Universidade do Minho
Escola de Engenharia
Departamento de Informática

João Pedro Pereira Fontes

Intelligent Medical Image Analysis

A *Deep Learning* approach to breast cancer diagnosis

April 2018

Universidade do Minho
Escola de Engenharia
Departamento de Informática

João Pedro Pereira Fontes

Intelligent Medical Image Analysis

A *Deep Learning* approach to breast cancer diagnosis

Master dissertation
Mestrado Integrado em Engenharia Informática

Dissertation supervised by
Professor Miguel Angel Guevara Lopez
Professor Luís Gonzaga Mendes Magalhães

April 2018

ACKNOWLEDGEMENTS

ABSTRACT

Once medical images were scanned and uploaded to a computer, researchers began to create automated medical imaging systems. From the 1970's to the 1990's, medical imaging was performed with sequential application of low-level pixel processing and mathematical modeling to solve specific tasks such as organ segmentation. At the end of the 1990's, supervised techniques began to appear, where data extracted from the images were used to train models and classification systems. One example is the use of automated classifiers to build support systems for cancer detection and diagnosis. This pattern recognition and / or Machine Learning approach is still very popular and represented a shift from systems that were completely human-engineered to computer-trained systems with the use of specific (manually drawn) features and automatically extracted from the training data (example). The next step is that the algorithms directly learn the characteristics of the pixels of the images. This is the basic concept of Deep Learning algorithms: multi-layered models that transform input data (images) into outputs (e.g. the presence or absence of pathological lesions or cancer).

This study intends to present ways of using Deep Learning algorithms in the analysis of medical images, in particular for the classification of pathological lesions representative of breast cancer phenotypes.

Keywords: Medical Image Analysis, Deep Learning, Artificial Intelligence, Breast Cancer Diagnosis

RESUMO

Assim que foi possível digitalizar e carregar imagens médicas num computador, os investigadores começaram a criar sistemas automatizados para análise de imagens médicas. No intervalo dos anos 70 até aos anos 90, a análise de imagens médicas foi feita com a aplicação sequencial de processamento de pixeis de baixo nível e modelação matemática para resolver tarefas específicas como, por exemplo, a segmentação de órgãos. No final dos anos 90, começam a aparecer as técnicas supervisionadas, onde os dados extraídos das imagens são usados para treinar modelos e sistemas de classificação. Um exemplo é o uso de classificadores automáticos para construir sistemas de apoio à deteção e diagnóstico do cancro. Esta abordagem de reconhecimento de padrões e/ou Machine Learning ainda é muito popular e representou uma mudança nos sistemas que eram completamente projetados por seres humanos para sistemas treinados por computadores com recurso ao uso de características específicas (manualmente desenhadas) e extraídas automaticamente dos dados de treino (exemplo). O passo seguinte a alcançar é que os algoritmos aprendam directamente as características dos pixeis das imagens. É este o conceito base dos algoritmos de Deep Learning: modelos (redes) compostos por muitas camadas que transformam dados de entrada (imagens) em saídas (por exemplo, a presença ou ausência de lesões patológicas ou cancro).

Pretende-se com este estudo apresentar formas de usar algoritmos de Deep Learning na análise de imagens médicas, em particular para a classificação de lesões patológicas representativas de fenotipos de cancro da mama.

Palavras-Chave: Análise de Imagem Médica, Deep Learning, Inteligência Artificial, Diagnóstico de Cancro da Mama

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

A

AI Artificial Intelligence. 1

ANN Artificial Neural Network. 2, 7

AUC Area Under the ROC Curve. 23

C

CADX Computer-Aided Diagnosis. 3, 5, 11, 19, 21, 27, 32

CC Craniocaudal. 28

CNN Convolutional Neural Network. 13, 14, 21, 22

D

DBN Deep Belief Network. 17

DL Deep Learning. 2, 21–24, 27–29, 31–33

DNN Deep Neural Networks. 12, 13

F

FC Fully-Connected. 14, 31–33

L

LSTM Long Short-Term Memory Unit. 15, 16

M

ML Machine Learning. 1, 22

MLO Medio-Lateral Oblique. 28

MLP Multi Layer Perceptron. 7, 23, 24, 32

R

RBM Restricted Boltzmann Machine. 16, 17

RELU Rectified Linear Unit. 31

RNN Recurrent Neural Network. 13, 15, 16

ROI Region Of Interest. 21, 22, 29, 30

S

SOA Service Oriented Architecture. 2

SVM Support Vector Machine. 22–24, 32

1

INTRODUCTION

Since computers first came up and started to replace humans in many tasks, the evolution of these techniques follows an exponential growth that presently is giving us the power to came up with results to solve our problems much more quickly and, very often, more accurately than only with human effort. This is possible with the use of Artificial Intelligence (AI). AI algorithms can now provide an environment in which computers can learn from training data and provide results that are more like those provided by a human.

AI is a branch of computer science that seeks to make computers have human-like behaviors, like think, react and learn [1]. These actions allow computers to solve problems that are normally associated with cognitive functions and that usually require some level of training to be solved.

Ever since AI is studied, one of the areas that gained a major role in this study is Machine Learning (ML). Machine Learning is a science that studies the ways that a computer can solve a problem without being explicitly programmed to do so or, in other words, to improve with experience [2]. To do this, the computer learns from a training set of data and adjusts itself to fit the results that are expected to be produced. The process of learning follows a technique that tries to minimize an error metric and, as a result, to maximize the algorithm's performance in responding to the questions asked. With this, it is possible to bring computers closer to perform tasks that until now were only possible to perform by humans, such as recognizing an image, video, sounds, etc.

With the introduction of Machine Learning, fields like Computer Vision and Natural Language Processing got a huge boost in its performance. Previous algorithms in these fields of study had a high complexity and required many resources in order to be able to provide results in a short time. Introducing Machine Learning, the algorithms have shifted the main workload to the training stage, releasing the classification stage (recognition) from this task. With this, it is possible to provide results faster than using traditional techniques, although it may have a lower degree of confidence, which depends on the algorithm's training.

Although Machine Learning algorithms produced very satisfactory results, there was a ceiling in the performance that these algorithms could not overpass and that delayed the developments of this area for some years. With the new developments in technology (i.e. the introduction of faster CPUs and GPUs) and the development of new Artificial Neural Network (ANN) algorithms, Machine Learning techniques gained a new breath and, with the introduction of new approaches like Deep Learning (DL), could overcome the ceiling that was imposed before and gave the results that were expected some years before. As we can see in figure 1, the difference between modern Deep Learning and older Machine Learning algorithms performance is huge when the amount of data is increased.

This new breath in Machine Learning algorithms given by the introduction of Service Oriented Architecture (SOA) Deep Learning models in the field of medical image analysis is the object of study of this dissertation. In specific, we aim to explore DL methods for breast cancer diagnosis. Deep Learning-based models have demonstrated to be successful in identifying and classifying pathological lesions in several medical imaging modalities (e.g. X-ray, PET-CT, ultrasound and MRI)

1.1 BREAST CANCER DIAGNOSIS

According to the NCI[1] Dictionary of Cancer Terms, Breast Cancer is a malign tumor that forms in breast tissue. Like any other cancer, cells become more and more abnormal, old or damaged cells survive when they should die, and new cells form when they are not needed. These extra cells can divide without stopping and may form growths called tumors. This originates a set of symptoms that may cause death to the person suffering from this disease if not treated well and as soon as possible. Breast cancer usually occurs in women, although it can also appear in men more rarely.

Breast cancer is one of the most spread chronic non-communicable diseases in the world. According to the American Cancer Society [3], breast cancer represents 25 percent of the new cancer diagnostics in women worldwide, being the greatest cause of death in world's developing regions.

[1] National Cancer Institute https://www.cancer.gov/types/breast, Accessed 04 November 2017

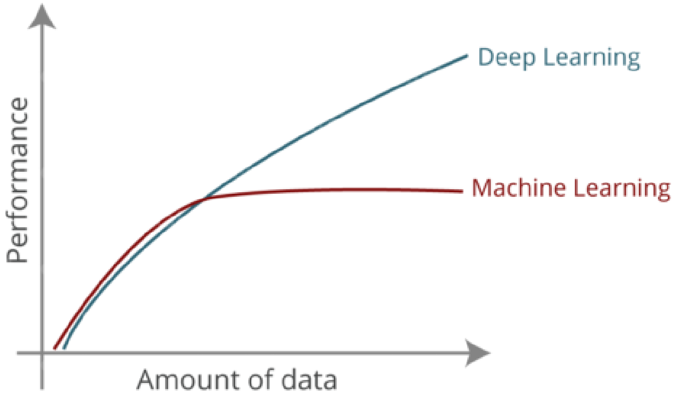

Figure 1.: Deep Learning performance compared to traditional Machine Learning algorithms. When the amount of data increases, traditional Machine Learning methods are not able to take advantage of that amount, stabilizing its performance in a ceiling. However, Deep Learning methods learn better with more data, being able to perform significantly better than traditional methods do.
Source.: http://bit.ly/2E6Iofb, Accessed 02 November 2017

Having this in mind, the purpose of this dissertation is to introduce Deep Learning functionalities to improve the current Computer-Aided Diagnosis (CADx) methods of breast cancer diagnosis. Current CADx methods already help medical professionals in their diagnostic tasks, improving diagnosis accuracy and achieving a earlier diagnosis, which gives a bigger change of recovery to patients. Having such a huge role in diagnosis, CADx methods need to be improved in order to achieve a better performance in classifying cancer phenotypes. Since current CADx methods use Machine Learning algorithms in their tasks, the introduction of Deep Learning can give a boost in these methods performance.

1.2 MOTIVATION AND MAIN AIMS

This work has its motivation based in the statistics presented on section 1.1. Watching the numbers being so large worldwide, it is necessary to provide better methods to help medical professionals in their diagnostic activities. Achieving an earlier and more accurate diagnosis will make the mortality numbers come down, helping to save more lives. The introduction of CADx systems can bring improvements in the early cancer detection process. With this, the main aims of this work are:

- To carry out a study of the algorithms and tools of Deep Learning , in specific for analysis of medical images;
- Implement selected Deep Learning algorithms on high-performance computing infrastructures (GRID, Cloud);
- Test the algorithms with higher performance in a benchmarking dataset of images of patients with breast cancer.

...

1.3 DOCUMENT STRUCTURE

This dissertation is organized in 6 chapters that describe the application area and present a solution to the problem of diagnosing breast cancer using Deep Learning .

Chapter 1 presents the topic of study and provides the motivation and main aims that lead to this work.

Chapter 2 provides some background and context to this work, by presenting the current state of the art of Deep Learning, including some of the most used methods and structures. It also adds some related work in the Deep Learning thematic, presenting some of its applications. Finally, it provides a survey on medical image analysis, showing the current applications and how could Deep Learning be helpful in this area.

Chapter 3 presents the problem treated in this dissertation, showing some existing approaches. It also describes the design and architecture of the proposed approach by this work.

Chapter 4 shows all details of implementation of our proposed approach, including the decisions made and the performance improving strategies, throughout the details of implementation in a high performance environment, and the outcomes presented to this work.

Chapter 5 presents a proof of concept of this work, by showing the tests made on this solution.

Finally, chapter 6 concludes this dissertation and presents prospects for future work.

2

BACKGROUND

This chapter provides some context to the work presented in this dissertation. Having this, this chapter is centered in presenting the concepts of Medical Image Analysis, with an emphasis in breast cancer diagnosis. It also presents the concepts behind Deep Learning, since its origins in Machine Learning, the current state of the art and, finally, how it can be used to build CADx systems. It also presents

2.1 MACHINE LEARNING BASICS

Machine Learning is a field of computer science that studies and tries to implement learning capabilities, like humans possess, into computers. Learning capabilities on computers can be defined as [4]: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.". In this definition we can find the concepts like *experience*, *class of tasks* and *performance measure*, that help understand what Machine Learning really is. Machine Learning algorithms try to improve its performance in a set of *tasks* with training (the act of building *experience*), in order to decrease the value of an error metric (used to *measure performance*). The lower is this error metric, the better is the performance of the algorithms. Machine Learning methods are generally separated into two different groups:

- *Supervised methods* - in these methods, algorithms are trained using a dataset that already answers to the questions asked. In other words, all the instances have a label that corresponds to the answer that has to be given for this class (category) of instances. This label is used for the algorithm to adjust itself to giving the right answer to the question asked. Examples of these methods are classification and regression methods;

- *Unsupervised methods* - unlike supervised methods, these strategies try to deduce the label that accompanies the records in supervised methods. These methods are useful to find patterns in data and when we don't have an annotated dataset to train a supervised model, to infer the classes for each record. Clustering methods are examples of these methods.

2.1.1 *Neural Networks*

Neural networks are Machine Learning algorithms that are inspired by the human brain structure. It forms the basis in major Deep Learning methods. These models are composed of neurons, that have activation functions and weights. Neurons are connected to each other, where the output of some neurons is the input of other(s), giving the model a network shape, like it is seen in figure 2.

WEIGHTS Basic training of neural networks will give the input connections of a neuron different weights, modifying the output value of a neuron. After this training phase, these weights will give a bigger role to more important connections. With this, the input value of a connection will be multiplied by the connection's weight. All values from input connections will be integrated and reduced to a single value (usually all values are summed). Additionally, it can be introduced a value (+1) called *bias*, that ensures that certain conditions are established for each node. Giving input values, the network will produce an output label (e.g. class) based on its training.

ACTIVATION When input arrives at the neuron, it is computed based on the weights and bias and an output value is calculated, also called the neuron activation. This activation will be the value outputted to all the neuron's connections. To produce this output, a transfer function is used to limit the output range and to introduce a non-linear component in the output value, usually giving the network the capacity to approximate any function [5]. The most used transfer functions are Softmax, hyperbolic tangent (tanh) or sigmoid. It is usual that transfer functions give an output in the interval $[0,1]$, but uses of functions with intervals $[-1,1]$ and even $[-\infty,+\infty]$ have proven to give good results in certain cases [6].

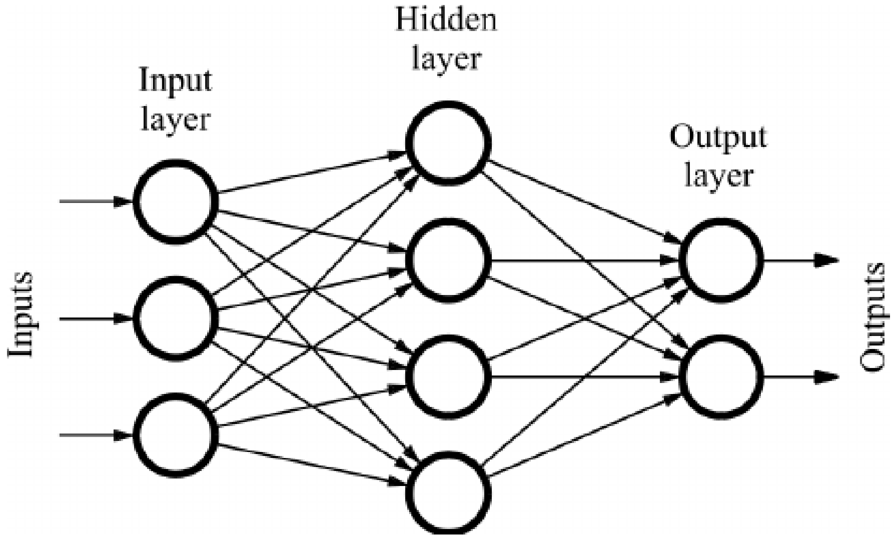

Figure 2.: Structure of a standard ANN. Computation starts in the input layer, passing through hidden layers and ending in the output layer. This results in a feed-forward architecture, where connections never form a cycle.
Source.: http://bit.ly/2EkesLZ, Accessed 14 November 2017

Therefore, in order to get the output $\boldsymbol{y}$ of a neuron n, with input $\boldsymbol{x}$:

$$\boldsymbol{y} = \sigma\left(\boldsymbol{W}_n \boldsymbol{x} + \boldsymbol{b}\right) \tag{1}$$

where $\boldsymbol{W}$ represents the weights, σ the activation/transfer function and $\boldsymbol{b}$ the bias. Therefore, using a neural network with multiple layers, also called Multi Layer Perceptron (MLP), the function computed in a neuron on layer n can be described by:

$$f\left(\boldsymbol{x}, \boldsymbol{W}, \boldsymbol{B}\right) = \sigma\left(\boldsymbol{W}_n\, \sigma\left(\boldsymbol{W}_{n-1} \ldots \sigma\left(\boldsymbol{W}_0 + \boldsymbol{B}_0\right) \ldots + \boldsymbol{B}_{n-1}\right) + \boldsymbol{B}_n\right) \tag{2}$$

where $\boldsymbol{B}$ represents the bias values for all neurons.

BACK-PROPAGATION Neural networks, like all Machine Learning methods, learn with experience. During its training phase, the system must be fitted in order to overcome the error generated in a prediction. The most used technique in network training is the back-propagation algorithm [7]. This algorithm propagates the error backward, updating connection weights based on the error value for each iteration. It starts on the output layer and goes all the way to the input layer, searching for

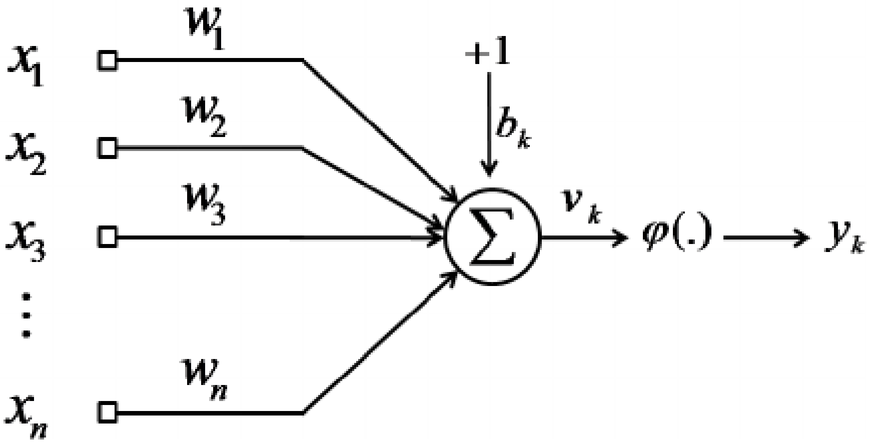

Figure 3.: Illustration of a neural network's neuron structure. When input x_n arrives to the neuron, it is multiplied by the connection's weight w_n and finally integrated with the rest of the input values, using the summation function. After the bias $\boldsymbol{b}_k$ is added, the resulting value v_k is passed to the activation/transfer function $\boldsymbol{\phi}$, producing output $\boldsymbol{y}_k$
Source.: http://bit.ly/2AErAcp, Accessed 15 November 2017

the minimum in the error function. It is based on the Gradient Descent technique, using a learning rate to give bigger or smaller steps in its iterations. A high learning rate will learn very quickly but it may cause the error not achieving a minimum. On the other hand, a low learning rate will be to slow and it will take a large number of epochs (iterations on the training dataset) to learn all the data. The perfect learning rate is a half-term between these two, giving the neural network the capacity to produce generalized predictions without the possibility to overfit the training set. The performance evolution during the training phase for both the training and validation sets can be seen in figure 4.

2.2 MEDICAL IMAGE ANALYSIS

Medical Image Analysis is a sub-field of computer vision that concerns with knowledge extraction from medical images/exams like X-ray, ultrasound or MRI [8]. After obtaining these images, computers analyze them in order to determine the presence or absence of lesions, only visible through the use of risky surgical interventions.

...

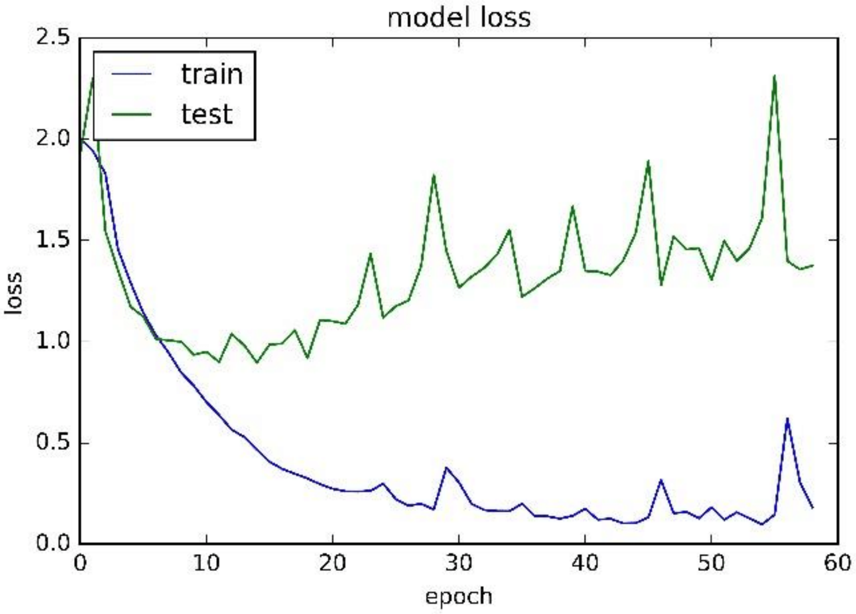

Figure 4.: Example of error evolution in a neural network training. Increasing the number of training iterations will result in a better performance in evaluating the training set. The perfect value of learning epochs is when the error when evaluating the validation data starts increasing, which gives the indication that the neural network is beginning to over-fit the training data [13], losing its capacity to generalize to other data other than the training one.
Source.: http://bit.ly/2D3wnXB, Accessed 21 November 2017

The process of analyzing these images began to be done only by experts, who would determine the existence or absence of these lesions through the visual analysis of the images obtained. With the advance of technology, medical image analysis through computer become an aid to specialists, since it facilitated diagnostic processes using computer vision algorithms to make certain processes easier, such as organ segmentation [9, 10] or lesion classification [11] . With the advent of machine learning algorithms, it was possible to teach the machines how to diagnose based on features extracted from the images obtained by medical imaging. By using variety-rich datasets, covering a significant number of possible cases, it would be possible to obtain good diagnostic results. An example proposed by López et al. in 2007 [12] was to identify cases of breast cancer using machine learning tools.

...

As was the norm for traditional machine learning algorithms, they reached a point where it was not possible to obtain better results than the ones obtained at the time, already shown in previous sections. With this, Deep Learning became the first aim in research in medical image analysis. As it will be shown in section 2.3, Deep Learning algorithms can learn features directly from raw data (e.g. image pixels), so the intervention of an expert for handcrafted feature extraction is no longer necessary for the algorithms to do its job. These algorithms can achieve a better performance than traditional machine learning techniques, which will lead to better results in diagnostic and to the salvation of many lives.

...

CLASSIFICATION Most Deep Learning approaches to medical image analysis tend to classification tasks. In here, one can find examples like image/exam classification (using the whole image, the output will be the existence or not of, e.g. a disease) or lesion classification (using an image, it will assign a class into parts of the image, giving a single image multiple classes).

SEGMENTATION Another task with great importance in medical image analysis where Deep Learning techniques are used is segmentation. In here, organ, subsection or lesion segmentation are the major areas of research. Using segmentation in diagnosis, the expert's work will be facilitated be the visual feedback created by the segmetation process. An example of brain matter and fluids segmentation is presented in figure 5.

...

2.2.1 *Breast Cancer Diagnosis*

Breast cancer diagnosis is one of the main topics in medical image analysis research nowadays. Since breast cancer is one of the most spreaded and deadly diseases in the world, researchers try to find new ways of diagnosing it sooner and accurately.

The most common types of breast cancer start in the ducts that carry milk to the nipple or in the glands that produce the milk. These cancers are called ductal and lobular cancers respectively [14]. These tumors can form a lump in the breast tissue that is visible in breast exams like mammography/mammogram or MRI.

...

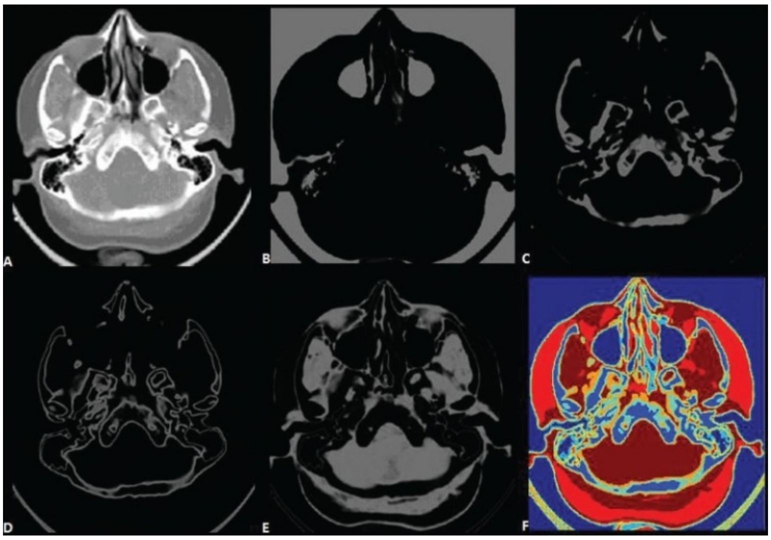

Figure 5.: Example of brain subsections segmentation. This image shows the original exam (CT-scan), the individual segments captured by the segmentation algorithm and the final segmented image using different colors.
Source.: http://bit.ly/2CClqv3, Accessed 6 January 2017

MAMMOGRAPHY/MAMMOGRAM Mammography is a method for breast cancer screening that uses X-rays to get a X-ray film of the breast [15]. This exam is the oldest in breast examining and is the primary exam for cancer detection. Some advances have been made in the late years that include:

- Digital Mammography - also called full film mammography, is a method that converts X-rays into mammographic pictures, producing a digital image that can be used in CADx systems without scanning. this method produces less error that the scanning of X-ray films;
- Breast Tomosynthesis - also called 3D mammography, is a method where multiple images are retrieved from the breast, using different angles to produce a 3D model after reconstruction.

These methods can be used by CADx systems to search for abnormal areas in the breast (e.g. abnormal masses) that could indicate cancer. CADx systems nowadays use primarily film and digital mammographies, although tomosynthesis is starting to appear in breast cancer research. Some of the uses of these exams are presented in section 3.1.

...

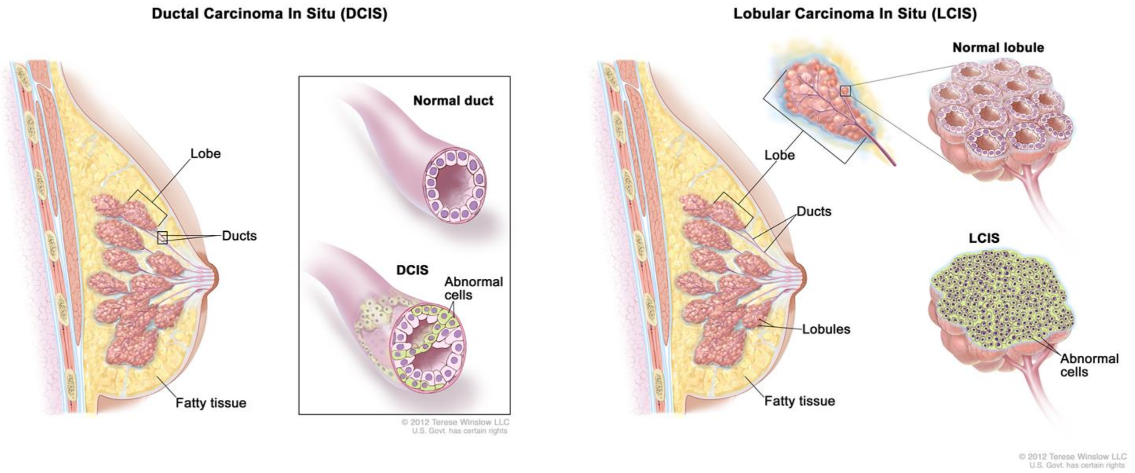
Ductal Carcinoma In Situ (DCIS)
Lobe
Ducts
Fatty tissue
Normal duct
DCIS
Abnormal cells
© 2012 Terese Winslow LLC
U.S. Govt. has certain rights
Lobular Carcinoma In Situ (LCIS)
Normal lobule
Lobe
Ducts
Lobules
Fatty tissue
LCIS
Abnormal cells
© 2012 Terese Winslow LLC
U.S. Govt. has certain rights

Figure 6.: Schemes for Ductal and Lobular Breast Carcinomas. Ductal Tumors form in the ducts that carry milk to the nipple. In the case of Lobular Tumors, they form in the breast lobules, the structures responsible for producing milk.
Source.: http://bit.ly/2DgnkFg, Accessed 11 Jan 2018

2.3 DEEP LEARNING

When traditional Machine Learning methods reached its performance ceiling, Deep Learning techniques appeared. Deep Learning is just an extension of Machine Learning methods, where the algorithms learn the true representation and features of data by itself, without being programmed or informed of which features to extract [16]. With this, Deep Learning methods can be fed with raw data (e.g image pixels) and automatically discover the correct features to extract and train itself based on those features.

Major Deep Learning techniques are based on Deep Neural Networks (DNN), an extension of traditional feed-forward artificial neural networks that present multiple hidden layers. Having a more complex structure, it will take more time to train these networks, although it will have a greater capability of learning. With a greater number of hidden layers, the input data is decomposed into simple data, allowing the network to learn more complex features and representations, acquiring a great generalization capability. The great complexity of this structures made them impractical to be used without an hardware acceleration. With the introduction of GPU computation, DNNs gained a huge boost in its training, which allowed them to be tested and improved. The most popular DNN approaches are Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), being these architectures the most widely used in image analysis. The following sections present some details about the most used Deep Learning methods.

2.3.1 *Convolutional Neural Networks*

CNNs are a DNN variant very useful in image analysis, being able to use convolution and pooling layers to extract features from the input images. Convolution layers will apply filters to extract features and pooling layers will highlight these features, increasing the level of the input to the next layer. After the feature extraction phase, the network uses fully-connected layers (i.e. simple feed-forward neural networks layers) that will perform the classification tasks based on the features extracted by the convolutional and pooling layers.

Some CNN architectures have become widely used and became references in the Deep Learning world. LeNet [17] consisted in an architecture with 2 convolutional layers that was well used in digit recognition. AlexNet [18] was very similar to LeNet, but introduced the ReLU function as the neurons' activation function instead of hyperbolic tangent or sigmoid functions, being now the most common function used in CNNs. This architecture won the ImageNet ILSVRC challenge in 2012[1]. GoogLeNet [19] was another architecture that become famous after winning the 2014 ImageNet ILSVRC challenge[2]. This architecture introduced an inception module, that caused the parameters in the network from 60M (in AlexNet) to 4M, reducing the usage of computation resources.

segmentation One of the most used applications of CNNs is segmentation in an image. It is one of the largely used applications in medical image analysis, in order for computer programs to be able to divide organs in an image. This is achieved by analyzing every pixel in the image and labeling it. The networks used in these tasks apply all the feature extraction filters like normal CNNs, but don't have the usual fully-connected layers for the classification phase at the end of the computation. Having only convolutional layers in its architecture, these networks

[1] See details about the 2012 ImageNet ILSVRC challenge here

[2] See details about the 2014 ImageNet ILSVRC challenge here

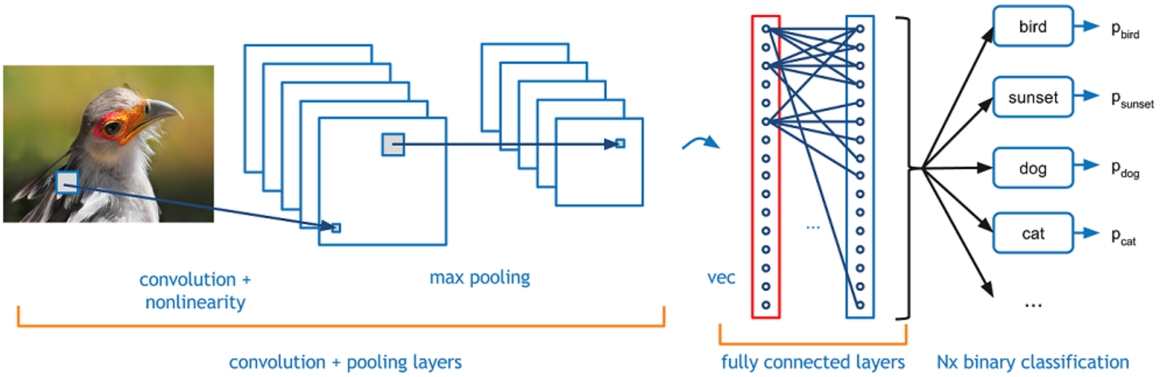

Figure 7.: Example of a CNN architecture. Given an input image, the CNN will output the probabilities that the input belongs to each of the classes trained by the model. In this model, the convolutional and pooling layers represent the feature extraction phase and the Fully-Connected (FC) layers represent the classification phase.
Source.: http://bit.ly/2CRmiQ3, Accessed 2 December 2017

are called Fully Convolutional Neural Networks. An example of segmentation using these networks can be seen here [20].

3D GENERALIZATION Although CNNs where made for 2D image analysis, the convolution process can be generalized for 3D. Instead of using filters in two dimensions, it is used a 3D version of convolution filters. These networks are useful in cases like video analysis [21, 22], 3D object recognition [23, 24, 25] or 3D image segmentation [26].

...

2.3.2 *Recurrent Neural Networks*

Recurrent neural networks are a network architecture where we can find direct cycles in its connections. These cycles allow these networks to have a characteristic that differentiates them from regular networks - memory. This characteristic comes because of the fact that past experiences and values influence computing. The value from past experiences is stored in nodes and is stored for the next computation. This characteristic is useful when analyzing data inserted in a time series, that has a correlation in time. So, in order to calculate a certain neuron state $\boldsymbol{y}$ at time t, the value from time t-1 needs to be used alongside the input value $\boldsymbol{x}$ on time t:

$$\boldsymbol{y}_t = \sigma\left(\boldsymbol{W}\boldsymbol{x}_t + \boldsymbol{R}\boldsymbol{y}_{t-1} + \boldsymbol{b}\right) \tag{3}$$

where $\boldsymbol{W}$ and $\boldsymbol{R}$ represent the weights and $\boldsymbol{b}$ the bias.

Since RNNs analyze time sequences, these networks need an extension of the back-propagation algorithm in order to link all time steps. This extension is called Back-Propagation Through Time. This algorithm, adjusts the weights using a gradient descent rule, like normal back-propagation does, but it relies in past . Since this rule relies on multiplications, and since RNNs use a great number of this operations, the gradients tend to vanish (to become zero), not allowing the network to learn. This problem is called the Vanishing Gradients Problem [27]. This problem emerged as one great obstacle in RNNs performance, only solved later by the introduction of Long Short-Term Memory Units (LSTMs).

LONG SHORT-TERM MEMORY UNITS LSTMs are a RNN variation that came in the 1990's [28] to solve the vanishing gradient problem by using a more constant error value, not allowing abrupt changes in it. By preserving the error, these networks can link the time series occurrences in a more efficient way, being well suited for classification tasks. The error is preserved because of the use of a memory cell and three gates: input, output and forget. The cell will store values from previous computations. The input gate will control what value will flow to the network. The forget will control the values that will be stored in the network. The output gate will control what value will be used to calculate the neuron's (considering a LSTM block a single neuron) activation. This workflow can be seen in figure 8

2.3.3 *Unsupervised Techniques*

Deep Learning techniques work better with more data. Using supervised techniques, with more data, comes a great outcome. All the data needs to be labeled so that the algorithm can learn using the answers provided. Having millions of records to label, the work becomes expensive. To overcome this obstacle, unsupervised techniques were created, so that the algorithms could learn from unlabeled data. These algorithms can also be used to create labels in a dataset when they

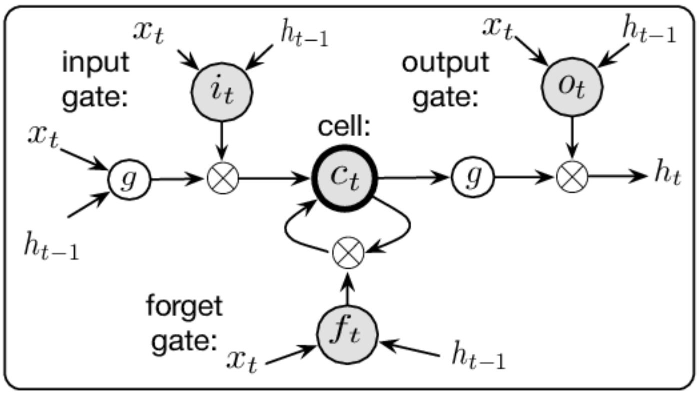

Figure 8.: Standard LSTM architecture. Input $\boldsymbol{X}_t$ flows into the cell and the gates, along with the value stored from last computation. The input cell will determine first which value will be passed to the cell, where the forget gate will control the amount of past information that will erased from memory. Finally, the output gate will determine the final value that will be outputted by the LSTM block.
Source.: http://bit.ly/2EkmODo, Accessed 16 December 2017

don't exist, in order to provide the labeled dataset to a supervised technique after that process. The following section provide some details about some of the most used unsupervised Deep Learning techniques like Restricted Boltzmann Machines, Deep Belief Networks and Auto-Encoders.

RESTRICTED BOLTZMANN MACHINES AND DEEP BELIEF NETWORKS Boltzmann Machines are stochastic RNNs and an example of Markov Random Fields [29] that can fit to a probability function. These networks are composed of two groups of units, visible and hidden units, connected between each other bidirectionally. Since this architecture could result in a great complexity, Restricted Boltzmann Machines (RBMs) where created. These networks do not allow connections in-layer, i.e. between units in the same layer. An example of this architecture can be seen in figure 9. Each node in a visible unit will receive a feature to learn (a pixel value in the case of an image), and pass that value to the next layer, multiplied ny the connection's weight. In the hidden layer, hidden units have associated a bias value, that will be added to the sum of the inputs received from the visible units. The final value goes through the activation function, producing the final output of that unit. So, with input x, output y_t of unit t will be:

$$y_t = \sigma(\boldsymbol{W}x + \boldsymbol{b}) \qquad (4)$$

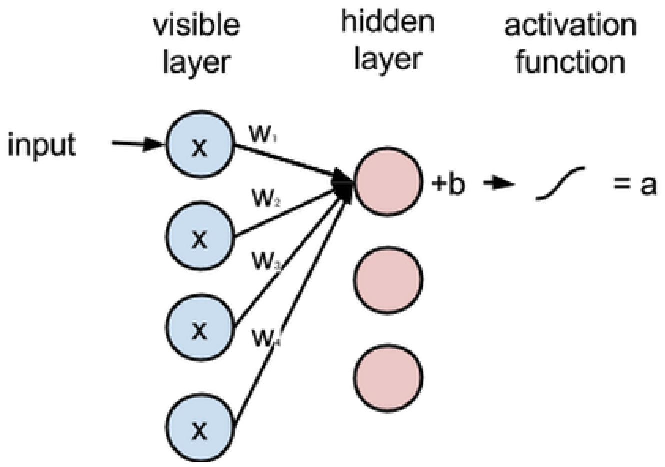

Figure 9.: Structure of a RBM. When input arrives to the visible units, it will multiplied by the weights and then passed to the hidden layer. In the hidden layer, a bias value will be added and the resulting output will feed an activation function to produce an output in the interval $[0, 1]$, i.e. a probability value.
Source.: http://bit.ly/2FfogIs, Accessed 28 December 2017

where $\boldsymbol{W}$ represents the weights, $\boldsymbol{b}$ the hidden unit's bias and σ the activation function.

RBMs can be stacked and create a network where the layers are RBM, producing a Deep Belief Network (DBN). Excluding the first and last layers from this network, all layers will perform as visible and hidden layer in the training phase. This stack of RBMs can used in supervised tasks, as classifiers, or unsupervised as part of auto-encoding systems, like the ones presented in the next section.

AUTO-ENCODERS Auto-Encoders are unsupervised learning algorithms that try to fit to the function $f(x) = x$. In other words, the network tries to fit itself in order that the input and output data match, constructing a compressed representation of the input data in its hidden layers. These algorithms process input in two phases, each one composed normally by a DBN:

1. Encoding - In this phase, the neural network will try to find a compressed feature vector that represents the input data;

2. Decoding - This phase will try to reconstruct the feature vector outputted from last phase to its initial stage, the input data.

These algorithms can be used in topic modeling [30]. In its operations, similar data will generate the same output, being useful in question-answer answer systems like Watson [31].

2.3.4 *Reinforcement Learning*

Reinforcement Learning is a set of unsupervised learning methods that study the way how intelligent agents take actions in order to maximize a certain reward. In these methods, an intelligent agent must find the correct answer to the problem by trying to find the possibilities based on the current environment state. The execution flow of basic reinforcement learning can be summarized in the following actions:

1. Agent performs action on environment;

2. Environment rewards the agent by its action;

3. Agent calculates the convenience of the action;

4. Agent performs a new action.

The key difference between supervised learning and reinforcement learning is the fact that reinforcement learning explores the environment in order to search for the best way to solve a problem. Supervised methods will fit to a way to solve that problem. Reinforcement learning methods will learn from experience in a trial-and-error strategy in order to maximize the final reward.

EXPLORATION/EXPLOITATION TRADE-OFF A key paradox in reinforcement learning methods is the exploration/exploitation trade-off. By exploring, an agent will choose something that it does not have information, but it can bring new and important knowledge. On the other hand, exploiting will secure expectations, because it will use already known things and will often produce expected results. When an agent first starts its interaction with the environment, it will explore in order to build its knowledge. After some time, it will exploit what it already knows, always looking for the maximum reward.

2.4 SUMMARY

In this chapter the key concepts behind this dissertation were presented. Since Deep Learning techniques were introduced, Machine Learning gained a new boost in research. With this, it is possible to build better CADx systems to help medical professionals in their diagnostic tasks. Having this in mind, this dissertation proposes a Deep Learning approach to breast cancer diagnosis, making use of the concepts presented in the sections above.

For a better understanding of the concepts presented in this state of the art review, a conceptual map is presented in figure 13 on appendix D.

3

BREAST CANCER DIAGNOSIS USING DEEP LEARNING

This chapter provides information about the problem addressed in this dissertation. Therefore, the information presented focus on showing the already existing approaches to solve the problem and presents our proposed approach, presenting its details and architecture.

3.1 EXISTING APPROACHES

Breast cancer diagnosis using CADx systems is one of the biggest interests in current research in medical image analysis. Since the introduction of Deep Learning methods, these have been used to classify mammographies and assist medical experts in their tasks.

The first uses of DL methods in CADx systems from breast cancer diagnosis were made by Sahiner et al. in 1996 [32], where a CNN was used to classify mass and normal breast tissue. This approach had a first phase of image preprocessing, where a Region Of Interest (ROI) was selected and a series of operations (e.g. filters) were applied in order to facilitate the learning process by the CNN. This two stage approach was used by other solutions, varying the preprocessing methods and the input image type, using mammographies, MRI or tomosynthesis (3D mammography). Mammographies are the most used exams for classification/segmentation tasks [32, 33, 34, 35]. MRI [36] and more lately tomosynthesis [37] also contains some approaches, not being so expressive for lack of samples for training of DL models.

Although recent approaches use features extracted with the use of DL models, there are some approaches [35, 38, 39] that combine DL classification with hand crafted features, designed by experts. All the knowledge extracted is used to train

ML and DL models and, sometimes, to achieve better results than using only one group of features for training.

3.2 PROPOSED SOLUTION

3.2.1 *Base Approach*

The base approach for this dissertation is the one presented by Arevalo et al. in 2015 [39]. This approach separates the classification task into three main stages, like already used by some of the approaches in section 3.1: (i) preprocessing of input images, (ii) a supervised feature learning using a DL model and finally (iii) the supervised learning of an Support Vector Machine (SVM) model to perform classification of benign and malign tumors.

The preprocessing phase, very often used in image recognition systems, tries to enhance image details, in order to facilitate the work of the feature extraction phase, increasing its performance. This stage, applied to the training dataset, uses the following processes:

1. The ROI is cropped, corresponding to a square around the lesion already outlined by an expert. This process will reduce image size, thus facilitating the training process;

2. A Data Augmentation process is applied to increase the number of samples by rotating the cropped images by 90, 180 and 270 degrees. This process will give the model more data to learn, increasing its training and possibly its performance. It will also prevent future overfitting, by creating a bigger diversity of data;

3. A global and local contrast normalization process is applied to all the samples. These filters are useful to overcome effects like difference of lightning between film images.

Once prepared, the images are used to train a CNN that creates a simpler representation (feature extraction process) that is later classified into benign or malign cancer using an SVM classifier. The use of this feature representation for the images is justified by the reduction in training time (it is cheaper to train an SVM using feature representation than using full image representation to train a CNN). This approach reached an Area Under the ROC Curve (AUC) of 0.822, outperforming the previous values already obtained by some of the authors when using computer vision methods like HOG[1] and HGD[2] descriptors to represent an image/exam [40]. Adding a set of hand crafted features, the model got a AUC value of 0.826.

3.2.2 *System Overview*

The objective of this dissertation is to take as base approach the solution presented in [39] and increase its performance in classifying benign and malign tumors using instances of the BCDR benchmarking dataset [41]. To achieve this performance improvement, the feature extraction model was augmented, with the use of a known architecture that has proven to have a great performance in feature extraction. In order to check the best combination of processes to overcome the problem presented, the following combinations were tested:

- Use of a Deep Learning model to perform feature extraction and also to perform the classification task, with the use of a Softmax classification layer as output of the DL model. When predicting a sample, the model will produce a binary classification (benign or malign);
- Use of a Deep Learning model to perform feature extraction and later feed those features to an SVM classifier. The use of this SVM model is justified with the fact that it is cheaper to train it then to train the fully-connected from the DL model;
- Use of a Deep Learning model to perform feature extraction and later feed those features alongside with a set of hand-crafted features to an SVM and a MLP classifier.

The scheme presented in figure 10 explains the main workflow of the system.

deep learning model A Deep Learning model is a great tool to perform both feature extraction and classification processes, as referred before. So, a DL model was developed to perform this tasks in our system. The model is based on Google's InceptionV3 architecture [42], which scheme can be seen on appendix A.1. This

[1] Histogram of Oriented Gradients
[2] Histogram of Gradient Divergence

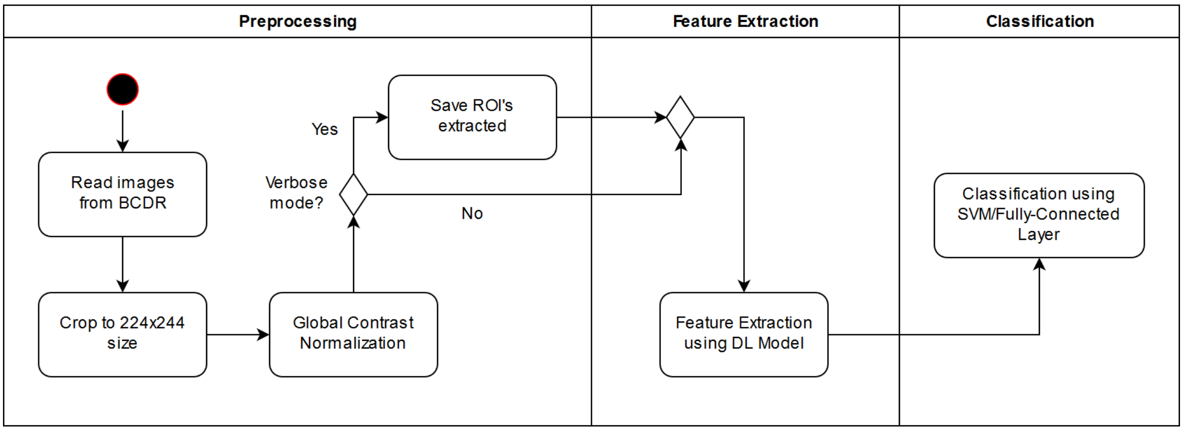

Figure 10.: System's main workflow

architecture is an update to GoogLeNet, winner of the 2014 ImageNet contest, that has also been trained on the ImageNet dataset, where it achieved a top-1 accuracy (correct answer is the the prediction with the highest probability) of 78.0% and a top-5 accuracy (correct answer is in the 5 predictions with higher probability) of 93.9%. Since this architecture can score high accuracy values in the 1000 class ImageNet dataset, it can be fine-tuned to our 2 class dataset and, theoretically, achieve similar values of accuracy.

CLASSIFIERS After the feature extraction process, these can be used to represent an image in a simpler way. With this, a simpler classifier (SVM, MLP, etc.) can be used to make a representation learning process. That is the main basis of neural networks: (i) feature learning in the hidden layers and (ii) classification of the features extracted using the output layer. The feature learning process gives a compressed representation of the input image, allowing us to reduce training times and to experiment different classifiers using the same features extracted using the DL model. With the use of hand-crafted features alongside the features extracted with the DL model, the performance can even increase, through the introduction of a human factor, since this set of features is extracted with the help of a medical professional.

3.3 SUMMARY

In this chapter the solution to the problem presented was detailed, presented some already existing solutions, alongside our proposed approach. Our approach follows an approach of feature classification, where these features are extracted using a Deep Learning model. Along with these features, there are included hand-crafted features, designed and extracted by experts, which adds the possibility of an improvement in performance/accuracy of the system. The following chapters will give the implementation details and the results obtained using our approach.

4

DEVELOPMENT

This chapter presents details about the development phase of this dissertation. In general, the information presented focus on development details about the three phases of the system's workflow (preprocessing, feature extraction and classification) presented on section 3.2.2. The following section will present all the implementation details about those three phases.

4.1 IMPLEMENTATION

This CADx system was developed within a system which details are specified in table 1 and all the code listings can be seen in the appendix C. This system allowed, with the use of 2 GPU's, to reduce training times of the Deep Learning models and all the classifiers. The system was developed using the Python language, because of its versatility and abundance of software libraries focused in the development of Deep Learning applications. The most well known libraries are:

- TensorFlow, developed by Google;
- Theano, developed by Université de Montréal;
- CogNitive TooKit, also known as CNTK, developed by Microsoft.
- Keras, an open source process that provides a high level wrapper to the libraries presented in the previous points.

For this system's implementation, the Keras library was selected, using TensorFlow backend, because of the simplicity of its API and the possibility of using highly optimized DL APi's like TensorFlow in a quick and easy way. The possible gain of performance with the use of native TensorFlow was not considered as a factor, because it could not overcome the gain in development time.

	CPU	**GPU**
Manufacturer	Intel®	NVIDIA®
Model	i7-7700	GeForce GTX 1080
#Devices	1	2
Release Date	Q1'17	Q2'16
μArch	Kaby Lake	Pascal
#Cores	4	2560
Frequency	3600	1607
L1 Cache	2 x 32KB/core	96KB/SM
L2 Cache	256KB/core	4MB
L3 Cache	8MB	N/A
Memory	16GB (RAM)	8GB
Max Memory Bandwidth	38.4GB/s	320GB/s

Table 1.: Development system's CPU[1] and GPU's[2] characteristics. This table presents the hardware that the systems used for the DL model training possess. The use of 2 GPU's made possible a theoretical gain of 2x in the training time.

4.1.1 *The BCDR Dataset*

The BCDR dataset [41] is a benchmarking dataset that contains annotated images from film (BCDR-FXX instances) and digital (BCDR-DXX instances) mammographies from patients in the northern region of Portugal. This dataset studies that contain both biopsy proven Medio-Lateral Oblique (MLO) and Craniocaudal (CC) views of the mammography. The lesions were manually segmented by medical experts at Centro Hospitalar São João and compiled in a CSV file that could be used to read and prepare the dataset by a computer program. Along with the outlines CSV, there is also a features CSV file, that contains a set of hand-crafted features that can also be used in classification processes. These features will be explored later on this document.

... melhorar ...

Having downloaded any of the BCDR instances (film or digital), we read the CSV into memory, that contains information like image filename, (x,y) outline points of

[2] CPU details: https://bit.ly/2uSSTlF
[2] GPU details: https://bit.ly/2IwD5Hh

the lesion and the classification (benign of malign). Python's libraries like *csv* and *Scikit-Image* were useful in this process.

4.1.2 *Preprocessing*

The preprocessing phase is a set of process that takes care of preparing the dataset images to be fed to feature extraction DL model. This phase uses the *OpenCV* and *Scikit-Image* libraries to implement these processes. The processes include cropping of the images to fit the input shape of the DL model, a data augmentation process and the application of filters to enhance features. The preprocessing pipeline can be seen in figure 11.

CROPPING The InceptionV3 architecture, which was used for the development of the DL model used in this system, requires that its input (the breast cancer images) has a resolution of 224x224. Having the (x,y) outline points, read from the outlines CSV file, we have the region were the lesion is located. With this, a 224x224 region is cropped around the ROI. If the lesion is located is a region bigger than 224x224, it is zoom down to fit the 224x224 resolution.

DATA AUGMENTATION In order to increase the dataset's domain and quantity of information, data augmentation processes are used. In this system's partical case, the process used to increase the dataset's domain was the creation of new images by the use of a set of rotations (0, 90, 180 and 270 degrees), both in a normal and horizontally flipped image. This creates 7 new samples for each image, thereby increasing the size and variability of the dataset. This type of process can also prevent the overfitting process with the increase in the training domain.

FEATURE ENHANCEMENT In order to have a better feature learning process, the features to be learned need to be visible in order for the convolution filters to seize those features. Therefore, image filters are applied to enhance those features. In Arevalo et al. [39] approach, the filters applied were the global contrast normalization and the local contrast normalization. In our approach, just the global contrast normalization was applied, because of the lack of performance gain with the addition of a local contrast normalization.

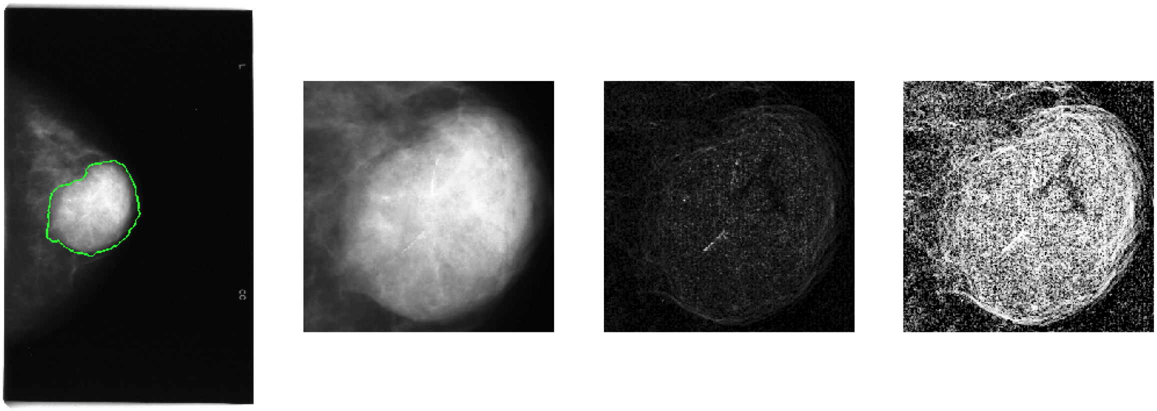

Figure 11.: Preprocessing stages. In the first image we can see the raw image with the lesion segmented. The second image is the result of the extraction of the ROI where the tumor is. The third image is the result of the computation of the intensity gradient. Finally, the fourth image represents the final image, after the histogram equalization process.

Global contrast normalization is a process used to normalize images exposed to different lightning conditions. This process subtracts the mean of the pixels in an image to every pixel. In formal terms, we can define the global contrast normalization as:

$$Y_{i,j} = X_{i,j} - \bar{x}, 0 < i, j < r \tag{5}$$

where X is the initial image, $\bar{x}$ represent the mean intensity of the pixels in image X, r is the dimension of the image (width and height are the same, assuming a square image), in this case 224, and Y is the resulting image.

In our approach, we use two processes to prepare the images for the feature learning process. Forst, after getting the ROI, we compute the intensity gradient of the image. Given a single-channel image (grayscale), this algorithm provides the maximum difference between each pixel and its immediate neighbours. A minimum gradient can be defined so that when a pixel has a lower intensity than that minimum, its gradient will be zero. The implementation of this algorithm in the Python language can be seen in appendix C.1.

After the computation of the intensity gradient, the histogram of the image will no be normalized. To solve this, an histogram equalization process is applied to make sure that the features are more easily distinguishable. The resulting images from each one of the preprocessing processes can be seen in figure 11.

4.1.3 *Feature Extraction Model*

The DL model developed, as already explained in section 3.2.2, follows the InceptionV3 architecture, developed by Google. This architecture contains approximately 24 million trainable parameters, far bellow than other architectures like VGG16 (≈140M), VGG19 (≈145M) or InceptionResNetV2 (≈55M), although it can obtain accuracy percentages in the same level.

Using the Keras framework, and more specifically the *keras.applications* module, we can have an implementation of an InceptionV3 model easily. With this method we can obtain a tuned model for the ImageNet dataset, with or without the classification layers (Softmax classifier at the end of the InceptionV3 model). Since the ImageNet dataset contains 1000 classes and our dataset only contains 2 (benign/-malign), we discarded the classification layer. With this, we can already test our approach, making a feature extraction process using the default weights from ImageNet fine-tune process.

Having the preprocessed images, its corresponding labels and the DL model, we can make a fine-tuning process of the model to our dataset. In order to do this, we added a Global Average Pooling layer to flatten the data resulting from the last convolutional layer of the InceptionV3 model, and three FC layers, with 1024, 512 and 2 nodes respectively. The first two FC layers have Rectified Linear Unit (ReLU) activation and the last layer is a Softmax classifier. As for the train/test division, it was used a 10-fold cross-validation strategy, splitting the dataset using 90-10% for training/testing. This division makes sure that the model will never test with a study from a patient that had some of its studies trained. To achieve this, the division is made by patient, and never by study.

The model was trained using a stochastic gradient descent learner, for 150 epochs each run. The training graphs, obtained using the TensorBoard tool, can be seen in appendix B.1.

PERFORMANCE IMPROVING In order to take advantage of both GPU's in our system, the Keras framework allows us to create a multi-GPU model, that it will replicate the model in both GPU's and take advantage of multi-GPU parallelism. To do that, the CPU needs to build the model in order for the model to be copied to both GPU's without corruptions. Using both GPU's, we gain, theoretically, a 2x speedup in training time, since both GPU's are equal.

4.1.4 *Classifiers*

Having the feature extractor ready for use, the classifiers (SVM and MLP) were developed. The 3 FC layers functioned as a MLP classifier, with 1024, 512 and 2 layers, respectively. As for the SVM classifier, it was trained using the features extracted from the penultimate layer (FC layer with 512 nodes) from the DL model.

...

HAND-CRAFTED FEATURES To try to achieve an increase in the SVM model classification performance, the hand-crafted features presented in 4.1.1 were added to the features extracted from the DL model. With this, the model was trained with 529 features: 512 from the penultimate layer of the DL model and 17 that came included in the BCDR dataset.

4.2 SUMMARY

In this chapter the details about the implementation of our CADx system were presented. The system workflow, presented in section 3.2.2 gave the guidelines for this development phase with its three-stage workflow: (i) preprocessing of images from the BCDR dataset, (ii) feature learning, using a Deep Learning model and (iii) a binary classification stage of the input mammographic images in benign or malign tumors, using a Softmax or SVM classifiers. All the stages were easily implemented recurring to the Python language and Keras framework, which gave us the opportunity of concentrating more in tuning the system to achieve a better performance than in the implementation details.

5

CASE STUDIES / EXPERIMENTS

This chapter presents the details of the experimental setup made for the evaluation of our system. This chapter's information focus on the DL model's training outcomes and the results obtained with the experimental setup made for our system's evaluation. Finally, the results obtained are discussed and analysed in section 5.3.

5.1 EXPERIMENT SETUP

The set of experiments made on our system using the BCDR benchmarking dataset involved 4 main tasks:

- Evaluation of the DL model before the fine-tuning process, using the default weights from the ImageNet dataset that came along with the default InceptionV3 model. We can make a feature extraction process using the DL model and later feed those features to an SVM model for binary classification (benign/malign);
- Evaluation of the DL model after a fine-tuning process on our dataset and using the FC layers and the Softmax classifier appended to this model for binary classification (benign/malign);
- Evaluation of the DL model after a fine-tuning process on our dataset. We can make a feature extraction process using the DL model and later feed those features to an SVM model for binary classification (benign/malign);
- Evaluation of the DL model after a fine-tuning process on our dataset, along with a set of hand-crafted features. We can make a feature extraction process using the DL model and later feed those features along with the set of hand-crafted features to an SVM model for binary classification (benign/malign);

5.2 RESULTS

…

5.3 DISCUSSION

…

5.4 SUMMARY

…

6

CONCLUSION

...

6.1 CONCLUSIONS

...

6.2 PROSPECT FOR FUTURE WORK

...

BIBLIOGRAPHY

[1] S. Russell and P. Norvig, *Artificial Intelligence : A Modern Approach*. Prentice Hall Series in Artificial Intelligence, Boston: Pearson, third ed., 2010.

[2] J. G. Carbonell, R. S. Michalski, and T. M. Mitchell, "An Overview of Machine Learning," in *Machine Learning*, pp. 3–23, Springer, 1983.

[3] A. C. Society, *Global Cancer Facts & Figures*. Atlanta: American Cancer Society, third ed., 2015.

[4] T. Mitchell, *Machine Learning*. McGraw-Hill international editions - computer science series, McGraw-Hill Education, 1997.

[5] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken, "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," *Neural networks*, vol. 6, no. 6, pp. 861–867, 1993.

[6] B. Karlik and A. V. Olgac, "Performance analysis of various activation functions in generalized MLP architectures of neural networks," *International Journal of Artificial Intelligence and Expert Systems*, vol. 1, no. 4, pp. 111–122, 2011.

[7] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533–536, 1986.

[8] A. P. Dhawan, *Medical Image Analysis*. IEEE Press series in biomedical engineering., John Wiley & Sons - IEEE Press, 2^a ed., 2011.

[9] N. Karssemeijer, J. T. O. Leon, and E. G. Eijkman, "Recognition of organs in CT-image sequences: a model guided approach," *Computers and biomedical research*, vol. 21, no. 5, pp. 434–448, 1988.

[10] R. Punia and S. Singh, "Review on machine learning techniques for automatic segmentation of liver images," *International Journal*, vol. 3, no. 4, 2013.

[11] W. H. Wolberg, W. N. Street, and O. L. Mangasarian, "Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates," *Cancer letters*, vol. 77, no. 2-3, pp. 163–171, 1994.

[12] Y. López, A. Novoa, M. A. Guevara, and A. Silva, "Breast cancer diagnosis based on a suitable combination of deformable models and artificial neural networks techniques," in *Iberoamerican Congress on Pattern Recognition*, pp. 803–811, Springer, 2007.

[13] J. E. Dayhoff and J. M. DeLeo, "Artificial neural networks," *Cancer*, vol. 91, no. S8, pp. 1615–1635, 2001.

[14] K. K. Hunt, M. C. Green, and T. A. Buchholz, "Diseases of the breast," *Sabiston Textbook of Surgery. 19th ed. St. Louis, MO: Elsevier Saunders*, 2012.

[15] "Mammography screening for breast cancer," *New England Journal of Medicine*, vol. 367, no. 21, p. e31, 2012. PMID: 23171121.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014.

[20] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[21] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[22] A. Diba, A. M. Pazandeh, and L. V. Gool, "Efficient two-stream motion and appearance 3D CNNs for video classification," *CoRR*, vol. abs/1608.08851, 2016.

[23] D. Maturana and S. Scherer, "Voxnet: A 3D convolutional neural network for real-time object recognition," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 922–928, IEEE, 2015.

[24] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3D object classification," in *Advances in Neural Information Processing Systems*, pp. 656–664, 2012.

[25] J. Huang and S. You, "Point cloud labeling using 3D convolutional neural network," in *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pp. 2670–2675, IEEE, 2016.

[26] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, "Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation," *Medical Image Analysis*, vol. 36, no. Supplement C, pp. 61 – 78, 2017.

[27] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6, no. 02, pp. 107–116, 1998.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[29] Y. A. Rozanov, *Markov Random Fields*, pp. 55–102. New York, NY: Springer New York, 1982.

[30] P. Mirowski, M. Ranzato, and Y. LeCun, "Dynamic auto-encoders for semantic indexing," in *Proceedings of the NIPS 2010 Workshop on Deep Learning*, pp. 1–9, 2010.

[31] R. High, "The era of cognitive systems: An inside look at ibm watson and how it works," *IBM Corporation, Redbooks*, 2012.

[32] B. Sahiner, H.-P. Chan, N. Petrick, D. Wei, M. A. Helvie, D. D. Adler, and M. M. Goodsitt, "Classification of mass and normal breast tissue: a convolution neural network classifier with spatial domain and texture images," *IEEE Transactions on Medical Imaging*, vol. 15, pp. 598–610, Oct 1996.

[33] P. Fonseca, J. Mendoza, J. Wainer, J. Ferrer, J. Pinto, J. Guerrero, and B. Castaneda, "Automatic breast density classification using a convolutional neural network architecture search procedure," in *Proc. of SPIE Vol*, vol. 9414, pp. 941428–1, 2015.

[34] B. Q. Huynh, H. Li, and M. L. Giger, "Digital mammographic tumor classification using transfer learning from deep convolutional neural networks," *Journal of Medical Imaging*, vol. 3, no. 3, pp. 034501–034501, 2016.

[35] N. Dhungel, G. Carneiro, and A. P. Bradley, *The Automated Learning of Deep Features for Breast Mass Classification from Mammograms*, pp. 106–114. Cham: Springer International Publishing, 2016.

[36] M. U. Dalmış, G. Litjens, K. Holland, A. Setio, R. Mann, N. Karssemeijer, and A. Gubern-Mérida, "Using deep learning to segment breast and fibroglandular tissue in mri volumes," *Medical Physics*, vol. 44, no. 2, pp. 533–546, 2017.

[37] S. V. Fotin, Y. Yin, H. Haldankar, J. W. Hoffmeister, and S. Periaswamy, "Detection of soft tissue densities from digital breast tomosynthesis: comparison of conventional and deep learning approaches," in *Proc. SPIE*, vol. 9785, 2016.

[38] T. Kooi, G. Litjens, B. van Ginneken, A. Gubern-Mérida, C. I. Sánchez, R. Mann, A. den Heeten, and N. Karssemeijer, "Large scale deep learning for computer aided detection of mammographic lesions," *Medical Image Analysis*, vol. 35, pp. 303 – 312, 2017.

[39] J. Arevalo, F. A. González, R. Ramos-Pollán, J. L. Oliveira, and M. A. Guevara Lopez, "Representation learning for mammography mass lesion classification with convolutional neural networks," *Computer methods and programs in biomedicine*, vol. 127, pp. 248–257, 2016.

[40] D. C. Moura and M. A. Guevara Lopez, "An evaluation of image descriptors combined with clinical data for breast cancer diagnosis," *International Journal of Computer Assisted Radiology and Surgery*, vol. 8, pp. 561–574, Jul 2013.

[41] M. A. Guevara Lopez, N. Posada, D. C. Moura, R. R. Pollán, J. M. F. Valiente, C. S. Ortega, M. Solar, G. Diaz-Herrero, I. Ramos, J. Loureiro, *et al.*, "BCDR: a breast cancer digital repository," in *15th International Conference on Experimental Mechanics*, 2012.

[42] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” *CoRR*, vol. abs/1512.00567, 2015.

INDEX OF TERMS

A

SUPPORT MATERIAL

A.1 INCEPTIONV3 ARCHITECTURE

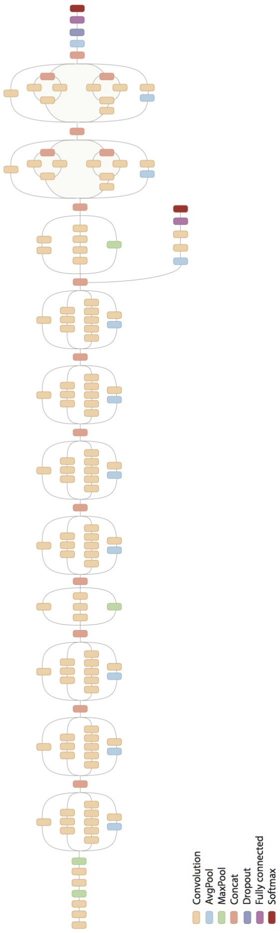

Figure 12.: InceptionV3 Architecture.

B

DETAILS OF RESULTS

B.1 TRAINING LOGS

...

C

LISTINGS

C.1 INTENSITY GRADIENT COMPUTATION IN PYTHON

```python
def compute_intensity_gradient(image, min_gradient=0):
    """
    Given an intensity single-channel image it computes the corresponding
    intensity gradient. The gradient image provides the maximum difference
    between each pixel and its immediate neighbours (in row or column)
    as a 2D image.

    Pixels where the gradient value is lower than min_gradient are set to zero
    """
    N,M = image.shape
    imWork = image.copy().astype(np.float16)
    gradImag = np.zeros_like(image)

    iWork_rowprev = np.pad(imWork[1:,:] - imWork[:-1,:], ((1,0),(0,0)), '
        constant')
    iWork_rownext = np.pad(imWork[:-1,:] - imWork[1:,:], ((0,1),(0,0)), '
        constant')

    iWork_colprev = np.pad(imWork[:,1:] - imWork[:,:-1], ((0,0),(1,0)), '
        constant')
    iWork_colnext = np.pad(imWork[:,:-1] - imWork[:,1:], ((0,0),(0,1)), '
        constant')

    canvas = np.zeros([N,M,4])
    canvas[:,:,0] = np.abs(iWork_rowprev)
    canvas[:,:,1] = np.abs(iWork_rownext)
    canvas[:,:,2] = np.abs(iWork_colprev)
    canvas[:,:,3] = np.abs(iWork_colnext)

    gradImag = np.amax(canvas, axis=-1)
    gradImag[gradImag<min_gradient]=0
    return gradImag
```

D

TOOLING

D.1 DISSERTATION CONCEPTUAL MAP

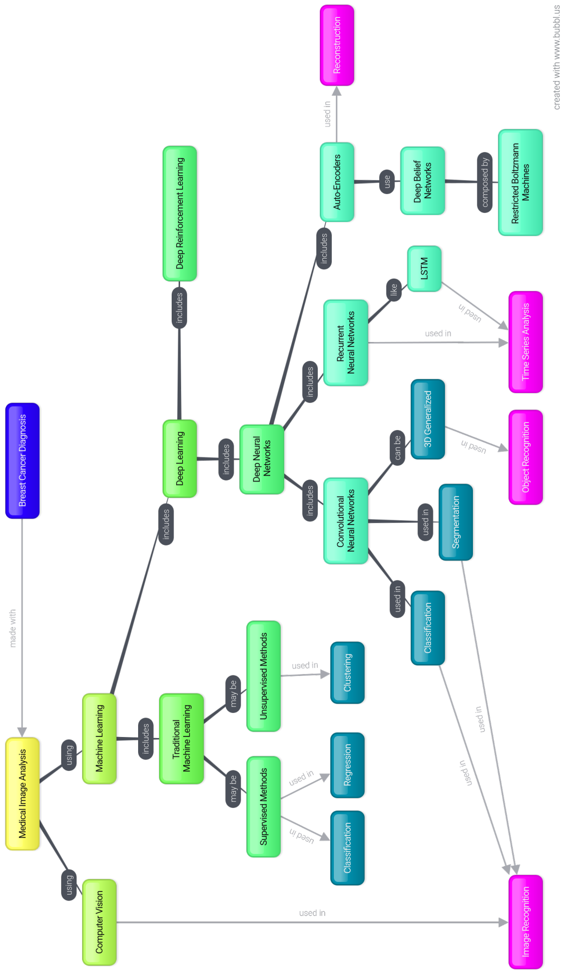

Figure 13.: Dissertation Conceptual Map.