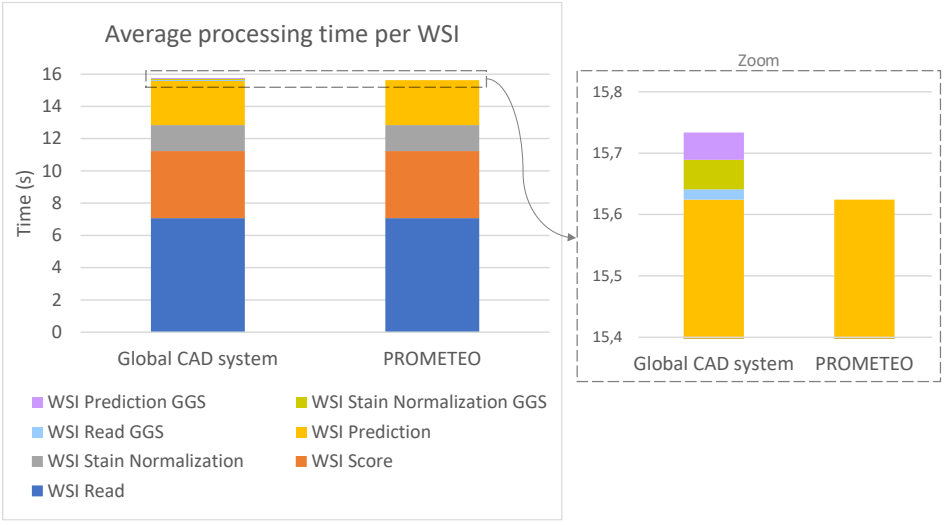

FIGURE 6.12: Combined average WSI processing time (in seconds) for the global CAD system compared to PROMETEO. The top part of the chart is zoomed in for better visualization of the difference between both.

TABLE 6.4: Global CAD system performance evaluation results. The average and standard deviation of the execution times (in seconds) are shown for each of the steps considered, both per patch and per WSI. The execution times were calculated using the barebone computer described in this section.

			Patch	**WSI**
PROMETEO	Read	Avg	0,000363	7,069082
		Std	0,000661	1,19244
	Score	Avg	0,003568	4,154812
		Std	0,000502	1,215193
	Stain Normalization	Avg	0,001377	1,61293
		Std	0,000685	0,909821
	Prediction	Avg	0,002321	2,787565
		Std	0,004105	0,82306
GGS CNN	Read GGS	Avg	0,002510123	0,016657
		Std	0,001031709	0,0315
	Stain Normalization GGS	Avg	0,007862633	0,047923
		Std	0,002607513	0,098615
	Prediction GGS	Avg	0,003468	0,044554
		Std	0,021389	0,083266
	Total		**0,02147 ± 0,030981**	**15,73352 ± 4,353897**

Chapter 7

Application of CAD systems to other medical image analysis: COVID-19 case of use

7.1 Introduction

The world has changed due to the emergence of the new virus from the coronavirus family (COVID-19), which was declared as a pandemic a few months after its appearance (Zheng et al., 2020). COVID-19 has had a high impact in every single field, such as in health, education, economics, labor, among others. Since the pandemic situation started during the development of this Thesis, we decided to contribute to the fight against COVID-19. To this end, the knowledge that was already acquired after carrying out the experiments presented in previous chapters, including the application of CNNs and CAMs, was applied for this purpose.

COVID-19 is the disease caused by the new severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Coronaviruses are an extensive family of viruses that may affect both humans and animals, causing problems to the respiratory system (World Health Organization, 2021). Other well-known human coronaviruses identified in the past are SARS-CoV and MERS-CoV, which have had around 8100 and 2500 confirmed cases, with a case fatality rate of around 9.2% and 37.1%, respectively (Cui et al., 2019; Momattin et al., 2019).

Globally, as of June 30, 2021, the number of confirmed deaths worldwide caused by COVID-19 has almost surpassed 3.5 million, with more than 215 countries, areas or territories affected and a total of more than 168 million confirmed cases (World Health Organization, 2021), plunging humanity into a severe state of fear whose outcome is still unknown.

COVID-19 spreads through direct contact with respiratory drops produced when an infected person coughs, speaks, sneezes, or even breaths. These droplets can enter the host's body through the nose, mouth and tear ducts, giving a

passage to the mucous membranes in the throat. Through this process of transmission, the virus reaches the respiratory tract.

Some studies have confirmed angiotensin receptor 2 (ACE2) as the receptor through which the virus enters the respiratory mucosa (Singhal, 2020). After reaching the lung alveoli, the virus starts to replicate itself, increasing the viral load within the host cell. Type II pneumocytes are destroyed, releasing specific inflammatory mediators. As a result, lungs might become inflamed, which could lead to pneumonia in the most severe cases (Hussain et al., 2020).

An early detection of COVID-19 is crucial to control outbreaks and prevent the virus from spreading. Current diagnostic tests for COVID-19 include reverse-transcription polymerase chain reaction (RT-PCR), real-time RT-PCR (rRT-PCR), and reverse-transcription loop-mediated isothermal amplification (RT-LAMP) (Zhai et al., 2020).

Patients who have been exposed to the virus and present severe symptoms, could still get a negative result in the RT-PCR test (Zhai et al., 2020; Kucirka et al., 2020; Li et al., 2020a). Therefore, in these cases, COVID-19 should be diagnosed with medical imaging techniques, such as X-ray or chest Computed Tomography (CT) (Zhai et al., 2020). Although CT has proved to be one of the most precise diagnostic methods for COVID-19 (Fang et al., 2020), it has some important limitations, including around 70$\times$ higher ionizing radiation than X-ray (Lin, 2010), its high cost, and the fact that it cannot be performed as a bedside test (Lu et al., 2020). Therefore, it is not routinely used in COVID-19 diagnosis (Self et al., 2013). Moreover, it is not suitable for monitoring the evolution of specific cases, particularly in critically ill patients. On the other hand, X-ray is a less sensitive modality in the detection of COVID-19 compared to CT with a reported baseline sensitivity of 69% (Jacobi et al., 2020). However, X-ray is a cheaper and faster alternative, and it is also available in most hospitals. Therefore, X-ray will likely be the primary imaging modality used for COVID-19 diagnosis and management. With high clinical suspicion for COVID-19 infection, positive X-ray findings can obviate the need for CT scanning (Jacobi et al., 2020). However, it is important to consider that these techniques may present limitations to particular patients such as pregnant women, since they could cause harm to unborn children (Ratnapalan et al., 2008).

The most common findings that radiologists look for when analyzing X-ray images for COVID-19 diagnosis are multiple, patchy, sub-segmental or segmental ground glass density shadows in both lungs (Jin et al., 2020). This process could be automated using CAD systems in order to aid experts when making a decision (Civit-Masot et al., 2020).

Although COVID-19 is a very recent topic, many researchers have carried out studies to find solutions during this crisis. In Shi et al., 2020, a review of recent AI-based CAD systems for COVID-19 diagnosis in CT and X-ray images is presented; however, since this study is based on X-ray images, the authors

only focused on the latter. Ghoshal et al. presented a Bayesian CNN to estimate the diagnosis uncertainty in COVID-19 prediction, distinguishing between COVID-19 and non-COVID-19 cases (other types of pneumonia and healthy patients), obtaining an accuracy of 92.9% (Ghoshal and Tucker, 2020). Narin et al. performed a binary classification between COVID-19 and normal cases comparing different DL models, achieving 98.0% accuracy with the ResNet50 model in the best case (Narin et al., 2020). Zhang et al. presented a ResNet-based model to classify COVID-19 (0.952 AUC), highlighting the pneumonia-affected regions by applying the Gradient-weighted Class Activation Mapping (Grad-CAM) method (Zhang et al., 2020). Finally, Wang et al. proposed a Deep CNN to classify between COVID-19, non-COVID-19 (distinguishing between viral and bacterial) and normal cases, obtaining an accuracy of 83.5% (Wang and Wong, 2020).

These studies achieved accurate solutions to fight against the COVID-19 pandemic. However, they have some limitations that should be considered. First of all, they used small datasets with less than 400 COVID-19 X-ray images in total in the best case. To validate the system, some of them only used 10 X-ray images for the COVID-19 class. Moreover, studies that proposed not only COVID-19 detection, but also locating affected areas of the lungs, did not include any ground truth comparison or medical supervision with the obtained results.

In this experiment, a DL-based CAD system, named COVID-XNet, which classifies between COVID-19 and normal frontal X-ray chest images is presented. The network focuses on specific regions of the lungs to perform a prediction and detect whether the patient has COVID-19. The output of the system can be then represented in a heatmap-like plot by performing the CAM algorithm, which locates the affected areas. The high reliability obtained in the results, which were supervised by a lung specialist, indicates that this system could be used to aid expert radiologists as a screening test for COVID-19 diagnosis in patients with clinical manifestations, helping them throughout this stage and to overcome this situation.

7.2 Materials and Methods

7.2.1 Dataset

In this work, different publicly-available datasets were taken into account to build a diverse and large collection of chest X-ray images from healthy patients and COVID-19 cases. Both posteroanterior (PA) and anteroposterior (AP) projections were considered, discarding lateral X-ray images.

For the COVID-19 class, chest X-ray images were obtained from the BIMCV-COVID19+ dataset, provided by the Medical Imaging Databank of the Valencia Region (BIMCV) (Vayá et al., 2020) and from the COVID-19 image data collection

from Cohen et al., 2020. On the other hand, for healthy patients, images were obtained from the PadChest dataset, also provided by BIMCV (Bustos et al., 2019). From the total number of images labeled as normal from this dataset, around the first 10% were used, since, otherwise, the imbalance between the number of cases for COVID-19 and healthy patients would have been very high. Therefore, a total of 2589 images from 1429 patients and 4337 images from 4337 patients were considered for COVID-19 and normal classes, respectively.

7.2.2 Methods

7.2.2.1 Preprocessing step

In order to reduce the large variability of these images, a preprocessing step, which included different techniques, was applied to the original images.

Firstly, all images were converted to grayscale. Since the original images came from different hospitals, and, consequently, from different X-ray machines, a histogram matching process was applied to every image, taking one of them as a reference (Gonzales and Woods, 2002). Therefore, all images in the dataset were similar in terms of histogram distribution.

Then, rib shadows were suppressed from the X-ray images with a pretrained autoencoder model developed by Chuong M. Huynh, which is publicly available in GitHub[1]. This makes it easier for the network to focus on relevant information within the lungs. Rib shadows suppression has been applied in other works related to lung cancer, pulmonary nodules and pneumonia detection in chest radiography, proving to be a useful approach to aid radiologists and machine learning systems when diagnosing lung related diseases (Qin et al., 2018; Soleymanpour, Pourreza, et al., 2011; Oda et al., 2009; Gordienko et al., 2018; Gusarev et al., 2017).

After this process, a contrast enhancement method called Contrast Limited Adaptive Histogram Equalization (CLAHE) was used to improve local contrast and enhance the image definition (Reza, 2004).

Figure 7.1 shows the whole preprocessing phase, where each algorithm's output is presented for three different examples.

7.2.2.2 Convolutional Neural Network

After applying the preprocessing step, the images obtained were used as input to a custom CNN model that was trained from scratch to classify between COVID-19 and normal cases. This model consists of the following set of layers: 5 convolutions, 4 max poolings, a GAP and a final softmax layer (see Figure 7.2). This custom model was selected by means of an Exhaustive Grid Search over the number of layers and kernel's sizes, prioritizing accuracy and computational

[1] www.github.com/hmchuong/ML-BoneSuppression

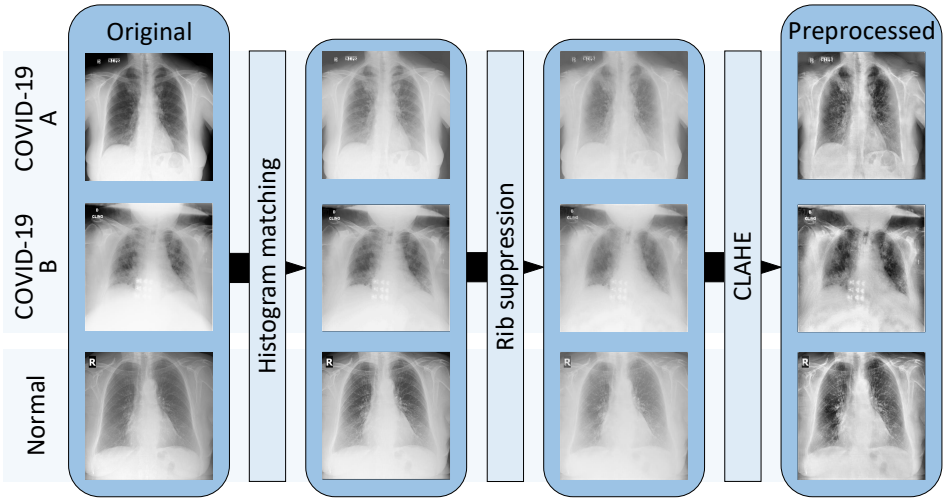

FIGURE 7.1: Preprocessing flowchart describing the different steps to obtain the final images for the dataset. COVID-19 A and B correspond to images from BIMCV-COVID19 and the COVID-19 image data collection from Cohen et al., respectively.

complexity. Layers were explored from 1 up to the maximum number of layers that allowed having features of over 1x1 pixels before the GAP layer. Kernel sizes were explored from 3x3 up to 11x11. The best configuration over all the different possibilities was the one selected.

7.2.2.3 Training and validating the system

To ensure that our model was generalizing well with data that it had not been trained with, a stratified 5-fold cross-validation was used to train and validate the network with all the images. This allowed obtaining more robust results on the different evaluation metrics. For this approach, the images were split in 5 different sets, taking into account that images from the same patient were only present in a single set. Then, the model was trained five times, where four out of the five sets were used for training and the remaining one for validation. Therefore, for each fold, 80% of the dataset was considered when training the system and the remaining 20% when validating it.

In order to increase the variability of the dataset, data augmentation techniques were used. Random rotations (up to a maximum of 15 degrees), width shift (up to 20%), height shift (up to 20%), shear (up to 20%), zoom (up to 20%) and horizontal flips were applied to the input images.

To reduce the computational complexity, input images were resized to 128×128 pixels. Since the dataset used in this experiment is unbalanced (i.e.,

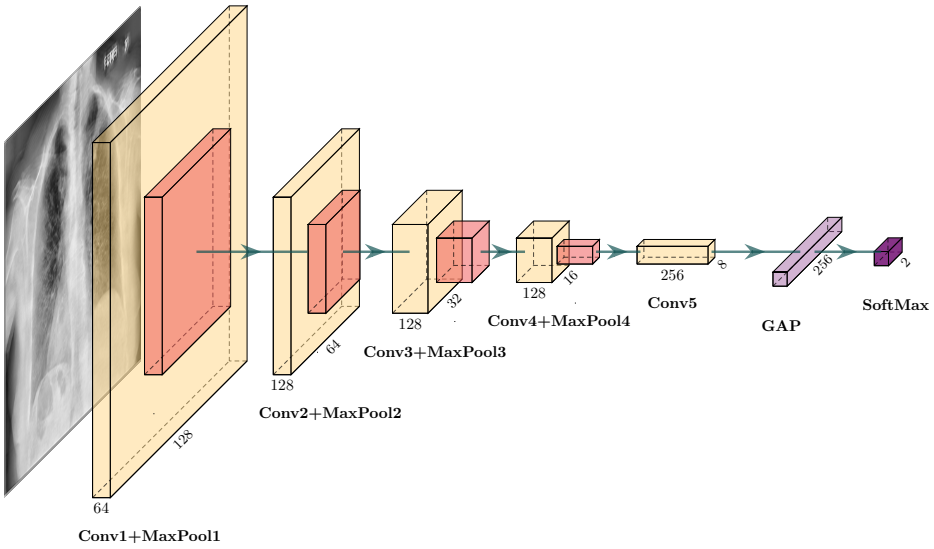

FIGURE 7.2: Diagram of COVID-XNet. It consists of 5 convolutional layers (Conv), 4 max pooling layers (MaxPool), a GAP layer and a softmax layer. Conv1, Conv2 and Conv3 use 5×5 kernel size, while Conv4 and Conv5 use 3×3. All MaxPool layers use 2×2 kernels.

there are more images corresponding to the normal class than to COVID-19), the class weight function was applied accordingly in Keras in order to give more importance to the COVID-19 class when training the network.

The following metrics were used to measure the performance of the COVID-19 detection system: sensitivity (Equation 3.7), specificity (Equation 3.8), precision (Equation 3.6) and F1-score (Equation 3.9); since the dataset is unbalanced, the balanced accuracy was also used. In addition, the AUCs of the ROC were calculated.

7.2.2.4 Postprocessing step

Since the relevant information for COVID-19 detection in frontal X-ray images only lies inside the lung area (Gordienko et al., 2018; Soleymanpour, Pourreza, et al., 2011), lungs were segmented from the original images in order to discard surrounding regions. With this process, CAMs (see section 6.2.2.3) only focus on this area and, therefore, clearer results in terms of visualization of the system's output are provided. This lung segmentation step was performed using a CNN based on the U-Net model (Ronneberger et al., 2015), which was used to

solve the Radiological Society of North America (RSNA®) Pneumonia Detection Challenge[2].

7.3 Results

7.3.1 Quantitative evaluation

The results achieved by the network after training and validating the CNN using the 5-fold cross-validation are summarized in Table 7.1. The ROC curve for each of the cross-validation folds are presented in Figure 7.3, which also reports their corresponding AUC values.

TABLE 7.1: Cross-validation results for each of the folds, where sensitivity, specificity, precision, F1-score, AUC and balanced accuracy are reported. The average of these metrics over the different folds are also shown.

Fold test	Actual classes	Predicted classes		Sensitivity	Specificity	Precision	F1-score	AUC	Balanced accuracy
		Normal	COVID-19						
1st fold	Normal	851	16	96.71%	98.15%	96.89%	96.8%	0.997	97.43%
		98.15%	1.85%						
	COVID-19	17	499						
		3.29%	96.71%						
2nd fold	Normal	839	28	94.00%	96.77%	94.54%	94.27%	0.990	95.38%
		96.77%	3.23%						
	COVID-19	31	485						
		6%	94%						
3rd fold	Normal	834	33	93.02%	96.19%	93.57%	93.29%	0.989	94.61%
		96.19%	3.81%						
	COVID-19	36	480						
		6.98%	93.02%						
4th fold	Normal	815	52	88.95%	94.00%	89.82%	89.39%	0.976	91.48%
		94%	6%						
	COVID-19	57	459						
		11.05%	88.95%						
5th fold	Normal	839	30	90%	96.55%	93.98%	91.94%	0.986	93.27%
		96.55%	3.45%						
	COVID-19	52	468						
		10%	90%						
Average	Normal	96.33%	3.67%	92.53%	96.33%	93.76%	93.14%	0.988	94.43%
	COVID-19	7.47%	92.53%						

As can be seen, the results achieved demonstrate that the system is able to generalize well, obtaining similar and stable results across the different folds. Each of these sets achieved balanced accuracies greater than 91%, and AUC values above 0.97, which confirms that the system is very reliable when performing the classification. After calculating the average of the metrics obtained over all the different cross-validation folds, the system achieved 92.53% sensitivity, 96.33% specificity, 93.76% precision, 93.14% F1-score, 94.43% balanced accuracy and an AUC value of 0.988.

[2]www.kaggle.com/eduardomineo/u-net-lung-segmentation-montgomery-shenzhen

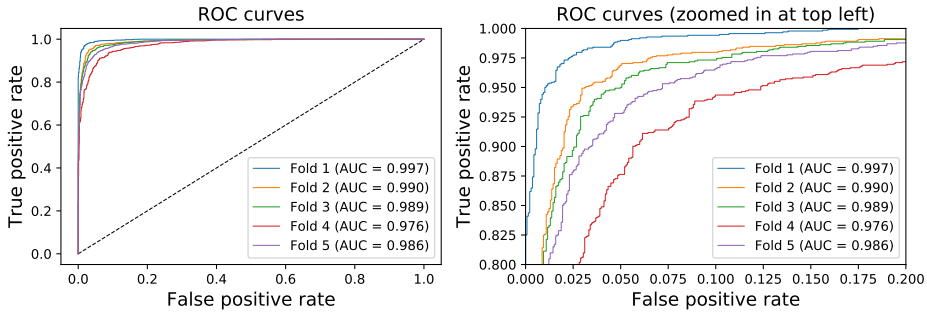

FIGURE 7.3: Left: ROC curve for each cross-validation set. Right: zoomed in at top left. AUC values are shown in the legend.

7.3.2 Qualitative evaluation

As introduced in section 6.2.2.3, CAMs are used to visualize what the network is focusing on represented in a heatmap when performing the classification. Figure 7.4 shows different input images and their corresponding CAM heatmaps obtained with COVID-XNet. The most relevant information that the network considered when performing the prediction for the COVID-19 class is highlighted in red, while regions that were not relevant for COVID-19 detection (considered as normal) are presented in dark blue.

The examples shown in Figure 7.4 present different cases that correspond to true positives (A–H), true negatives (I–K) and false positives (L). The heatmaps obtained for the true positive cases were compared to the ground truth descriptions provided in the datasets in order to verify whether the system was highlighting the correct regions inside the lung area. It is important to mention that these results were also validated by a lung specialist.

The ground truth corresponding to Figure 7.4-A reports patchy ground-glass opacities in right upper and lower lung zones and patchy consolidation in left middle to lower lung zones. Furthermore, several calcified granulomas were incidentally noted in the left upper lung zone. Figure 7.4-B shows a right interstitial paracardiac thickening with a tendency to cavitation in its most cranial portion, along with a mild right hilar enlargement. Figure 7.4-.C presents consolidations in the base of the right hemithorax and an interstitial pattern that affects most of that lung. Moreover, a small pseudonodular consolidation is presented in the left paracardiac region which could suggest another affected area. In Figure 7.4-D, the ground truth describes the presence of right upper lobe opacity. The report for Figure 7.4-E details the existence of alveolar infiltrates in the right upper and lower lobe, and also in the left parahilar area. As can be seen in the corresponding heatmaps for these cases (A–E), relevant areas described in the ground truths were detected by the system. In the case shown in Figure 7.4-F,

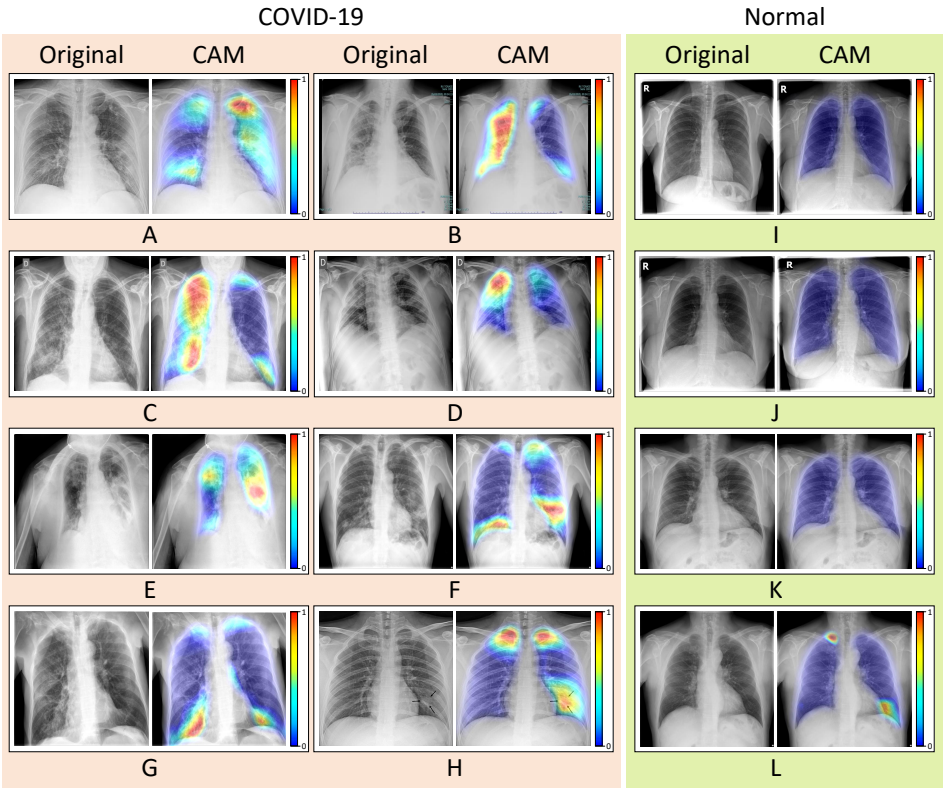

FIGURE 7.4: CAM obtained for the COVID-19 class together with their corresponding original images. Images A–H represent COVID-19 cases, while I–L correspond to healthy patients. CAMs are represented with heatmaps, where the most relevant regions for COVID-19 detection are highlighted in red.

the patient is reported to present opacities in the base of the right lung and in the left middle and lower lung zones. The output heatmap matches this description, along with a smaller region in the left upper area which is not mentioned in the report. Lower and middle to upper right lobe consolidations are reported in Figure 7.4-G, together with a mild small consolidation in the left lower lobe. In this case, the system was not able to detect consolidations in the middle to upper right lung area. Finally, the ground truth of Figure 7.4-H reports COVID-19 pneumonia manifesting as a single nodular lesion. The AP chest radiograph shows a single nodular consolidation (black arrows) in the left lower lung zone. In the latter case, the system detected consolidations marked by ground truth arrows, but it also mistakenly highlighted upper areas in both lungs.

For normal cases (I–L), the system did not detect any relevant COVID-19 area, except for Figure 7.4-L, where two small regions were highlighted.

These results prove that, even when training the system with a large unbalanced dataset obtained from different sources, the proposed custom model is learning specific characteristics and patterns appropriately.

The results obtained in this chapter have been published in Applied Sciences journal as "COVID-XNet: A Custom Deep Learning System to Diagnose and Locate COVID-19 in Chest X-ray Images" (Duran-Lopez et al., 2020b). More details of this publication can be found in Appendix D.

Chapter 8

Discussion

In this chapter, the main contributions achieved in the previous chapters are discussed, comparing them to other state-of-the-art approaches, along with future works that could be investigated as the next steps of this Thesis.

As presented in Chapter 3, a DL-based CAD system, called PROMETEO, was developed for discriminating between malignant and normal regions in WSIs. PROMETEO was trained, validated and tested with 100×100 patches extracted from PCa tissue images. These patches were filtered with a novel patch extraction and scoring algorithm, which removed unwanted areas, such as pen marks and external agents, and then normalized using Reinhard's stain normalization.

As can be clearly seen from the results obtained in Section 3.2.2.3, both the stain-normalized version of the dataset and the original one achieved very high recognition rates on the PCa detection task. However, the second one performed slightly better than the former in the 3-fold cross-validation step and in the final test, which can be due to different reasons. First of all, color differences could be one of the factors that the network learns for distinguishing between malignant and normal patches, since the H&E stain makes malignant regions tend to a more purple-like color. Then, normalizing all patches to a target color could imply losing relevant information for the classification.

However, when testing with different hospitals whose WSIs were not used to train the system and which present different color variations, the results changed when comparing both approaches. In that case, the mean specificity of both is still around 95%. However, when looking at the standard deviation of the specificity, the difference is clearer: the stain-normalized was more stable while achieving almost the same result. This could be caused by the fact that, thanks to the normalization, the patches were more homogeneous in terms of color, and the network was able to extract more relevant features based on the cell structures (which are more complex to detect than color differences) during the training phase. Hence, the stain-normalization could make the system more robust and stable to images from new hospitals and scanners where color variations exist. This idea was confirmed when performing the Student's t-test over the

percentage of malignant area of the tissue predicted by the CNN for normal and malignant WSIs without applying stain-normalization, which showed that there was no statistically significant difference between the two classes for two out of the three external sources. These results were also studied in Otálora et al., 2019, where the authors concluded that training a deep CNN with stain-normalized images did not improve the results and, in some cases, they were worse than the baseline. However, the authors stated that this technique improved the generalization of the CNN for classification tasks using digital pathology images. Our results also confirm this idea regarding the application of stain normalization to PCa histopathological images.

State-of-the-art works, such as Kwak and Hewitt, 2017, Litjens et al., 2016, Campanella et al., 2019 and Ström et al., 2020, also performed a classification between normal and malignant tissue. However, since these works performed the classification and obtained the metrics at a different level (core-level and slide-level), results cannot be directly compared to the ones achieved by PROMETEO. Moreover, those works use different datasets, which also does not allow a strict and fair comparison with the results obtained in the aforementioned experiment. Therefore, PROMETEO was compared with different well-known pre-trained models, which were tested on the dataset we used. Some of them obtained similar results after a fine-tuning process, as presented in Table 3.5. However, in terms of performance, due to the higher complexity of these models compared to our proposal, the average time that they take to predict a single patch and a single WSI is higher than that of PROMETEO, as presented in Chapter 4. This means that, when using our network, the CAD system would be able to process more WSIs in the same amount of time than any of the other models that were tested, as well as to achieve a slightly higher accuracy. These pre-trained models were much faster to train than our network, since it was trained from scratch, and their accuracies are close to ours. However, since the training process is a step that only has to be done once, it is worth to have a longer training process in order to obtain a lighter computational algorithm with better accuracy for its production phase (predicting every new WSI that is processed in a hospital).

A comprehensive comparison with other state-of-the-art works in terms of the number of operations (OPS) performed by the network was also performed. Based on the number and size of the layers, PROMETEO performs around 350 MOPS (10^6 OPS) per patch. Other works that present high accuracies for a binary classification task in WSIs, such as Toro et al., 2017; Campanella et al., 2019 and Ström et al., 2020, use well-known models that, based on Canziani et al., 2016, perform more than 1 GOPS (10^9 OPS) per input patch. The custom model presented in Litjens et al., 2016 needs to perform more than 660 MOPS per patch. PROMETEO outperforms other state-of-the-art works in terms of computational complexity for a binary classification task in prostate histopathological images between normal and malignant WSIs.

The use of transfer learning in CNNs has become a commonplace technique

for medical image analysis. Most of the current research focuses on using this approach for avoiding the problem of having to design, train and validate a custom CNN model for a specific task. This has proved to achieve state-of-the-art results in many different fields and has also accelerated the process of training a custom CNN from scratch (Zhuang et al., 2019). However, when using this technique, very deep CNNs are commonly considered, which, as demonstrated, leads to a higher computational cost when predicting an input image and, therefore, a slower processing time. Some specific tasks could benefit from designing shallower custom CNN models from scratch, such as DL-based PCa screening, providing a faster response to the pathologists in order to help them in this laborious process. With the increase on the number of cases and the mortality produced by PCa, this factor could become even more relevant in the future.

As an alternative, cloud computing has provided powerful computational resources to big data processing and ML models (Zhang et al., 2019). Recent works have focused on accelerating CNN-based medical image processing tasks by using cloud solutions. While it is true that processing images using GPU and Tensor Processing Units (TPUs) in the cloud is faster than in any local edge-computing device, there is an aspect that is not commonly taken into account when stating this fact: the time required to upload the image to the cloud. This depends on many factors and it is not easy to predict. Moreover, when digitizing histological images, scanners store them in a local hard drive using around 1 GB for each of them. As an example, with an upload speed of 300 Mbps, it would take more than 27 seconds in ideal conditions just for uploading the WSI to the cloud, which is more than the time it would take to fully process the image in a local platform using the system proposed in this Thesis.

The results obtained in the PCa detection task were expanded in order to perform a Gleason pattern recognition over the malignant patches, providing more relevant information to pathologists. The results obtained for the Gleason pattern recognition task presented in Chapter 6 demonstrate that a custom CAD system as the one proposed could help reducing inter-observer variability (around 30%, as reported in Berg et al., 2011) when analyzing PCa in WSIs. Interesting results can also be drawn from the confusion matrix presented in Figure 6.5. The proposed Gleason pattern classification CAD system has a higher uncertainty between Gleason patterns 3 and 4 (83.30% and 75.90% accuracy, respectively). On the other hand, pattern 5 is correctly classified in most of the cases (95.36% accuracy). Furthermore, when the proposed system performed predictions over patches labeled as Gleason pattern 3, it mistakenly predicted 16.49% of them as pattern 4 (only 0.21% of pattern 3 cases were assigned with pattern 5). The same happened when the system did not classified Gleason pattern 4 patches correctly, which, in all of the cases, were classified as pattern 3 (0% of the errors committed for patches labeled as Gleason pattern 4 corresponded to the system classifying them as pattern 5). This higher

uncertainty between Gleason patterns 3 and 4 is also present in a traditional scenario when expert pathologists analyze PCa histopathological images, as reported by Arvaniti et al., 2018b and Salmo, 2015.

As mentioned in Section 1.2.5.1, the authors in Berg et al., 2011 proposed the re-evaluation of PCa patients with a primary low Gleason score diagnosed. This double effort, which would require different pathologists to analyze the same images in order to agree with each other and reduce the aforementioned variability, is not a very efficient practice, and CAD systems as the one proposed and developed could be very useful for this task. This does not mean that the proposed system could replace pathologists, since these kind of DL-based approaches are still far from being perfect. However, they could definitely be used to perform tedious and time-consuming tasks, which would only need to be supervised and reviewed afterwards, reducing pathologists' fatigue and, therefore, error probability.

The evaluated Gleason pattern recognition system, which achieved 84.60% balanced accuracy, leads to developing a global CAD system combining all the different parts presented in this Thesis in order to perform a complete report of input WSIs. This way, an input image is classified as either malignant or normal by means of the novel W&D network presented in Chapter 5, and, for the former case, a heatmap highlighting the malignant regions of the tissue and their corresponding Gleason pattern is generated. The different networks and processes that are combined to form the global CAD system allow deeper feature extraction while having a low latency in terms of execution time, with the Gleason pattern classification CNN representing less than 1% of the total amount of time needed. Future works will continue expanding this idea by including a new block at the output of the current system in order to assign a slide-level GGS score to the input WSI, following a similar approach than the one considered for performing patch-aggregation with the W&D model over the results obtained at patch level with PROMETEO. This way, the complete report given by the system would provide both global and spatial GGS information, which would be of great use to expert pathologists. Current efforts focus on improving the results obtained in the Gleason pattern recognition CNN by adding more variability to the dataset. For this purpose, we are collaborating closely with a team of pathologists from the Hospital Clínic Barcelona in order to validate the output heatmaps obtained by the system and to get a higher amount of labeled WSIs.

All the aforementioned algorithms and processes were applied to COVID-19 detection in order to contribute to the fight against this pandemic situation (Chapter 7). The proposed system could be useful as a screening test for COVID-19 diagnosis in combination with patients' clinical manifestations and/or laboratory results to discard severe cases and decide whether the patient should be hospitalized. The performance of the system when predicting new unseen images shows that the model generalizes well, proving that COVID-XNet could be the first step for developing a universal CAD system for COVID-19 diagnosis

in X-ray images. This study, by no means, presents a solution that is currently ready for its production phase. More tests and improvements should be performed before considering the use of any DL solution in hospitals. COVID-XNet was never conceived as a replacement for human radiologists, but as a tool to aid them and contribute to the fight against COVID-19.

To conclude, this Thesis has presented novel contributions in the field of medical image analysis and, in particular, in PCa diagnosis in histopathological images. Some of the results improve previous state-of-the-art solutions in terms of performance (Appendices A and B), and others are implementations that, to the best of the author and the advisors' knowledge, have been developed for the first time for this purpose (Appendices C and D), becoming important contributions that have been published (or are under review) in high impact factor journals. Some new lines of research have been opened with these results, allowing numerous possibilities for future works, some of which have been presented in this discussion.

Chapter 9

Conclusions

In this Thesis, which has been presented throughout this document, the following contributions and conclusions are highlighted:

- A study of PCa has been carried out, including the histopathology, the epidemiology, the main causes and the diagnosis procedure.
- An in-depth study of DL algorithms and CNNs has been performed, along with the current most widely used frameworks for designing, training and evaluating these kind of networks.
- A novel CNN-based CAD system for discriminating between malignant and normal regions in WSIs has been developed. A set of WSIs annotated by pathologists were obtained to create the dataset, which was used to train, validate and test a custom 9-layer CNN, called PROMETEO. This network is able to generate a heatmap of the input WSI, indicating the regions that the network detected as malignant. A novel patch scoring algorithm, which removed unwanted patches, was also developed. In addition, the impact of applying a stain-normalization algorithm was studied, proving to be relevant for the generalization of the model when predicting WSIs from different sources due to the color variations produced by the H&E stain.
- A novel benchmark has been designed in order to measure the processing and prediction time of a CNN architecture for a PCa screening task. The proposed benchmark was used to perform a comprehensive evaluation of PROMETEO on different computing platforms to measure the impact that their hardware components have on the WSI processing time. The benchmark was run with different state-of-the-art CNN models, comparing them in terms of average prediction time. The proposed model outperforms other widely-used state-of-the-art CNN architectures, while achieving better results on the same dataset.
- A novel ML-based algorithm has been developed in order to classify PCa WSIs as normal or malignant at global slide level based on a previous patch-level classification. For this purpose, a W&D model, which combines both

linear model components (wide) and neural network components (deep) was designed. Different processed features were extracted, which were then used as input to the proposed W&D model. The proposed model was compared with other state-of-the-art methods, proving that the W&D network performs better in terms of accuracy, sensitivity, F1 score and AUC. To the best of the author's knowledge, this was the first time that a W&D network was used to medical image analysis.

- A custom CNN model has been designed, trained and evaluated for classifying between the three main different Gleason patterns. The proposed system generates CAMs from the input patches, highlighting the tissue regions that the system considered as relevant to perform the classification, allowing for a more precise heatmap of the malignant areas in the WSI.
- A global PCa diagnosis CAD system has been developed, combining the different subsystems developed in this Thesis. The proposed system performs a sequence of processes to the input WSI, acting first as a screening method and, for those diagnosed as malignant, it reports the output of the Gleason pattern recognition system with the corresponding heatmap. After measuring the performance of the global CAD system when analyzing a WSI, the impact of the Gleason pattern classification system represents a negligible amount of time of the whole process. To the best of the author's knowledge, this aspect makes the proposed global CAD system the fastest among the state-of-the-art PCa detection studies.
- In order to contribute to the fight against COVID-19, the knowledge acquired when developing the aforementioned CAD systems for PCa diagnosis was used for a COVID-19 detection task in chest X-ray images. A custom CAD system, called COVID-XNet, was developed. X-ray images were preprocessed with different methods in order to enhance the relevant information. CAMs were used to visualize the relevant features that the system considered for COVID-19 detection. The output of the system for the evaluation set was compared and verified with its corresponding ground truths and validated by a lung specialist.

Chapter 10

Bibliography

Abadi, Martín, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. (2016). "Tensorflow: A system for large-scale machine learning". In: *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pp. 265–283.

Abraham, Ajith (2005). "Artificial neural networks". In: *Handbook of measuring system design*.

Agarap, Abien Fred (2018). "Deep learning using rectified linear units (relu)". In: *arXiv preprint arXiv:1803.08375*.

Ahmad, Jamil, Haleem Farman, and Zahoor Jan (2019). "Deep learning methods and applications". In: *Deep Learning: Convergence to Big Data Analytics*. Springer, pp. 31–42.

Alpaydin, Ethem (2016). *Machine learning: the new AI*. MIT press.

Altaf, Fouzia, Syed MS Islam, Naveed Akhtar, and Naeem Khalid Janjua (2019). "Going deep in medical image analysis: concepts, methods, challenges, and future directions". In: *IEEE Access* 7, pp. 99540–99572.

American Society of Clinical Oncology (2019). "World Cancer Day 2019: emphasis on early detection". In: *Florham Park, New Jersey: ASCO*.

Amin, Mahul B and Satish K Tickoo (2016). *Diagnostic Pathology: Genitourinary E-Book*. Elsevier Health Sciences.

Anwar, Syed Muhammad, Muhammad Majid, Adnan Qayyum, Muhammad Awais, Majdi Alnowami, and Muhammad Khurram Khan (2018). "Medical image analysis using convolutional neural networks: a review". In: *Journal of medical systems* 42.11, pp. 1–13.

Arel, Itamar, Derek C Rose, and Thomas P Karnowski (2010). "Deep machine learning-a new frontier in artificial intelligence research [research frontier]". In: *IEEE computational intelligence magazine* 5.4, pp. 13–18.

Arvaniti, Eirini, Kim Fricker, Michael Moret, Niels Rupp, Thomas Hermanns, Christian Fankhauser, Norbert Wey, Peter Wild, Jan Hendrik Rüschoff, and Manfred Claassen (2018a). "Replication Data for: Automated Gleason grading of prostate cancer tissue microarrays via deep learning." Version V1.

In: DOI: 10.7910/DVN/OCYCMP. URL: https://doi.org/10.7910/DVN/OCYCMP.

Arvaniti, Eirini, Kim S Fricker, Michael Moret, Niels Rupp, Thomas Hermanns, Christian Fankhauser, Norbert Wey, Peter J Wild, Jan H Rueschoff, and Manfred Claassen (2018b). "Automated Gleason grading of prostate cancer tissue microarrays via deep learning". In: *Scientific Reports* 8.1, pp. 1–11.

Aunan, Jan R, William C Cho, and Kjetil Søreide (2017). "The biology of aging and cancer: a brief overview of shared and divergent molecular hallmarks". In: *Aging and disease* 8.5, p. 628.

Baldin, Rosimeri Kuhl Svoboda, Raul Alberto Anselmi Júnior, Marina Azevedo, Ana Paula Martins Sebastião, Mário Montemor, Luiz Fernando Tullio, Luiz Felipe de Paula Soares, and Lúcia de Noronha (2015). "Interobserver variability in histological diagnosis of serrated colorectal polyps". In: *Journal of Coloproctology (Rio de Janeiro)* 35.4, pp. 193–197.

Bassett, Danielle S and Michael S Gazzaniga (2011). "Understanding complexity in the human brain". In: *Trends in cognitive sciences* 15.5, pp. 200–209.

Bast Jr, Robert C, James F Holland, and Emil Frei Iii (2010). *Holland-Frei cancer medicine 8*. Vol. 8. PMPH-USA.

Bechis, Seth K, Peter R Carroll, and Matthew R Cooperberg (2011). "Impact of age at diagnosis on prostate cancer treatment and survival". In: *Journal of Clinical Oncology* 29.2, p. 235.

Bennett, Betsy D and William A Gardner (1988). "Prostatic crystalloids". In: *JAMA* 260.15, pp. 2287–2287.

Berg, Kasper Drimer, Birgitte Grønkaer Toft, Martin Andreas Røder, Klaus Brasso, Ben Vainer, and Peter Iversen (2011). "Prostate needle biopsies: interobserver variation and clinical consequences of histopathological re-evaluation". In: *Apmis* 119.4-5, pp. 239–246.

Bergstra, James, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio (2010). "Theano: a CPU and GPU math expression compiler". In: *Proceedings of the Python for scientific computing conference (SciPy)*. Vol. 4. 3. Austin, TX, pp. 1–7.

Bhavsar, Anil and Sadhna Verma (2014). "Anatomic imaging of the prostate". In: *BioMed research international* 2014.

Björndahl, Lars and Ulrik Kvist (2011). "A model for the importance of zinc in the dynamics of human sperm chromatin stabilization after ejaculation in relation to sperm DNA vulnerability". In: *Systems biology in reproductive medicine* 57.1-2, pp. 86–92.

Borley, Nigel and Mark R Feneley (2009). "Prostate cancer: diagnosis and staging". In: *Asian journal of andrology* 11.1, p. 74.

Bostwick, David G, Harry B Burke, Daniel Djakiew, Susan Euling, Shuk-mei Ho, Joseph Landolph, Howard Morrison, Babasaheb Sonawane, Tiffany Shifflett, David J Waters, et al. (2004). "Human prostate cancer risk factors". In: *Cancer:*

Interdisciplinary International Journal of the American Cancer Society 101.S10, pp. 2371–2490.

Bray, Freddie, Jacques Ferlay, Isabelle Soerjomataram, Rebecca L Siegel, Lindsey A Torre, and Ahmedin Jemal (2018). "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries". In: *CA: a cancer journal for clinicians* 68.6, pp. 394–424.

Breiman, Leo (2001). "Random forests". In: *Machine learning* 45.1, pp. 5–32.

Brown, Guy C (2015). "Living too long: the current focus of medical research on increasing the quantity, rather than the quality, of life is damaging our health and harming the economy". In: *EMBO reports* 16.2, pp. 137–141.

Bulten, Wouter, Hans Pinckaers, Hester van Boven, Robert Vink, Thomas de Bel, Bram van Ginneken, Jeroen van der Laak, Christina Hulsbergen-van de Kaa, and Geert Litjens (2020). "Automated deep-learning system for Gleason grading of prostate cancer using biopsies: a diagnostic study". In: *The Lancet Oncology*.

Bustos, Aurelia, Antonio Pertusa, Jose-Maria Salinas, and Maria de la Iglesia-Vayá (2019). "Padchest: A large chest X-ray image dataset with multi-label annotated reports". In: *arXiv preprint arXiv:1901.07441*. https://bimcv.cipf.es/bimcv-projects/padchest/ (accessed on June 30, 2021).

Caini, Saverio, Sara Gandini, Maria Dudas, Viviane Bremer, Ettore Severi, and Alin Gherasim (2014). "Sexually transmitted infections and prostate cancer risk: a systematic review and meta-analysis". In: *Cancer epidemiology* 38.4, pp. 329–338.

Campanella, Gabriele, Matthew G Hanna, Luke Geneslaw, Allen Miraflor, Vitor Werneck Krauss Silva, Klaus J Busam, Edi Brogi, Victor E Reuter, David S Klimstra, and Thomas J Fuchs (2019). "Clinical-grade computational pathology using weakly supervised deep learning on whole slide images". In: *Nature Medicine* 25.8, pp. 1301–1309.

Canziani, Alfredo, Adam Paszke, and Eugenio Culurciello (2016). "An analysis of deep neural network models for practical applications". In: *arXiv preprint arXiv:1605.07678*.

Castro, Elena and Rosalind Eeles (2012). "The role of BRCA1 and BRCA2 in prostate cancer". In: *Asian journal of andrology* 14.3, p. 409.

Chan, John KC (2014). "The wonderful colors of the hematoxylin–eosin stain in diagnostic surgical pathology". In: *International journal of surgical pathology* 22.1, pp. 12–32.

Chang, Chin-Chen, Hong-Hao Chen, Yeun-Chung Chang, Ming-Yang Yang, Chung-Ming Lo, Wei-Chun Ko, Yee-Fan Lee, Kao-Lang Liu, and Ruey-Feng Chang (2017). "Computer-aided diagnosis of liver tumors on computed tomography images". In: *Computer methods and programs in biomedicine* 145, pp. 45–51.

Chen, Ni and Qiao Zhou (2016). "The evolving Gleason grading system". In: *Chinese Journal of Cancer Research* 28.1, p. 58.

Cheng, Heng-Tze, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. (2016). "Wide & deep learning for recommender systems". In: *Proceedings of the 1st workshop on deep learning for recommender systems*, pp. 7–10.

Chollet, François et al. (2015). *Keras*. https://keras.io.

Chollet, François (2017). "Xception: Deep learning with depthwise separable convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258.

Ciompi, Francesco, Oscar Geessink, Babak Ehteshami Bejnordi, Gabriel Silva De Souza, Alexi Baidoshvili, Geert Litjens, Bram Van Ginneken, Iris Nagtegaal, and Jeroen Van Der Laak (2017). "The importance of stain normalization in colorectal tissue classification with convolutional networks". In: *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. IEEE, pp. 160–163.

Civit-Masot, Javier, Francisco Luna-Perejón, Manuel Domínguez Morales, and Anton Civit (2020). "Deep Learning system for COVID-19 diagnosis aid using X-ray pulmonary images". In: *Applied Sciences* 10.13, p. 4640.

Cohen, Joseph Paul, Paul Morrison, and Lan Dao (2020). "COVID-19 image data collection". In: *arXiv 2003.11597*. https://github.com/ieee8023/covid-chestxray-dataset (accessed on June 30, 2021).

Cui, Jie, Fang Li, and Zheng-Li Shi (2019). "Origin and evolution of pathogenic coronaviruses". In: *Nature Reviews Microbiology* 17.3, pp. 181–192.

De Magalhães, João Pedro (2013). "How ageing processes influence cancer". In: *Nature Reviews Cancer* 13.5, pp. 357–365.

Deng, J., W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei (2009). "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255.

Dixon, J, P Chow, and J Gosling (1999). "Anatomy and function of the prostate gland". In: *Textbook of prostatitis*, pp. 33–46.

Doi, Kunio (2005). "Current status and future potential of computer-aided diagnosis in medical imaging". In: *The British journal of radiology* 78.suppl_1, s3–s19.

Doi, Kunio (2007). "Computer-aided diagnosis in medical imaging: historical review, current status and future potential". In: *Computerized medical imaging and graphics* 31.4-5, pp. 198–211.

Doyle, Scott, Mark Hwang, Kinsuk Shah, Anant Madabhushi, Michael Feldman, and John Tomaszeweski (2007). "Automated grading of prostate cancer using architectural and textural image features". In: *2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. IEEE, pp. 1284–1287.

Duran-Lopez, Lourdes, Juan P Dominguez-Morales, Antonio Felix Conde-Martin, Saturnino Vicente-Diaz, and Alejandro Linares-Barranco (2020a).

"PROMETEO: A CNN-Based Computer-Aided Diagnosis System for WSI Prostate Cancer Detection". In: *IEEE Access* 8, pp. 128613–128628.

Duran-Lopez, Lourdes, Juan P. Dominguez-Morales, Antonio Rios-Navarro, Daniel Gutierrez-Galan, Angel Jimenez-Fernandez, Saturnino Vicente-Diaz, and Alejandro Linares-Barranco (2021). "Performance Evaluation of Deep Learning-Based Prostate Cancer Screening Methods in Histopathological Images: Measuring the Impact of the Model's Complexity on Its Processing Speed". In: *Sensors* 21.4. ISSN: 1424-8220. DOI: 10.3390/s21041122. URL: https://www.mdpi.com/1424-8220/21/4/1122.

Duran-Lopez, Lourdes, Juan Pedro Dominguez-Morales, Jesús Corral-Jaime, Saturnino Vicente-Diaz, and Alejandro Linares-Barranco (2020b). "COVID-XNet: a custom deep learning system to diagnose and locate COVID-19 in chest X-ray images". In: *Applied Sciences* 10.16, p. 5683.

El-Baz, Ayman, Garth M Beache, Georgy Gimel'farb, Kenji Suzuki, Kazunori Okada, Ahmed Elnakib, Ahmed Soliman, and Behnoush Abdollahi (2013). "Computer-aided diagnosis systems for lung cancer: challenges and methodologies". In: *International journal of biomedical imaging* 2013.

Epstein, Jonathan I, William C Allsbrook Jr, Mahul B Amin, Lars L Egevad, ISUP Grading Committee, et al. (2005). "The 2005 International Society of Urological Pathology (ISUP) consensus conference on Gleason grading of prostatic carcinoma". In: *The American journal of surgical pathology* 29.9, pp. 1228–1242.

Ewing, Charles M, Anna M Ray, Ethan M Lange, Kimberly A Zuhlke, Christiane M Robbins, Waibhav D Tembe, Kathleen E Wiley, Sarah D Isaacs, Dorhyun Johng, Yunfei Wang, et al. (2012). "Germline mutations in HOXB13 and prostate-cancer risk". In: *New England Journal of Medicine* 366.2, pp. 141–149.

Fang, Yicheng, Huangqi Zhang, Jicheng Xie, Minjie Lin, Lingjun Ying, Peipei Pang, and Wenbin Ji (2020). "Sensitivity of chest CT for COVID-19: comparison to RT-PCR". In: *Radiology*, p. 200432.

Faust, Oliver, U Rajendra Acharya, Vidya K Sudarshan, Ru San Tan, Chai Hong Yeong, Filippo Molinari, and Kwan Hoong Ng (2017). "Computer aided diagnosis of coronary artery disease, myocardial infarction and carotid atherosclerosis using ultrasound images: a review". In: *Physica Medica* 33, pp. 1–15.

Ferlay, Jacques, M Colombet, I Soerjomataram, C Mathers, DM Parkin, M Piñeros, A Znaor, and F Bray (2019). "Estimating the global cancer incidence and mortality in 2018: GLOBOCAN sources and methods". In: *International journal of cancer* 144.8, pp. 1941–1953.

Fleshner, Neil, P Scott Bagnell, Laurence Klotz, and Vasundara Venkateswaran (2004). "Dietary fat and prostate cancer". In: *The Journal of urology* 171.2, S19–S24.

Ghoshal, Biraja and Allan Tucker (2020). "Estimating uncertainty and interpretability in deep learning for coronavirus (COVID-19) detection". In: *arXiv preprint arXiv:2003.10769*.

Gomes, Douglas S, Simone S Porto, Débora Balabram, and Helenice Gobbi (2014). "Inter-observer variability between general pathologists and a specialist in breast pathology in the diagnosis of lobular neoplasia, columnar cell lesions, atypical ductal hyperplasia and ductal carcinoma in situ of the breast". In: *Diagnostic pathology* 9.1, pp. 1–9.

Gonzales, Rafael C and Richard E Woods (2002). "Digital Image Processing". In: Prentice Hall: Upper Saddle River, NJ, USA.

Gordetsky, Jennifer and Jonathan Epstein (2016). "Grading of prostatic adenocarcinoma: current state and prognostic implications". In: *Diagnostic pathology* 11.1, pp. 1–8.

Gordienko, Yu, Peng Gang, Jiang Hui, Wei Zeng, Yu Kochura, Oleg Alienin, Oleksandr Rokovyi, and Sergii Stirenko (2018). "Deep learning with lung segmentation and bone shadow exclusion techniques for chest X-ray analysis of lung cancer". In: *Advances in Intelligent Systems and Computing, Proceedings of the International Conference on Computer Science, Engineering and Education Applications*. Springer: Berlin/Heidelberg, Germany, pp. 638–647.

Gulland, Anne (2016). *Global life expectancy increases by five years*.

Gurcan, Metin N, Laura E Boucheron, Ali Can, Anant Madabhushi, Nasir M Rajpoot, and Bulent Yener (2009). "Histopathological image analysis: A review". In: *IEEE reviews in biomedical engineering* 2, pp. 147–171.

Gusarev, Maxim, Ramil Kuleev, Adil Khan, Adin Ramirez Rivera, and Asad Masood Khattak (2017). "Deep learning models for bone suppression in chest radiographs". In: *2017 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pp. 1–7.

Guyton, Arthur and John Hall (2006). *Textbook of medical physiology, 11th*.

Haenlein, Michael and Andreas Kaplan (2019). "A brief history of artificial intelligence: On the past, present, and future of artificial intelligence". In: *California management review* 61.4, pp. 5–14.

Hakama, Matti, Anssi Auvinen, Nicholas E Day, and Anthony B Miller (2007). "Sensitivity in cancer screening". In: *Journal of Medical Screening* 14.4, pp. 174–177.

Hassan, AU, G Hassan, and ZR Zubeida (2013). "Aims and objectives of histological studies of prostate". In: *Universal Journalof Clinical Medicine* 1.2, pp. 13–21.

Hawkes, Nigel (2019). *Cancer survival data emphasise importance of early diagnosis*.

Hayman, Samantha (1999). "The mcculloch-pitts model". In: *IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No. 99CH36339)*. Vol. 6. IEEE, pp. 4438–4439.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Hiom, SC (2015). "Diagnosing cancer earlier: reviewing the evidence for improving cancer survival." In: *British journal of cancer* 112, S1–5.

Hou, Le, Dimitris Samaras, Tahsin M Kurc, Yi Gao, James E Davis, and Joel H Saltz (2016). "Patch-based convolutional neural network for whole slide tissue image classification". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2424–2433.

Howard, Andrew G, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam (2017). "Mobilenets: Efficient convolutional neural networks for mobile vision applications". In: *arXiv preprint arXiv:1704.04861*.

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger (2017). "Densely connected convolutional networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708.

Humphrey, Peter A (2017). "Histopathology of prostate cancer". In: *Cold Spring Harbor perspectives in medicine* 7.10, a030411.

Hussain, Azhar, Jasndeep Kaler, Elsa Tabrez, Salma Tabrez, and Shams SM Tabrez (2020). "Novel COVID-19: A comprehensive review of transmission, manifestation, and pathogenesis". In: *Cureus* 12.5.

Huttenlocher, Peter R (2009). *Neural plasticity*. Harvard University Press.

Jacobi, Adam, Michael Chung, Adam Bernheim, and Corey Eber (2020). "Portable chest X-ray in coronavirus disease-19 (COVID-19): A pictorial review". In: *Clinical Imaging*.

Jalalian, Afsaneh, Syamsiah BT Mashohor, Hajjah Rozi Mahmud, M Iqbal B Saripan, Abdul Rahman B Ramli, and Babak Karasfi (2013). "Computer-aided detection/diagnosis of breast cancer in mammography and ultrasound: a review". In: *Clinical imaging* 37.3, pp. 420–426.

Jia, Yangqing, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell (2014). "Caffe: Convolutional architecture for fast feature embedding". In: *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 675–678.

Jiang, Liangxiao, Zhihua Cai, Dianhong Wang, and Siwei Jiang (2007). "Survey of improving k-nearest-neighbor for classification". In: *Fourth international conference on fuzzy systems and knowledge discovery (FSKD 2007)*. Vol. 1. IEEE, pp. 679–683.

Jin, Ying-Hui, Lin Cai, Zhen-Shun Cheng, Hong Cheng, Tong Deng, Yi-Pin Fan, Cheng Fang, Di Huang, Lu-Qi Huang, Qiao Huang, et al. (2020). "A rapid advice guideline for the diagnosis and treatment of 2019 novel coronavirus (2019-nCoV) infected pneumonia (standard version)". In: *Military Medical Research* 7.1, p. 4.

Khan, Salman, Hossein Rahmani, Syed Afaq Ali Shah, and Mohammed Bennamoun (2018). "A guide to convolutional neural networks for computer vision". In: *Synthesis Lectures on Computer Vision* 8.1, pp. 1–207.

Kiefer, Jack, Jacob Wolfowitz, et al. (1952). "Stochastic estimation of the maximum of a regression function". In: *The Annals of Mathematical Statistics* 23.3, pp. 462–466.

Kim, Tae-Yun, Jaebum Son, and Kwang-Gi Kim (2011). "The recent progress in quantitative medical image analysis for computer aided diagnosis systems". In: *Healthcare informatics research* 17.3, p. 143.

Krieger, Nancy, Robert A Hiatt, Richard W Sagebiel, Wallace H Clark Jr, and Martin C Mihm Jr (1994). "Inter-observer variability among pathologists' evaluation of malignant melanoma: effects upon an analytic study". In: *Journal of clinical epidemiology* 47.8, pp. 897–902.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25, pp. 1097–1105.

Kucirka, Lauren M, Stephen A Lauer, Oliver Laeyendecker, Denali Boon, and Justin Lessler (2020). "Variation in false-negative rate of reverse transcriptase polymerase chain reaction–based SARS-CoV-2 tests by time since exposure". In: *Annals of Internal Medicine*.

Kwak, Jin Tae and Stephen M Hewitt (2017). "Nuclear architecture analysis of prostate cancer via convolutional neural networks". In: *IEEE Access* 5, pp. 18526–18533.

Kwast, Th H Van der, C Lopes, C Santonja, CG Pihl, I Neetens, Pekka Martikainen, S Di Lollo, L Bubendorf, and RF Hoedemaeker (2003). "Guidelines for processing and reporting of prostatic needle biopsies". In: *Journal of clinical pathology* 56.5, pp. 336–340.

Lavery, Anita, Roger S. Kirby, and Simon Chowdhury (2016). "Prostate cancer". In: *Medicine* 44.1. Oncology, pp. 47–51. ISSN: 1357-3039. DOI: https://doi.org/10.1016/j.mpmed.2015.10.001. URL: https://www.sciencedirect.com/science/article/pii/S1357303915002662.

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.

Lee, Christine H, Oluyemi Akin-Olugbade, and Alexander Kirschenbaum (2011). "Overview of prostate anatomy, histology, and pathology." In: *Endocrinology and metabolism clinics of North America* 40.3, pp. 565–75.

Lessells, Alastair M, Rodney A Burnett, S Rosalind Howatson, Stephen Lang, Frederick D Lee, Kathryn M McLaren, E Robert Nairn, Simon A Ogston, Alistair J Robertson, John G Simpson, et al. (1997). "Observer variability in the histopathological reporting of needle biopsy specimens of the prostate". In: *Human Pathology* 28.6, pp. 646–649.

Leung, Henry and Simon Haykin (1991). "The complex backpropagation algorithm". In: *IEEE Transactions on signal processing* 39.9, pp. 2101–2104.

Li, Wenyuan, Jiayun Li, Karthik V Sarma, King Chung Ho, Shiwen Shen, Beatrice S Knudsen, Arkadiusz Gertych, and Corey W Arnold (2018). "Path

R-CNN for prostate cancer diagnosis and Gleason grading of histological images". In: *IEEE Transactions on Medical Imaging* 38.4, pp. 945–954.

Li, Yafang, Lin Yao, Jiawei Li, Lei Chen, Yiyan Song, Zhifang Cai, and Chunhua Yang (2020a). "Stability issues of RT-PCR testing of SARS-CoV-2 for hospitalized patients clinically diagnosed with COVID-19". In: *Journal of medical virology*.

Li, Yuanzhi, Colin Wei, and Tengyu Ma (2019). "Towards explaining the regularization effect of initial large learning rate in training neural networks". In: *arXiv preprint arXiv:1907.04595*.

Li, Zewen, Wenjie Yang, Shouheng Peng, and Fan Liu (2020b). "A survey of convolutional neural networks: analysis, applications, and prospects". In: *arXiv preprint arXiv:2004.02806*.

Lin, Eugene C (2010). "Radiation risk from medical imaging". In: *Mayo Clinic Proceedings*. Vol. 85. 12. Elsevier: Amsterdam, The Nederlands, pp. 1142–1146.

Litjens, Geert, Clara I Sánchez, Nadya Timofeeva, Meyke Hermsen, Iris Nagtegaal, Iringo Kovacs, Christina Hulsbergen-Van De Kaa, Peter Bult, Bram Van Ginneken, and Jeroen Van Der Laak (2016). "Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis". In: *Scientific Reports* 6, p. 26286.

Lozano, Rafael, Mohsen Naghavi, Kyle Foreman, Stephen Lim, Kenji Shibuya, Victor Aboyans, Jerry Abraham, Timothy Adair, Rakesh Aggarwal, Stephanie Y Ahn, et al. (2012). "Global and regional mortality from 235 causes of death for 20 age groups in 1990 and 2010: a systematic analysis for the Global Burden of Disease Study 2010". In: *The lancet* 380.9859, pp. 2095–2128.

Lu, Wuzhu, Shushan Zhang, Binghui Chen, Jiaxin Chen, Jianzhong Xian, Yuhong Lin, Hong Shan, and Zhong Zhen Su (2020). "A clinical study of noninvasive assessment of lung lesions in patients with coronavirus disease-19 (COVID-19) by bedside ultrasound". In: *Ultraschall in der Medizin-European Journal of Ultrasound*.

Magee, Derek, Darren Treanor, Doreen Crellin, Mike Shires, Katherine Smith, Kevin Mohee, and Philip Quirke (2009). "Colour normalisation in digital histopathology images". In: *Proc. Optical Tissue Image analysis in Microscopy, Histopathology and Endoscopy (MICCAI Workshop)*. Vol. 100. Citeseer, pp. 100–111.

McLean, M, J Srigley, D Banerjee, P Warde, and Y Hao (1997). "Interobserver variation in prostate cancer Gleason scoring: are there implications for the design of clinical trials and treatment strategies?" In: *Clinical Oncology* 9.4, pp. 222–225.

Mesquita, Jolien M Bueno-de, DSA Nuyten, J Wesseling, H van Tinteren, SC Linn, and MJ van De Vijver (2010). "The impact of inter-observer variation in pathological assessment of node-negative breast cancer on clinical risk

assessment and patient selection for adjuvant systemic treatment". In: *Annals of oncology* 21.1, pp. 40–47.

Minsky, Marvin and Seymour A Papert (2017). *Perceptrons: An introduction to computational geometry*. MIT press.

Momattin, Hisham, Anfal Y Al-Ali, and Jaffar A Al-Tawfiq (2019). "A Systematic Review of therapeutic agents for the treatment of the Middle East Respiratory Syndrome Coronavirus (MERS-CoV)". In: *Travel medicine and infectious disease* 30, pp. 9–18.

Narin, Ali, Ceren Kaya, and Ziynet Pamuk (2020). "Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks". In: *arXiv preprint arXiv:2003.10849*.

National Collaborating Centre for Cancer (UK) (2008). "Prostate cancer: diagnosis and treatment". In:

Nir, Guy, Soheil Hor, Davood Karimi, Ladan Fazli, Brian F Skinnider, Peyman Tavassoli, Dmitry Turbin, Carlos F Villamil, Gang Wang, R Storey Wilson, et al. (2018). "Automatic grading of prostate cancer in digitized histopathology images: Learning from multiple experts". In: *Medical image analysis* 50, pp. 167–180.

Oda, Seitaro, Kazuo Awai, Kenji Suzuki, Yumi Yanaga, Yoshinori Funama, Heber MacMahon, and Yasuyuki Yamashita (2009). "Performance of radiologists in detection of small pulmonary nodules on chest radiographs: effect of rib suppression with a massive-training artificial neural network". In: *American Journal of Roentgenology* 193.5, W397–W402.

Otálora, Sebastian, Manfredo Atzori, Vincent Andrearczyk, Amjad Khan, and Henning Müller (2019). "Staining invariant features for improving generalization of deep convolutional neural networks in computational pathology". In: *Frontiers in Bioengineering and Biotechnology* 7, p. 198.

Otsu, N. (1979). "A Threshold Selection Method from Gray-Level Histograms". In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1, pp. 62–66.

Ott, JJ, A Ullrich, and AB Miller (2009). "The importance of early symptom recognition in the context of early detection and cancer survival". In: *European Journal of Cancer* 45.16, pp. 2743–2748.

Ozten, Nur, Katherine Vega, Joachim Liehr, Xi Huang, Lori Horton, Ercole L Cavalieri, Eleanor G Rogan, and Maarten C Bosland (2019). "Role of estrogen in androgen-induced prostate carcinogenesis in NBL rats". In: *Hormones and Cancer* 10.2, pp. 77–88.

Pantanowitz, Liron (2010). "Digital images and the future of digital pathology". In: *Journal of pathology informatics* 1.

Parent, Marie-Élise, Mark S Goldberg, Dan L Crouse, Nancy A Ross, Hong Chen, Marie-France Valois, and Alexandre Liautaud (2013). "Traffic-related air pollution and prostate cancer risk: a case–control study in Montreal, Canada". In: *Occupational and environmental medicine* 70.7, pp. 511–518.

Parloff, Roger (2016). "Why deep learning is suddenly changing your life". In: *Fortune. New York: Time Inc.*

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. (2019). "Pytorch: An imperative style, high-performance deep learning library". In: *arXiv preprint arXiv:1912.01703*.

Pienta, Kenneth J and Peggy S Esper (1993). "Risk factors for prostate cancer". In: *Annals of internal medicine* 118.10, pp. 793–803.

Pierorazio, Phillip M, Patrick C Walsh, Alan W Partin, and Jonathan I Epstein (2013). "Prognostic G leason grade grouping: data based on the modified G leason scoring system". In: *BJU international* 111.5, pp. 753–760.

Poola, Indrasen (2017). "How artificial intelligence in impacting real life everyday". In: *International Journal for Advance Research and Development* 2.10, pp. 96–100.

Presti Jr, Joseph C (2003). "Prostate biopsy: how many cores are enough?" In: *Urologic Oncology: Seminars and Original Investigations*. Vol. 21. 2. Elsevier, pp. 135–140.

Qin, Chunli, Demin Yao, Yonghong Shi, and Zhijian Song (2018). "Computer-aided detection in chest radiography based on artificial intelligence: a survey". In: *Biomedical engineering online* 17.1, p. 113.

Ramírez, J, JM Górriz, F Segovia, R Chaves, D Salas-Gonzalez, M López, I Álvarez, and P Padilla (2010). "Computer aided diagnosis system for the Alzheimer's disease based on partial least squares and random forest SPECT image classification". In: *Neuroscience letters* 472.2, pp. 99–103.

Ratnapalan, Savithiri, Yedidia Bentur, and Gideon Koren (2008). "Doctor, will that X-ray harm my unborn child?" In: *Cmaj* 179.12, pp. 1293–1296.

Rawla, Prashanth (2019). "Epidemiology of prostate cancer". In: *World journal of oncology* 10.2, p. 63.

Razzak, Muhammad Imran, Saeeda Naz, and Ahmad Zaib (2018). "Deep learning for medical image processing: Overview, challenges and the future". In: *Classification in BioApps*, pp. 323–350.

Reiner, Bruce I and Elizabeth Krupinski (2012). "The insidious problem of fatigue in medical imaging practice". In: *Journal of digital imaging* 25.1, pp. 3–6.

Reinhard, Erik, Michael Adhikhmin, Bruce Gooch, and Peter Shirley (2001). "Color transfer between images". In: *IEEE Computer Graphics and Applications* 21.5, pp. 34–41.

Ren, Jian, Evita Sadimin, David J Foran, and Xin Qi (2017). "Computer aided analysis of prostate histopathology images to support a refined Gleason grading system". In: *Medical Imaging 2017: Image Processing*. Vol. 10133. International Society for Optics and Photonics, p. 101331V.

Renshaw, Andrew A (1997). "Adequate tissue sampling of prostate core needle biopsies". In: *American journal of clinical pathology* 107.1, pp. 26–29.

Reza, Ali M (2004). "Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement". In: *Journal of VLSI signal processing systems for signal, image and video technology* 38.1, pp. 35–44.

Ritchie, Hannah and Max Roser (2018). "Causes of Death". In: *Our World in Data*. https://ourworldindata.org/causes-of-death.

Rojas, Raul (1996). "The backpropagation algorithm". In: *Neural networks*. Springer, pp. 149–182.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-net: Convolutional networks for biomedical image segmentation". In: *Lecture Notes in Computer Science, Proceedings of the International Conference on Medical image computing and computer-assisted intervention*. Springer: Berlin/Heidelberg, Germany, pp. 234–241.

Rosenblatt, Frank (1958). "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6, p. 386.

Roser, Max, Esteban Ortiz-Ospina, and Hannah Ritchie (2013). "Life expectancy". In: *Our World in Data*.

Roy, Kaushiki, Debapriya Banik, Debotosh Bhattacharjee, and Mita Nasipuri (2019). "Patch-based system for Classification of Breast Histology images using deep learning". In: *Computerized Medical Imaging and Graphics* 71, pp. 90–103.

Roy, Santanu, Alok kumar Jain, Shyam Lal, and Jyoti Kini (2018). "A study about color normalization methods for histopathology images". In: *Micron* 114, pp. 42–61.

Ruddon, Raymond W (2007). *Cancer biology*. Oxford University Press.

Ruderman, Daniel L, Thomas W Cronin, and Chuan-Chin Chiao (1998). "Statistics of cone responses to natural images: implications for visual coding". In: *Journal of the Optical Society of America A* 15.8, pp. 2036–2045.

Saitoh, Hiroshi, Miho Hida, Takao Shimbo, Kazuyoshi Nakamura, Jun Yamagata, and Takeshi Satoh (1984). "Metastatic patterns of prostatic cancer: correlation between sites and number of organs involved". In: *Cancer* 54.12, pp. 3078–3084.

Salmo, Emile N (2015). "An audit of inter-observer variability in Gleason grading of prostate cancer biopsies: The experience of central pathology review in the North West of England". In: *Integr Cancer Sci Ther* 2.2, pp. 104–106.

Santos, Marcel Koenigkam, José Raniery Ferreira Júnior, Danilo Tadao Wada, Ariane Priscilla Magalhães Tenório, Marcello Henrique Nogueira Barbosa, and Paulo Mazzoncini de Azevedo Marques (2019). "Artificial intelligence, machine learning, computer-aided diagnosis, and radiomics: advances in imaging towards to precision medicine". In: *Radiologia brasileira* 52.6, pp. 387–396.

Sathya, Ramadass and Annamma Abraham (2013). "Comparison of supervised and unsupervised learning algorithms for pattern classification". In: *International Journal of Advanced Research in Artificial Intelligence* 2.2, pp. 34–38.

Schalken, Jack A and Geert van Leenders (2003). "Cellular and molecular biology of the prostate: stem cell biology". In: *Urology* 62.5, pp. 11–20.

Schlegel, Daniel (2015). "Deep machine learning on Gpu". In: *University of Heidelber-Ziti* 12.

Self, Wesley H, D Mark Courtney, Candace D McNaughton, Richard G Wunderink, and Jeffrey A Kline (2013). "High discordance of chest X-ray and computed tomography for detection of pulmonary opacities in ED patients: implications for diagnosing pneumonia". In: *The American journal of emergency medicine* 31.2, pp. 401–405.

Selman, Steven H (2011). "The McNeal prostate: a review". In: *Urology* 78.6, pp. 1224–1228.

Shang, Wenling, Kihyuk Sohn, Diogo Almeida, and Honglak Lee (2016). "Understanding and improving convolutional neural networks via concatenated rectified linear units". In: *international conference on machine learning*. PMLR, pp. 2217–2225.

Sharma, Sagar (2017). "Activation functions in neural networks". In: *towards data science* 6.

Shi, Feng, Jun Wang, Jun Shi, Ziyan Wu, Qian Wang, Zhenyu Tang, Kelei He, Yinghuan Shi, and Dinggang Shen (2020). "Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19". In: *IEEE Reviews in Biomedical Engineering*.

Shung, K Kirk, Michael Smith, and Benjamin MW Tsui (2012). *Principles of medical imaging*. Academic Press.

Silva-Rodríguez, Julio, Adrián Colomer, María A Sales, Rafael Molina, and Valery Naranjo (2020). "Going deeper through the Gleason scoring scale: An automatic end-to-end system for histology prostate grading and cribriform pattern detection". In: *Computer Methods and Programs in Biomedicine* 195, p. 105637.

Simon, Haykin (1999). *Neural networks: a comprehensive foundation*. Prentice hall.

Simonyan, Karen and Andrew Zisserman (2014). "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556*.

Singhal, Tanu (2020). "A review of coronavirus disease-2019 (COVID-19)". In: *The Indian Journal of Pediatrics*, pp. 1–6.

Soleymanpour, Elaheh, Hamid Reza Pourreza, et al. (2011). "Fully automatic lung segmentation and rib suppression methods to improve nodule detection in chest radiographs". In: *Journal of medical signals and sensors* 1.3, p. 191.

Specht, Donald F et al. (1991). "A general regression neural network". In: *IEEE transactions on neural networks* 2.6, pp. 568–576.

Stearns, Stephen C and Jacob C Koella (2008). *Evolution in health and disease*. Oxford University Press.

Stratton, Michael R, Peter J Campbell, and P Andrew Futreal (2009). "The cancer genome". In: *Nature* 458.7239, pp. 719–724.

Ström, Peter, Kimmo Kartasalo, Henrik Olsson, Leslie Solorzano, Brett Delahunt, Daniel M Berney, David G Bostwick, Andrew J Evans, David J Grignon, Peter A Humphrey, et al. (2020). "Artificial intelligence for diagnosis and grading of prostate cancer in biopsies: a population-based, diagnostic study". In: *The Lancet Oncology*.

Sutcliffe, Siobhan (2010). "Sexually transmitted infections and risk of prostate cancer: review of historical and emerging hypotheses". In: *Future oncology* 6.8, pp. 1289–1311.

Swallow, Thomas, Simon Chowdhury, and Roger S Kirby (2012). "Cancer of the prostate gland". In: *Medicine* 40.1, pp. 10–13.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi (2017). "Inception-v4, inception-resnet and the impact of residual connections on learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1.

Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich (2015). "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9.

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna (2016). "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826.

Taylor, Marcia L, AG Mainous, Brian J Wells, et al. (2005). "Prostate cancer and sexually transmitted diseases: a meta-analysis". In: *FAMILY MEDICINE-KANSAS CITY-* 37.7, p. 506.

Toro, Oscar Jiménez del, Manfredo Atzori, Sebastian Otálora, Mats Andersson, Kristian Eurén, Martin Hedlund, Peter Rönnquist, and Henning Müller (2017). "Convolutional neural networks for an automatic classification of prostate tissue slides with high-grade Gleason score". In: *Proc. SPIE Medical Imaging 2017: Digital Pathology*. Vol. 10140. International Society for Optics and Photonics, 101400O.

Vayá, Maria de la Iglesia, Jose Manuel Saborit, Joaquim Angel Montell, Antonio Pertusa, Aurelia Bustos, Miguel Cazorla, Joaquin Galant, Xavier Barber, Domingo Orozco-Beltrán, Francisco Garcia, et al. (2020). "BIMCV COVID-19+: a large annotated dataset of RX and CT images from COVID-19 patients". In: *arXiv preprint arXiv:2006.01174*. https://bimcv.cipf.es/bimcv-projects/bimcv-covid19/ (accessed on June 30, 2021).

Velonas, Vicki M, Henry H Woo, Cristobal G dos Remedios, and Stephen J Assinder (2013). "Current status of biomarkers for prostate cancer". In: *International journal of molecular sciences* 14.6, pp. 11034–11060.

Verbrugge, Lois M, Deborah L Wingard, and Haworth Continuing Features Submission (1987). "Sex differentials in health and mortality". In: *Women & health* 12.2, pp. 103–145.

Vesal, Sulaiman, Nishant Ravikumar, AmirAbbas Davari, Stephan Ellmann, and Andreas Maier (2018). "Classification of breast cancer histology images using transfer learning". In: *International Conference Image Analysis and Recognition*. Springer, pp. 812–819.

Wagner Jr, Henry N and Peter S Conti (1991). "Advances in medical imaging for cancer diagnosis and treatment". In: *Cancer* 67.S4, pp. 1121–1128.

Wang, Linda and Alexander Wong (2020). "Covid-net: A tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images". In: *arXiv preprint arXiv:2003.09871*.

Wang, Lipo (2005). *Support vector machines: theory and applications*. Vol. 177. Springer Science & Business Media.

Weinreb, Jeffrey C, Jelle O Barentsz, Peter L Choyke, Francois Cornud, Masoom A Haider, Katarzyna J Macura, Daniel Margolis, Mitchell D Schnall, Faina Shtern, Clare M Tempany, et al. (2016). "PI-RADS prostate imaging–reporting and data system: 2015, version 2". In: *European urology* 69.1, pp. 16–40.

Wetter, Axel and Thomas J. Vogl (2016). "The Prostate". In: *Diagnostic and Interventional Radiology*. Ed. by Thomas J. Vogl, Wolfgang Reith, and Ernst J. Rummeny. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 917–927. ISBN: 978-3-662-44037-7. DOI: 10.1007/978-3-662-44037-7_31. URL: https://doi.org/10.1007/978-3-662-44037-7_31.

World Health Organization (2018). "Latest global cancer data: Cancer burden rises to 18.1 million new cases and 9.6 million cancer deaths in 2018". In: *International Agency for Research on Cancer. Geneva: World Health Organization*.

World Health Organization (2021). "Coronavirus disease (COVID-19) pandemic". In: https://www.who.int/emergencies/diseases/novel-coronavirus-2019 (accessed on June 30, 2021).

Wu, CP and FL Gu (1991). "The prostate in eunuchs." In: *Progress in clinical and biological research* 370, pp. 249–255.

Yan, Chaoyang, Kazuaki Nakane, Xiangxue Wang, Yao Fu, Haoda Lu, Xiangshan Fan, Michael D Feldman, Anant Madabhushi, and Jun Xu (2020). "Automated Gleason grading on prostate biopsy slides by statistical representations of homology profile". In: *Computer Methods and Programs in Biomedicine* 194, p. 105528.

Yanase, Juri and Evangelos Triantaphyllou (2019). "A systematic survey of computer-aided diagnosis in medicine: Past and present developments". In: *Expert Systems with Applications* 138, p. 112821.

Yegnanarayana, Bayya (2009). *Artificial neural networks*. PHI Learning Pvt. Ltd.

You, Wenpeng and Maciej Henneberg (2018). "Cancer incidence increasing globally: the role of relaxed natural selection". In: *Evolutionary applications* 11.2, pp. 140–152.

Zarocostas, John (2010). "Global cancer cases and deaths are set to rise by 70% in next 20 years". In: *BMJ: British Medical Journal (Online)* 340.

Zeiler, Matthew D (2012). "Adadelta: an adaptive learning rate method". In: *arXiv preprint arXiv:1212.5701*.

Zhai, Pan, Yanbing Ding, Xia Wu, Junke Long, Yanjun Zhong, and Yiming Li (2020). "The epidemiology, diagnosis and treatment of COVID-19". In: *International journal of antimicrobial agents*, p. 105955.

Zhang, Guoqiang Peter (2000). "Neural networks for classification: a survey". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 30.4, pp. 451–462.

Zhang, Jianpeng, Yutong Xie, Yi Li, Chunhua Shen, and Yong Xia (2020). "COVID-19 screening on chest X-ray images using deep learning based anomaly detection". In: *arXiv preprint arXiv:2003.12338*.

Zhang, Qingchen, Changchuan Bai, Zhikui Chen, Peng Li, Hang Yu, Shuo Wang, and He Gao (2019). "Deep learning models for diagnosing spleen and stomach diseases in smart Chinese medicine with cloud computing". In: *Concurrency and Computation: Practice and Experience*, e5252.

Zhang, Shi-Jun, Hai-Ning Qian, Yan Zhao, Kai Sun, Hui-Qing Wang, Guo-Qing Liang, Feng-Hua Li, and Zheng Li (2013). "Relationship between age and prostate size". In: *Asian journal of andrology* 15.1, p. 116.

Zheng, Ying-Ying, Yi-Tong Ma, Jin-Ying Zhang, and Xiang Xie (2020). "COVID-19 and the cardiovascular system". In: *Nature Reviews Cardiology* 17.5, pp. 259–260.

Zhou, Bolei, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba (2014). "Object detectors emerge in deep scene CNNs". In: *arXiv preprint arXiv:1412.6856*.

Zhou, Bolei, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba (2016). "Learning deep features for discriminative localization". In: *Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 27-30 June 2016*, pp. 2921–2929.

Zhuang, Fuzhen, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He (2019). "A comprehensive survey on transfer learning". In: *arXiv preprint arXiv:1911.02685*.

Part II

Set of papers

Appendix A

PROMETEO: A CNN-Based Computer-Aided Diagnosis System for WSI Prostate Cancer Detection

Authors

- Lourdes Duran-Lopez
- Juan Pedro Dominguez-Morales
- Antonio Felix Conde-Martin
- Saturnino Vicente-Diaz
- Alejandro Linares-Barranco

Publication

Journal: IEEE Access

Volume: 8

Publisher: IEEE

Date: July 2020

Pages: 128613-128628

ISSN: 2169-3536

Link: https://ieeexplore.ieee.org/abstract/document/9139241

Appendix B

Performance Evaluation of Deep Learning-Based Prostate Cancer Screening Methods in Histopathological Images: Measuring the Impact of the Model's Complexity on Its Processing Speed

Authors

- Lourdes Duran-Lopez
- Juan Pedro Dominguez-Morales
- Antonio Rios-Navarro
- Daniel Gutierrez-Galan
- Angel Jimenez-Fernandez
- Saturnino Vicente-Diaz
- Alejandro Linares-Barranco

Publication

Journal: Sensors

Volume: 21, **Pages:** 1122-1134

Publisher: MDPI

Date: February 2021

ISSN: 1424-8220

Link: https://www.mdpi.com/1424-8220/21/4/1122

Appendix C

Wide & Deep neural network model for patch aggregation in CNN-based prostate cancer detection systems

Authors

- Lourdes Duran-Lopez
- Juan Pedro Dominguez-Morales
- Daniel Gutierrez-Galan
- Antonio Rios-Navarro
- Angel Jimenez-Fernandez
- Saturnino Vicente Diaz
- Alejandro Linares-Barranco

Publication

Journal: arXiv

Publisher: Cornell University

Date: May 2021

Link: https://arxiv.org/abs/2105.09974

Appendix D

COVID-XNet: A Custom Deep Learning System to Diagnose and Locate COVID-19 in Chest X-ray Images

Authors

- Lourdes Duran-Lopez
- Juan Pedro Dominguez-Morales
- Jesus Corral-Jaime
- Saturnino Vicente-Diaz
- Alejandro Linares-Barranco

Publication

Journal: Applied Sciences

Volume: 10

Publisher: MDPI

Date: August 2020

Pages: 5683-5694

ISSN: 2076-3417

Link: https://www.mdpi.com/2076-3417/10/16/5683