UCLouvain

École polytechnique de Louvain

Deep learning for colorectal cancer classification using dual energy CT

Author: **Thomas KHMIELNITZKY**
Supervisor: **Benoît MACQ**
Readers: **Etienne DANSE, John LEE, Benoît MACQ**
Academic year 2020–2021
Master [120] in Electrical Engineering

Abstract

The objective is to predict the presence of budding, tumor stage, microsatellite instability (MSI) status, vascular/lymphatic permeation, peri-nervous sheathing, KRAS/BRAF mutations, and grade of colon cancer on preoperative dual-energy CT imaging using radiomic analysis. This retrospective study consisted of radiomic analysis of preoperative dual-energy CT imaging of patients undergoing colon cancer resection. Radiologist Etienne Danse manually segmented the tumor region on dual-energy CT images. Pre-processing included resolution homogenization, correction of segmentations, and conversion of the RGB color images to a new color management system. 700 traditional radiomic features, 22428 traditional radiomic features after filtering (MM features) as well as 366864 deep features were extracted from the tumor region. Several prediction models were developed by varying the pre-processing method and the classifier used. The validation method was a partially nested cross validation. The performance of the models was evaluated using AUC, metrics (accuracy, F1, precision, recall), learning curve. From a total of 72 patients, 31 were segmented and 28 were finally selected in the final dataset. Most prediction models performed better with data pre-processing including segmentation correction and resolution homogenization. Most of the models did not require MM features and deep features. The models for prediction of budding and grade showed more than encouraging results (respectively, AUC of 0.92 with 80% accuracy and AUC of 0.97 with 90% accuracy). Preoperative prediction of the presence of budding, tumor stage, MSI status, vascular/lymphatic permeation, peri-nervous sheathing, KRAS/BRAF mutations and grade of colon cancer by radiomic analysis of the preoperative DECT scan adds specificity to the clinical assessment and may contribute to individualized treatment selection. In addition, the new color management system appears to concentrate information from the effective atomic number map into almost 3 times less data in some cases.

Acknowledgments

First of all, I would like to thank Mr. Macq and Elliot Brion for their patience and their follow-up throughout the year. They allowed me to direct my research towards exciting subjects while giving me great confidence. I also thank Maxime Laurent and Guillaume Gheysen for their proofreading all along this month of July and August in order to advise me on a coherent structure to give to this thesis.

Then, I thank Junia Bertsch and my father for their attentive listening with this constant desire to understand and support me during this trying period.

Finally I thank Mr. Danse for his patience and his curiosity on this project. A project that would surely never have seen the light of day without his active application, whether for the quality of the manual segmentations provided or for his reactivity in answering during the week or on weekends!

Contents

List of Abbreviations v
List of Figures vi
List of Tables ix

Introduction **1**

1 Introduction **1**
1.1 State of the art 2
1.1.1 Image acquisition 3
1.1.2 Segmentation 6
1.1.3 Pre-Processing 6
1.1.4 Features extraction 6
1.1.5 Features Selection 7
1.1.6 Outcome Prediction 8
1.2 Summary of Chapter 1 9

2 Materials and Methods **11**
2.1 Data Acquisition and Segmentation 11
2.1.1 Final Dataset 12
2.2 Pre-processing 13
2.2.1 Homogenization format / size / orientation 14
2.2.2 Rigid registration 14
2.2.3 RGB to HSV 15
2.2.4 Cropping 17
2.2.5 Homogenise the resolution among the different cases 18
2.2.6 Segmentation correction 21
2.3 Features extraction (Part 1: Traditional features) 21
2.3.1 Shape features 21
2.3.2 First-Order statistical features 22
2.3.3 Higher-Order statistical features 22
2.4 Features extraction (Part 2: MM features) 23
2.4.1 Theoretical Foundations 24
2.4.2 Erosion and Dilation 25
2.4.3 Opening and Closing 26
2.4.4 MM features 27

2.5 Features extraction (Part 3: Deep features) . . . 29
2.5.1 Overfitting . . . 30
2.5.2 Pre-training . . . 32
2.5.3 Training . . . 32
2.5.4 Augmentations . . . 34
2.5.5 Optimizers . . . 35
2.5.6 Deep features extraction . . . 39
2.6 Features Selection . . . 40
2.7 Prediction Outcome . . . 43
2.7.1 Model for Prediction . . . 43
2.7.2 Algorithm Validation . . . 50
2.8 Summary of Chapter 2 . . . 51

3 Results **53**
3.1 Choice of models . . . 53
3.1.1 Pre-processing techniques comparison . . . 53
3.1.2 Combinaisons comparaison . . . 55
3.1.3 Final models . . . 61
3.2 Models Evaluation . . . 64
3.2.1 Model metrics plot . . . 64
3.2.2 Confusion matrix . . . 65
3.2.3 Learning curves . . . 65
3.2.4 ROC plot . . . 65

4 Discussion and Conclusion **69**
4.1 Interpretations . . . 69
4.2 Implications . . . 70
4.3 Limitations and Recommendations . . . 71
4.3.1 Segmentation . . . 71
4.3.2 Pre-processing . . . 71
4.3.3 Feature extraction . . . 71
4.3.4 Feature selection . . . 72

Appendices **73**

A Colorectal cancer **74**

B Image types after DECT acquisition **76**
B.1 Monochromatic image . . . 76
B.2 VNC . . . 77
B.3 Z_{eff} . . . 77
B.4 Map iode . . . 78

C List of patients **79**

D Features from Traditional Radiomics **82**
D.1 Shape . . . 82
D.2 First Order . . . 82

D.3 Higher Order features 83

E Model summary: VGG16 **87**

Bibliography **89**

List of Abbreviations

AUC Area Under the Curve

CRC Colorectal cancer

CT Computed Tomography

ctDNA Circulating tumor DNA

DECT Dual-Energy Computed Tomography

DFT Discrete Fourier transform

GLCM Gray Level Co-occurrence Matrix

GLDM Gray Level Dependence Matrix

GLRLM Gray Level Run Length Matrix

GLSZM Gray Level Size Zone

HSV Hue, Saturation, Value

HU Hounsfield scale

KNN K-nearest neighbors

MCC Matthews Correlation Coefficient

MM Mathematical Morphology

MRI Magnetic Resonance Imaging

MSI Microsatellite Instability

NRRD Nearly raw raster data

PCA Principal Component Analysis

PET Positron Emission Tomography

RFE Recursive Feature Elimination

RGB red, green, blue

ROI Region of Interest

SE Structuring Element

SGD Stochastic gradient descent

SMOTE synthetic minority oversampling technique

SVM Support Vector Machine

US Ultrasound

VGG Visual Geometry Group

VNC Virtual non-contrast

List of Figures

1.1 Representation of traditional radiomics 4

2.1 Diagram showing the data acquisition and segmentation process 11
2.2 Colour gauge informing which colour corresponds to which effective atomic number. 12
2.3 Diagram showing the data pre-processing 13
2.4 a) Slice of the exported DICOM segmentation with a shift that prevents radiomics from working properly; b) Shifted DICOM segmentation slice using rigid registration requiring a reference image to rely on; c) A slice of the 3D image used as a reference for the segmentation so that the segmentation can be aligned with the other image types via rigid registration 15
2.5 Example of false colors. From left to right, the original image, the result of a marginal median filter, and of a vector (lexicographical) median filter. 15
2.6 Example of false colors. From left to right, the original image with the appearance of false 'colours' such as grey, the result of a marginal erosion filter, and of a vector (lexicographical) erosion filter. 16
2.7 Example of false colors. From left to right, the original image with the appearance of false 'colours' such as white, the result of a marginal dilation filter, and of a vector (lexicographical) dilation filter. 16
2.8 Tumor of sample n°1 after cropping from the YZ, XZ, XY plane viewpoint. . . 18
2.9 Pictures from left to right: Low-pass Butterworth filter gains $H(u, v)$ as a function of frequency (u, v) 19
2.10 Pictures from left to right: Image of the region of interest after pixel spacing homogenisation; Image of the region of interest after homogenising the pixel spacing and using the Low-pass Butterworth filter 19
2.11 Succession of 3 slices of the same body level ($z = 0$ for lower body and $z = z_{max}$ upper body): segmentation after resolution homogenization; monochromatic low keV image, monochromatic low keV image with resolution homogenization and Butterworth low pass filtering. 20
2.12 Illustrative examples of basic SEs with increasing size λ 25
2.13 Grey-scale erosion with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$ 25
2.14 Grey-scale dilation with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$ 26
2.15 Grey-scale opening with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$ 26

2.16 Grey-scale closing with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$ . . . 27
2.17 Grey-scale opening with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$ for a), b), c), d) . . . 27
2.18 **a)** 120 keV image; **b)** $\gamma_{\lambda=1}(f) - \gamma_{\lambda=3}(f)$; **c)** $\gamma_{\lambda=3}(f) - \gamma_{\lambda=5}(f)$ . . . 28
2.19 Picture from a texture dataset representing corrugated steel. Dimensions: 512 × 512. . . . 29
2.20 The resulting non normalized morphological covariance $K^{\overrightarrow{v(f)}}$ plots for 2 directions where f is the picture representing the corrugated steel. . . . 29
2.21 Diagram showing the Deep features extraction process . . . 30
2.22 a) A standard neural network with two hidden layers; b) Example of a network after inactivation of some randomly selected neurons produced by applying dropout to the network on the left. [45] . . . 31
2.23 model vgg16 . . . 32
2.24 Pictures from Dataset 1 related to classes such as a) tumor; b) stroma; c) complex; d) lympho; e) debris; f) mucosa; g) adipose; h) empty . . . 33
2.25 Textures from Dataset 2 linked to different classes such as a) brick; b) fabric; c) fence; d) floor; e) ground; f) metal; g) misc; h) roof; i) siding; j) skin; k) ;stone l) test; m) wall; n) wood . . . 35
2.26 2 cases of stochastic gradient descent where the step applied iteratively with the help of mini-batches is either too small (probable convergence towards a local minimum) or too large (global minimum difficult to reach due to the divergence) 36
2.27 Diagram showing the Features selection process . . . 42
2.28 A rough guide on how to approach problems with classification estimators to try on the data [56, 57]. . . . 43
2.29 Pictorial explanation of the classifier *knn* [17] . . . 44
2.30 Pictorial explanation of the classifier *Naive Bayes* [17] . . . 45
2.31 Pictorial explanation of the classifier *Logistic Regression* [17] . . . 46
2.32 Pictorial explanation of the classifier *SVM* . . . 47
2.33 Maximum-margin hyperplane and margins for an SVM trained with samples from two classes [60]. . . . 48
2.34 Pictorial explanation of the classifiers *Decision Tree* and *Random forest* [17] . . . 50
2.35 Schematic approach used to apply feature selection, pre-processing selection and classifier selection. . . . 51

3.1 The first two PCAs of a set of deep features extracted from a pre-trained vgg16 model by imagenet. The two classes separate methylated and unmethylated glioblastoma patients . . . 55
3.2 from left to right: a) image of a dataset intended to train a model without augmentation; b) the same image augmented via the rotation principle. A 45° rotation is applied in this case . . . 58
3.3 a) confusion matrix, b) precision and c) loss of the model trained by dataset 1 only; d) confusion matrix, e) precision and f) loss of the model trained by dataset 2 then 1; g) confusion matrix, h) precision and i) loss of the model trained by dataset 2 only; j) confusion matrix, k) precision and l) loss of the model trained by dataset 1 then 2. . . . 59

3.4 Features selection, Final selection of features for each prediction. Analysis of the features that appeared most often in the significant feature set. In red, features that appeared enough times to be selected in the final feature set for KRAS/BRAF mutations (a) traditional and b) deep) ; vascular/lymphatic permeation (c) traditional) ; MSI status (d) traditional) ; peri-nerve sheathing (e) traditional) ; budding (f) traditional and g) MM) ; stage (h) traditional) ; grade (i) traditional, j) MM, and k) deep). 62
3.5 Confusion Matrix, on the basis of the final set of features, generated with a repeated Stratified K Fold as a cross-validation generator for a) KRAS/BRAF mutations b) Vascular/lymphatic permeation c) MSI status d) Peri-nerve sheathing e) Budding f) Stage g) Grade 66
3.6 Learning curves, on the basis of the final set of features, generated with a repeated Stratified K Fold as a cross-validation generator for a) KRAS/BRAF mutations b) Vascular/lymphatic permeation c) MSI status d) Peri-nerve sheathing e) Budding f) Stage g) Grade 67
3.7 ROCs, obtained on the basis of the final set of features, generated with a repeated Stratified K Fold as a cross-validation generator for a) KRAS/BRAF mutations b) Vascular/lymphatic permeation c) MSI status d) Peri-nerve sheathing e) Budding f) Stage g) Grade 68

4.1 1) Caecum; 2) Ascending colon; 3) Transverse colon; 4) Descending colon; 5) Sigmoide colon; 6) Rectum 72

B.1 Types of images acquired with the DECT. a) monochromatic image at low Kev (40keV); monochromatic image at high Kev (40keV) 76
B.2 Type of images acquired with the DECT. Image with virtual non contrast (VnC) 77
B.3 Type of images acquired with the DECT. A map of the atomic number Z_{eff} studying the distribution of a material within a tissue 77
B.4 Type of images acquired with the DECT. An iodine concentration chart to measure the number of mg of iodine per ml at within the tissue 78

E.1 calculation details of the convolutional layer conv2d_10 [78] 88

List of Tables

1.1 Pros and cons of clinical imaging techniques (from an imaging point of view) [18, 19, 21] 4

2.1 Table of each case (patient) including metadata (resolution), clinical features (gender, age, tumour location) as well as different sets of output values (stage, grade, vascular/lymphatic permeation, peri-nervous sheathing, budding, micro-satellite instability and KRAS/BRAF mutations) 13
2.2 (a) Table representing a 4x4 image composed of pixels with 4 gray levels, (b) Table CM listing the pairs of similar pixels of the image **(a)**. These pairs of pixels are defined by the orientation that the pixels form (θ=0°here), the distance between them (δ=1 here), the value of the two pixels [42]. 22
2.3 (a) Table representing a 4x4 image composed of pixels with 4 gray levels, (b) Table RLM listing the groups of consecutive similar pixels in the image (a). These groups of pixels are defined by the orientation that the pixels form (θ=0°here), the value and the number of these consecutive pixels [42]. 22
2.4 (a) Table representing a 4x4 image composed of pixels with 4 gray levels, (b) Table $LSZM$ listing the areas of similar pixels in the image (a). These pixel groups are defined by the number and common value of these pixels [42]. . . . 23
2.5 (a) Table representing a 5x5 image composed of pixels with 5 gray levels, (b) Table LDM listing the dependency of pixels similar to a central pixel of the image (a). These pixel groups are defined by the number of pixels around the central one and the common value of these pixels [42]. 23
2.6 Dataset 1 and Dataset 2 34

3.1 The pre-process column distinguishes between pre-processes integrating a resolution homogenization (R for yes and noR for no), and a segmentation correction (S for yes and noS for no). Classifier informs about the classifier used. Average Ranking calculates the "place on the podium". A pre-processing type that is the best for each classifier used will have an average ranking of 1. A pre-processing type that is always the worst for each classifier will have an average ranking of 4 because this table only includes 4 pre-processing types. . . 54
3.2 model pretrained imagenet, trained by dataset 1. The four columns group the results obtained with (Aug) or without (No Aug) augmentation during training. Augmentation always includes the flip and a rotation. Either the rotation can be done with any angle, or it is limited to multiple 90° rotation angles or multiple 90° rotation angles with an additional 5° margin. 56

3.3 model pretrained imagenet, trained by dataset 2 (balanced) and then trained by dataset 1. The four columns group the results obtained with (Aug) or without (No Aug) augmentation during training. Augmentation always includes the flip and a rotation. Either the rotation can be done with any angle, or it is limited to multiple 90° rotation angles or multiple 90° rotation angles with an additional 5° margin. 56
3.4 model pretrained imagenet, trained by dataset 2 (balanced). The four columns group the results obtained with (Aug) or without (No Aug) augmentation during training. Augmentation always includes the flip and a rotation. Either the rotation can be done with any angle, or it is limited to multiple 90° rotation angles or multiple 90° rotation angles with an additional 5° margin. 57
3.5 model pretrained imagenet, trained by dataset 1 and then trained by dataset 2 (balanced). The four columns group the results obtained with (Aug) or without (No Aug) augmentation during training. Augmentation always includes the flip and a rotation. Either the rotation can be done with any angle, or it is limited to multiple 90° rotation angles or multiple 90° rotation angles with an additional 5° margin. 57
3.6 Table of results (average accuracies) obtained for different feature set combinations. Average accuracies include a final feature set of 3, 5 or 7 features. 60
3.7 Summary of all previous steps: the features chosen for each prediction model with the most suitable hyperparameters (if any). 61
3.8 distribution of features in percent: The traditional features, The HSV channels of the effective atomic number map, The different types of filtering applied to the image before feature extraction, The selected planes from which the deep features are extracted, and the combination of spectral data (n°1: 120 keV, 40 keV, vnc; n°2: 120 keV, 40 keV, iodine; n°3: Z_{eff} h/hsv, Z_{eff} s/hsv, Z_{eff} v/hsv) 63
3.9 Table grouping the different output sets and data quantities for each class . . . 64
3.10 Metrics obtained after the partially nested cross validation for each target . . . 64

D.1 Shape features 82
D.2 First Order features 82
D.3 (a) picture, (b) Example of a CM for a 4x4 image and for 4 gray levels with for a displacement vector (0,1):a distance δ=1 (considering pixels with a distance of 1 pixel from each other) and angle θ=0°, with matrix named C(i,j) [42] 83
D.4 First Order 83
D.5 (a) picture, (b) Example of a RLM for a 4x4 image in 0°direction and for 4 gray levels with matrix named R(i,j) [42] 84
D.6 GLRLM 84
D.7 (a) picture, (b) Example of a LSZM for a 4x4 image and for 4 gray levels with matrix named Z(i,j) 85
D.8 GLSZM 85
D.9 (a) picture, (b) Example of a LDM for a 5x5 image and for 5 gray levels with matrix named N(i,j) 86
D.10 GLDM 86

Chapter 1

Introduction

CRC (colorectal cancer) is the third most common cancer and the second most common cause of cancer death worldwide.

Recently, advances in therapeutic strategies have played a crucial role in improving survival [1, 2]. The onset and development of CRC is accompanied by a series of genetic abnormalities, among which microsatellite instability (MSI), KRAS/BRAF mutations, Vascular/lymphatic permeation, Peri-nervous sheathing and budding (see appendix A for further development of these abnormalities). Genetic profiling of tumours is a powerful tool that allows to obtain more diagnostic clues and guide treatment strategies (endoscopy, surgery, neoadjuvant chemotherapy[1]).

For example, achieving **MSI status** is necessary because MSI CRC tissues have special biological behaviours, they are more likely to have a better prognosis and benefit from immunotherapy[2]. Furthermore, cancers with MSI status may be resistant to fluorouracil chemotherapy [4].

Determining the **stage** of the tumour can also be useful for the treatment stage. For early stage (stages I and II), resection surgery is considered the most common treatment option, while chemotherapy is usually the main treatment option for patients with advanced CRC (stages III and IV) [5].

Tumor budding has been recognized as an excellent index of aggressiveness in rectal cancer [6]. It is associated with a risk of lymph node metastasis in patients with superficial colorectal cancer. When the cancer is infiltrating and reaches the lymph nodes, surgical treatment is recommended, with lymph node dissection and is associated with "adjuvant chemotherapy". Finally, if the colon cancer has metastasised, the treatment will consist of surgery of the colon completed by surgery of the metastasised organs, followed or preceded by chemotherapy and sometimes targeted therapy [2].

[1] Neo-adjuvant chemotherapy is given before surgery or radiotherapy. Its aim is to reduce the size of the tumour. Adjuvant chemotherapy is administered after surgery or radiotherapy with the aim of eliminating any residual cancer cells [3].

[2] Immunotherapy: Treatment to increase or induce the body's immunity by injecting antibodies or antigens

KRAS/BRAF mutations predict a lack of response to cetuximab and panitumumab, which are anti-epidermal growth factor receptor (EGFR) monoclonal antibodies [7, 8]. Therefore, before or during treatment, identification of KRAS/BRAF mutational status is crucial to predict therapeutic effect and determine individual treatment strategies for colorectal cancer patients. To do this, the patient is genotyped. This practice is a reference in the medical field. However, collected tissues after biopsy may not represent genotypic changes since collection and samples may be limited by intratumoral heterogeneity [8].

Thus, the development of a **non-invasive**, easily repeatable method that can reflect intratumoural heterogeneity to help identify gene mutation status is important to provide a complement to real-time histological assessment. Since DNA from each of our cells circulates in small quantities in our blood, tumour DNA can be found in the blood if the patient has cancer. Tumour DNA can be found in the bloodstream if the patient has cancer. This is called circulating tumour DNA (ctDNA). Circulating DNA analysis could be a non-invasive method of genotype analysis in colorectal cancer [8]. However, there are limitations related to the inaccuracies of the method. ctDNA can be found in the bloodstream following the destruction of a tumour - the body's defence system can then successfully catch and eradicate the threat. Low signal-to-noise ratio may prevent ctDNA detection. The method is not yet fully reliable [9].

The use of advanced medical imaging to complement traditional diagnostic methods has the potential to better assess the spatial heterogeneity and change over time of tumours. **Radiomics** is an emerging approach that converts medical images into usable data and generates thousands of quantitative features [10]. The combined analysis of a multiple feature panel has been used in the prediction or prognosis of colorectal cancer, head and neck cancer, and lung cancer [4, 11, 12].

Meanwhile, the advanced deep learning method has become the main approach for radiomic analyses based on big data medical imaging [13, 14]. Theoretically, the combination of dual-energy CT and the deep learning method can potentially improve the predictive performance. The detection of certain anomalies is therefore an integral part of the approach to personalize the therapy. The aim of this study is to build a radiomic feature set based on dual energy CT as a means of acquiring medical images, using deep learning and many other concepts to optimise preoperative assessment and consequently develop the best possible targeted therapies.

1.1 State of the art

In conventional radiology practice, except for a few measurements like size and volume, the imaging data sets are generally evaluated visually or qualitatively [1]. This approach not only involves intra- and interobserver variability but also leaves a very large amount of hidden data in the medical images unused. For example, two patients could have tumors with quite different qualitative characteristics such as size, shape, boundaries and heterogeneity. The

[1] Qualitative data is data that cannot be assigned a value or characteristic. Quantitative data are data that can be measured (height, weight...) or tracked (temperature...) [15]

survival of the patients in this scenario will probably be different even though the tumors have histopathologically similar features. If one could have predicted the prognosis of the patients before any intervention or treatment, the management of the patients would be different. This is actually called *precision medicine*.

In order to have the most optimal treatment possible, it is vital to get an accurate diagnosis from the image analysis. Indeed, good pre-operative treatment increases the probability of a successful operation. The characteristics of cancer are currently defined from a biopsy [1] of the tumour. The development of a non-invasive, cost-effective method of predicting tumour characteristics could be useful to clinicians to provide more diagnostic clues and guide subsequent treatment strategies. The radiomics technique is therefore used [16]. The primary purpose of the radiomics is to extract as much and meaningful hidden objective features as possible to be used in decision support. There are several ways to obtain radiomics tools: the paid software programs, free software programs as GUI (with a graphical interface), MaZda, LIFEx, PyRadiomics, IBEX. Some of them use AI like Waikato, WEKA, Orange data mining software, RapidMiner, Rattle in R statistics and Deep learning studio [14, 17].

Radiomic analysis converts medical images into usable high-dimensional data to quantitatively and comprehensively describe tissue characteristics from the imaging [4].

These radiomic features are further used in creating statistical models with an intent to provide support for individualized diagnosis and management in a variety of organs and systems such as brain, pituitary gland, lung, heart, liver, kidney, adrenal gland, prostate and colon [4, 11, 12].

Radiomics can be applied to various imaging techniques including computed tomography with or without variable energy, magnetic resonance imaging (MRI), positron-emission tomography, X-ray, and ultrasonography. Radiomics is divided into 6 main parts. First, there is **data acquisition**. Then a **segmentation** of the tumour must be applied in order to focus mainly on the tumour itself. A **pre-processing** is applied to the images in order to optimise the quality of the following process. Once this step is finished, we **extract features** from the tumour. These features provide hidden information that can potentially be decisive in predicting outcomes. In order to sort out the useful and useless features for a good prediction, the next step aims at **reducing the set of features** on which we work. Finally, with the remaining features, we use the most adequate classification model to obtain the best possible **predictions** (e.g. defining the presence or not of certain anomalies in a non-invasive way).

1.1.1 Image acquisition

Radiomics is a fundamental application when it comes to computed tomography (CT), magnetic resonance (MR) or positron emission tomography (PET) studies. Indeed, the human eye is not able to extract all the information held in these complex biomedical images [16].

[1] A biopsy is an examination in which tissue is removed for analysis under a microscope

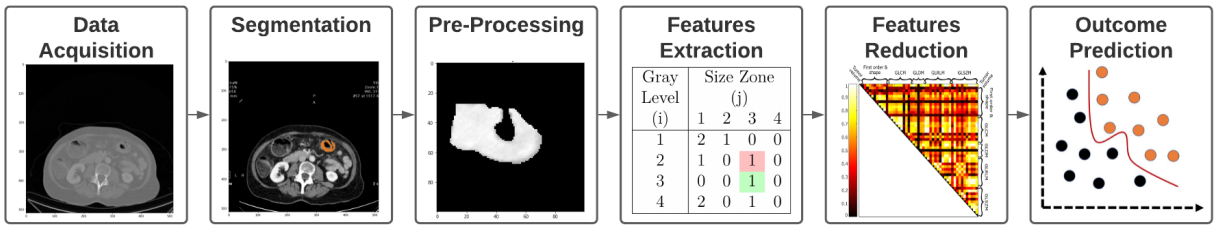


Figure 1.1: Representation of traditional radiomics

First, Magnetic Resonance Imaging (MRI) is a type of scanner that uses strong magnetic fields and radio waves to produce detailed images of the inside of the body.

Then we have Computed Tomography (CT) which is a diagnostic tool that uses a series of x-rays and a computer to produce a 3D image of soft tissue and bone.

Third, Positron Emission Tomography (PET) is an imaging test that reveals the function of tissues and organs. PET uses a radioactive drug (tracer) to show this activity. This test can sometimes detect a disease before it appears on other imaging tests.

The use of different **image acquisition** and processing techniques (e.g. acquisition of some data via MRI and others via CT) can have a significant impact on radiomics. Indeed, radiomics extracts its features by studying the relationship that voxels [1] have with each other and depending on the techniques used, the resolution, image noise, etc. may vary from one data to another. Computed tomography, for example, is a type of imaging that has a much better resolution than PET imaging (of the order of $100\,\mu m$ for CT imaging against a resolution of $5\,mm$ for PET imaging [18]). These differences can therefore lead to inconsistent results in radiomic analyses of independent data sets, which is one of the main problems of radiomics. It is therefore important to have some standardisation in the data acquisition process [19]. Within the same acquisition technique, differences in the image may also be added related to the acquisition parameters.

Standard phantoms are used to evaluate imaging performance and to determine the extent to which image quality depends on the technique adopted (technique characterized by acquisition settings). The standard phantom is an artificial structure that mimics the properties of human tissue. It is scanned on several machines in order to characterize the scan result against a known physical standard [20]. The use of these standard phantoms can therefore provide useful information on the parameters potentially affecting the image texture.

The respective pros and cons of each modality are described in Table 1.1.

Table 1.1: Pros and cons of clinical imaging techniques (from an imaging point of view) [18, 19, 21]

Techniques	**Pros**	**Cons**

[1] The voxel (portmanteau word created by contracting "volume" and "element") is to 3D what the pixel is to 2D (portmanteau word created by contracting "picture" and "element").

CT	• Accurately spatial information ($100\mu m - mm$) • Electron density information for dosimetry[1]	• Sub-optimal soft tissue imaging
MRI	• True multi-planar capability (able to obtain images in any oblique plane). • Superior soft tissue imaging with excellent spatial resolution	• Limited availability of suitable phantoms • Variable scanner signal intensity (variability of voxels value) → normalisation needed OR some radiomic features not usable
PET	• May have diagnostic value detecting metastatic lesions that would have been missed on conventional imaging	• Spatial resolution ($4.5 - 5\,mm$ at the center) → may require an increased number of patients to generate a meaningful association with clinical endpoints • Calibration of scanners and standardisation of protocols emerges but not widely applied

Given the increasing number of applications in clinical diagnosis, dual-energy computed tomography (DECT) has been considered an important step in CT imaging as it provides quantitative measurements to characterise lesions [22]. It allows for improved diagnostic techniques and reduced exposure to radiation. DECT can generate accurate iodine-based material decomposition (MD) images, which can reflect the vascularity of various tissues by measuring the concentration (CI) of the contrast medium (iodine) [22]. All of these image types are discussed in more detail in Appendix B. It is since 2015 that the first scientific articles proposing the use of DECT imaging in radiomics appear.

Tomography provides a 3-dimensional (3D) description of the internal structures of an object from a series of 2-dimensional radiographs. The use of an **iodinated contrast agent** is required during CT imaging. It allows better visualisation of certain target areas. For example the tumour region, due to the presence of a high atomic number element, iodine, strongly attenuates X-rays.

The **Hounsfield scale** is the quantitative scale for describing radiodensity [23]. It is the unit for what we call the *CT number*. This CT number is a calculated value reflecting the X-ray attenuation coefficient in an image voxel. To calculate this value, we use the **linear attenuation coefficient (μ)** which is a constant that describes the fraction of attenuated incident photons in a monoenergetic beam per unit thickness of a material. To go into a little more detail, the unit of measurement *HU* is based on a scale created by linearly transforming

[1]Dosimetry: quantitative determination of the absorbed dose to an organism, the energy received per unit mass as a result of exposure to ionizing radiation.

the original *linear attenuation coefficient* measurement into a scale where the radiodensity of distilled water at standard temperature and pressure (STP) is set to 0 HU and that of air to -1000 HU. Below is the definition of HU:

$$HU = 1000 \cdot \frac{\mu - \mu_{water}}{\mu_{water}} \tag{1.1}$$

Different regions of interest in the body (e.g. blood, bone,...) do not react in the same way to X-rays of variable energy. These different reactions allow us to identify the nature of the regions of interest. For instance, iodine in blood shows higher CT values at lower keV, while fat reveals lower CT values at lower keV. In contrast, muscle demonstrates almost constant CT values in range of 40–190 keV [22].

1.1.2 Segmentation

Image **Segmentation** is used to focus only on the volume of interest (VOI). This VOI is a cancerous area in the radiomic case. Focusing only on it reduces the computer workload. This process is challenging because of the fact that some tumors have a very unclear margin. The manual segmentation is a reference provided that it is performed by experts, which is very time-consuming. On the other hand, manual segmentation is subject to intra- and inter-reader variability, leading to radiomic feature reproducibility problems. To avoid this variability, a few automatic and semi-automatic methods have been described as follows: active contour (snake) methods, level set methods, region-based methods, graph-based methods, and deep learning-based methods [24].

1.1.3 Pre-Processing

There are some parameters to consider when starting a radiomic process. This is the **pre-processing**. The most important of those that need to be dealt with in any imaging modality are the size of the pixel or voxels and number of the gray levels. Pixel resampling can be done using various interpolation methods such as linear and cubic B-spline interpolation [10, 17].

1.1.4 Features extraction

The next step is to **extract features**. It is important to consider that, with the exception of some histogram or first-order features, attempting to define each radiomic feature in a clinical context is likely to result in failure. There are two categories of radiomic features: *traditionnal* and *deep features* [17]. The first ones are predefined or hand-crafted features, being created by human image processing experts.

It can be divided into five groups: size and shape based–features, descriptors of the image intensity histogram, descriptors of the relationships between image voxels derived textures, and fractal [1] features.

One way to extend this process is to extract the same features on these images after filtering. Image filtering is therefore a way to increase the number of image types from which features can be extracted. Searchers from the University of Louvain-La-Neuve worked on the integration of mathematical morphology [25, 26] into their feature extraction based on the work of Sébastien Lefèvre and Erchan Aptoula [27]. The objective is in part to use non-linear, non-invertible and indempotent morphological operators to filter images and then extract features.

The second category of radiomic features is deep features, which has gained popularity nowadays because some deep learning algorithms design and select the features themselves for a given task within its layers, without need for any human intervention [28, 29]. Some recent works have also suggested the superiority of the deep features to traditional features [28]. Work has also been undertaken using 3D convolutional neural networks [30].

Experiments were conducted on the possibility of combining traditional and deep features extracted from CT scans. The prediction accuracy was found to be higher after the combination. [31]. It was also found in a research that clinical data (tumour location, age, etc.) used as features gave more robust prediction models [32].

1.1.5 Features Selection

When features are extracted, we can begin the Radiomic data handling.

The purpose of dimension reduction is to retain some meaningful properties of the original data. Working in high-dimensional spaces can be undesirable for several reasons: it could lead to overfitting and can quickly become computationally difficult. There are different approches such as feature reproducibility analysis, collinearity analysis, algorithm-based feature selection, and cluster analysis.

Feature reproducibility analysis should be performed to assess features that are sensitive to segmentation variability. For example, if radiologist A segments the tumour of patient x and another radiologist B segments the same patient x, it is likely that the segmentations will not be completely identical. Some differences in the way of segmenting may have a negative impact on some of the extracted features. As these features are sensitive to small variations in the segmentations, it is better to ignore them. This analysis can therefore only be performed if several professionals have segmented the same tumours.

Collinearity analysis is another way of dimension reduction because a very large number of the features have similar information and the extent of which is called the strength of collinearity.

[1] Fractal: Which represents fragmentary forms, revealing similar patterns at increasingly fine scales of observation

If two features have the same information, then it is not necessary to consider both features in the feature selection because only one of them is sufficient.

There are various algorithms-based feature selection with different functions such as least absolute shrinkage and selection operator, correlation-based feature selection algorithm, ReliefF, and Gini index. The researchers should experiment with these algorithms for achieving the best results.

The most confusing issue in dimension reduction is the final number of features that should be achieved. Although there is no guideline about this, it would be good to reduce the total number of features at least to one-tenth of the total labeled data [17].

It should be noted that radiologist Etienne Danse and Charline Jopart had already attempted to determine whether a link could be established between spectral data and the various parameters to be predicted such as grade, presence of mutation, MSI status, etc. Their approach consists in extracting features such as the mean, the maximum, the minimum value of the HU,... It appears that a 4-parameter model (tumour thickness, portal HU, VNC HU and effective Z) present in both observers, seems to be efficient in detecting MSI status with a sensitivity of 89% and a specificity of 88%.

1.1.6 Outcome Prediction

Model development can be done using various algorithms. The most common algorithms are *k-nearest neighbors*, *naive Bayes*, *logistic regression*, *support vector machine*, *decision tree*, *random forest*, *neural networks*, and *deep learning*. The main reason for using AI in radiomics is its better capability of handling a massive amount of data compared with the traditional statistical methods. AI algorithms are essentially used for classification problems. The field of radiomics needs much more powerful analytic tools, and AI appears to be a potential candidate for this purpose. Although the concept of AI goes back to 1950s, it has gained momentum since 2000 because of the advances in computational power. Today, AI technology provides numerous indispensable tools for intelligent data analysis for solving several medical problems, particularly for diagnostic issues. These algorithms can also be combined with meta-classifiers or ensemble techniques like adaptive boosting and bootstrap aggregation to enhance generalizability.

In order to obtain the best possible results, classification parameters need to be adjusted. To do it, it is necessary to have a validation dataset in addition to the training and test datasets. The model is initially fitted to a training dataset. The validation dataset provides an evaluation of a model fitted on the training dataset while tuning the model's hyperparameters. Finally, the test dataset is a dataset used to provide an unbiased evaluation of a final model fitted on the training dataset . The most common internal validation techniques that can be encountered in the literature are k-fold, leave-one-out cross-validation, and hold-out. In addition, there are much more sophisticated techniques such as random subsampling, bootstrap cross-validation, and nested cross-validation.

The sample size is an important factor to avoid some problems in model fitting. Nonetheless, in case of limited or small data, it should be known that there are some well-known augmentation techniques (e.g., image transformation in *deep learning*, SMOTE = synthetic minority over-sampling) to be considered as well.

Performance evaluation of the classifications is generally done using the area under the receiver operating characteristic curve (AUC). To define AUC, it is necessary to first define the receiver operating characteristic curve (ROC) which is obtained by placing the sensitivity (true positive rate TP) on the y-axis against the specificity (false positive rate FP) on the x-axis. An ROC curve that follows the diagonal line y=x produces false positive results at the same rate as true positives. This means that the model does not have the ability to separate the classes. The ideal for a model is to have as many true positives as possible and as few false positives as possible. Therefore, a good classification model has a good measure of separability calculated as the area under the ROC curve (AUC). AUC is an overall measure of the ability of a test to distinguish the presence and absence of a specific condition. An AUC of 0.5 represents a test that is unable to discriminate (i.e. no better than chance), while an AUC of 1.0 represents a test with perfect discrimination. It should be kept in mind that AUC might be a poor performance evaluator if the data set has a class imbalance. For this reason, other performance measures can be used such as:

- $Accuracy = \frac{TP+TN}{TP+FP+FN+TN}$
- $Specificity = \frac{TN}{TN+FP}$
- $Sensitivity/Recall = \frac{TP}{TP+FN}$
- $Precision = \frac{TP}{TP+FP}$
- $F1Score = 2 * \frac{(Recall*Precision)}{(Recall+Precision)}$
- $MCC = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$

Where TP is the true positive rate, FP the false positive rate, TN the true negative rate and FN the false negative rate.

It should be supplied for further assessment. In multiple comparisons, the multiplicity problem needs to be addressed. The best performing and stable classifier or classifiers are generally selected for the clinical application of interest.

1.2 Summary of Chapter 1

This first chapter will have allowed us to see more clearly the objectives of this thesis: to build 7 models aiming at detecting different specificities of the tumor allowing to better

"personalize" the treatment of the patients. The different steps of radiomics, the process used to reach our goal, are also developed.

Chapter 2

Materials and Methods

2.1 Data Acquisition and Segmentation

***In the framework of this master thesis**, we will use a new dataset exported from St-Luc Hospital with the great help of radiologist Etienne Danse and his colleagues. A dataset providing complete spectral imaging and characteristics (anomalies, stage, grade, etc) of each colorectal tumour patient. The segmentation of tumours will be entirely supervised by the radiologist Etienne Danse with the main tool, the 3DSlicer application.*

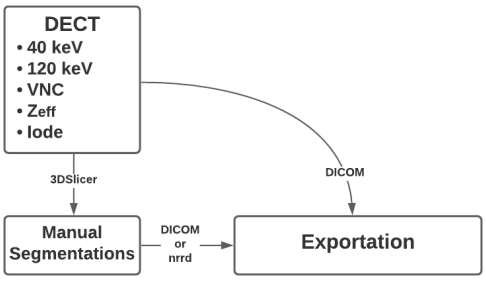


Figure 2.1: Diagram showing the data acquisition and segmentation process

The dataset is composed for each of the 72 patients of the Monochromatic images at 40 keV and 120 keV, the iodine map, the VNC, the effective atomic numbers Z_{eff}. All these DECT images are detailed in the appendix B. As for the segmentations, they were applied to 31 patients. The manual segmentation is added in a coloured form on the tomography at 120 keV as shown in figure 2.4 or provided in NRRD format in binary form. It was performed by radiologist Etienne Danse.

It was arbitrarily chosen to work with monoenergetic virtual imaging with a X-ray energy spectra of 40 keV and 120 keV (see fig. figure B.1). In other words, **a very low and very high energy spectrum**. As mentioned above, the fact that certain regions of the body do not react in the same way to X-ray emission at different energies means that different areas can be

highlighted with images at 40keV and 120keV.

2.1.1 Final Dataset

Initially, the data without segmentation was not taken into account. The data not having slices thickness equal to $3mm$ (but a thickness of $2mm$) were also not kept. An interpolation from $2mm$ to $3mm$ slices thickness is possible but there is no guarantee of quality. Finally, some tumours were hidden by certain informations (e.g. the colour gauge informing which colour corresponds to which effective atomic number visible in figure 2.2).

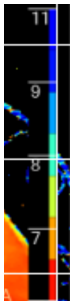


Figure 2.2: Colour gauge informing which colour corresponds to which effective atomic number.

We did not take the risk of falsifying the features with this kind of data. Finally, it should be noted that medicine tends to distinguish between seniors and juniors. The World Health Organisation statistically defines individuals as elderly from the age of 60 onwards and it would therefore be preferable to ignore patients younger than this age. The medical world, however, notes a "milestone at 45 or 50 years" [33, 34]. We will therefore consider senior status from the age of 45. But being in lack of example, we will nevertheless include a 43 years old patient.

From the initial 72 patients, we finally arrived at 28 patients.

Cases	Pixel size [mm]	Sex	Age	Localisation	Stage	H.G.	Perm.	E.P.	Bud.	MSI	Mutations
1	0.78125	W	84	Colonic angle L	3	1	1	0	1	0	/
2	0.80859375	M	84	Recto-sigmoid	4	0	1	1	1	0	/
4	0.828125	M	62	Sigmoide	4	0	1	1	1	0	0
4vrais	0.578125	W	70	Caecum	4	1	1	1	/	0	/
6	0.828125	M	60	Sigmoide	3	0	1	1	/	0	/
7	0.697265625	W	65	Colonic angle L	2	0	0	0	/	0	/
9	0.595703125	M	66	Sigmoide	2	0	0	0	1	0	0
11	0.82421875	M	73	Colon R	2	0	0	0	1	0	/
12	0.697265625	M	91	Sigmoide	3	0	1	/	0	0	1
51	0.630859375	W	77	Caecum	2	1	0	1	1	0	/
52	0.736328125	M	77	Sigmoide	4	1	1	0	0	0	0
53	0.703125	W	76	Caecum	2	1	1	0	/	1	1
54	0.662109375	W	56	Caecum	1	1	0	0	/	0	/
55	0.716796875	M	58	Splenic angle	2	0	1	0	1	1	0
56	0.697265625	M	85	Sigmoide	3	/	0	0	0	0	1
57	0.697265625	W	68	Colon R	2	/	0	0	0	1	1
58	0.673828125	W	72	Caecum	2	0	0	0	0	0	/
59	0.78125	W	85	Caecum	2	0	1	0	0	1	1
60	0.640625	W	43	Sigmoide	4	/	/	0	0	0	1
61	0.650390625	M	81	Sigmoide	2	1	1	1	1	0	/
62	0.69921875	M	66	Colon R	2	0	0	1	1	0	1
63	0.654296875	W	46	Transverse colon	2	0	1	1	1	0	1
64	0.8515625	W	71	Caecum	2	1	1	/	1	1	1
66	0.619140625	W	82	Caecum	2	0	0	0	/	0	1
67	0.5625	W	90	Colon R	2	0	0	0	0	0	/
69	0.69140625	W	65	Transverse colon	3	0	0	0	0	0	/
70	0.5703125	W	75	Caecum	2	0	0	0	/	1	1
71	0.7109375	M	88	Sigmoide	2	0	0	0	0	0	1

Table 2.1: Table of each case (patient) including metadata (resolution), clinical features (gender, age, tumour location) as well as different sets of output values (stage, grade, vascular/lymphatic permeation, peri-nervous sheathing, budding, micro-satellite instability and KRAS/BRAF mutations)

2.2 Pre-processing

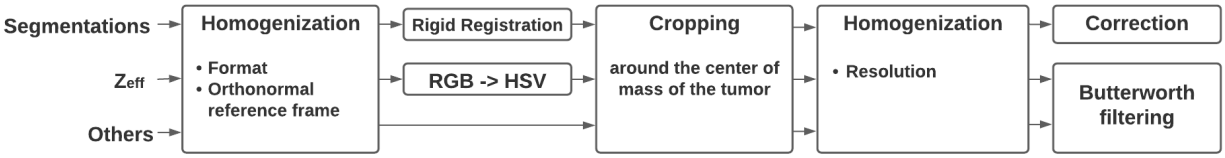


Figure 2.3: Diagram showing the data pre-processing

In this section, attention will be paid to the number of grey levels but also more particularly to the homogenisation of resolution. The images do not have the same **resolution**. This is due to the fact that in diagnostic imaging, the image thickness and pixel size are systematically adapted to each patient in order to optimize the radiation dose and image quality. It will be necessary to homogenize the resolution of the initial dataset [35]. This master thesis will study the precise cases where resolution homogenisation in spite of information loss (with a low-pass filter) can be favourable to better results.

The pre-processing will also ensure compatibility between segmentation and spectral data. The export was not done in one go. Some of files do not have the **same format** (DICOM or NRRD). This can lead to complications. The first segmentations were exported in DICOM format while the last segmentations are in NRRD format. Segmentations in DICOM format have undergone a **shift**. Image processing is necessary to match the rest of the 3D images. Indeed, the principle of radiomics is to highlight the tumour using the segmentation. The shift in the image would not allow this practice.

Once all the images of all the cases seem to be homogeneous, we will apply a **crop** on them in order to have smaller dimensions to work with and to optimise the running time. One step, for example, is to extract features via a deep neural network (more details in the section 2.5). It is therefore preferable to work with a smaller number of images with a smaller size in order to save a significant amount of time.

Finally, pre-processing includes a special treatment of coloured images and segmentation correction. An aim of this thesis will be to show that this segmentation correction can improve the prediction results.

2.2.1 Homogenization format / size / orientation

A homogenization is therefore set such that:

- For simplicity, we will now work in an orthonormal frame $(0, X, Y, Z)$ where the XY plane will correspond to a slice. The dimensions of each greyscale image 3D are $512 \times 512 \times Height$ voxels. With $z = 0$, the slice in the lowest part of the body and $z = height_{max}$, the slice in the highest part of the body.

- All DICOM images are converted to NRRD files. An NRRD file is contained in a single file, unlike DICOM. This makes NRRD a more convenient for sharing. NRRD files also do not store patient information whereas DICOM files do. This conversion is done in the following way: we "stack" the slices of the DICOM file in order (from lowest to highest height) on top of each other to finally form a 3-dimensional *array* in *Python* that can now be more easily converted into NRRD format.

The first segmentations were not provided in the form of 3-dimensional binary-valued arrays. A binarity where a value of "1" indicates a tumour voxel and a value of "0" in the opposite case. Indeed, segmentations provided in DICOM format represents a tomography with a photon energy of 120 keV where only the tumour is coloured (cf figure 2.4).

2.2.2 Rigid registration

Registration is a technique that consists of "image matching" in order to be able to compare or combine their respective information. In radiomics, we have to match the segmentation of the tumour with each type of image because the goal of radiomic is to segment the tumor to analyse it. If the segmentation is shifted in relation to the other images, a good segmentation of the tumor is impossible (see figure 2.4).

We apply rigid registration on the segmentation image. We don't work with the 3D image but just one slice because we suppose that the shift is only experienced on the XY plane and no shift is experienced along the Z axis. To do so, cross-correlation [36] and a 2nd order Taylor expansion [37] are used to measure the offset between the two images .

As we can see on figure 2.4, the segmented tumor is put in color. However, a pixel with a grayscale (black and white) has an identical value for the R,G and B channels. A grayscale pixel therefore fulfills the following condition:

$$R = G = B \tag{2.1}$$

It is therefore easy to extract the tumor only from this kind of image: it is enough to recover the voxels that do not respect the condition of the equation 2.1. In order to recover only a binary mask (element of the array with a value of "1" if it includes the tumour and "0" if it does not), it is sufficient to keep only the pixels respecting the following condition:

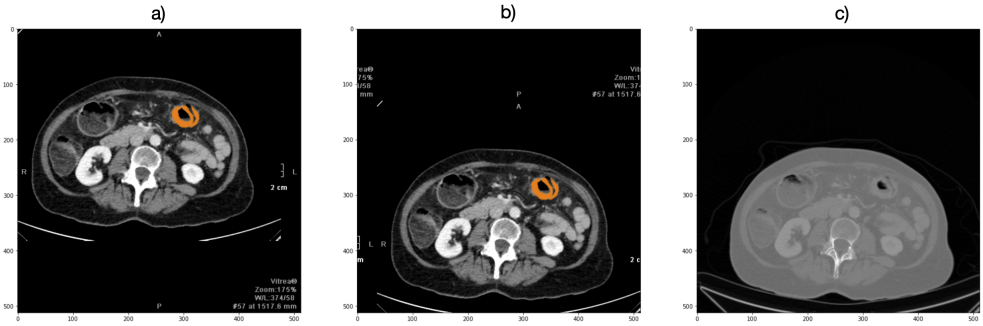


Figure 2.4: a) Slice of the exported DICOM segmentation with a shift that prevents radiomics from working properly; b) Shifted DICOM segmentation slice using rigid registration requiring a reference image to rely on; c) A slice of the 3D image used as a reference for the segmentation so that the segmentation can be aligned with the other image types via rigid registration

$$R \neq G \quad \text{or} \quad R \neq B \quad \text{or} \quad G \neq B \tag{2.2}$$

2.2.3 RGB to HSV

To return to the treatment that we will apply to the coloured images, some types of images also do not seem to have been exploited. Our data are, among other things, composed of effective atomic numbers (Z_{eff}) given as **coloured images**. In the case where the image is colored, a dimension is added for the 3 primary colors (RGB). The next step after the pre-processing is the extraction of features and it can apply the extraction only on three dimensional images (not 4). So the first idea would be to apply the extraction to each color channel **separately** to work with three dimensional images. The problem is that as explained in the document of Sebastien Lefèvre [27], applying a feature extraction on the 3 channels separately suggests that the channels are independent but it is not the case. And applying a filter on the 3 channels separately during the MM feature extraction is not a reasonable solution because it would inevitably lead to the appearance of false colors. Here is a simple example of the phenomenon in the case where we apply an median filter on a colored image (fig. 2.5). The middle picture contains the colour yellow as a false colour and the figure on the right does not contain any.

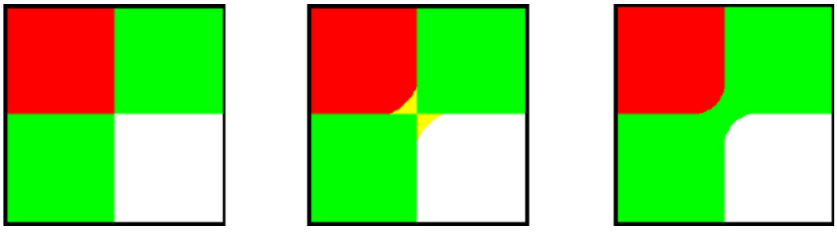


Figure 2.5: Example of false colors. From left to right, the original image, the result of a marginal median filter, and of a vector (lexicographical) median filter.

In fact, the colors corresponds to a certain value. For example the red colors correspond to low value effective atomic numbers and blue colors correspond to high value effective atomic numbers. By applying separately the same filter on the 3 channels, we take the risk to have incoherent features extracted. The solution is to apply a conversion of the image while passing

from a format RGB to **HSV (hue-saturation-value)**. This technique has the particularity to take into account the dependence which could have the channels R,G,B. The information is organized differently. An example directly applied on a colored medical image can be seen in figures 2.6 and 2.7. As we can see on figure 2.6, we first try to apply erosion on every channel (RGB) separately. For information, erosion allows filtering in such a way that an original pixel of an image is replaced by the local minimum of a certain area where the original pixel is in the centre of this area. All this will be developed in the section 2.4 . As said before, we thus exclude the hypothesis that the channels are dependant. The problem is that the color corresponds to a certain value. In this example, blue ($R = 0;\ G = 0;\ B = 255$) corresponds to the highest atomic number equal to 11. Red ($R = 255;\ G = 0;\ B = 0$) corresponds to the lowest atomic number equal to 5. The intermediate values are represented by colors composed of a mix of the 3 primary colors. For example, ($R = 255;\ G = 204;\ B = 153$) corresponds to the salmon red shade. We see that the filtered image of the middle (see fig. 2.6) does not correspond to what we wanted to obtain, i.e.: an erosion on a colored image, in spite of the erosion applied on each channel. Indeed, we can see grey appearing as a false colour because it is not present in the original image. On the filtered image on the right, we applied erosion on **hue**, **saturation** and **value** of the image. We effectively see that the filtered image has a dominant color which is red: the color corresponding to the lowest value. It is the result we expected.

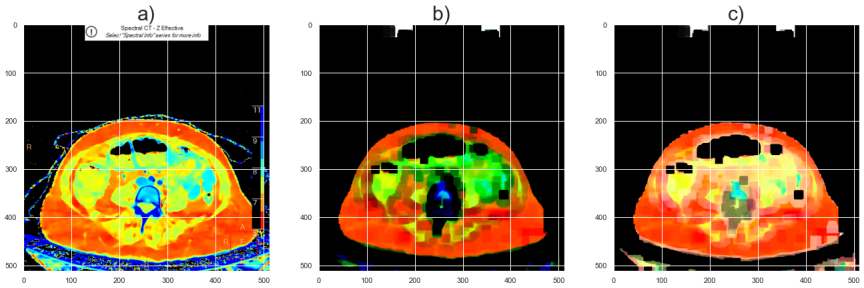


Figure 2.6: Example of false colors. From left to right, the original image with the appearance of false 'colours' such as grey, the result of a marginal erosion filter, and of a vector (lexicographical) erosion filter.

And on figure 2.7, we apply a dilation. Dilation allows the image to be filtered in such a way that an original pixel is replaced by a local maximum. But here again, the concept will be developed in depth in the section 2.4. Also here, the filtered image on the right give us coherent results because of the fact that the dominant color is blue (corresponding to the highest atomic numbers). In the middle image, the false colour appearing is white.

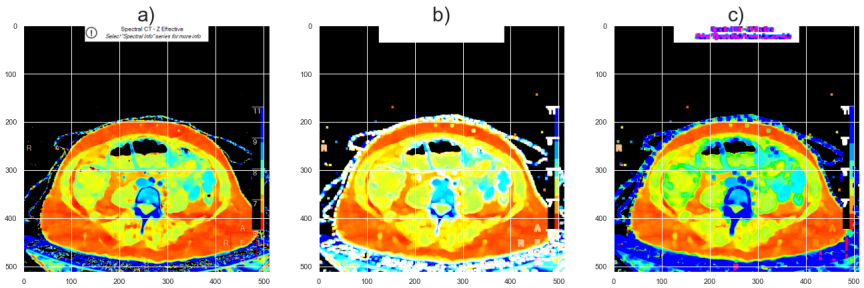


Figure 2.7: Example of false colors. From left to right, the original image with the appearance of false 'colours' such as white, the result of a marginal dilation filter, and of a vector (lexicographical) dilation filter.

2.2.4 Cropping

Then, to optimize running time, we reduce the dimension of the image. It is important to not lose any information. That is why we do not simply downsample the image 3D. The technique consists in making the three-dimensional image contain the tumour in its centre. From then on, the extremities of the 3D image are no longer taken into account, as only the tumour is to be analysed and it is now located in the centre. To do it, a translation is applied on the basis of the segmentation composed only of voxels with a value of "1" for the tumour and "0" otherwise. First of all, we have to find the center of mass of the three-dimensional image (the segmentation) by considering the pixel value as a mass.

Let us denote

- $h : E \to T_h$ where h, the segmentation, is a digital image highlighting the location of the tumor, E is the discrete coordinate grid (the set is $\mathbb{N}^3$ because it is a 3D image) and T_h is the set of possible image values (in our case, we work with a binary image values. So $T_h = \{0, 1\}$).
- $f : E \to T$ where f is a digital image, E is the discrete coordinate grid (the set is $\mathbb{N}^3$) and T is the set of possible image values (T is defined on $\mathbb{R}$).

We thus obtain the coordinates of the center of mass below:

$$X_{CM} = \frac{\sum_{p \in E} x_p \cdot h(p)}{\sum_{p \in E} h(p)} \tag{2.3}$$

$$Y_{CM} = \frac{\sum_{p \in E} y_p \cdot h(p)}{\sum_{p \in E} h(p)} \tag{2.4}$$

$$Z_{CM} = \frac{\sum_{p \in E} z_p \cdot h(p)}{\sum_{p \in E} h(p)} \tag{2.5}$$

Where $(X_{CM}; Y_{CM}; Z_{CM})$ are the coordinates of the center of mass, p is the coordinates of a pixel composing the tumour and the mass $h(p)$ of the pixel p.

When it is done we translate the tumor to the center of the 3D image of size $l_x \times l_y \times l_z$:

$$f_{translated}(p) = f(l_x/2 - X_{CM}; l_y/2 - Y_{CM}; l_z/2 - Z_{CM}) \tag{2.6}$$

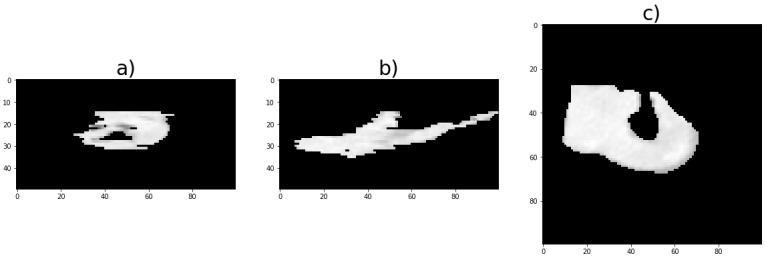


Figure 2.8: Tumor of sample n°1 after cropping from the YZ, XZ, XY plane viewpoint.

Due to the fact that the tumor is centered, we can crop the 3D image. Thanks to the display of a 3D image (figure 2.8), we can observe that the image is overwritten on axis Z (because the width of the slices is fixed at 3 mm).

That means that a voxel has not a 'volume' proportional on all axis. These information is taken into account, we decide to crop the 3D images of size $512 \times 512 \times l_z$ and keep $150 \times 150 \times 50$ (figure 2.11). There are, however, some exceptions in the size of the cropping for larger tumours.

2.2.5 Homogenise the resolution among the different cases

Having consistent pixel sizes is important for the evaluation of textural features that relate intensity and spatial information in radiomic studies. For example, a difference in resolution between two cases will not give relevant information on tumour volume or even texture as the degree of image detail is not the same. To correct for the effects of variable pixel size, one possibility is to combine resampling of the image with Butterworth filtering in the frequency domain. In other words, we impose a resolution on all cases and then apply a low-pass filter to smooth them. Researchers [35] have tested this correction on CT scans of lung cancer patients reconstructed with pixel sizes ranging from 0.59 to 0.98 mm. After pixel size matching, it was shown that with the filtering correction, 8 out of 8 patients were correctly grouped, compared to only 2 out of 8 without the correction.

Applying a correction based on resampling and Butterworth low-pass filtering in the frequency domain effectively reduced variability in CT radiomics features caused by variations in pixel size.

For measuring agreement when the variable of interest is continuous (e.g., size of the tumour), we use the overall concordance correlation coefficient (OCCC) which is more appropriate than other indices. The OCCC assesses the agreement of a single measured value (in this case, radiomics features) with multiple subjects (patients/examples) by multiple observers (Fields of view reconstruction). The OCCC ξ is given by [35]:

$$\xi = \frac{2\sum_{j=1}^{J-1}\sum_{k=j+1}^{J} S_{jk}}{(J-1)\sum_{j=1}^{J} S_j^2 + J\sum_{j=1}^{J}(\bar{M}_j - \bar{M})^2} \tag{2.7}$$

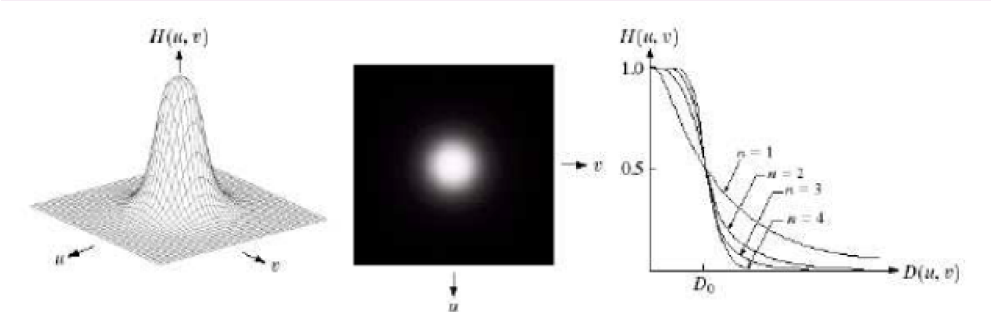


Figure 2.9: Pictures from left to right: Low-pass Butterworth filter gains $H(u, v)$ as a function of frequency (u, v)

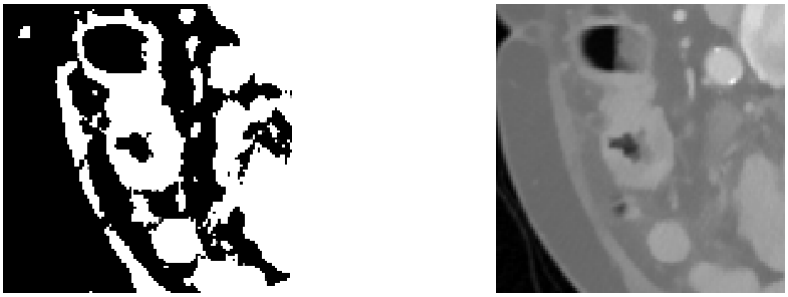


Figure 2.10: Pictures from left to right: Image of the region of interest after pixel spacing homogenisation; Image of the region of interest after homogenising the pixel spacing and using the Low-pass Butterworth filter

Where J is the number of observers (= Fields of view), S_{jk} is the covariance of the features of observers j and k. S_j is the sample standard deviation for FOV j, $\bar{M}_j$ is the mean value of a certain feature of all the patients/examples with a FOV j, and $\bar{M}$ is the mean value of the means for each FOV.

To reduce the information discrepancy, we filtered each slice of the ROI in frequency space using 2D, second-order Butterworth low-pass filters.

The Butterworth low-pass filter of order n is defined by [38]:

$$H(u, v) = \frac{1}{1 + (\frac{\sqrt{u^2+v^2}}{D_0})^{2n}} \tag{2.8}$$

where , D_0 is the critical frequency.

As can be seen in Figure 2.9, the frequency components are more attenuated the further the pair (u, v) is from the origin. We also see that the larger n is, the greater the attenuation of high frequencies. The filter is known to cause less blurring (less smooth contours) than with an ideal low-pass filter [39, 40, 41].

On figure 2.11, we can finally see an example of what we obtain after a resampling to fix the resolution of all cases at $1mm/pixel$ and a cropping.

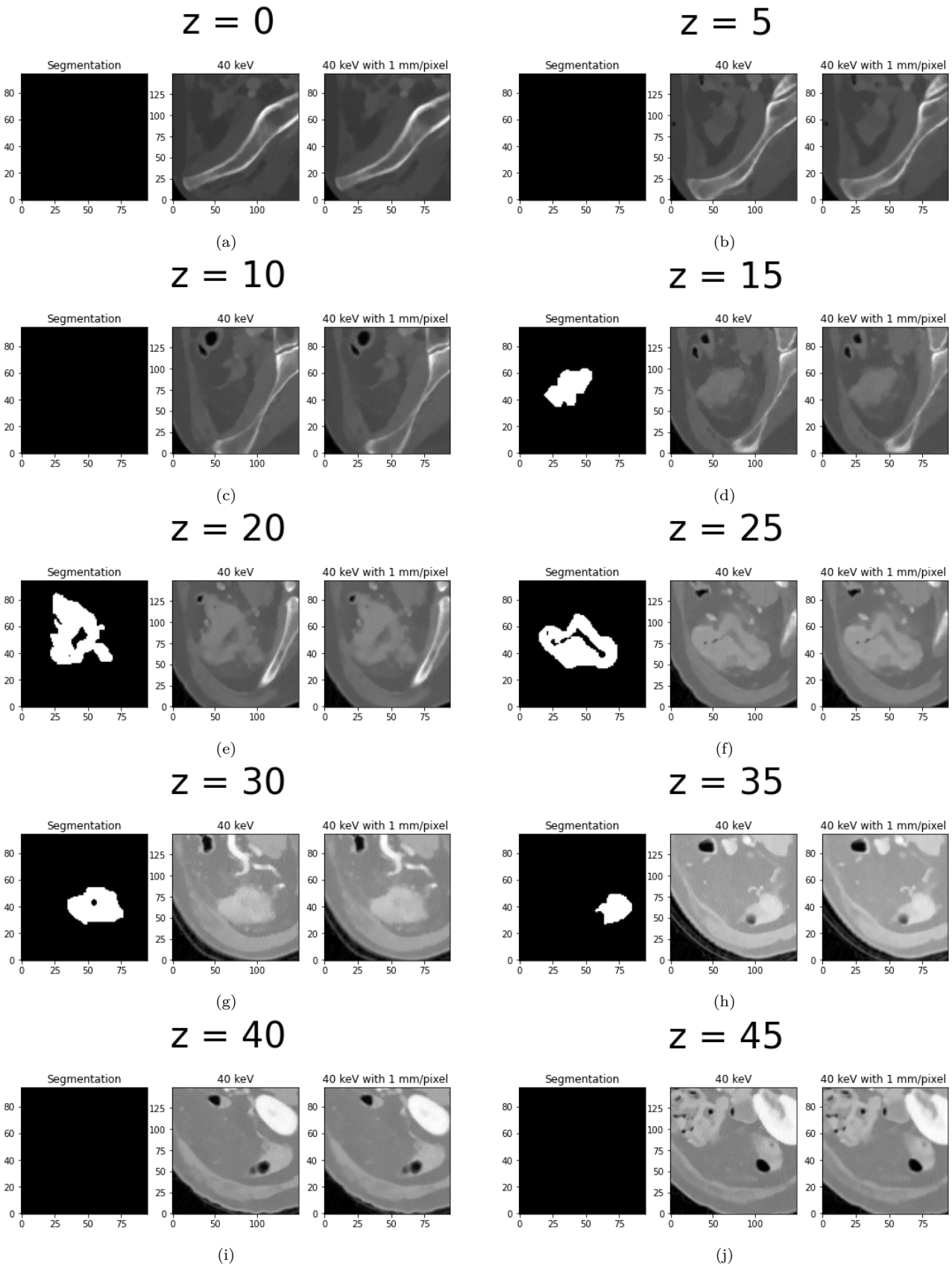


Figure 2.11: Succession of 3 slices of the same body level ($z = 0$ for lower body and $z = z_{max}$ upper body): segmentation after resolution homogenization; monochromatic low keV image, monochromatic low keV image with resolution homogenization and Butterworth low pass filtering.

2.2.6 Segmentation correction

This step allows to correct the possible errors made during the manual segmentations. **A voxel with negative HU values indicates the presence of fatty tissue**. Classically, a malignant tumour of the colon or rectum is dominated by a tissue component with a positive density, higher than the density measured in a healthy colonic wall. That there is an occasional very small fat component is conceivable on a microscopic level, but in clinical practice it is not reported on CT.

To summarise this information: a tumour is not supposed to be composed of fat that is visible on a CT scan. Therefore, **the voxels composing the tumour and having the HU as the unit of measurement, cannot logically have a negative value**.

Once this information is taken into account, any voxel with a potential negative value can be reduced to 0 at the segmentation level so as not to be taken into account when analyzing the tumor. A tumour is not supposed to be composed of negative voxels with HU as the unit of measurement, but an imperfect segmentation could inadvertently include fat.

2.3 Features extraction (Part 1: Traditional features)

***In the framework of this master thesis**, The extraction of traditional features will be done using the Pyradiomics package provided by Python. The feature extraction on all exported spectral imagery (including of course the effective atomic numbers represented as HSV).*

The first type of features are predefined or created by hand [10]. They are also called *traditional features*. Quantitative features are generally classified into the following subgroups:

2.3.1 Shape features

Shape features describe the shape of the plotted region of interest (ROI) and its geometric properties such as volume, maximum diameter in different orthogonal directions, maximum surface area, tumour compactness and sphericity. For example, the surface-to-volume ratio of a spiculated[1] tumour will have higher values than that of a round tumour of similar volume. For full details of the extracted features, see table D.1.

[1]Spiculated: any structure in the shape of a spike or needle point.

2.3.2 First-Order statistical features

First-order statistical features describe the distribution of values of individual voxels without regard to spatial relationships. These are histogram-based properties that account for the mean, median, maximum and minimum values of voxel intensities in the image, as well as their skewness, kurtosis, uniformity and randomness (entropy). For full details of the extracted features, see table D.2.

2.3.3 Higher-Order statistical features

Higher-order statistical features include the so-called textural features, which are obtained by calculating the statistical interrelationships between neighbouring voxels. They provide a measure of the spatial arrangement of voxel intensities, and thus of the heterogeneity within the tumour itself. These features can be derived from the grey level co-occurrence matrix (GLCM) at table 2.2, which quantifies the incidence of voxels with the same intensity at a predetermined distance along a fixed direction. For full details of the extracted features, see table D.4.

1	2	3	4
1	3	4	4
3	2	2	2
2	1	4	1

Gray Level (i)	Cooccurrences (j)			
	1	2	3	4
1	0	1	1	3
2	1	4	2	0
3	1	2	0	2
4	3	0	2	2

Table 2.2: (a) Table representing a 4x4 image composed of pixels with 4 gray levels, (b) Table CM listing the pairs of similar pixels of the image **(a)**. These pairs of pixels are defined by the orientation that the pixels form (θ=0°here), the distance between them (δ=1 here), the value of the two pixels [42].

The grey level run length matrix (GLRLM) at table 2.3, which quantifies consecutive voxels with the same intensity along fixed directions. For full details of the extracted features, see table D.6.

1	2	3	4
1	3	4	4
3	2	2	2
2	1	4	1

Gray Level (i)	Run Length (j)			
	1	2	3	4
1	4	0	0	0
2	1	0	1	0
3	3	0	0	0
4	3	1	0	0

Table 2.3: (a) Table representing a 4x4 image composed of pixels with 4 gray levels, (b) Table RLM listing the groups of consecutive similar pixels in the image (a). These groups of pixels are defined by the orientation that the pixels form (θ=0°here), the value and the number of these consecutive pixels [42].

The Gray Level Size Zone (GLSZM) at table 2.4 quantifies gray level zones in an image. A

gray level zone is defined as a the number of connected voxels that share the same gray level intensity. For full details of the extracted features, see table D.8.

1	2	3	4
1	3	4	4
3	2	2	2
4	1	4	1

Gray Level (i)	Size Zone (j)			
	1	2	3	4
1	2	1	0	0
2	1	0	1	0
3	0	0	1	0
4	2	0	1	0

Table 2.4: (a) Table representing a 4x4 image composed of pixels with 4 gray levels, (b) Table *LSZM* listing the areas of similar pixels in the image (a). These pixel groups are defined by the number and common value of these pixels [42].

Finally, the grey level dependency matrix (GLDM) at table 2.5 quantifies the grey level dependencies in an image. GLDM actually studies the dependence of neighbouring voxels on a central voxel. A neighbouring voxel of grey level j is considered dependent on the central voxel of grey level i if $|i - j| \leq \alpha$. Here, α is set to 0. A voxel is therefore dependent on the central voxel if it has the same grey level. For full details of the extracted features, see table D.10.

5	2	5	4	4
3	3	3	1	3
2	1	1	1	3
4	2	2	2	3
3	5	3	3	2

Gray Level (i)	gray lvl dpcy (j)			
	0	1	2	3
1	0	1	2	1
2	1	2	3	0
3	1	4	4	0
4	1	2	0	0
5	3	0	0	0

Table 2.5: (a) Table representing a 5x5 image composed of pixels with 5 gray levels, (b) Table *LDM* listing the dependency of pixels similar to a central pixel of the image (a). These pixel groups are defined by the number of pixels around the central one and the common value of these pixels [42].

2.4 Features extraction (Part 2: MM features)

***In the framework of this master thesis**, We will use MM features. As mentioned earlier, the corrected segmentation will not take into account the fat surrounding the tumour. But of course, we do not exclude the possibility that the fat surrounding the tumour could give information about the type of anomalies, the grade, the stage of the tumour. This is why the use of mathematical morphology is present in feature extraction. Erosion, for example, captures the local minimum over a predefined area. This type of filter allows in a way to include the surrounding fat and to estimate its importance in the identification of the tumour. Morphological covariance compares the similarity of 2 points at a certain predefined distance. This tool could be used to study the similarity of cancerous and non-cancerous tissue. Along with color and shape, texture constitutes one of the three fundamental properties of objects in our threedimensional*

(3D) world. In this section the main goal will expose the basis of the Mathematical Morphology [27]. This technique is a nonlinear analysis framework based on complete lattice theory[1].

2.4.1 Theoretical Foundations

Before the presentation of every morphological operator (**Erosion**, **Dilation**, **Opening** and **Closing**) , let's clarify some notations that will be used in the rest of this section and the specificity's of the complete lattice. As a reminder, complete lattice is a partially ordered set in which all subsets have both a supremum and an infimum.

From the lattice theory viewpoint, let us denote the operator[2] $f : E \to T$ where f is a digital image, E is the discrete coordinate grid (the set is $\mathbb{N}^3$ in our case) and T is the set of values that the pixels in this grid can take (in our case, T is defined on $\mathbb{R}$). It should be noted that the images we work on are not limited to the tumour itself and are therefore composed of positive and negative voxel values. In other words, the images to be filtered have not yet undergone the "masking" of a segmentation.

A complete lattice is defined from three elements:

- A **partially ordered set** $(T, \geq)$. It means that every pair of scalars are related via $\geq$, so $\geq$ is a total order and $(T, \geq)$ is a chain. A natural order of scalars is formed for the grey-scale images.
- an **infimum** or greatest lower bound $\wedge$, which is most often computed as the *minimum* operator (this choice will also be made here for the sake of simplicity). For example, $\wedge f(p)$ will give us the greatest lower bound of the digital image f.
- a **supremum** or least upper bound $\vee$, which is similarly most often computed as the *maximum* operator.

Basically, MM relies on the spatial analysis of images through a pattern called *structuring element* (SE) and consists of a set of nonlinear operators that are applied on the images considering this SE. See on figure 2.12 for some basic SEs. A structuring element SE could for example be a square of size 5 ($\lambda = 5$) and the operator could be an erosion ϵ applied on an image f.

The SE noted b when defined as a set on E.

[1] Complete lattice is a partially ordered set in which all subsets have both a supremum and an infimum.

[2] An operator is an application between two topological vector spaces. A topological vector space is a space with a topological structure associated with a vector space structure. For example, the image f is an operator where the first topological vector space $\boldsymbol{E}$ is a grid of size $m \times n$ composed of pixels. And the second topological vector space $\boldsymbol{T}$ is the set of values that the pixels in this grid can take.

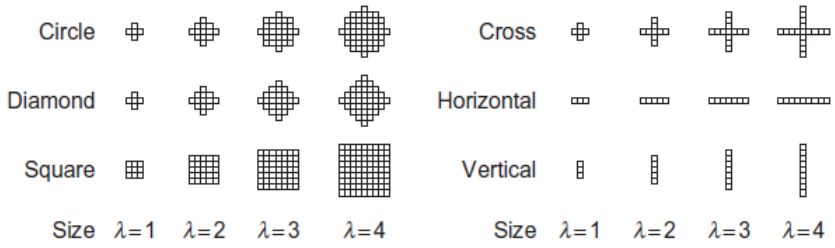


Figure 2.12: Illustrative examples of basic SEs with increasing size λ

2.4.2 Erosion and Dilation

From these theoretical requirements, one can define the two basic morphological operators.

We first have the *erosion*:

$$\epsilon_b(f)(p) = \wedge_{q \in b} f(p+q),\ p \in E \tag{2.9}$$

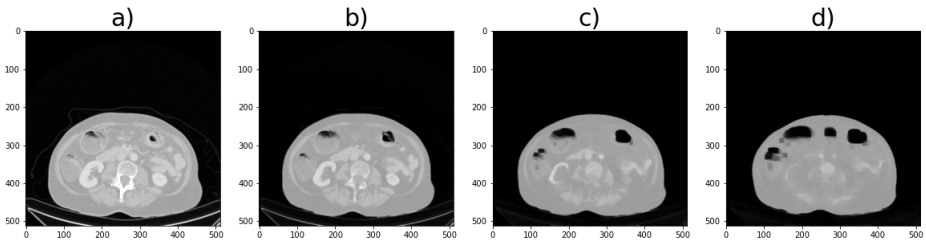


Figure 2.13: Grey-scale erosion with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$

Where p is the pixel coordinates. The coordinates within the SE b are denoted by q (defined in the same space as p). For every pixel $f(p)$ with coordinates p of the image f, we take a look around the pixel (most precisely the pixels $f(p+q)$) and replace the pixel value $f(p)$ by the infinimum which can also be defined as the **local minimum**.

The other main morphological operator is called *dilation*:

$$\delta_b(f)(p) = \vee_{q \in b} f(p-q),\ p \in E \tag{2.10}$$

Here the result is an image where each pixel is associated with the **local maximum** in the neighborhood of the pixel $f(p)$ defined by the SE b.

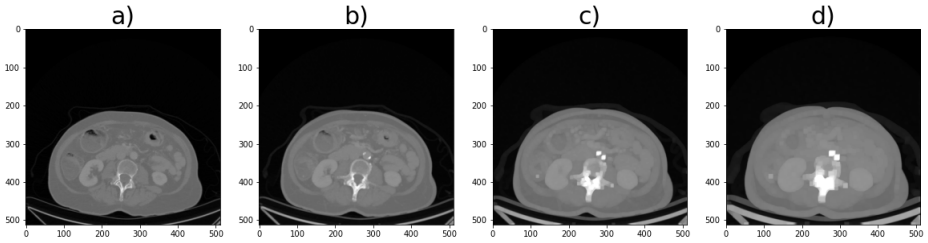


Figure 2.14: Grey-scale dilation with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$

2.4.3 Opening and Closing

Thanks to erosion and dilaton, we can build other morphological operators. *Opening* is defined by:

$$\gamma_b(f)(p) = \delta_{\breve{b}}(\epsilon_b(f)) \tag{2.11}$$

Where $\breve{b}$ is the reflected SE such that $\breve{b} = \{-q | q \in b\}$.

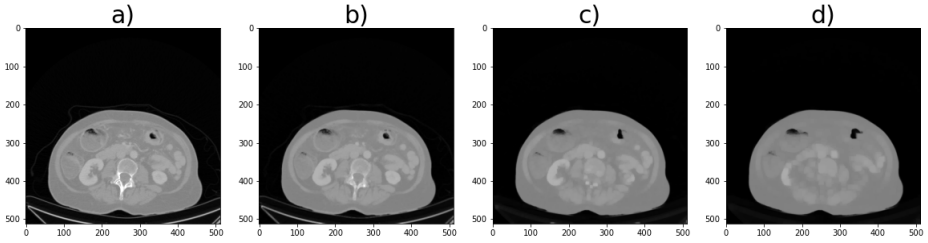


Figure 2.15: Grey-scale opening with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$

This operator is used to remove local maxima and return filtered images that are lower than the input image

Closing is defined by:

$$\varphi_b(f)(p) = \epsilon_{\breve{b}}(\delta_b(f)) \tag{2.12}$$

This operator is used to remove local minima and returns filtered images that are higher than the input image.

The main concern with these two morphological filters is their very strong sensitivity to the SE shape, which will have a straight influence on the shapes visible in the filtered image.

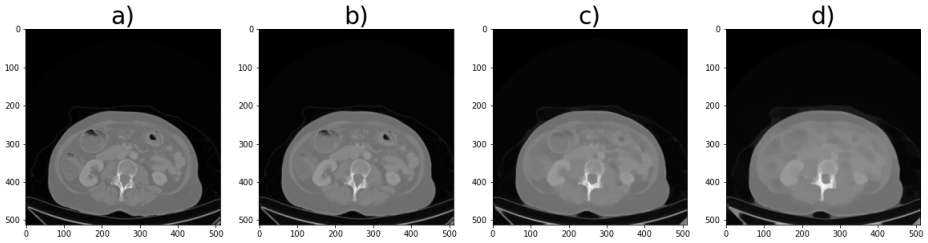


Figure 2.16: Grey-scale closing with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$

2.4.4 MM features

Morphological analysis excels at the exploitation of spatial relationships among pixels, and possesses a large number of tools capable of extracting size and shape information.

The two main morphological tools used for texture analysis are *granulometry* and *morphological covariance*, and both are based on the common principle of *morphological series*. These are successive series of filtered images with less and less detail. The first one studies the amount of detail removed by applying successively basic morphological operators along with SEs of various sizes, whereas the latter is the morphological equivalent of the standard covariance operator. As explained above, the extraction of MM-related features is done after applying a morphological filtering on the image (three-dimensional in this master thesis) with SEs of variable size.

Let us denote by b_λ the SE b of size λ and write γ_λ as a shortcut for γ_{b_λ}. Here, γ is defined as an opening but nothing prevents us from working with a closure, an erosion or a dilation. We thus have the series $\Pi^\gamma(f)$ of successive openings γ on the input image f as:

$$\Pi^\gamma(f) = \left\{\Pi^\gamma_\lambda(f) | \Pi^\gamma_\lambda(f) = \gamma_\lambda(f)\right\}_{0 \leq \lambda \leq n'} \tag{2.13}$$

As we can see on figure 2.17, that the higher the lambda, the less detailed the image

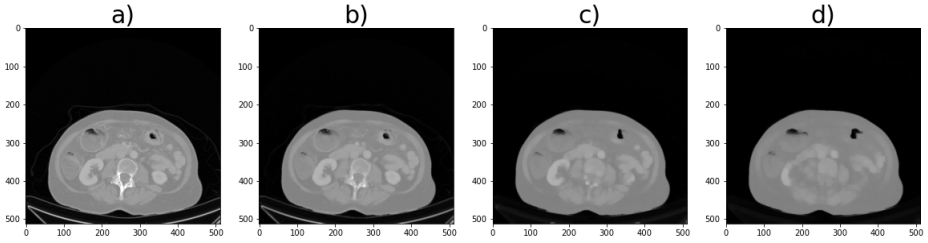


Figure 2.17: Grey-scale opening with square-shaped SE $\blacksquare_\lambda$ of increasing size λ as applied on one image of our dataset. With $\lambda = \{0, 1, 3, 5\}$ for a), b), c), d)

Instead of focusing on filtered images, one can also emphasize the details removed after each opening, thus building a differential series which we will denote as Δ_γ. For every element of the differential serie, we substract the filtered image $\gamma_\lambda(f)$ by $\gamma_{\lambda-1}(f)$:

$$\Delta^\gamma(f) = \left\{\Delta^\gamma_\lambda(f) | \Pi^\gamma_{\lambda-1}(f) - \Pi^\gamma_\lambda(f)\right\}_{0 \leq \delta \leq n'} \tag{2.14}$$

We can see the results on figure 2.18

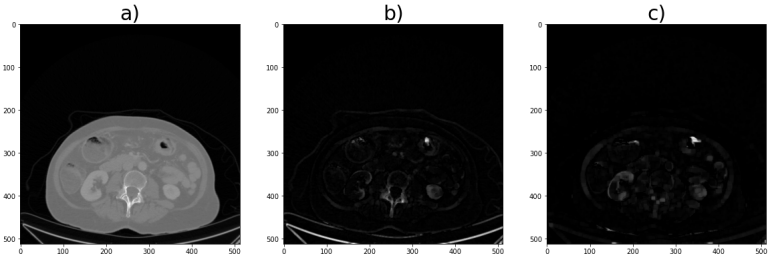


Figure 2.18: **a)** 120 keV image; **b)** $\gamma_{\lambda=1}(f) - \gamma_{\lambda=3}(f)$; **c)** $\gamma_{\lambda=3}(f) - \gamma_{\lambda=5}(f)$

From these two series Π and Δ, it is possible to compute morphological texture features related to the distribution of primitive sizes within a texture image.

The most basic of these features is *granulometry*, which is built by first gathering the values of the series Π^γ and then summing the pixels of the images after filtering:

$$\Omega^\gamma(f) = \left\{ \Omega^\gamma_\lambda(f) | \Omega^\gamma_\lambda(f) = \sum_{p \in E} \Pi^\gamma_\lambda(f)(p) \right\}_{0 \leq \lambda \leq n} \tag{2.15}$$

Of course, one is by no means limited to using only the image volume, as higher-order statistical moments can be computed to form the final feature vector, some of the usual of which include the mean, variance, skewness, and kurtosis [25, 26, 27]. These features are referred to as *granulometric moments*. In our case, we will go even further by applying on the filtered image an extraction of features of order higher than 1 (See section: *Features extraction (Part 1: Traditional features)*).

Let us now introduce the morphological covariance.

$$K^{\overrightarrow{v}}(f) = \left\{ K^{\overrightarrow{v}}_\lambda(f) | K^{\overrightarrow{v}}_\lambda(f) = \sum_{p \in E} \Pi^\epsilon_{\lambda, \overrightarrow{v}}(f)(p) \right\}_{0 \leq \lambda \leq n} \tag{2.16}$$

In summary, this part consists of taking a voxel of an image f and comparing at the same distance from this voxel several other voxels with each other. Here we will only work with the voxels of the tumour. p is the voxel and E is the set of voxels that will be processed (those of the tumour). The correlation function is given by [43]:

$$\epsilon_{\lambda, \overrightarrow{v}} = f(p - \lambda \overrightarrow{v}) \cdot f(p + \lambda \overrightarrow{v}) \tag{2.17}$$

To illustrate the usefulness of this tool, we take for example an image composed of repetitions as shown in figure 2.19, we have a figure with a non-zero spatial frequency. The recurrence in

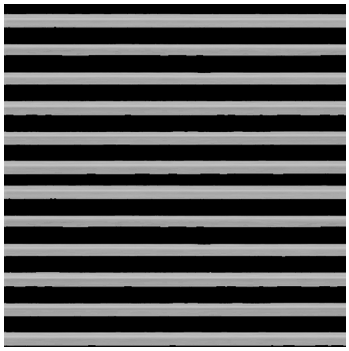


Figure 2.19: Picture from a texture dataset representing corrugated steel. Dimensions: 512 × 512.

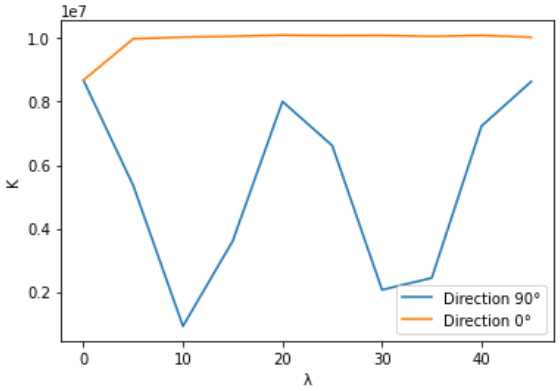


Figure 2.20: The resulting non normalized morphological covariance $K^{\overrightarrow{v(f)}}$ plots for 2 directions where f is the picture representing the corrugated steel.

the image is captured when the comparison of points is done in a certain direction. Here, the direction must be 90° if we want to perceive the spatial frequency.

A basic morphological covariance therefore compares 2 voxels. This pair forms a certain direction. In the previous example the morphological covariance requires that the pixel pair is oriented in a certain way to perceive the spatial frequency. In this master thesis we assume that there is no difference in the results of one orientation of the pairs compared to another. What we are really interested in is to study the relationship of voxels at a certain distance from each other, not the direction of the pair they form. We therefore decided to work with a morphological covariance grouping several orientations rather than analysing them separately. To do this, we sum the morphological covariances of the different orientations to give importance to the distance between the two analyzed voxels and not not to the orientation they form in pairs.

$$K(f) = \sum_{\theta \in \{0°, 45°, 90°, 135°\}} K^{\overrightarrow{v}}(f) \tag{2.18}$$

where θ is the orientation of the vector $\overrightarrow{v}$.

Finally, a standardisation is made by dividing the K obtained by the volume of the tumour. Thus, the high K values are only related to the high morphological covariance within the tumour and its surroundings. Tumour size is no longer a factor in a high K value.

2.5 Features extraction (Part 3: Deep features)

In the framework of this master thesis, *models have been learned to extract deep features. The idea of this Master Thesis is to use an unusual dataset to train our models. Here,*

we seek to prove that a model trained on a dataset composed exclusively of textures (such as wood, stone, metal, concrete,..) can yield other types of deep features that can make a difference. Note that each of these models will be different in the optimizers, augmentations and datasets used.

An innovative way to use DECT data for deep feature extraction is to combine different types of scans (e.g. iodine map, 40 keV, 120 keV) at the locations normally reserved for RGB channels in a pre-trained model. In research done on the Prediction of Malignant Nodules [31] for example, there is a use of the VGG16 pre-trained model in deep feature extraction. But since set of data is only composed of one type of greyscale images, the authors choose to add a dimension by duplicating his image three times. He then has an input for each of the RGB colour channels.

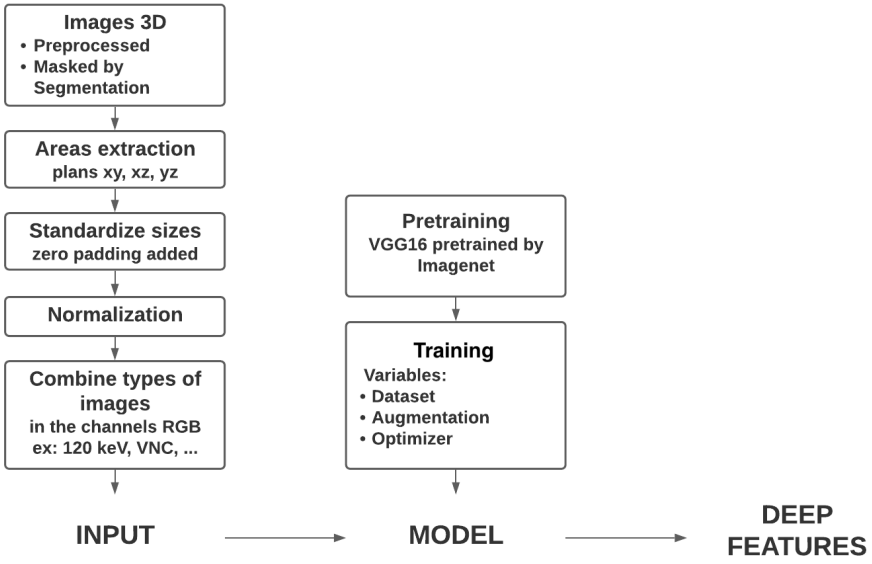


Figure 2.21: Diagram showing the Deep features extraction process

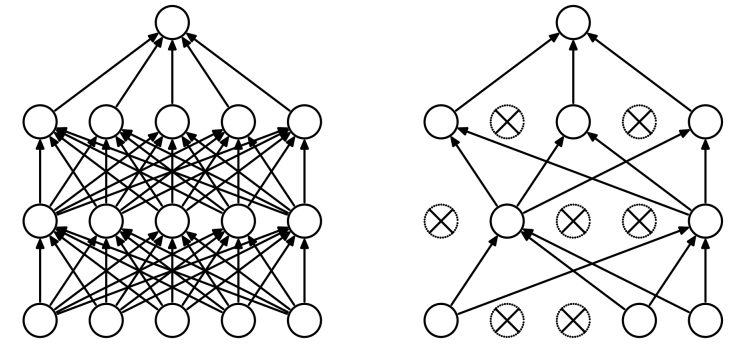
In this section, the objective is to explain in a little more detail all the steps leading to the extraction of our deep features.

We will first explain in more detail what overfitting is and what methods have been used to avoid it. We will develop the pre-processing, transfer learning and augmentation methods and finally we will look at the different types of optimizers used to train our models.

2.5.1 Overfitting

Overfitting occurs when a model tries to model the training data too well. Overfitting occurs when a model learns so much detail and noise from the training data that it negatively impacts the performance of the model on the new data tested with the validation set and then the test set. There are some techniques to avoid this [44].

Dropout is a regularisation technique that cancels the activation values of randomly selected neurons during training. This constraint forces the network to learn more robust features rather than relying on the predictive ability of a small subset of neurons in the network.



(a) Standard Neural Net (b) After applying dropout.

Figure 2.22: a) A standard neural network with two hidden layers; b) Example of a network after inactivation of some randomly selected neurons produced by applying dropout to the network on the left. [45]

Transfer Learning works by training a network on a big dataset such as ImageNet [1] and then using those weights as the initial weights in a new classification task. Typically, just the weights in convolutional layers are copied, rather than the entire network including fully-connected layers. This is very effective since many image datasets share low-level spatial characteristics that are better learned with big data. Understanding the relationship between transferred data domains is an ongoing research task. If, for example, we are trying to train a model that can distinguish between pictures of tractors and motorbikes, we can base our training on a model that distinguishes between cars and bicycles, so as not to "start from scratch".

Pretraining is conceptually very similar to transfer learning. In Pretraining, the network architecture is defined and then trained on a big dataset such as ImageNet. This differs from Transfer Learning because in Transfer Learning, the network architecture such as VGG-16 or ResNet must be transferred as well as the weights. Pretraining enables the initialization of weights using big datasets, while still enabling flexibility in network architecture design.

Data Augmentation, Unlike the techniques mentioned above, data augmentation prevents overfitting by addressing the real problem: the too-small training dataset. It assumes that more information can be extracted from the original data set through augmentations. These augmentations artificially inflate the size of the training dataset, either by data warping or by oversampling. Augmentations by data deformation transform existing images in such a way as to preserve their label. An example of this is an augmentation that allows a model to recognise a cat even after a slight deformation. Other techniques include geometric and colour augmentation, random deletion[2] in natural language processing, and adversarial training[3].

[1]ImageNet is a database of images organised in a hierarchy currently composed of names only, in which each node of the hierarchy is represented by hundreds and thousands of images. Currently, ImageNet has an average of over five hundred images per node.

[2]Random deletion: Augmentation technique in natural language processing consisting in deleting words randomly

[3]Adversarial training: technique that attempts to fool models by supplying deceptive input.

Augmentations by oversampling create synthetic instances and add them to the training set. This includes image blending [44].

2.5.2 Pre-training

The deep features extraction approach we applied uses the intermediate activations of the VGG16 convolutional neural network from the second last fully-connected layer. The VGG16 network is already pre-trained thanks to the ImageNet database. Studies [46] have repeatedly proved that pre-trained models can accelerate the training convergence speed.

VGG (Visual Geometry Group) neural networks are known for their ability to localise and classify track [47]. VGG16 has 13 convolutionnal layers, 5 maxpool layers, and 3 fully connected layers. Initially, the model has 1000 outputs, one output per image category in the ImagNet database. Input to the model are color images with a with a default size of $224 \times 224 \times 3$. The image is passed through a stack of convolutionnal layers, i.e. the convolution layers grouped between each max-pool layer. The image is padded in order to maintain a spatial resolution between every convolutionnal layers of a same block. At the end of each block, the image is max-pooled.

At each convolution layer, several filters are used (see the number of filters per convolution layer and precisions in Appendix E).

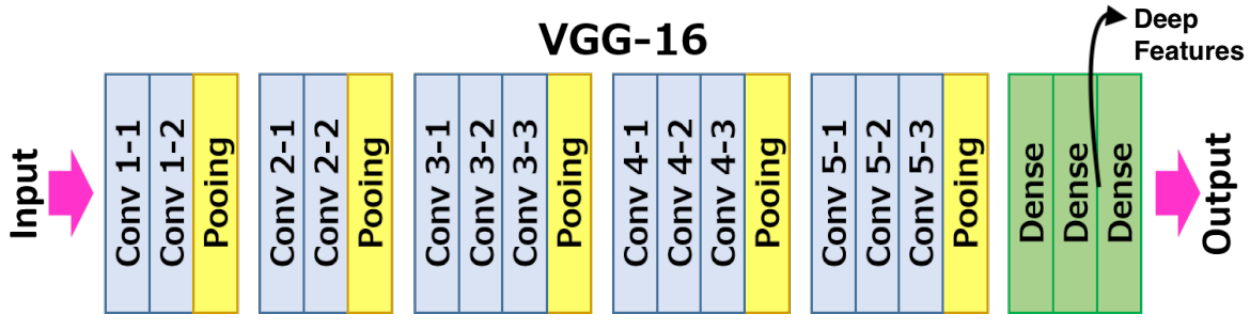


Figure 2.23: model vgg16

Previous searchers had already employed deep feature extraction and it turns out that tissue analysis based on deep learning predicts the outcome of colorectal cancer [29].

2.5.3 Training

The trainings will be applied on the last 3 layers and be based on 2 datasets which are quite different in their composition. The first dataset in question is composed of 8 classes with some examples visible on figure 2.24. This dataset is composed of images specific to human tissue from research on the national lung disease screening trial [48].

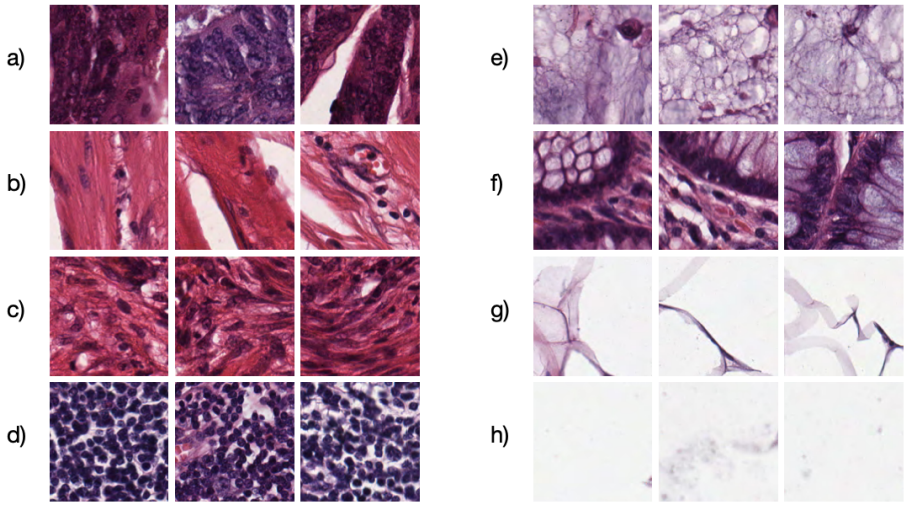


Figure 2.24: Pictures from Dataset 1 related to classes such as a) tumor; b) stroma; c) complex; d) lympho; e) debris; f) mucosa; g) adipose; h) empty

The second dataset is composed exclusively of textures as we can see on the figure 2.25. This type of abstract image dataset could also be important because what we are primarily asking our model to do is to extract the deep features of our biomedical images related to the tumour texture.

We observe first of all, by looking at tables 2.6, that dataset 1 has much more homogeneous and high numbers of images in each of its classes. The classification will therefore be much easier and more precise.

A second observation we can make from the tables specifying the number of images per class, is that the second dataset is much less supplied and that there is a very good chance that training models from the second dataset will give less good results due to the lack of data. In addition, the data set has unbalanced classes. The solution here is to estimate class weights for unbalanced datasets. This will cause the model to "pay more attention" to examples of an under-represented class. To do this, we will draw on the work of King, Gary, and Langche Zeng [49, 50]:

$$\omega_c = \frac{1}{size_c} * \frac{\sum_{i=0}^{C-1} size_i}{C} \tag{2.19}$$

Where ω_c is the weights assigned to the class c, C is the number of classes, $size_c$ is the number of elements in the class c. First, we generate, for every class c, a weight ω_c. The lower the number of elements in the class, the higher the weight assigned to the class.

In spite of all these observations, the second dataset was nevertheless chosen for the varied number of classes and thus the diverse textures it offers.

The different strategies will therefore consist of proving/verifying that one or more additional

	Training	Validation	Test
tumor	437	63	125
stroma	437	63	125
complex	437	63	125
lympho	437	63	125
debris	437	63	125
mucosa	437	63	125
adipose	437	63	125
empty	437	63	125

	Training	Validation	Test
brick	31	5	9
fabric	27	4	8
fence	3	1	2
floor	6	1	2
ground	31	5	9
metal	16	3	5
misc	6	1	2
roof	16	3	5
siding	18	3	6
skin	3	1	2
stone	24	4	8
test	1	1	1
wall	21	3	6
wood	58	9	17

Table 2.6: Dataset 1 and Dataset 2

trainings to the pre-training can really have an impact on the quality of the decisions taken. In order to push our thinking even further, we will also check whether combined training (i.e. a model trained by both dataset 1 and dataset 2) can be useful.

2.5.4 Augmentations

Many application areas do not have access to big data, such as medical image analysis. One solution to the problem of limited data is data augmentation. Data augmentation encompasses a series of techniques that improve the size and also the quality of training datasets (if done intelligently) so that better deep learning models can be built. [44]

The safety of a data augmentation method refers to the likelihood that it will preserve the label after transformation. For example, a photo of a dog with the label "dog" should not be recognised as a cat and therefore have the label "cat" after augmentation. Rotations and flips are generally safe for ImageNet challenges such as cat and dog, but not for number recognition tasks such as 6 and 9. A non-label-preserving transformation could potentially enhance the model's ability to produce a response indicating that it is not confident in its prediction. It is important to consider the "safety" of an augmentation. This is somewhat domain-dependent, which presents a challenge for the development of generalisable augmentation policies.

On reflection, augmentation can only be deepened on a limited number of features. This is because a tumour is analysed according to its shape and texture. Using an augmentation that varies its size, shape and texture could distort the accuracy of the classifications. The augmentation used with the second dataset, which is more focused on textures, could have included cropping, but we preferred to limit ourselves to rotation (0-90°) and inversion (for the sake of simplicity). We will look at different ways of dealing with the rotation-related increase.

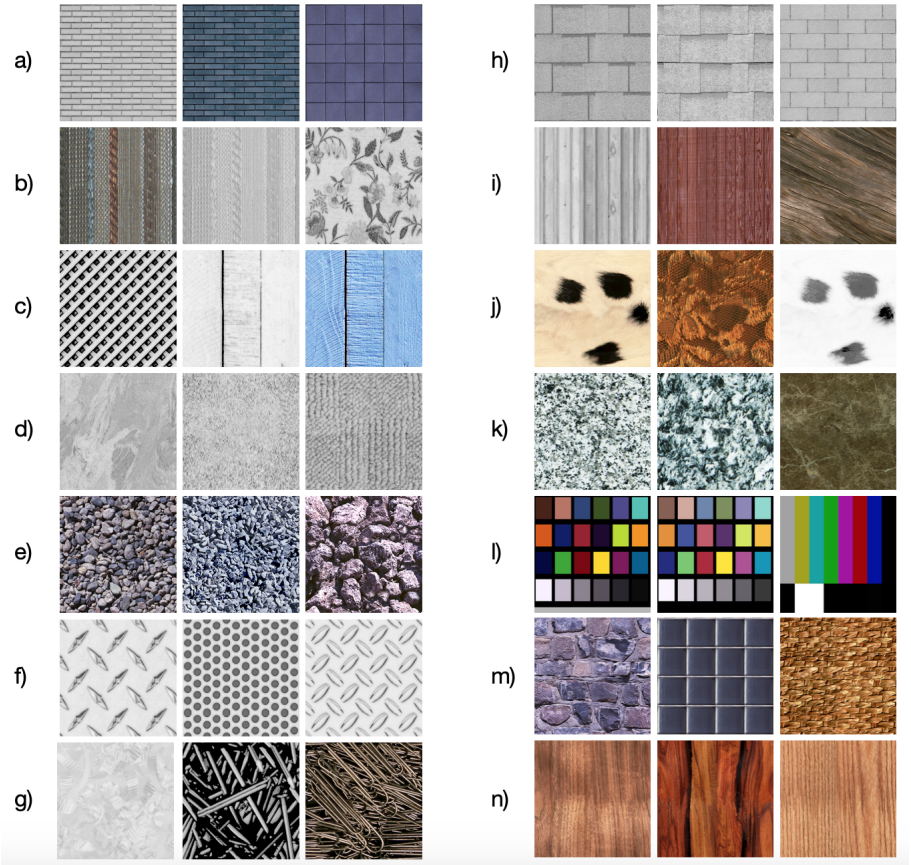


Figure 2.25: Textures from Dataset 2 linked to different classes such as a) brick; b) fabric; c) fence; d) floor; e) ground; f) metal; g) misc; h) roof; i) siding; j) skin; k) ;stone l) test; m) wall; n) wood

First, rotation can be done with any angle, or we limit rotation to angles of rotation multiples of 90° and finally multiples of 90° with an additional margin of 5°.

2.5.5 Optimizers

Optimizers define how neural networks learn. They find the values of parameters such that a loss function is at its lowest. The big question is the following: which is the best optimizer? That depends on the kind of problem that we are trying to solve: instant segmentation, semantic analysis, machine translation, image generation. Many problems out there with different types of losses.

We will choose the model optimizer according to the results obtained during the test phase.

2.5.5.1 SGD (With or Without Momentum)

The gradient descent involves taking small steps iteratively until we reach the correct weight. The problem here is that the weight is only updated once after seeing the entire dataset. So the loss function $\sum_1^m L_m(w_t) = f_t(w_t)$ is typically large. The steps are necessarily big and the ideal weight is then hard to reach. The solution is to update the parameters more frequently. Like in the case of stochastic gradient descent. It updates weights after seeing mini-batch (update parameters only after a few samples) instead of the entire data set.

Instead of using only the gradient of the current step to guide the search, momentum μ also accumulates the gradient of the past steps to determine the direction to go. For the complete algorithm, see Algorithm 1:

Algorithm 1 SGD [51]

Require: Learning rate $\eta > 0$, momentum $\beta > 0$
  while w_t not converged **do**
    $p_t \leftarrow \beta p_{t-1} + \nabla_w f_t(w_t)$
    $w_{t+1} \leftarrow w_t - \eta p_t$
  end while

The first term is the gradient that is retained from previous iterations. This retained gradient is multiplied by a value called *Coefficient of Momentum* which is the percentage of the gradient retained every iteration. In this thesis, we learn the model with momentum's $\mu = \{0, 0.5, 0.9\}$.

For the second part, we subtract the gradient of the loss function with respect to the weights multiplied by η, the *learning rate*. The learning rate controls how quickly the model is adapted to the problem. Smaller learning rates require more training epochs given the smaller changes made to the weights each update, whereas larger learning rates result in rapid changes and require fewer training epochs. A learning rate that is too large can cause the model to converge too quickly to a suboptimal solution, whereas a learning rate that is too small can cause the process to get stuck (fig 2.26).

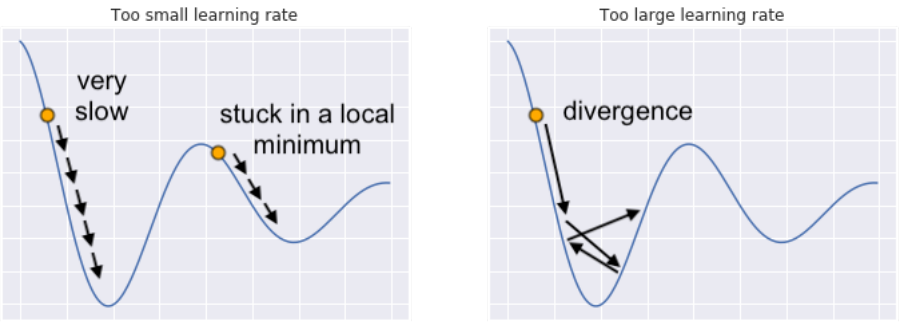


Figure 2.26: 2 cases of stochastic gradient descent where the step applied iteratively with the help of mini-batches is either too small (probable convergence towards a local minimum) or too large (global minimum difficult to reach due to the divergence)

The gradient is a vector which gives us the direction in which loss function has the steepest

ascent. The direction of steepest descent is the direction exactly opposite to the gradient, and that is why we are subtracting the gradient vector from the weights vector.

A widely used technique in gradient descent is to have a variable learning rate, rather than a fixed one. Initially, we can afford a large learning rate. But later on, we want to slow down as we approach a minima. An approach that implements this strategy is called Simulated annealing, or decaying learning rate. We don't use it in our models.

2.5.5.2 RMSprop

In this case, the learning rate is employed differently. Indeed, RMSProp choses **a different learning rate for each parameter** w^j. We apply the following algorithm 2 for each parameter w^j:

Algorithm 2 RMS [52]

Require: Learning rate $\eta > 0$, numerical stabilizer $\epsilon > 0$, Discounting factor $1 \geq \rho \geq 0$
  while w_t not converged **do**
    $g_t \leftarrow \nabla_w f_t(w_t)$
    $\nu_t \leftarrow \rho\nu_{t-1} + (1-\rho)g_t^2$
    $\Delta w_t \leftarrow \frac{\eta}{\sqrt{\nu_t+\epsilon}} g_t$
    $w_{t+1} \leftarrow w_t + \Delta w_t$
  end while

Note that η is the initial learning rate, g_t is the gradient $\nabla_w f_t(w_t)$ at time t along w. g_t corresponds to the component of the gradient along the direction represented by the parameter we are updating.

The hyperparameter μ defines the importance of the current gradient g_t in the determination of w_{t+1}^j. The reason why we use exponential average is because it helps us weigh the more recent gradient updates more than the less recent ones. The term *exponential* is employed because the weightage of previous terms falls exponentially.

We also remark that if we are in a case where $\nu_t << \nu_{t-1}$, it means that we are getting closer to an optimum. If we are 'downhill', we are getting closer to the minimum, if we 'go up', we deviate from the maximum. As we can see when updating the step, $\Delta w_t \leftarrow \frac{\eta}{\sqrt{\nu_t+\epsilon}} g_t$, it will be smaller to avoid to overtaking the minimum.

2.5.5.3 Adam and Adamax

Adam can take different steps for different parameters and with momentum for every parameter it can also lead to a faster convergence. Because of its speed and accuracy, this optimizer can be used for many projects.

For each parameter w^j, Here is what to do:

Algorithm 3 Adam

Require: Learning rate $\eta > 0$, momentum $0 < \beta_1,\beta_2 < 1$, numerical stabilizer $\epsilon > 0$, Discounting factor $1 \geq \rho \geq 0$

while w_t not converged **do**

$g_t \leftarrow \nabla_w f_t(w_t)$

$\nu_t \leftarrow \beta_1 \nu_{t-1} - (1-\beta_1) g_t$

$\hat{\nu_t} \leftarrow \frac{\nu_t}{1-\beta_1}$

$s_t \leftarrow \beta_2 s_{t-1} - (1-\beta_2) g_t^2$

$\hat{s_t} \leftarrow \frac{s_t}{1-\beta_2}$

$w_{t+1} \leftarrow w_t - \frac{\eta}{\sqrt{\hat{s_t}+\epsilon}} \hat{\nu_t}$

end while

j is not written in the equations for clarity. Here, η is the initial learing rate, g_t is the gradient at time t along w^j, ν_t is the exponentional average of gradients along w_j. s_t is the exponentional average of squares of gradients along w_j. β_1 and β_2 are simply hyperparameters. The epsilon ϵ is added for numerical stability (especially to get rid of division by zero when $s_t = 0$)

For Adam's method, the update rule for individual weights w_{t+1} is to scale their gradients ν_t inversely proportional to a (scaled) L2 norm of their individual current and past gradients $\sqrt{\hat{s_t}}$. Adamax's method generalizes the L_2 norm[1] based update rule to a L_p norm[2] based update rule [53]. We then have as expression of s_t

$$s_t = \beta_2^p s_{t-1} + (1-\beta_2^p) \mid g_t \mid^p \tag{2.20}$$

$$= (1-\beta_2^p) \sum_{i=1}^{t} \beta_2^{p-i} \mid g_i \mid^p \tag{2.21}$$

With the Adamax method [53], the stepsize at time t is inversely proportional to $s_t^{1/p}$. Note that the decay term is here equivalently parameterised as β_2^p instead of β_2. if we extend p to infinity, and define $u_t = lim_{p\to\infty}(s_t)^{1/p}$, then, taking the expression of s_t defined in equation 2.21, we obtain the expression of u_t:

$$u_t = lim_{p\to\infty} \left(\beta_2^p s_{t-1} + (1-\beta_2^p) \mid g_t \mid^p\right)^{1/p} \tag{2.22}$$

$$= lim_{p\to\infty}(1-\beta_2^p)^{1/p} \left(\sum_{i=1}^{t} \beta_2^{p(t-i)} \cdot \mid g_i \mid^p\right)^{1/p} \tag{2.23}$$

We can replace $lim_{p\to\infty}(1-\beta_2^p)^{1/p}$ by 1 because the exponent $1/p$ tends to 0.

$$= lim_{p\to\infty} \left(\sum_{i=1}^{t} \beta_2^{p(t-i)} \cdot \mid g_i \mid^p\right)^{1/p} \tag{2.24}$$

[1] L_2 ***norm*** of a vector X is defined as $|X|_2 = \sqrt{\sum_{k=1}^{n} |x_k|^2} = (\sum_{k=1}^{n} |x_k|^2)^{1/2}$

[2] L_p ***norm*** of a vector X is defined as $|X|_p = \sqrt[p]{\sum_{k=1}^{n} |x_k|^p} = (\sum_{k=1}^{n} |x_k|^p)^{1/p}$

Given the fact that every term of the sum is subjected to a power p tending to infinity, if a term is bigger than the others, it becomes infinitely bigger than the others with an exponent $p = \infty$. That's why we can conclude that:

$$u_t = max(\beta_2^{t-1} \mid g_1 \mid, \beta_2^{t-2} \mid g_2 \mid, ..., \beta_2 \mid g_{t-1} \mid, \mid g_t \mid) \tag{2.25}$$

$$u_t = max(\beta_2 \cdot u_{t-1}, \mid g_t \mid) \tag{2.26}$$

The updated parameter w at time $t + 1$ is then equal to:

$$w_{t+1} = w_t - \frac{\eta}{u_t + \epsilon}\hat{\nu}_t \tag{2.27}$$

Similarly to Adam, the epsilon ϵ is added for numerical stability (especially to get rid of division by zero when $u_t = 0$)

2.5.6 Deep features extraction

As a reminder, the next step is to extract the underlying features of a tumour. Now that we are familiar with the VGG16, note that the deep features will be extracted at the penultimate layer (the second dense layer).

To do this, the first step is to choose only one slice to work on. Indeed, a tumour is normally supposed to be studied on its whole shape but we are limited to only one slice since our models only analyse 2D images. The choice of the slice was the one with the largest tumour presence. Simply analyze the binary segmentation and select the slice with the most voxels with a value of "1". As a reminder, the voxels with a value of "1" are the voxels that represent a part of the tumour. This step is to be done on the 3 planes XY, XZ and YZ. In this way, an approximate analysis of the 3D shape can be made.

Once we have chosen a slice of the tumour to be analysed in each plane, it is necessary to set the value of all the pixels that do not make up the tumour to "0", so that the analysis will be exclusively focused on the tumour and will be more accurate.

As mentioned in subsection 2.5.2, the input to the model is by default an image of dimensions $[224 \times 224 \times 3]$. A used area coming from the XY plane for example, is most often of size $[150 \times 150]$ without homogenization of the resolution but with a smaller and variable size in case of homogenization. In order to have the right dimensions, zero value edges have been added.We therefore have grey scale images of size 224×224. In order to have a fully adequate input, we have the choice of duplicating the image in order to have an image of size $224 \times 224 \times 3$ or experimenting. The idea is to combine three types of images. For example, combine 120 keV

monoenergetic imaging, 40 keV and an iodine map. Since all the images of different types for the same case are not offset from each other, this "superposition" is possible.

Now that we have an image with the right dimensions, we need to change the "pixel" values so that the standards set for the input are fully respected. Indeed, the input is composed of images with pixel values ranging from 0 to 255, which are exclusively integers. Note that the values of each voxel of the spectral data will be scaled so that the window of values of each voxel of the spectral data is between 0 and 255. To do this, the maximum and minimum values found in all examples of all spectral data will be used as references for the scalings.

2.6 Features Selection

***In the framework of this master thesis**, rather than choosing to work exclusively with traditional features, deep or otherwise, the choice here is to combine several kinds of extraction. This master thesis will try to demonstrate the impact of different feature combinations, classical and/or deep, in the radiomics process. The question is how do we choose the features that will subsequently make up the training set. A features selection is applied.*

Given the fact that the segmentation was only done by the radiologist Etienne Danse and no one else, Feature reproducibility analysis can therefore not be studied. Other feature selection techniques can nevertheless be used. We will first reduce the features using a Collinearity analysis. In addition, the use of feature importance scores can provide insight into the dataset.

The features selection (Figure 2.27) here will consist of working on fewer variables/features for several reasons. Firstly, to save time. In addition, too many features would inevitably lead to overfitting.

The logical steps are first to minimise the correlation between different features as much as possible. Indeed, the aim is to avoid redundancy in the information we provide to the model. In other words, if two features give exactly the same information, why not keep only one? Here, this reflection applies to a much larger set. To avoid working only with similar features, we use autocorrelation (Pearson correlation coefficients). The idea is to work with a matrix of all possible relationships between two features in the set. If the Pearson correlation coefficients are too high, then the information provided by the features is too similar. An elegant method is to group all features that are highly correlated (Pearson correlation coefficient greater than 0.85) with each other and to keep in each group only the feature with the highest importance in the prediction performance. It is important to note that at this stage the combination of traditional, deep, clinical and MM features is not yet achieved: the selection of each feature set is performed independently of the other feature sets. In the case of a too large feature set (e.g. about 20,000 deep features), generating the Pearson correlation coefficients and the matrix as a whole becomes more complicated to design. We therefore decided to apply a **feature selection beforehand** for sets that are too large. We divide the set of features into several subsets to which we apply a feature selection using the importance of the features by **recursive feature elimination (RFE)** [54]. First, an estimator is trained on the initial feature set and

the importance of each feature is obtained. Then, the least important features are removed from the current feature set. Feature importance is scored either using the provided machine learning model (e.g., decision trees provide importance scores) or using a more general approach that is independent of the full model [55]. This procedure is repeated recursively on the new set until the desired number of features to be selected is finally reached.

When all correlated variables have been filtered to keep only the most important ones, we sort this set again to keep only the most important variables in the set itself.

The concept is to combine different sets. A dimensionality reduction will finally be applied on the combined sets. We again use the Recursive feature elimination. For this method, we combine all uncorrelated feature sets and decide to keep only 3, 5 or 7 features for the prediction model.

After this step, we will have a model defined on the basis of features from all possible types of extraction.

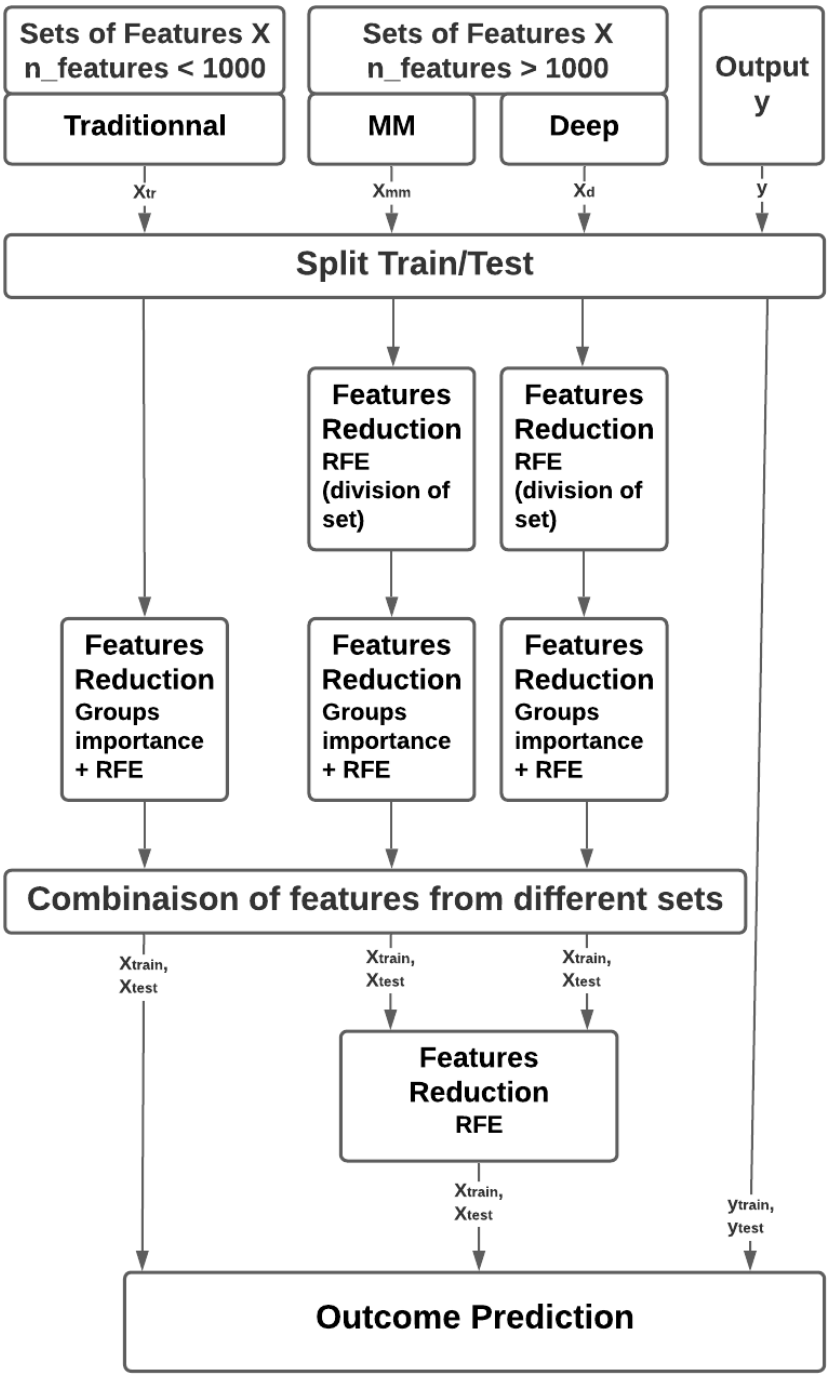


Figure 2.27: Diagram showing the Features selection process

2.7 Prediction Outcome

***In the framework of this master thesis**, we will try to find the most appropriate model (K-Nearest Neighbors, Naive Bayes, Logistic Regression, Support Vector Machine, Decision Tree and Random Forest) for each type of prediction to be treated (microsatellite instability (MSI), KRAS/BRAF mutations, vascular/lymphatic permeation, peri-nervous sheathing and budding). Each of these models has hyperparameters that will also be studied for optimisation.*

2.7.1 Model for Prediction

The models are chosen mainly on the basis of the size of the dataset. It is not necessary to use a neural network as a classifier because the number of samples we have available is not large enough (100000 >>> 28). We will therefore limit ourselves to working with classical models: KNN,Naive Bayes, Logistic Regression, SVM, Decision Tree and Random Forest.

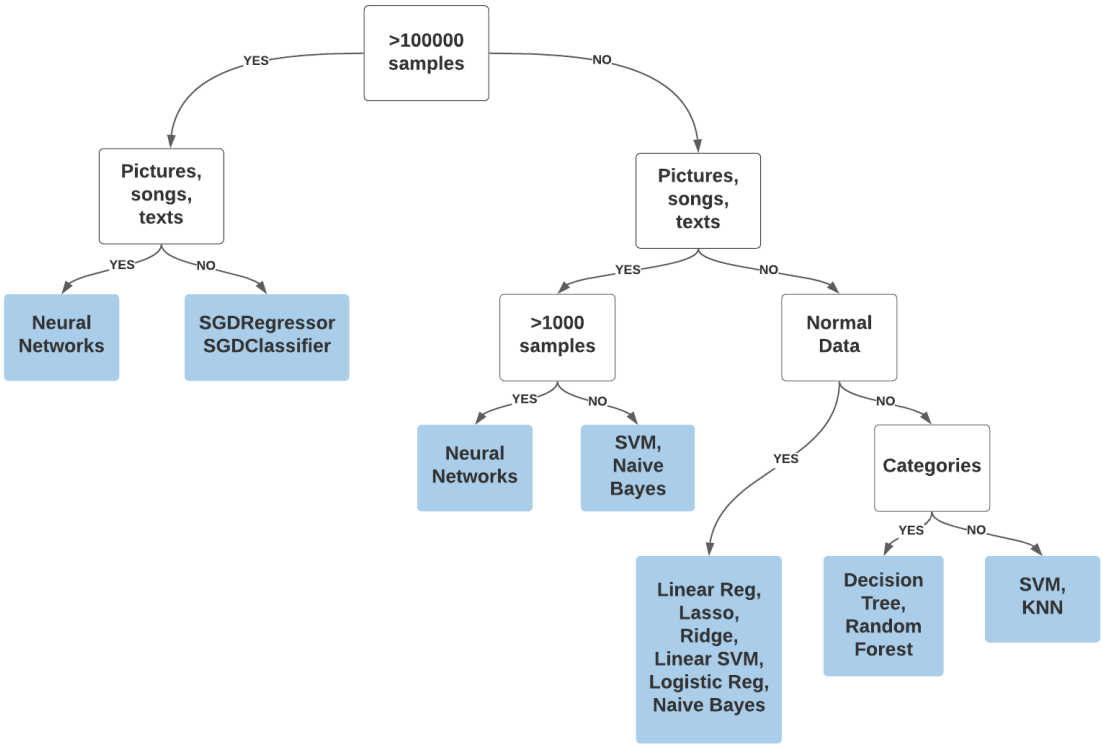


Figure 2.28: A rough guide on how to approach problems with classification estimators to try on the data [56, 57].

2.7.1.1 K-Nearest Neighbors (K-NN)

The k-nearest neighbour method is a supervised learning method [58]. The goal is to classify a new cell by looking at the k nearest annotated cells: the nearest neighbors. k must not be a multiple of the number of classes. If there are only 2 classes, then k must be odd.

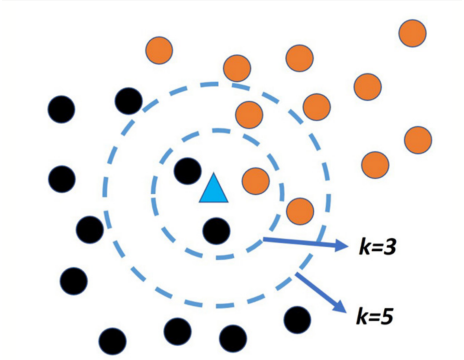


Figure 2.29: Pictorial explanation of the classifier *knn* [17]

The **advantage** of this classification is that it requires almost no assumptions about the data. It has a non-parametric approach. Nothing has to be deduced from the data, except k and possibly $D()$ if we take into account the distance of the neighbours. Therefore, more distant neighbours will have less impact on the choice of classification. In the case where near neighbours have more impact than far neighbours, it is reasonable to use the Minkowski metric (p-norm). In $\mathbb{R}^q$:

$$\|x' - x_j\|^p = \left(\sum_{i=1}^{q} |(x_i)' - (x_i)_j|^p \right), \tag{2.28}$$

where q is the number of features, p is a positive constant which is usually equal to 1 (Manhattan Distance) or 2 (Euclidian distance).

It is then sufficient to calculate the class probabilities for each class. That is the probability that a new sample is in one of the possible classes. For example, in a binary classification problem (class is 0 or 1):

$$p(class = 0) = \frac{count(class = 0)}{count(class = 0) + count(class = 1)} \tag{2.29}$$

A **drawback** is the complexity in searching the nearest neighbors for each sample. Indeed, KNN works well with a small number of input variables (number of features), but struggles when the number of inputs is very large. In high dimensions, points that may be similar can have very large distances.

An other **drawback** of classification is that a "majority vote" phenomenon occurs if the classes are asymmetric. For example, a class A that is much more frequent than another class B. Since class A is dominant, it will be more common that the k nearest neighbours of an sample to be classified belong to class A. The vote is therefore biased because it is a majority.

2.7.1.2 Naive Bayes

Naive Bayesian classification is a type of simple probabilistic Bayesian classification based on Bayes' theorem with strong (naive) independence of assumptions. In other words, this probabilistic model is a "statistically independent features model".

Suppose $p(C|F_1, F_2, ..., F_p)$ the probabilistic Bayesian classifier where p is the number of features F and C is a dependent class variable, and using Bayes' theorem, we write:

$$p(C|F_1, F_2, ..., F_p) = \frac{p(C)p(F_1, ..., F_p|C)}{p(F_1, ..., F_p)} \tag{2.30}$$

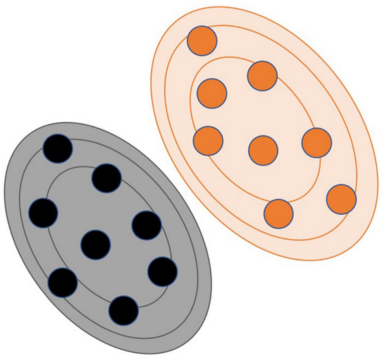


Figure 2.30: Pictorial explanation of the classifier *Naive Bayes* [17]

The greatest strength of this classifier, in the context of this project, is that when the independence assumption is verified, the Naive Bayes classifier outperforms other models such as logistic regression and requires less training data.

The biggest **limitation** of Naive Bayes is the assumption of independent features. In real life, it is almost impossible to get a set of features that are completely independent.

2.7.1.3 Logistic Regression

Logistic regression [59] is a binomial regression model. The relationship between the final decision and the independent variables is expressed as:

$$\sigma(\beta^T X) = \frac{1}{1 + e^{-\beta^T x}} \tag{2.31}$$

where β is a vector of parameters of size $p \times 1$ with p the number of features. x is a sample with a size $p \times n$ with n, the number of training samples. So $\beta^T x = \sum_{i=1}^{p} \beta_i x_i = \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p$

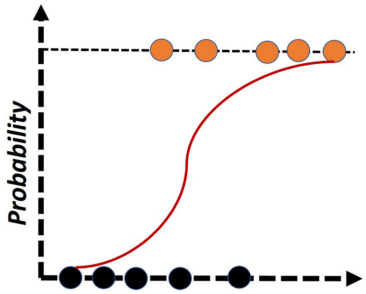


Figure 2.31: Pictorial explanation of the classifier *Logistic Regression* [17]

The weights β are found thanks to the maximum likelihood procedure. A cost function $J(\beta)$ represents optimization objective. $J(\beta)$ is created and minimized so an accurate model with minimum error is developed.

Knowing that:

$$P(Y = 1|X = x) = \sigma(\beta^T x) \tag{2.32}$$

$$P(Y = 0|X = x) = 1 - \sigma(\beta^T x) \tag{2.33}$$

The two functions are compressed into a single one :

$$P(Y = y|X = x) = \sigma(\beta^T x)^y \cdot [1 - \sigma(\beta^T x)]^{(1-y)} \tag{2.34}$$

Now we know the probability mass function, we can write the likelihood of all the data:

$$L(\beta) = \Pi_{i=1}^{n} \sigma(\beta^T x^{(i)})^{y^{(i)}} \cdot \left[1 - \sigma(\beta^T x^{(i)})^{1-y^{(i)}}\right] \tag{2.35}$$

adding a log gives the log likelihood for the logistic regression:

$$LL(\beta) = \sum_{i=1}^{n} \left[y^{(i)} log(\sigma(x^{(i)})) + (1 - y^{(i)}) log(1 - \sigma(x^{(i)}))\right] \tag{2.36}$$

where n is equal to the number of samples, $y^{(i)}$ and $x^{(i)}$ are the target value and the sample of case i.

To reduce the cost value, we can use the gradient descent to minimize $LL(\beta)$. At each iteration, we need to run the gradient descent function on each parameter β_j. We then apply the following algorithm 4:

where n in the number of training samples, and $\sum_{i=1}^{n} (y^{(i)} - \sigma(\beta^T x^{(i)})) x_j^{(i)} = \frac{\partial LL(\beta)}{\partial \beta_j}$

Algorithm 4 Logistic Regression

Require: $\alpha > 0$
  while β_j not converged **do**
  $\beta_j \leftarrow \beta_j - \alpha \sum_{i=1}^{n} (y^{(i)} - \sigma(\beta^T x^{(i)})) x_j^{(i)}$
  end while

2.7.1.4 Support Vector Machine (SVM)

SVM is a machine learning algorithm that can be used to solve classification, regression and anomaly detection problems. It is known for its strong theoretical guarantees and its great flexibility.

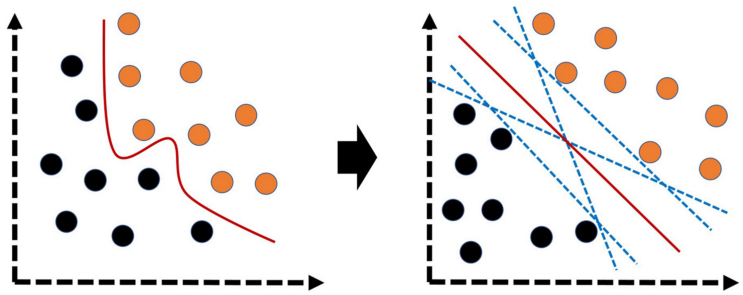


Figure 2.32: Pictorial explanation of the classifier *SVM* [17]

The aim is to separate the data into classes using as "simple" a boundary as possible, so that the distance between the different groups of data and the boundary separating them is maximum. This distance is also called the "margin". Positive and negative samples are separated with as wide a band (margin) as possible between the 2 classes. The 2 conditions $\vec{w} \cdot \vec{x_+} + b \geq 1$ and $\vec{w} \cdot \vec{x_-} + b \leq -1$ can be combined into one as follows:

$$y_i(\vec{w} \cdot \vec{x_i} + b) \geq 1, , \quad i = 1, ..., n \tag{2.37}$$

with n equal to the number of training examples. We suppose here that $y = \{-1, 1\}$.

To calculate the width of the margin, we will apply a dot product between the difference of the support vectors ($\vec{x_+}$ and $\vec{x_-}$) and the normalized w vector. We know that the shortest distance between a point and a hyperplane is perpendicular to the plane, and hence, parallel to $\frac{\vec{w}}{\|\vec{w}\|}$. Knowing that, we can measure the margin width by subtract $\vec{x_+} \cdot \frac{\vec{w}}{\|\vec{w}\|}$ by $\vec{x_-} \cdot \frac{\vec{w}}{\|\vec{w}\|}$:

$$width = (x_+ - x_-) \cdot \frac{\vec{w}}{\|\vec{w}\|} = ((1 - b) - (-1 - b)) \frac{1}{\|\vec{w}\|} = \frac{2}{\|\vec{w}\|} \tag{2.38}$$

Maximising the width of the margin $\frac{2}{\|\vec{w}\|}$ is therefore equivalent to minimising $\|\vec{w}\|$. The technique of margin maximisation ensures greater robustness to noise, and therefore a more

generalisable model. Let's take the example of the Hard margin SVM which does not allow mis-classification errors. Hard margin linear SVM solves the following optimization problem:

$$\min_{w,b} \frac{1}{2}\|w\|^2 \tag{2.39}$$

The factor $\frac{1}{2}$ and the exposant 2 are set for mathematical conveniance.

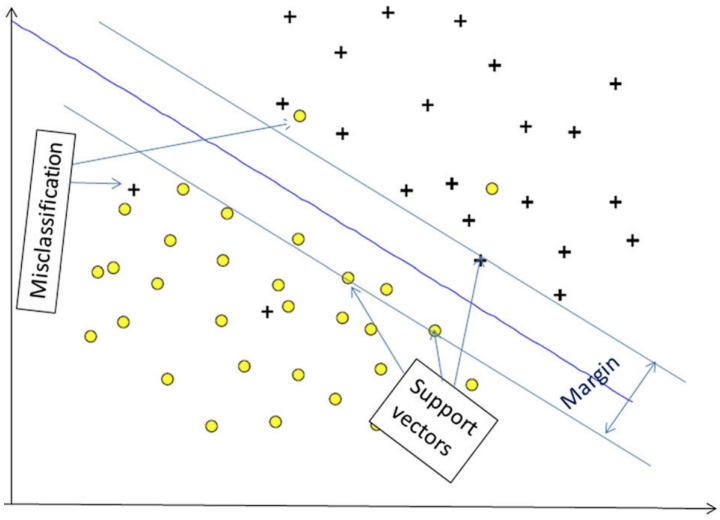


Figure 2.33: Maximum-margin hyperplane and margins for an SVM trained with samples from two classes [60].

From these basic conditions (eq 2.37 and 2.39), a Lagrangian function L is formed:

$$L(w,b,\alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i [y^{(i)}(w^T x^{(i)} + b) - 1] \tag{2.40}$$

with α_i, a Lagrangian multiplier greater than zero. The goal now is to minimize this Lagrangian function as a function of the weights w and b. After optimization as a function of w and b, a new form of the Lagrangian function is obtained [61]:

$$L(\alpha) = \sum_i \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i \cdot x_j \tag{2.41}$$

The notion of a frontier assumes that the data are linearly separable, which is rarely the case. To remedy this problem, we use the *kernel trick* consisting in reconsidering the problem in a higher dimensional space. *Kernels* are used to separate the data by projecting them into a higher-dimensional vector space called *feature space*. We therefore use a transformation

$$K(\vec{x_i}, \vec{x_j}) = \Phi(\vec{x_i}) \cdot \Phi(\vec{x_j}) = e^{-\frac{\|\vec{x_i} - \vec{x_j}\|}{\sigma}} \tag{2.42}$$

With a standard deviation σ chosen that should not be too small at the risk of shrunking right around the sample which implies overfitting.

Finally, it should be mentioned that outliers may be present. To deal with this possibility, the term $C \sum_i \xi_i$ is added to the equation 2.39, where C is a hyperparameter to fix and $\sum_i \xi_i$ is the sum of the classification errors. When C is small, classification mistakes are given less importance and focus is more on maximizing the margin, whereas when C is large, the focus is more on avoiding misclassification at the expense of keeping the margin small.

2.7.1.5 Decision Tree and Random Forest

Decision Tree: In order to create a decision tree, it is necessary to set up efficient decisions to make a classification in such a way that the decision tree has as few decisions as possible to make [62]. We first start by looking at how well features predicts weather or not the right classification of the samples. We do this step for every feature. We then build p little trees with two leaves. Every leaf contains the number of the samples belonging to a certain class respecting the condition (See figure 2.35).

If a leaf contains only the samples of one class, say class A, this means that all samples of other classes such as B, C do not meet the conditions of the leaf except for samples of class A. Class A is then said to be pure. There are several ways to quantify the Impurity of the leaves. We have for example the Gini impurity which is calculated like this:

$$GI = 1 - \sum_{i=1}^{C} (p_i)^2 \tag{2.43}$$

where GI is the impurity of one leaf, C is the number of possible classes and p_i the probability a sample made it to this leaf belongs to a class i. The total Gini impurity is the weighted average of Gini impurities for the leaves. If we take again the example of pure class A, then $GI = 1 - (p_A^2 + p_B^2 + p_C^3) = 1 - (1 + 0 + 0) = 0$. Finally, each new decision is chosen from the moment when its Gini impurity is the lowest. Samples belonging to pure leaves are therefore no longer included in the decisions of the lower levels.

The impurity should also be calculated with the entropy:

$$E(S) = \sum_{i}^{C} -p_i log_2(p_i) \tag{2.44}$$

Random forest allows to be as simple as the decision trees but with flexibility. The idea is to work with bootstrapped[1] dataset. Then a random subset of features at each step is considered. These two steps are repeated. A wide variety of trees is built.

[1]Bootstrapping: Statistical inference method based on multiple data replication from the studied dataset

Then we run the data (a new sample) in all the built trees. The class with the most "votes" will be the class defined for the new sample [63].

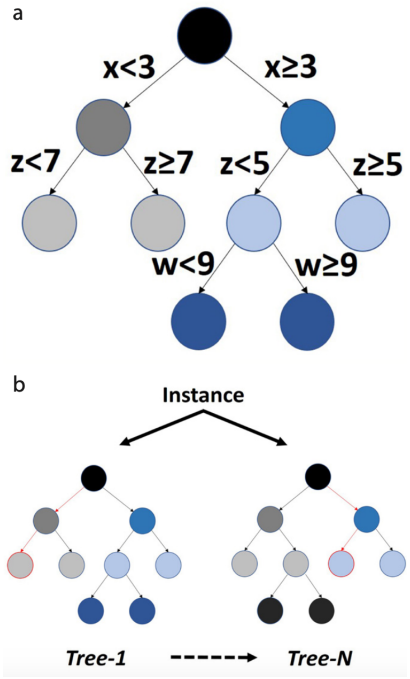


Figure 2.34: Pictorial explanation of the classifiers *Decision Tree* and *Random forest* [17]

2.7.2 Algorithm Validation

For the purposes of this master thesis, one way to examine the performance of the model is to randomly keep a portion of the data set blind. This portion of the data is intended to be used only for prediction (blind data). This subset is not intended to train the models but to test them. These approaches require the collection or retention of a significant amount of data for validation. These methods are rarely used in research with human participants, where data collection is usually associated with high costs. [64]. Given the lack of data in the dataset, we make the choice to use our (full) dataset with a limited number of samples to evaluate the trained model with a validation method and develop the model evaluation metrics.

The validation method used is a **partially nested cross validation**. This means that a first part of the process is applied in a non-nested manner. In our case, feature selection, model evaluation, and pre-processing to be employed were performed on the pooled training and test data, rather than in each cross-validation fold. Although this is a fairly common practice, the results of the partially nested validation showed that selecting features in a non-nested manner yields significantly biased results [64]. However, this method will be used because it is less computationally demanding and because of our limitation of examples.

The algorithm (partially nested cross-validation) is therefore divided into two main steps: evaluation of all models with multiple combinations (determines whether the feature set includes

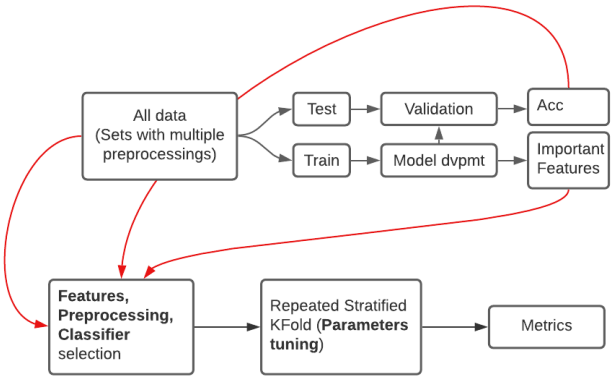


Figure 2.35: Schematic approach used to apply feature selection, pre-processing selection and classifier selection.

deep features or not, etc.) and multiple pre-processings (with segmentation correction or not, with resolution or not). This first step is repeated m times and the performance of all models is calculated as an average of the classification performance. The criterion for model selection is based on accuracy for the first step. However, metrics such as sensitivity and specificity would have been welcome. The train/test split is done m times randomly with the condition of always having enough examples of each class in the training set (more than one example of each class). Once the models are sorted based on their accuracy, the choices about pre-processing, feature combination and classifier selection are made.

We can then focus on tuning the hyperparameters of the final models and extracting the corresponding metrics. Due to the fact that most of the classes are not balanced, the repeated stratified fold method is used. This method consists of repeating the cross-validation process several times with an imposed stratification for each fold. Indeed, the folds are made keeping the percentage of samples for each class. We do not risk to find a training set without a certain class. A confusion matrix, the ROC and the learning curve are also generated. Note that the choice of the hyperparameters of the final models uses the $F1 - Score$ metric as selection criterion. This seems much more appropriate in the sense that it is the harmonic mean of precision and recall.

Remark: For the first step, cross validation with repeated stratified folds was not used but seemed to be a more efficient method. The reason for this non-use is related to the important running time required for the first step. The step would have been even longer with a cross validation with repeated stratified folds. Moreover, the methods are relatively similar.

2.8 Summary of Chapter 2

This chapter 2 gives a more complete explanation of the processes used. It explains in detail the sub-steps to acquire the data being DECT examples composed of colorectal tumors. The chapter develops the different pre-processing techniques used to improve the prediction. The feature extraction will be done in 3 different ways: traditional, with DECT filtering and with deep learning. Finally, we will specify the method used to select our features, our classification

models and the adequate pre-processing.

Chapter 3

Results

In order to find the most suitable model, there will be two steps. First of all, We will do a "grid" by analysing the results in the hope of finding certain trends such as a type of classifier that performs better than the others, features that is more adequate for prediction, a more constant accuracy in certain predictions, etc. This method is inspired by a work [65] in a similar situation to the one we are facing: working with a very large number of features despite our small number of examples. Then, a classical but complete analysis of the chosen models will be done (accuracy, sensitivity, AUC,...) [66]

3.1 Choice of models

In the first step of the partially nested cross validation, recall that we are looking for the right classifier, pre-processing and features.

3.1.1 Pre-processing techniques comparison

The choice of pre-processing and classifier is done in the following way: the prediction accuracy of several models (different classifiers with different types of pre-processing) is measured (see Table 3.1). Note that the feature set used to compare the different pre-processing techniques does not include the other feature types (deep and MM). Only the traditional features will be used to estimate which pre-processing and classifier is more suitable. Also, since we do not know how many features will be decisive, the accuracies in the table are an average of the accuracies made with a final set of features composed of 3, 5, 7 features (with 8 iterations). When we talk about a final set, we are talking about a set after feature selection.

A first observation to make is that for each case, it seems best to apply the segmentation correction. BRAF/KRAS mutations seem to be best predicted when no resolution is homogenized

CRC	PreProc	Classifier						Average Ranking
		KNN	NB	LG	SVM	DT	RF	
Mutation BRAF/ KRAS	R + S	60.8 ± 18.2	65.0 ± 8.8	55.8 ± 11.8	64.2 ± 14.4	56.7 ± 15.2	71.7 ± 13.0	**2**
	noR + S	45.8 ± 17.2	66.7 ± 9.6	54.2 ± 22.4	55.83 ± 19.6	50 ± 25.0	57.5 ± 19.0	3.67
	R + noS	60 ± 16.6	67.5 ± 9.8	58.3 ± 13.1	58.3 ± 11.7	62.5 ± 15.9	62.5 ± 15.9	2.17
	noR + noS	60 ± 22.8	70 ± 10.2	51.7 ± 20.4	62.5 ± 19.8	70.8 ± 22.0	68.3 ± 22.0	2.17
Permeation	R + S	50.9 ± 14.6	57.4 ± 13.4	56.0 ± 14.0	49.5 ± 12.6	53.2 ± 15.4	56.0 ± 10.0	**1.5**
	noR + S	52.8 ± 20.6	48.6 ± 17.0	45.4 ± 14.6	42.6 ± 14	49.5 ± 20	50 ± 18.6	3
	R + noS	54.17 ± 14.4	56.9 ± 16.2	49.5 ± 9.8	47.2 ± 13.2	47.7 ± 13.4	44.4 ± 10.4	2.667
	noR + noS	43.0 ± 9.4	44.9 ± 20	55.1 ± 12.0	51.9 ± 12.2	44.0 ± 8.4	52.8 ± 7.6	2.83
MSI	R + S	57.4 ± 14.6	66.67 ± 13.8	65.74 ± 14.6	65.3 ± 20	63.0 ± 14.2	59.7 ± 16.0	2.5
	noR + S	62 ± 17.8	69 ± 18.0	64.8 ± 14.2	65.3 ± 14	62.5 ± 16.4	67.6 ± 15.4	**1.83**
	R + noS	56.5 ± 12.2	59.7 ± 10.3	54.6 ± 12.2	63.9 ± 13.6	64.4 ± 15.8	54.6 ± 12.6	3.667
	noR + noS	59.3 ± 14.2	66.2 ± 10.0	59.7 ± 12.2	64.8 ± 10.2	67.1 ± 18.4	67.6 ± 14.2	2
Peri-Nervous Sheating	R + S	49.0 ± 15.6	55.2 ± 12.8	39.1 ± 14.8	44.3 ± 17.6	46.4 ± 11.4	47.4 ± 13.2	2.83
	noR + S	46.9 ± 13.0	55.2 ± 16.8	33.9 ± 12.4	42.7 ± 19.4	33.9 ± 18.6	34.4 ± 16.2	4
	R + noS	48.4 ± 16.2	65.6 ± 19.6	47.4 ± 11.6	57.8 ± 18.4	57.3 ± 11.6	58.3 ± 10.8	**1.33**
	noR + noS	56.8 ± 12.8	55.7 ± 10.4	45.3 ± 16.4	56.8 ± 13.2	54.7 ± 12.6	52.6 ± 15.2	1.833
Budding	R + S	45.2 ± 15.6	35.1 ± 15.2	40.5 ± 14.4	39.3 ± 12.2	44.6 ± 16.0	44.0 ± 15.2	2
	noR + S	45.2 ± 16.2	39.3 ± 12.0	35.7 ± 12.6	42.3 ± 13.6	36.9 ± 11.8	43.5 ± 17.2	2.67
	R + noS	47.0 ± 14.2	42.3 ± 13.6	38.7 ± 16.6	45.2 ± 17.2	39.9 ± 16.6	47.0 ± 16	**1.33**
	noR + noS	41.1 ± 15.4	33.3 ± 16.2	29.2 ± 9.0	35.1 ± 17.8	31.0 ± 19.6	26.8 ± 12.2	4
Stage	R + S	47.7 ± 19.6	60.6 ± 11.4	56.9 ± 12.4	53.2 ± 11.4	60.2 ± 15.0	60.2 ± 6.1	**1.33**
	noR + S	37 ± 16.6	45.8 ± 11.4	32.4 ± 12.6	42.1 ± 15.4	34.3 ± 14.2	35.6 ± 20.0	4
	R + noS	56 ± 16.8	55.6 ± 12.2	44.0 ± 15.6	50.9 ± 14.6	43.5 ± 13.4	46.3 ± 16.0	2.17
	noR + noS	50.5 ± 12.2	48.1 ± 9.0	41.2 ± 17.4	49.5 ± 17.6	46.8 ± 10.8	46.8 ± 10.4	2.5
Grade	R + S	58.3 ± 14.2	64.1 ± 17.0	47.9 ± 16.4	54.2 ± 18.4	66.67 ± 8.8	67.2 ± 17.2	**1.67**
	noR + S	42.2 ± 16.0	58.9 ± 19.0	37.5 ± 13.2	37.0 ± 14.6	49.0 ± 16.4	49.0 ± 12.2	3.83
	R + noS	51 ± 15.6	64.4 ± 24.4	46.4 ± 15.0	55.2 ± 16.8	60.4 ± 19.8	50 ± 19.8	2.33
	noR + noS	43.2 ± 19.8	64.5 ± 16.2	51.0 ± 12.2	45.8 ± 16.4	68.2 ± 13.2	67.2 ± 12.2	2.17

Table 3.1: The pre-process column distinguishes between pre-processes integrating a resolution homogenization (R for yes and noR for no), and a segmentation correction (S for yes and noS for no). Classifier informs about the classifier used. Average Ranking calculates the "place on the podium". A pre-processing type that is the best for each classifier used will have an average ranking of 1. A pre-processing type that is always the worst for each classifier will have an average ranking of 4 because this table only includes 4 pre-processing types.

with a *Random Forrest* classifier. We choose this classifier because the pre-processing with the best average ranking includes resolution and segmentation correction. In the case where these pre-processing techniques are used, the *Random Forrest* classifier gives the results with the best accuracy (71.7 ± 13.0). In summary, we first choose the pre-processing and then the classifier. This reasoning is also applied for the following classifier choices. Predictions on vascular/lymphatic permeations will be made with a Naive Bayes model. From the ranking, it appears that the most important pre-processing step is resolution homogenization. The MSI status does not seem to require homogenization to get the best results. This is because MSI status is primarily recognized for the homogeneity of texture measured in tumors that tend to be more indolent than MSS tumors [32]. The peri-nerve sheath is best predicted with a pre-processing that does not take into account the segmentation correction. The model chosen is again the Naive Bayes model. Budding appears to be the least predictable pattern. This is because it is defined as the presence of single tumor cells or small clusters of up to 5 cells in the tumor stroma. Because the initial resolution can vary between 0.5 mm/pixel and 1 mm/pixel, it is not possible to perceive budding by DECT and textural analysis of the tumor as a whole does not seem to give reliable information about this phenomenon. We do not believe that we can form a reliable predictive model or at least not with traditional features alone. Nevertheless, the approach will be pursued with a set of features with resolution homogenization but without segmentation correction. For the classifier, we will choose the KNN. For the determination of the step, it is important to note that this is the only case containing three classes and not two. It is not surprising that the average accuracy is worse because the classifier has to make a choice not with 2 classes but with 3, which leaves it more chances to make a mistake. The chosen model is Naive Bayes with a pre-processing including a homogenization of the resolution and a correction of the segmentation. The first places in the ranking are occupied by the pre-processing including homogenization. This can easily be justified by the fact that the grade is defined, among other

things, by the size of the tumor (more details in appendice A). Pixel homogenization allows a more valid comparison of tumor sizes.

3.1.2 Combinaisons comparaison

The next step is to check whether the addition of the deep features and MM features to the feature set improves the accuracy of the predictions or not. Note that the deep features were extracted from models trained differently. Some of the trained models were trained with a dataset composed of human tissues images while the second dataset is composed of images with various textures but with no apparent link with colorectal tumors.

3.1.2.1 Models Learning

To be sure of the efficiency of adding the deep features in the basic feature set, and while waiting to acquire the DECT data, tests were performed on another dataset. The dataset consists of Glioblastoma[1] cases. The target includes methylated and unmethylated cases.

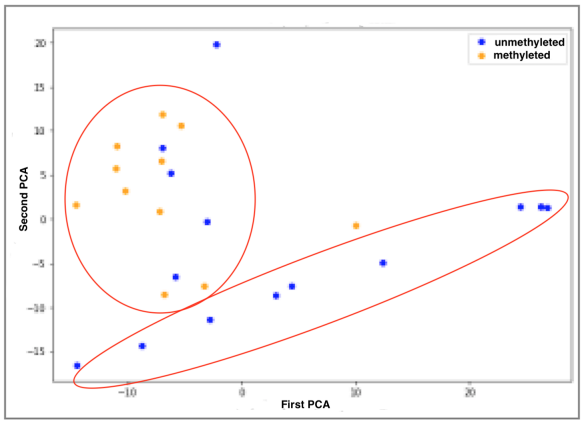


Figure 3.1: The first two PCAs of a set of deep features extracted from a pre-trained vgg16 model by imagenet. The two classes separate methylated and unmethylated glioblastoma patients

Figure 3.1 shows the PCA[2] of a set of features related in a graph. We can see that 2 groups are formed representing approximately the methylenated and non-methylenated classes. The results were relatively satisfactory and encouraged us to continue our in-depth study of deep features.

[1]Glioblastoma is the most common brain cancer in adults. It is caused by the abnormal proliferation of cells in the central nervous system.

[2]PCA: A technique that transforms variables that are correlated with each other into new variables that are decorrelated from each other. These new variables are called *principal components*. This reduces the number of variables and makes the information less redundant.

	No Aug	Aug (all angles)	Aug (k*90°)	Aug (k*90°+[-5;5])
SGD (1e-2)	89	86	90	89
SGD (1e-2) momentum 0.5	90	87	88	90
SGD (1e-2) momentum 0.9	89	87	**91**	88
SGD (1e-3)	89	88	90	90
SGD (1e-3) momentum 0.5	**91**	87	89	89
SGD (1e-3) momentum 0.9	90	85	90	90
RMSprop (1e-3)	**91**	86	**91**	89
Adam (1e-2)	**91**	88	90	89
Adamax (2e-3)	90	88	90	89

Table 3.2: model pretrained imagenet, trained by dataset 1. The four columns group the results obtained with (Aug) or without (No Aug) augmentation during training. Augmentation always includes the flip and a rotation. Either the rotation can be done with any angle, or it is limited to multiple 90° rotation angles or multiple 90° rotation angles with an additional 5° margin.

	No Aug	Aug (all angles)	Aug (k*90°)	Aug (k*90°+[-5;5])
SGD (1e-2)	89	85	90	89
SGD (1e-2) momentum 0.5	**91**	87	89	90
SGD (1e-2) momentum 0.9	88	85	88	88
SGD (1e-3)	90	88	90	90
SGD (1e-3) momentum 0.5	90	87	89	89
SGD (1e-3) momentum 0.9	**91**	86	90	90
RMSprop (1e-3)	90	89	**91**	89
Adam (1e-2)	89	85	88	89
Adamax (2e-3)	**91**	89	90	89

Table 3.3: model pretrained imagenet, trained by dataset 2 (balanced) and then trained by dataset 1. The four columns group the results obtained with (Aug) or without (No Aug) augmentation during training. Augmentation always includes the flip and a rotation. Either the rotation can be done with any angle, or it is limited to multiple 90° rotation angles or multiple 90° rotation angles with an additional 5° margin.

Four models were chosen to serve as deep feature extractors. The first model was trained by dataset 1, the second by dataset 2 and 1, the third by dataset 2 and 2 and the last one by dataset 1 and 2.

As shown in Table 3.2 and those that follow, the models were trained with different augmentations and optimizers. Thus, a model was chosen based on the accuracy of the test set predictions. Since several models appear to have the same level of accuracy, the loss during validation was the distinguishing feature. The chosen model has an SGD optimizer with a learning rate $\eta = 10^{-3}$ and a momentum $\mu = 0.5$ with an augmentation composed of flip, and multiple 90°rotation angles.

For the model trained by dataset 2 and then 1, we notice that the results (Table 3.3) are

	No Aug	Aug (all angles)	Aug (k*90°)	Aug (k*90°+[-5;5])
SGD (1e-2)	71	65	66	70
SGD (1e-2) momentum 0.5	67	49	71	65
SGD (1e-2) momentum 0.9	68	44	63	65
SGD (1e-3)	63	45	57	62
SGD (1e-3) momentum 0.5	66	56	66	67
SGD (1e-3) momentum 0.9	70	55	67	67
RMSprop (1e-3)	67	61	66	67
Adam (1e-2)	66	65	**73**	67
Adamax (2e-3)	67	59	65	66

Table 3.4: model pretrained imagenet, trained by dataset 2 (balanced). The four columns group the results obtained with (Aug) or without (No Aug) augmentation during training. Augmentation always includes the flip and a rotation. Either the rotation can be done with any angle, or it is limited to multiple 90° rotation angles or multiple 90° rotation angles with an additional 5° margin.

	No Aug	Aug (all angles)	Aug (k*90°)	Aug (k*90°+[-5;5])
SGD (1e-2)	68	60	72	71
SGD (1e-2) momentum 0.5	71	56	**73**	71
SGD (1e-2) momentum 0.9	68	51	63	**73**
SGD (1e-3)	63	51	61	55
SGD (1e-3) momentum 0.5	63	51	62	63
SGD (1e-3) momentum 0.9	71	61	60	65
RMSprop (1e-3)	71	62	72	67
Adam (1e-2)	68	61	65	67
Adamax (2e-3)	67	59	68	70

Table 3.5: model pretrained imagenet, trained by dataset 1 and then trained by dataset 2 (balanced). The four columns group the results obtained with (Aug) or without (No Aug) augmentation during training. Augmentation always includes the flip and a rotation. Either the rotation can be done with any angle, or it is limited to multiple 90° rotation angles or multiple 90° rotation angles with an additional 5° margin.

relatively similar to the models trained only by dataset 1 (previous table 3.2). Here the chosen model has a *Adamax* optimizer with a learning rate $\eta = 2 * 10^{-3}$ not augmented.

For the model only trained by dataset 2 (Table 3.4), the chosen model has a *Adam* optimizer with a learning rate $\eta = 10^{-2}$ with the augmentation including image rotations with multiple 90° rotation angles.

For the model only trained by dataset 1 and 2 (Table 3.5), the chosen model has a *SGD* optimizer with a learning rate $\eta = 10^{-2}$ and a momentum $\mu = 0.5$ with the augmentation including image rotations with multiple 90° rotation angles.

We can see the learning curve, the loss and confusion matrix of these models presented in Figures 3.3.

It seems that the models tend to have slightly better predictions when the augmentations are limited to image rotations with multiple 90° rotation angles as well as flips. **Our guess** as to why this is the case is that rotations with multiple 90° rotation angles cause black corners to appear in the augmented image that can distort the classification (see figure 3.2).

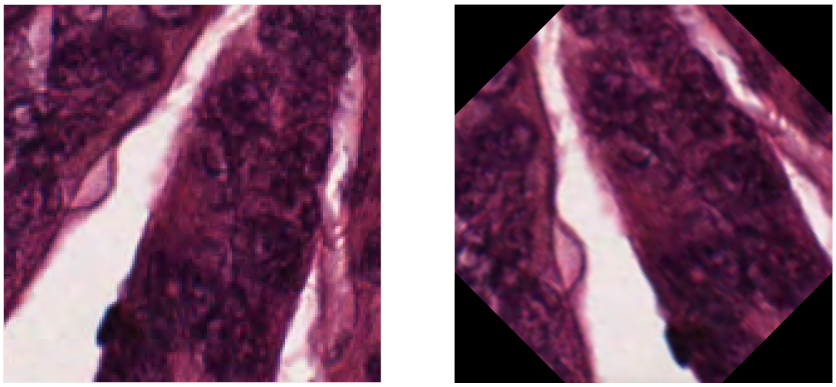


Figure 3.2: from left to right: a) image of a dataset intended to train a model without augmentation; b) the same image augmented via the rotation principle. A 45° rotation is applied in this case

.

3.1.2.2 Deep features choice

Again, the combination of feature types (traditional + MM + deep) will be determined based on the accuracy. We observe in the majority of cases (Table 3.6), that the best predictions do not require especially MM and deep features. Note that Budding gives **slightly** better results with the integration of MM and deep features.

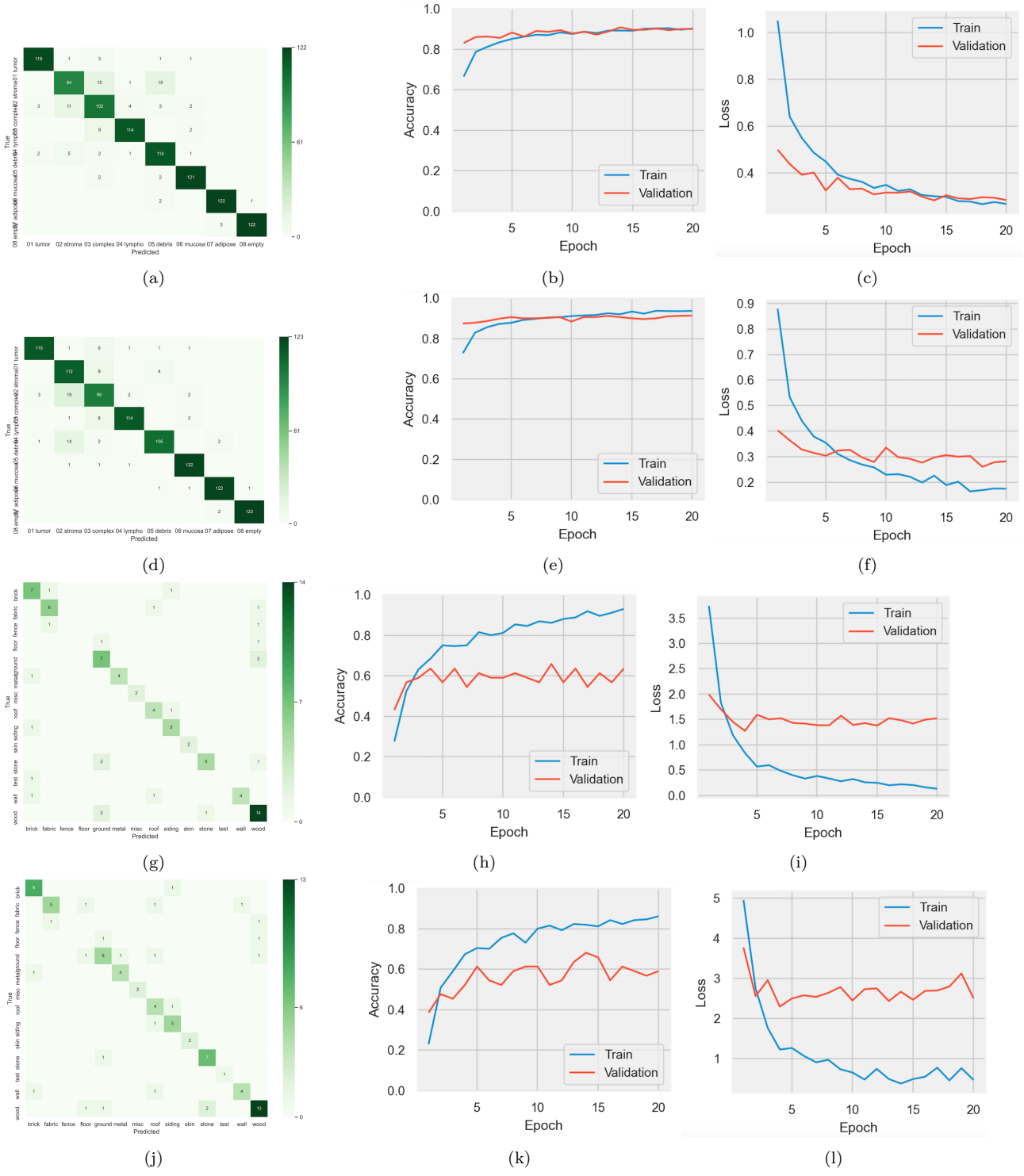


Figure 3.3: a) confusion matrix, b) precision and c) loss of the model trained by dataset 1 only; d) confusion matrix, e) precision and f) loss of the model trained by dataset 2 then 1; g) confusion matrix, h) precision and i) loss of the model trained by dataset 2 only; j) confusion matrix, k) precision and l) loss of the model trained by dataset 1 then 2.

CRC	Deep Features extractor choice					
	No Combinaison	Dataset 1	Dataset 2+1	Dataset 2	Dataset 1+2	No Learning
Mut. BRAF /KRAS	71.7±13.4	66.7±15.2	71.7±18.2	75.8 ± 9.6	72.5±13.8	65.8±21.6
Permeation	57.3±13.6	51.8±19.8	45.8±12.0	51.9±11.4	50 ± 11.0	49.1 ± 9.8
MSI	69.0±14.0	39.8±22.0	44.4±11.6	49.1±13.8	54.2±18.6	59.2±11.8
Peri-Nerv. Sheating	65.6±20.4	49.5±19.6	59.4±19.4	53.7±17.2	52.6±19.8	/
Budding	47 ± 14.8	44.1±14.4	49.4±16.0	44.6±19.6	44.0±16.2	/
Stage	60.6±11.8	48.6±18.4	39.8±24.2	31.5±21.4	35.7±18.6	32.8±20.0
Grade	67.2±17.4	60.4±18.2	59.9±14.0	62.5±15.0	59.9 ± 18	57.3±18.8

Table 3.6: Table of results (average accuracies) obtained for different feature set combinations. Average accuracies include a final feature set of 3, 5 or 7 features.

CRC	Features type	Features selected	Classifier	Hyper-parameters
Mut. BRAF /KRAS	Trad	• original_glcm_InverseVariance 40 • original_glszm_SizeZoneNonUniformityNormalized 40 • original_glszm_SizeZoneNonUniformityNormalized z h/hsv • original_glszm_ZoneEntropy vnc	Rand. Tree	• criterion: entropy • max depth: 20 • n estimators: 50
	MM	/		
	DL	• Resolution yz Applied Combinaison of types n°0 model n°2 deep feature n°3727		
Permeation	Trad	• original_glszm_GrayLevelNonUniformityNormalized z h/hsv • original_shape_Maximum2DDiameterColumn 40 • original_firstorder_Minimum 120 • original_glcm_ClusterShade z h/hsv • original_gldm_SmallDependenceEmphasis iode • original_glszm_ZoneEntropy z h/hsv	Naive B.	/
	MM	/		
	DL	/		
MSI	Trad	• original_firstorder_InterquartileRange iode • original_glcm_ClusterShade z h/hsv • original_gldm_DependenceVariance z s/hsv • original_glszm_SmallAreaEmphasis z h/hsv • original_glszm_SmallAreaHighGrayLevelEmphasis iode • original_glszm_SmallAreaLowGrayLevelEmphasis iode	Naive B.	/
	MM	/		
	DL	/		
Peri-Nerv. Sheating	Trad	• original_firstorder_Skewness iode • original_glcm_ClusterShade z h/hsv	Naive B.	/
	MM	/		
	DL	/		
Budding	Trad	• original_glszm_SizeZoneNonUniformityNormalized vnc • original_firstorder_Skewness iode • original_glszm_SizeZoneNonUniformityNormalized 120	KNN	• metric: manhattan • n neighbors: 3 • weights: uniform
	MM	• original_glrlm_ShortRunLowGrayLevelEmphasis z s/hsv dilation size kernel:2.0 PI		
	DL	/		
Stage	Trad	• original_glcm_InverseVariance 40 • original_glszm_GrayLevelNonUniformityNormalized z h/hsv • original_shape_Elongation 40	Naive B.	/
	MM	/		
	DL	/		
Grade	Trad	• original_glcm_Correlation 120	Rand. Tree	• criterion: gini • max_depth: 10 • n estimators: 20
	MM	• original_glcm_Correlation 120		
	DL	• Resolution yz Applied Combinaison of types n°0 model n°0 deep feature n°2222		

Table 3.7: Summary of all previous steps: the features chosen for each prediction model with the most suitable hyperparameters (if any).

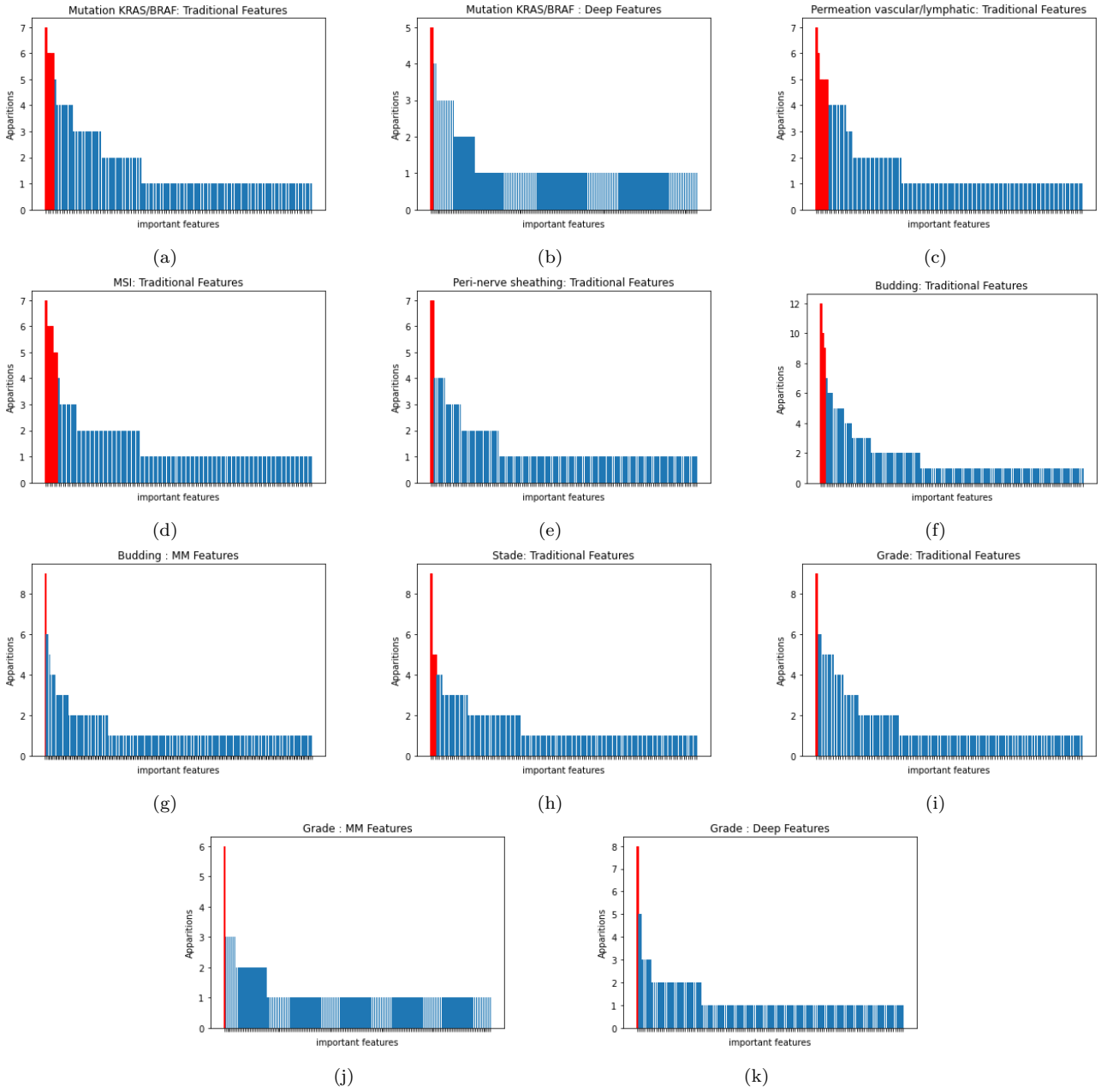
3.1.3 Final models

Once the type of pre-processing to be applied, the ideal model to be used as well as the optimal combination, a choice in the features to be finally used to train the model. To do this, the process (split training/test, dimensionality reduction, prediction) is iterated 20 times. During these 20 iterations, the 10 features of each extraction mode (traditional, deep, MM) are listed. **The features that appeared most often during these iterations are selected**.

Remember that this feature selection is performed on the full dataset and that selecting features in a non-nested manner could give significantly biased results [64].

Table 3.7 summarizes all the previous steps. It shows the selected features, the selected classifier and its hyperparameters.

The table 3.8 informs about the presence of certain types of features as a percentage. Two main observations can be made. The first is that if we look a little closer at the features extracted from the effective atomic numbers in the HSV form, we see that the importance extracted from the features is mainly in the **hue** part. This information is very interesting because for the permeation or the stage, this change of format could have the effect of a **compression** on the effective atomic numbers. This would imply working with less data but with almost as much



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k)

Figure 3.4: Features selection, Final selection of features for each prediction. Analysis of the features that appeared most often in the significant feature set. In red, features that appeared enough times to be selected in the final feature set for KRAS/BRAF mutations (a) traditional and b) deep) ; vascular/lymphatic permeation (c) traditional) ; MSI status (d) traditional) ; peri-nerve sheathing (e) traditional) ; budding (f) traditional and g) MM) ; stage (h) traditional) ; grade (i) traditional, j) MM, and k) deep).

information.

The second observation is at the level of deep features. We can notice that most of the important deep features have been extracted from the XZ and YZ planes. This fact is somewhat surprising considering that the XY plane is supposed to gather more information and the XZ and YZ planes, due to the thickness of the cut, give distorted images.

Other observations have also been made. As far as the traditional features are concerned, the features based on the shape and GLRLM seem to be not very important. As for the others, their importance varies according to the target. On the MM features side, it seems that the features based on morphological covariance are not efficient (they have no importance). As for the combinations of spectral data applied during the extraction of deep features, none of these combinations really seem to stand out.

		Bud.	Stage	MSI	Mut.	Perm.	E.P.	Grade
Trad	shape	6	18	4	17	18	12	16
	first order	21	16.5	18.5	8	15	27	9
	gldm	13	13.5	12	10	7	9	13
	glszm	29	32	31.5	32	32	21	27
	glcm	24	16	23	19	17	25	27
	glrlm	6	4	11	16	13	9	7
HSV	h/hsv	65	71	45	43	71	53	56
	s/hsv	20	17	39	35	14	23	25
	v/hsv	14	12	17	22	14	24	19
Filtering	erosion	14	/	/	28	/	/	11
	dilation	45	/	/	16	/	/	32
	opening	16	/	/	30	/	/	27
	closing	25	/	/	26	/	/	29
Plane DL	XY	4	/	/	6	/	/	6
	XZ	71	/	/	22	/	/	43
	YZ	26	/	/	72	/	/	52
Combinaison DL	n°1	25	/	/	39.5	/	/	40
	n°2	30	/	/	23.5	/	/	32
	n°3	45	/	/	37	/	/	28

Table 3.8: distribution of features in percent: The traditional features, The HSV channels of the effective atomic number map, The different types of filtering applied to the image before feature extraction, The selected planes from which the deep features are extracted, and the combination of spectral data (n°1: 120 keV, 40 keV, vnc; n°2: 120 keV, 40 keV, iodine; n°3: Z_{eff} h/hsv, Z_{eff} s/hsv, Z_{eff} v/hsv)

3.2 Models Evaluation

3.2.1 Model metrics plot

In Table 3.10 the metrics used to evaluate the performance of a model. Here, the evaluation calculates the macro-average of the metrics. The $F1score$ with $average='macro'$ for example, calculates the $F1score$ for each class and returns the average of the scores obtained.

As a reminder, **accuracy** is calculated as the fraction of correct predictions out of the total number of predictions. The problem is that it is enough for the classes not to be in balance (more examples in class A than in class B) for these results to be insufficient. In our case, most of the classes are highly unbalanced (table 3.9).

If the accuracy is optimal, this implies that the model has no false positives. In the case of permeation, we observe that the results are close to 0.5. This means that these models make correct predictions every other time. $F1score$ and *recall* are also close to the 0.5 for this prediction model. This means that half of the real positives are classified correctly and that the weighted average of *precision* and *recall* is also equal to 0.5. The permeation does not have an imbalance in its classes being 2. The selected features simply do not allow to distinguish the cases of permeation.

Stage				Grade			Permeation		E.P.			Budding			MSI		Mutations		
1	2	3	4	Low	High	/	N	Y	N	Y	/	N	Y	/	N	Y	N	Y	/
1	17	5	4	17	8	2	14	13	17	8	2	9	11	7	21	6	4	11	12

Table 3.9: Table grouping the different output sets and data quantities for each class

Target	accuracy	F1	precision	recall
Mutation BRAF/KRAS	0.82	0.77	0.74	0.82
Permeation	0.52	0.45	0.44	0.525
MSI	0.77	0.66	0.66	0.68
Peri-Nerv. Sheating	0.81	0.72	0.71	0.77
Budding	0.8	0.76	0.73	0.8
Stage	0.53	0.32	0.28	0.40
Grade	0.9	0.85	0.83	0.88

Table 3.10: Metrics obtained after the partially nested cross validation for each target

3.2.2 Confusion matrix

The classifiers overestimate the class *Second Stage* in the prediction. This is probably related to the fact that stage 2 examples are present in large majority compared to stage 3 and 4.

3.2.3 Learning curves

By observing the training curves, we can highlight 3 types of curves: those that are increasing, those that are decreasing and those that are stagnant. The increasing ones (stage in figure 3.6.f, Mutation KRAS/BRAF in figure 3.6.a, Budding in figure 3.6.e, Permeation in figure 3.6.b) suggest that an increase in the number of available examples would continue to improve the model performance. A flattening curve (Grade in figure 3.6.g) means that the model is in an overfitting zone at the end of training. A shapeless downward curve (MSI status in figure 3.6.c and perinervous sheathing in figure 3.6.d seeming to stagnate at 0.5) often means that the data set is too complex for the model. Underfitting can be observed when the algorithm is not able to model the training data or the new data, it consistently obtains high error values not allowing the learning curve to increase. The choice of features is not the right one or its number is not consistent enough.

3.2.4 ROC plot

Although there is no precise area under the ROC curve that describes good discrimination, a rule of thumb states that $AUC = 0.5$ suggests no discrimination. $AUC = 0.5 - 0.7$ suggests poor discrimination, not much better than flipping a coin. $AUC = 0.7 - 0.8$ is acceptable discrimination, $AUC = 0.8 - 0.9$ is excellent discrimination. And finally, $AUC > 0.9$ is exceptional discrimination [67].

The prediction models related to KRAS/BRAF mutation and Permeation and potentially stage can be considered as totally ineffective. The models for MSI status and Permeation are poor whereas the models detecting budding and defining grade are found to offer exceptional discrimination.

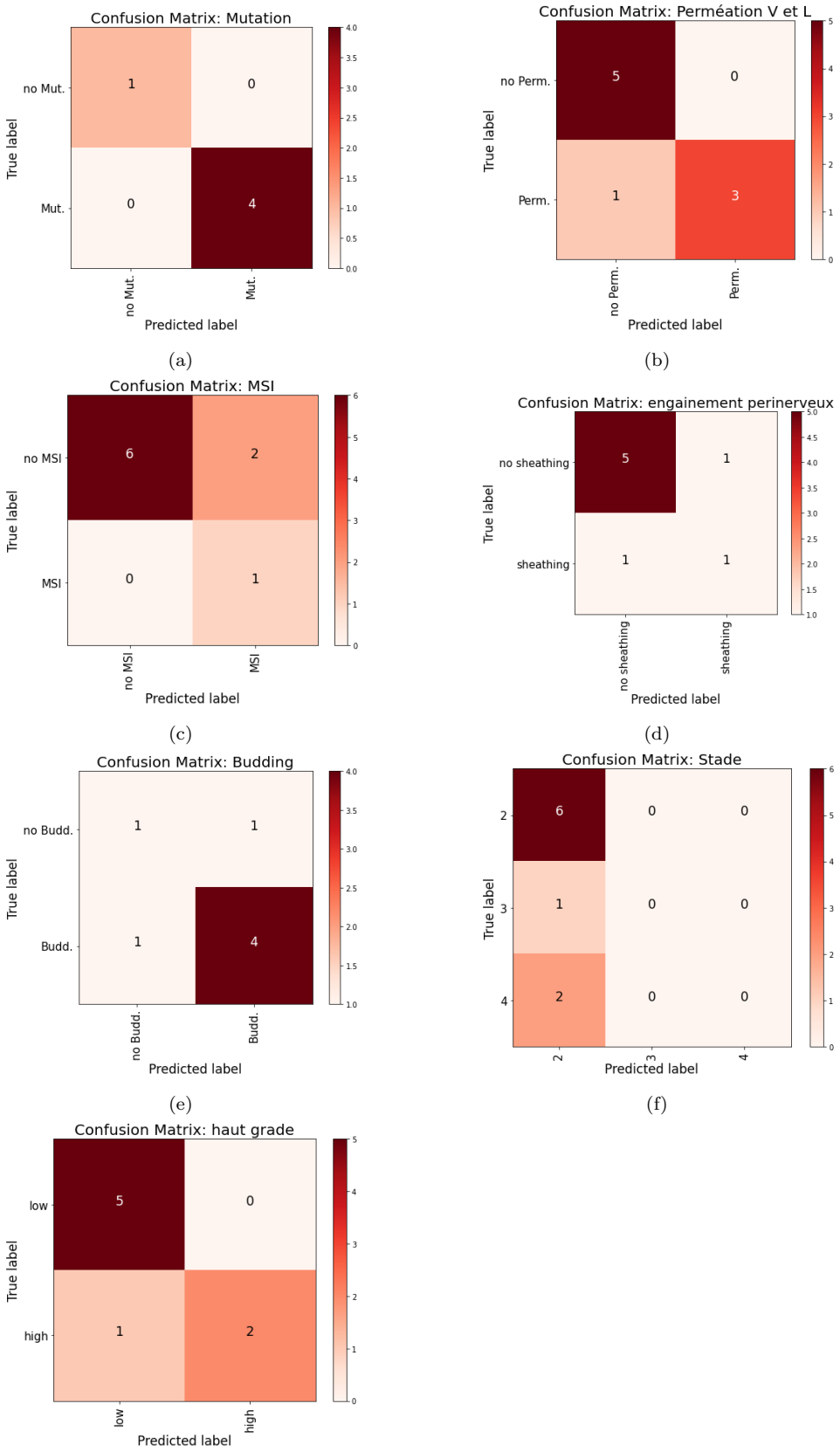
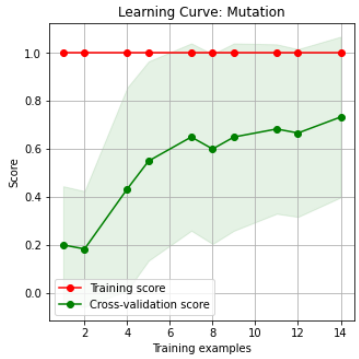
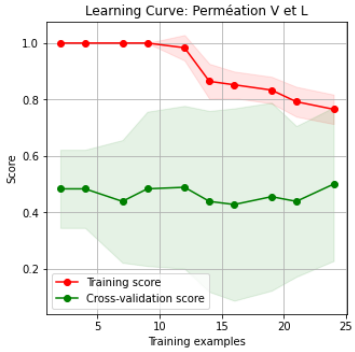


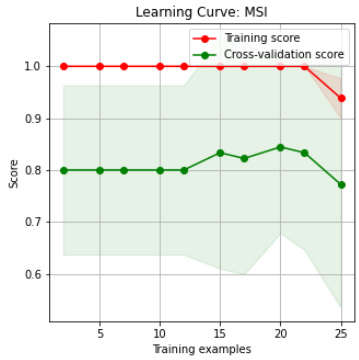
Figure 3.5: Confusion Matrix, on the basis of the final set of features, generated with a repeated Stratified K Fold as a cross-validation generator for a) KRAS/BRAF mutations b) Vascular/lymphatic permeation c) MSI status d) Peri-nerve sheathing e) Budding f) Stage g) Grade



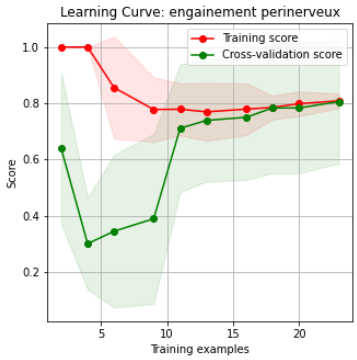
(a)



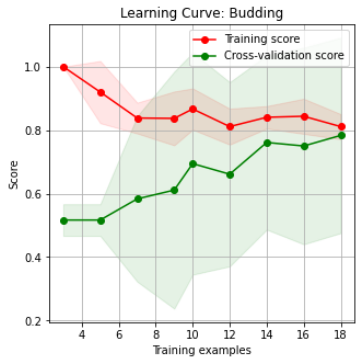
(b)



(c)

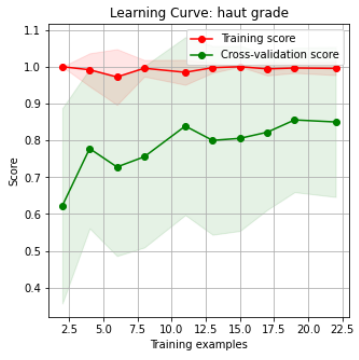


(d)



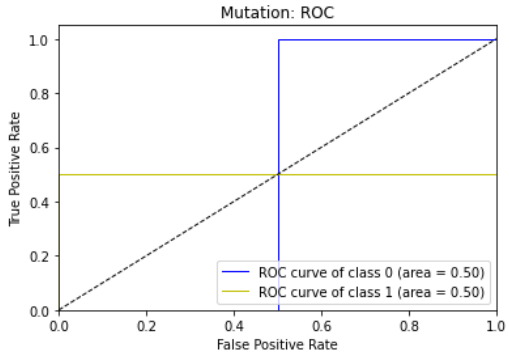
(e)

(f)

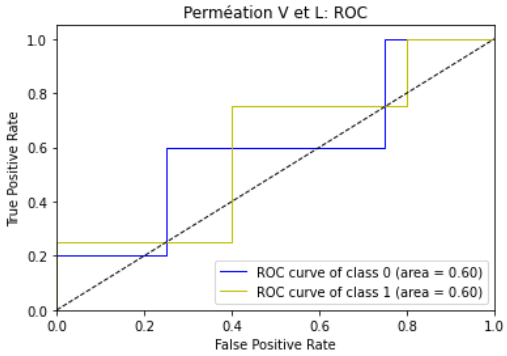


(g)

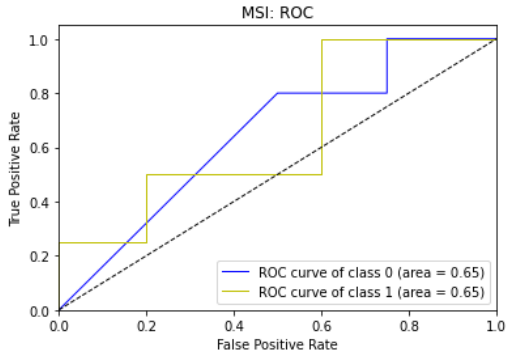
Figure 3.6: Learning curves, on the basis of the final set of features, generated with a repeated Stratified K Fold as a cross-validation generator for a) KRAS/BRAF mutations b) Vascular/lymphatic permeation c) MSI status d) Peri-nerve sheathing e) Budding f) Stage g) Grade



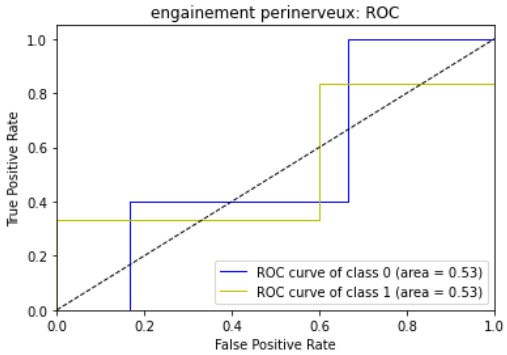
(a)



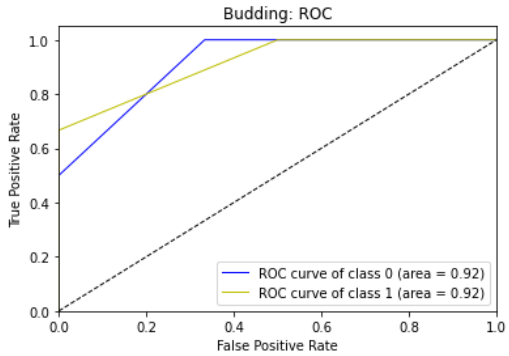
(b)



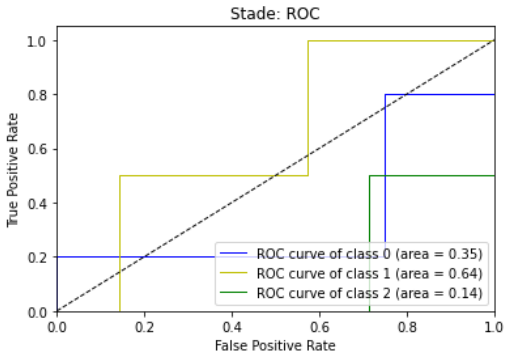
(c)



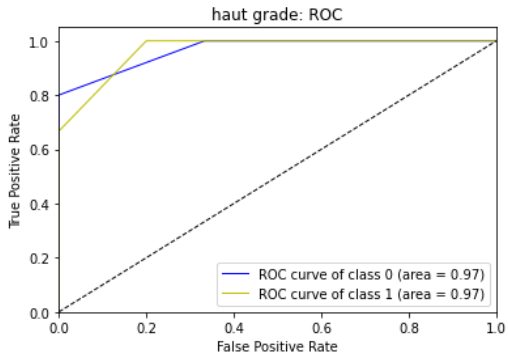
(d)



(e)



(f)



(g)

Figure 3.7: ROCs, obtained on the basis of the final set of features, generated with a repeated Stratified K Fold as a cross-validation generator for a) KRAS/BRAF mutations b) Vascular/lymphatic permeation c) MSI status d) Peri-nerve sheathing e) Budding f) Stage g) Grade

Chapter 4

Discussion and Conclusion

4.1 Interpretations

We recall us, before starting this discussion, that the validation method used is a partially nested cross validation. This means that the first part of the process is applied in a non-nested manner. The features, the pre-processing and the classifier are chosen in this first part. The selection factor (accuracy) could have been replaced by sensitivity, specificity or F1 score, since most of the classes are not balanced. Moreover, cross-validation is not used in these steps. Cross-validation is a common solution when the available data sets are limited [64]. The method used here is to repeat the modeling process a number of times and randomly split our dataset into training and test sets each time. This step imposes the presence of minority classes. It was applied to reinforce the presence of underrepresented classes in the training set. Since the method was repeated several times with multiple random train/test splits, we believe we can rely on the results to make our choice on pre-processing, model selection, and feature selection. Working with repeated stratified folds is the most appropriate solution. However, the running time is even more important with repeated stratified folds and in view of the similarity of the methods, we felt it was worthwhile not to use it.

SMOTE (Synthetic Minority Over-Sampling Technique) was not used here because it does not attenuate the bias towards the classification in the majority class for most classifiers when data are high-dimensional [68]. Moreover, the SMOTE must be applied on the training set and not the whole data set to be efficient [69]. This is not feasible with the current limited data. Indeed, SMOTE is based on the creation of surface data on the basis of existing data. The technique is based on the KNN algorithm. It is recommended to work with at least 5 neighbors [70]. This is not feasible with so little dataset. For example, BRAF/KRAS mutations have a class of 4 samples.

At the end of this first step, we can have an idea of the importance of the features in the final feature set.

Regarding the traditional features, the shape-based and GLRLM features do not seem to be very important. We would have thought that features based on shape would be more important in view of how certain classes are distinguished by their size, sphericity, etc. As for the others, their importance varies depending on the target. After RGB -> HSV conversion, we observe that the majority of the important features are related to the hue and not to the saturation and the value. This is perfectly logical considering the fact that hue contains the information extracted from the effective atomic numbers in a more concise way. It contains directly the effective atomic numbers but at a different scale: the voxel values do not vary between 5 and 11 but between 0 and 255.

On the MM features side, it seems that morphological covariance based features are not effective (they have no importance). MM features seem to be interesting for targets with classes that differ in tumor area. Indeed, budding is defined by the presence of single cells or clusters of up to 5 tumor cells at the tumor invasion front that are distinct from the rest of the tumor. In high grade, cancer cells are distinct from healthy cells and therefore grow more rapidly. As for the retained MM features (for grade determination and budding detection), they are all related to dilation in a PI series. This seems surprising given how the classes differ from each other. We would therefore have expected a Delta series with erosion focusing exclusively on the extremities of the tumor and its surroundings.

The extraction of deep features seems unreliable in the case of the second dataset, the number of images is far too small. We also think, in the case of dataset 2, that the *cropping augmentation* could have been integrated to give better results. Moreover, these features are extracted on the basis of a 2D image and not on a 3D shape. This extraction is however done on 3 planes but the fact that the voxels do not have a cubic shape implies that some planes are bound to give less useful and less significant deep features. As for the combinations of spectral data applied during deep feature extraction, none of these combinations really seem to stand out. The deep features are used in the detection of BRAF/KRAS mutation and in the determination of the grade. In both cases, these features are extracted from the combination of 40keV-120keV-vnc image types. This suggests that some combinations of image types are more determinant than others and therefore really useful for prediction.

4.2 Implications

Most classification models for CRC are generally trained to detect MSI status [4, 32]. Other studies have focused on BRAF/KRAF mutations [71], grade [72]. However, very few models (to our knowledge) are trained to define grade and detect budding. Our master thesis could therefore serve as a basis for further research on grade and budding. Moreover, our new pre-processing technique seems to give better overall results. This could lead researchers to improve their pre-processing. The RGB -> HSV conversion seems to concentrate information from the effective atomic number map into almost 3 times less data in some cases. Again, this could help future works to work on a less heavy but equally efficient database.

4.3 Limitations and Recommendations

We will now expose different limitations that we have faced and then propose a solution for future works. Moreover, we propose new ways of thinking.

4.3.1 Segmentation

One possibility, in order to speed up the time needed to carry out a manual segmentation, would be to support the radiologist with an artificial intelligence trained to carry out segmentations. The radiologist would then only have a role as examiner and corrector. The time needed to validate a segmentation would ideally be optimised and more examples would be available in the dataset. A more global analysis would then be possible. Indeed, once an AI trained in segmentation is reliable, why settle for the tumor as such when a segmentation of each organ and thus a much more complete radiomic analysis is possible?

4.3.2 Pre-processing

It would be recommended to do a more thorough study on the correction of segmentations as well as on the real usefulness of changing the computer color coding system (RGB to HSV).

4.3.3 Feature extraction

In addition to traditional, traditionnal after filtering (MM) and deep features, feature fusion could include clinical features that have proven to give more robust models. These features include location, age and gender of the patient. Successful results may encourage the medical field to provide a more comprehensive set of clinical features (smokers or non-smokers, diet, ..) that may detect unexpected links between the patient's lifestyle and tumour.

Previous researches having for goal to predict the microsatellite instability (MSI) in colorectal cancer at initial computed tomography evaluation [32], did the combination model of radiomic features and clinical features and had the best discriminatory ability in both the training cohort (AUC 0.80) and test cohort (AUC 0.79), higher than that of the model with radiomic features alone. Other researchers have discovered differences in the molecular mechanisms captured, highlighting, among other things, distinctions in the progression between left and right colon tumors [73]. In our case, we have information about: sex, age and localisation of the tumor as we can see on Table 2.1 (and the full table in the Appendix C). In order to make the location feature usable in our models, the location could be defined in binary form by separate the left and right side cancers. Right colon cancers are located in the cecum, ascending colon, right colonic angle (between ascending and transverse colon) and/or transverse colon, while left

colon cancers are located in the left colonic angle (between transverse and descending colon), descending colon and/or sigmoid colon.

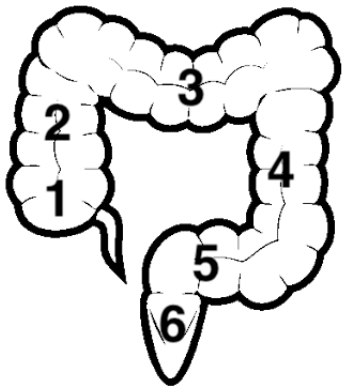


Figure 4.1: 1) Caecum; 2) Ascending colon; 3) Transverse colon; 4) Descending colon; 5) Sigmoide colon; 6) Rectum

4.3.4 Feature selection

In the scientific article outlining all the popular methods for carrying out a radiomic process, it is mentioned that in order to obtain a good selection of features, feature reproducibility analysis should be used. This method consists of seeing which features are most likely to vary with variation in segmentation. In order to check the reproducibility of the features as well as possible, there should have been at least 2 segmentations of the same tumour (made by 2 different people) [17]. This was not our case because only radiologist Etienne Danse performed the manual segmentations.

Appendices

Appendix A

Colorectal cancer

Here are some characteristics of colorectal cancer:

Cancer grade: Type of tumour classification based on the appearance of the cancer cells. Knowing the grade of the cancer helps doctors predict how fast the cancer will grow and how likely it is to spread. The grade is usually given a number from 1 to 3 or 4. The higher the number, the more different the cancer cells look from healthy cells and the faster they grow. [74]

Micro-satellite instabilities (MSI): Tumours with genetic instability account for 15 % of CRCs (colorectal cancer). To better understand what this anomaly is, let's define the microsatellite. A microsatellite is a DNA sequence formed by a continuous repetition of particular patterns. Specifically, MSIs carry mutations in the genes of the MMR (MisMatch Repair) system involved in the repair of DNA replication errors (DNA mismatch repair). When the MMR system is altered, the erroneous microsatellite sequences accumulate, leading to MSI and the early development of CRC. Achieving MSI status is necessary because MSI CRC tissues have special biological behaviours, they are more likely to have a better prognosis and benefit from immunotherapy[1]. These tumours are mainly found in the right colon. Furthermore, this anomaly is more common in female patients. [75].

KRAS mutations: In normal cells, KRAS serves as a hub for signals that lead to cell growth. Indeed, the KRAS gene encodes a small protein that acts as an inactivation signal for surface receptors such as the Epidermal Growth Factor Receptor (EGFR). Epidermal Growth Factor Receptors are found on the surface of tumour cells (but also in normal cells) and their role is to send a growth signal to the cell nucleus. With a KRAS mutation, the signal is too strong and cells grow without being told to do so: this leads to cancer. The mutation is found in 30-50% of colorectal and other tumours.

BRAF mutations: This mutation produces a mutated protein that then acts in an uncontrolled manner within the cancer cell, promoting the development of cancer. The BRAF mutation is relatively common in cancers, affecting 7 % of all solid tumour types. In the case of

[1]Immunotherapy: Treatment to increase or induce the body's immunity by injecting antibodies or antigens

colorectal cancer, the search for the BRAF mutation is carried out for prognostic reasons.

Stage of the cancer: The stage of the cancer allows doctors to know how much cancer there is in the body, where it is and how far it has spread. This information helps them decide what treatments to use. Cancer can spread within the organ in which it originated, to nearby lymph nodes or to distant sites. Various tests can be carried out to determine the stage of the cancer. The stage is given a number from 1 to 4. A stage 1 cancer is usually small and has not spread to other sites than where it originated. The higher the stage number, the larger the size of the tumour or the extent of its spread. A stage 4 cancer has usually spread to distant sites [74]. The stage is determined according to different criteria:

- size of the tumour (Tis, T1, T2, T3, T4a)
- possible involvement of lymph nodes (N0, N1a, N1b, N1c, N2a, N2b)
- the existence of one or more metastases. Metastasis is cell growth that occurs at a distance from the primary site of growth and without direct contact with it.

Vascular/lymphatic permeation: Let us first define permeation which is the penetration of a permeate (liquid, gas or vapour) through a solid. It is important to know whether the tumour is metastatic. If it is, the permeation may be vascular or lymphatic. In other words, the tumour can spread because of permeation through the lymphatic or vascular system.

Peri-nervous sheathing: Infiltration of tumour cells around and within the nerve. There is a correlation between the presence of peri-nervous sheathing and high stage, the tumour size (>4cm) and the presence of lymphatic emboli (= presence of tumour cells within lymphatic structures) and therefore gives a poor prognosis.

Budding: Presence, at the invasion front[1] of the tumour, of single cells or clusters of up to 5 tumour cells that stand out from the rest of the tumour.

[1]Invasion front: The invasion is the direct extension and penetration by cancer cells into neighboring tissues

Appendix B

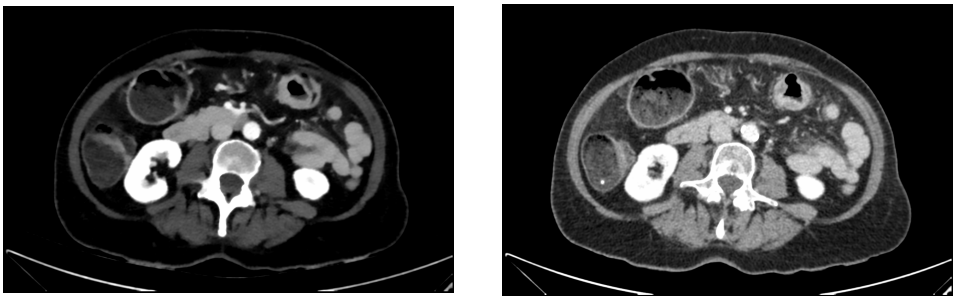
Image types after DECT acquisition

B.1 Monochromatic image

X-rays are produced by the interaction of electrons (emitted by a cathode and more precisely a filament, usually made of tungsten, heated by the passage of an electric current) with a metal target. These electrons are accelerated by a potential difference in the filament and directed towards the metal target (anode or anticathode). The production of X-ray photons is due to the rapid deceleration of the electrons as they impact on the target.

Depending on the energy of the radiation and therefore, on the potential difference, the X-rays emitted will not be the same. They are differentiated by the wavelength of the particles making up the X-ray. We use a energy spectrum to visualize the intensity of particles with a wavelength λ_m, λ_n, etc.. [76]

Monochromatic image is an image with a **single energy peak** (for example 40 keV or 120 keV).



(a) (b)

Figure B.1: Types of images acquired with the DECT. a) monochromatic image at low Kev (40keV); monochromatic image at high Kev (40keV)

B.2 VNC

Virtual non-contrast imaging is an image post-processing technique used to create 'non-contrast' images of contrast-enhanced scans via the subtraction of iodine. It is an imaging technique unique to dual energy CT (see figure B.2).

VNC imaging may replace a pre-contrast scan and substantially reduce radiation exposure [22].

Figure B.2: Type of images acquired with the DECT. Image with virtual non contrast (VnC)

B.3 Z_{eff}

In the context of this thesis, the effective atomic number (Z_{eff}) is the average atomic number for a compound or mixture of materials. In our case, the mixture of materials turns out to encompass all the materials in a region varying in area depending on the image resolution. Effective atomic number can be calculated from dual-energy CT data with small errors of 1.7 %.

See on figure B.3 a slice of the map of the effective atomic number Z_{eff}.

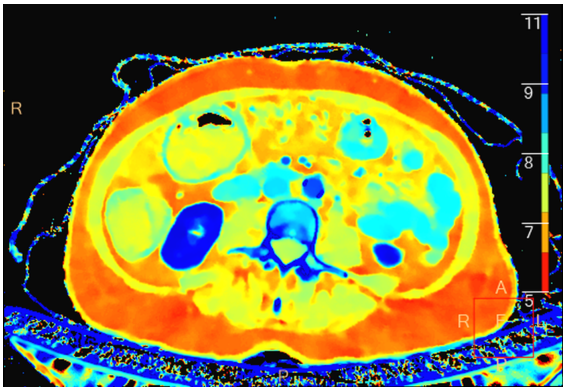


Figure B.3: Type of images acquired with the DECT. A map of the atomic number Z_{eff} studying the distribution of a material within a tissue

B.4 Map iode

Iodine images play a critical role in DECT's ability to improve lesion conspicuity. Iodine images detect and quantify iodine within each image voxel, allowing for detection of even a small amount of enhancement within a lesion (see figure B.4).

The iodine map appears to have a significant impact on MSI prediction.

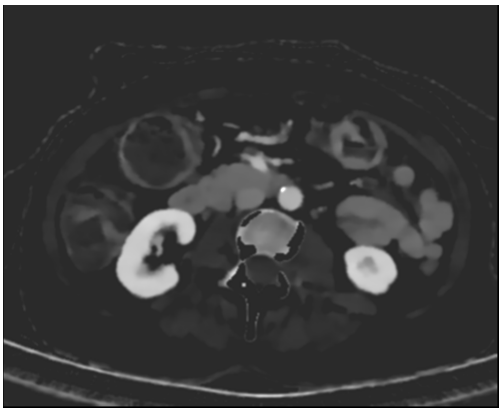


Figure B.4: Type of images acquired with the DECT. An iodine concentration chart to measure the number of mg of iodine per ml at within the tissue

Appendix C

List of patients

Case	Seg.	Usable	Pix. size [mm]	thckns [mm]	Sex	Age	Localisation	Stage	H.G.	Perm.	E.P.	Bud.	MSI	Mut.
1	Yes	Yes	0.78125	3	W	84	Angle colique G	3	1	1	0	1	0	/
2	Yes	Yes	0.80859375	3	M	84	Recto-sigmoide	4	0	1	1	1	0	/
3	Yes	No	0.78125	3	W	69	Colon D	3	1	1	1	0	1	1
4	Yes	Yes	0.828125	3	M	62	Sigmoïde	4	0	1	1	1	0	0
4vrais	Yes	Yes	0.578125	3	W	70	Caecum	4	1	1	1	/	0	/
5	No	Yes	0.78125	3	M	63	Colo-sigmoide	3	0	1	1	1	0	/
6	Yes	Yes	0.828125	3	M	60	Sigmoide	3	0	1	1	/	0	/
7	Yes	Yes	0.697265625	3	W	65	Angle colique G	2	0	0	0	/	0	/
8	No	Yes	0.78125	3	W	63	Sigmoïde	3	0	1	0	1	0	/
9	Yes	Yes	0.595703125	3	M	66	Sigmoïde	2	0	0	0	1	0	0
10	Yes	Yes	0.736328125	3	M	76	Colon G	2	0	0	1	/	0	/
11	Yes	Yes	0.82421875	3	M	73	Colon D	2	0	0	0	1	0	/
12	Yes	Yes	0.697265625	3	M	91	Sigmoïde	3	0	1	/	0	0	1
13	No	Yes	0.580078125		W	53	Colon D	2	0	1	0	1	0	/
14	No	Yes	0.896484375		M	71	Sigmoïde	4	0	1	1	/	0	0
15	No	Yes	0.650390625		W	82	Sigmoïde	2	0	1	1	1	0	/

16	No	Yes	0.66796875	W	75	Colon D	2	0	1	1	1	0	/
17	No	Yes	0.677734375	M	87	Recto-sigmoide	4	0	0	1	1	0	/
18	No	Yes	0.716796875	W	79	Colon D	2	0	0	0	1	0	/
19	No	Yes	0.6015625	W	58	Sigmoide	4	0	1	/	1	0	1
20	No	Yes	0.78125	M	62	Sigmoide	3	1	1	1	/	0	/
21	No	Yes	0.625	W	46	Sigmoïde	3	1	0	0	1	0	/
22	No	No		W	82	Sigmoïde	4	0	1	1	1	0	1
23	No	No		W	34	Transverse	4	1	1	1	1	0	1
24	No	No		W	91	Recto-sigmoïde	1	0	0	0	/	0	/
25	No	No		W	69	Colon D	2	0	1	0	/	0	1
26	No	No		M	85	Caecum	4	0	1	1	/	0	1
27	No	No		W	78	Caecum	3	1	1	1	1	0	0
28	No	No		W	82	Sigmoïde	3	0	1	/	1	0	1
29	No	No		M	80	Sigmoïde	2	0	0	0	0	/	1
30	No	No		W	32	Transverse	4	1	1	1	/	0	1
31	No	No		W	67	Sigmoïde	4	0	1	1	/	0	1
32	No	No		W	62	Colon D	2	0	1	1	1	0	/
33	No	No		W	73	Colon D	4	1	1	1	/	0	1
34	No	No		M	76	Caecum	2	0	1	1	1	0	/
35	No	No		W	85	Colon transverse	2	0	1	1	/	0	/
36	No	No		W	94	Colon transverse	3	0	1	/	/	0	0
37	No	No		W	69	Colon G	4	0	1	1	1	0	1
38	No	No		M	74	Sigmoide	4	0	1	1	1	0	1
39	No	No		M	81	Colon G	3	0	1	1	/	0	1
40	No	No		W	60	Caecum	2	0	0	0	0	0	/
41	No	No		W	72	Colon transverse	2	1	0	1	/	1	1
42	No	No		M	63	Recto-sigmoïde	2	0	0	0	1	0	/
43	No	No		W	59	Colon D	2	1	1	1	1	0	/
44	No	No		W	73	Ceacum	3	0	1	0	/	0	/
45	No	No		W	85	Colon D	3	1	0	1	1	1	/
46	No	No		W	92	Jonction iléo-caecale	2	0	1	/	0	0	/

47	No	No			W	73	Colon D	2	1	1	1	1	0	1
48	No	No			M	57	Sigmoïde	4	0	1	1	/	0	0
49	No	No			W	77	Colon transverse	4	0	0	1	/	0	1
50	Yes	No			M	89	Colon G	2	0	0	0	1	0	/
51	Yes	Yes	0.630859375	3	W	77	Caecum	2	1	0	1	1	0	/
52	Yes	Yes	0.736328125	3	M	77	Sigmoïde	4	1	1	0	0	0	0
53	Yes	Yes	0.703125	3	W	76	Caecum	2	1	1	0	/	1	1
54	Yes	Yes	0.662109375	3	W	56	Caecum	1	1	0	0	/	0	/
55	Yes	Yes	0.716796875	3	M	58	Angle splénique	2	0	1	0	1	1	0
56	Yes	Yes	0.697265625	3	M	85	Sigmoïde	3	/	0	0	0	0	1
57	Yes	Yes	0.697265625	3	W	68	Colon D	2	/	0	0	0	1	1
58	Yes	Yes	0.673828125	3	W	72	Caecum	2	0	0	0	0	0	/
59	Yes	Yes	0.78125	3	W	85	Caecum	2	0	1	0	0	1	1
60	Yes	Yes	0.640625	3	W	43	Sigmoïde	4	/	/	0	0	0	1
61	Yes	Yes	0.650390625	3	M	81	Sigmoïde	2	1	1	1	1	0	/
62	Yes	Yes	0.69921875	3	M	66	Colon D	2	0	0	1	1	0	1
63	Yes	Yes	0.654296875	3	W	46	Colon transverse	2	0	1	1	1	0	1
64	Yes	Yes	0.8515625	3	W	71	Caecum	2	1	1	/	1	1	1
65	Yes	Yes	0.904296875	2	W	60	Colon D	4	0	0	0	0	0	1
66	Yes	Yes	0.619140625	3	W	82	Caecum	2	0	0	0	/	0	1
67	Yes	Yes	0.5625	3	W	90	Colon D	2	0	0	0	0	0	/
68	No	Yes	0.708984375	3	M	78	Jonction iléo-caecale	3	1	1	1	1	0	0
69	Yes	Yes	0.69140625	3	W	65	Colon transverse	3	0	0	0	0	0	/
70	Yes	Yes	0.5703125	3	W	75	Caecum	2	0	0	0	/	1	1
71	Yes	Yes	0.7109375	3	M	88	Sigmoïde	2	0	0	0	0	0	1

Appendix D

Features from Traditional Radiomics

D.1 Shape

Table D.1: Shape features

Features	ISO Formulas
Mesh volume	$\mu = \sum_{i=0}^{G-1} i.p(i)$
Voxel volume	$\sigma^2 = \sum_{i=0}^{G-1} (i-\mu)^2.p(i)$

On Table D.1, $p(i)$ corresponds to the probability of occurrence of a voxel with a luminosity value equal to i. Where $G-1$ is the maximum possible value for a voxel.

D.2 First Order

Table D.2: First Order features

Features	ISO Formulas
Mean Intensity	$\mu = \sum_{i=0}^{G-1} i.p(i)$
Variance	$\sigma^2 = \sum_{i=0}^{G-1} (i-\mu)^2.p(i)$
Skewness	$\mu_3 = \frac{1}{\sigma^3} \sum_{i=0}^{G-1} (i-\mu)^3.p(i)$
Kurtosis	$\mu_4 = \frac{1}{\sigma^4} \sum_{i=0}^{G-1} (i-\mu)^4.p(i)$
Energy	$E = \sum_{i=0}^{G-1} [p(i)]^2$
Entropy	$H = \sum_{i=0}^{G-1} p(i) log_2[p(i)]$
COV	$cov = \frac{\sigma}{\mu}$

On Table D.2, $p(i)$ corresponds to the probability of occurrence of a voxel with a luminosity value equal to i. Where $G-1$ is the maximum possible value for a voxel.

D.3 Higher Order features

From each of the image-based extracted matrices (**GLCM, GLRLM, GLSZM, GLDM**), a number of features can be extracted as listed below.

Remark: The examples (tables D.3, D.5, D.7 and D.9) are illustrated in several steps. The first part is an illustrative "image" with pixels having a brightness with a value ranging from 0 to 4. The second part is the matrix generated using the various texture-related techniques mentioned above.

Gray-Level Co-Occurrence Matrix - GLCM

1	2	3	4
1	3	4	4
3	2	2	2
2	1	4	1

Gray Level (i)	Cooccurrences (j)			
	1	2	3	4
1	0	1	1	3
2	1	4	2	0
3	1	2	0	2
4	3	0	2	2

Table D.3: (a) picture, (b) Example of a CM for a 4x4 image and for 4 gray levels with for a displacement vector (0,1):a distance δ=1 (considering pixels with a distance of 1 pixel from each other) and angle θ=0°, with matrix named C(i,j) [42]

Table D.4: First Order

Features	ISO Formulas
Variance	$\sum_{i,j}^{G} C(i,j)[(i-\mu_x)^2 + (j-\mu_y)^2]$
Energy	$\sum_{i,j}^{G} C(i,j)^2$
Entropy	$-\sum_{i,j}^{G} C(i,j) log(C(i,j))$
Correlation	$\sum_{i,j}^{G} C(i,j) \times \frac{(i-\mu_x)(j-\mu_y)}{\mu_x \mu_y}$
Dissimilarity	$\sum_{i,j}^{G} C(i,j)\|i-j\|$
Contrast	$\sum_{i,j}^{G} C(i,j)(i-j)^2$
Homogeneity	$\sum_{i,j}^{G} \frac{C(i,j)}{1+\|i-j\|}$
IDM	$\sum_{i,j}^{G} \frac{C(i,j)}{1+(i-j)^2}$
Cluster Shade	$\sum_{i,j}^{G} C(i,j) \times (i+j-\mu_x-\mu_y)^3$
Cluster Tendency	$\sum_{i,j}^{G} C(i,j) \times (i+j-\mu_x-\mu_y)^4$

On Table D.4, $p(i)$ corresponds to the probability of occurrence of a voxel with a luminosity value equal to i. Where $G-1$ is the maximum possible value for a voxel [25].

Grey-Level Run Length Matrix - GLRLM

1	2	3	4
1	3	4	4
3	2	2	2
2	1	4	1

Gray Level (i)	Run Length (j)			
	1	2	3	4
1	4	0	0	0
2	1	0	1	0
3	3	0	0	0
4	3	1	0	0

Table D.5: (a) picture, (b) Example of a RLM for a 4x4 image in 0°direction and for 4 gray levels with matrix named R(i,j) [42]

Table D.6: GLRLM

Features	ISO Formulas
SRE for Small Run Emphasis	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{R(i,j)}{j^2}$
LRE for Large Run Emphasis	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}R(i,j)^2$
LGRE for Low Intensity Run Emphasis	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{R(i,j)}{i^2}$
HGRE for High Intensity Run Emphasis	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}R(i,j)\times i^2$
SRLGE (SRE combined with LGRE)	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{R(i,j)}{i^2\times j^2}$
SRHGE (SRE combined with HGRE)	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{R(i,j)\times i^2}{j^2}$
LRLGE (LRE combined with LGRE)	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{R(i,j)\times j^2}{i^2}$
LRHGE(LRE combined with HGRE)	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}R(i,j)\times i^2\times j^2$
GLNUr for Gray Level Non-Uniformity	$\frac{1}{N_r}\sum_{i=1}^{G}\left[\sum_{j=1}^{N}R(i,j)\right]^2$
RLNU for Run Length Non-uniformity	$\frac{1}{N_r}\sum_{j=1}^{N}\left[\sum_{i=1}^{G}R(i,j)\right]^2$
RP for Run Percentage	$N_r/\sum_{i=1}^{G}\sum_{j=1}^{N}(R(i,j)\times j)$

On Table D.6, $p(i)$ corresponds to the probability of occurrence of a voxel with a luminosity value equal to i. Where $G-1$ is the maximum possible value for a voxel [25].

Gray Level Size Zone Matrix - GLSZM

1	2	3	4
1	3	4	4
3	2	2	2
4	1	4	1

Gray Level (i)	Size Zone (j)			
	1	2	3	4
1	2	1	0	0
2	1	0	1	0
3	0	0	1	0
4	2	0	1	0

Table D.7: (a) picture, (b) Example of a LSZM for a 4x4 image and for 4 gray levels with matrix named Z(i,j)

Table D.8: GLSZM

Features	ISO Formulas
SZE for Small Run Emphasis	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{Z(i,j)}{j^2}$
LZE for Large Run Emphasis	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}Z(i,j)\times j^2$
LGZE for Low Intensity Run Emphasis	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{Z(i,j)}{i^2}$
HGZE for High Intensity Run Emphasis	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}Z(i,j)\times i^2$
SZLGE (SZE combined with LGZE)	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{Z(i,j)}{i^2\times j^2}$
SZHGE (SZE combined with HGZE)	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{Z(i,j)\times i^2}{j^2}$
LZLGE (LZE combined with LGZE)	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{Z(i,j)\times j^2}{i^2}$
LZHGE(LZE combined with HGZE)	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}Z(i,j)\times i^2\times j^2$
GLNUz for Gray Level Non-Uniformity	$\frac{1}{N_r}\sum_{i=1}^{G}\left[\sum_{j=1}^{N}Z(i,j)\right]^2$
ZLNU for Zone Length Non-uniformity	$\frac{1}{N_r}\sum_{j=1}^{N}\left[\sum_{i=1}^{G}Z(i,j)\right]^2$
ZP for Zone Percentage	$N_r/\sum_{i=1}^{G}\sum_{j=1}^{N}(Z(i,j)\times j)$

On Table D.8, $p(i)$ corresponds to the probability of occurrence of a voxel with a luminosity value equal to i. Where $G-1$ is the maximum possible value for a voxel [25].

Gray Level Dependence Matrix - GLDM

5	2	5	4	4
3	3	3	1	3
2	1	1	1	3
4	2	2	2	3
3	5	3	3	2

Gray Level (i)	gray lvl dpcy (j)			
	0	1	2	3
1	0	1	2	1
2	1	2	3	0
3	1	4	4	0
4	1	2	0	0
5	3	0	0	0

Table D.9: (a) picture, (b) Example of a LDM for a 5x5 image and for 5 gray levels with matrix named N(i,j)

Table D.10: GLDM

Features	ISO Formulas
Coarseness	$1/\sum_{i=1}(N(i,1)(i,2))$
Contrast	$\frac{1}{E(G-1)}\left[\sum_{i=1}\sum_{j=1}N(i,j)\times N(j,1)\times(i-j)^2\right]\times\left[\sum_{i=1}N(i,2)\right]$
Busyness	$\frac{1}{N_r}\sum_{i=1}^{G}\sum_{j=1}^{N}\frac{Z(i,j)}{i^2}$
Complexity	$\sum_{i=1}\sum_{j=1}\frac{\lvert i-j\rvert}{E(N(i,1)+N(j,1))}\times(N(i,1)\times N(i,2)+(N(j,1)\times N(j,2)))$
Strength	$\sum_{i=1}\sum_{j=1}(N(i,1)+N(j,1))(i-j)^2/\sum_{i=1}N(i,2)$

On Table D.10, $p(i)$ corresponds to the probability of occurrence of a voxel with a luminosity value equal to i. Where $G-1$ is the maximum possible value for a voxel [25].

Appendix E

Model summary: VGG16

Model VGG16 summary

Layer name	Output shape
input_1	(224, 224, 3)
conv2d_1	(224, 224, 64)
conv2d_2	(224, 224, 64)
MaxPooling2d_1	(112, 112, 64)
conv2d_3	(112, 112, 128)
conv2d_4	(112, 112, 128)
MaxPooling2d_2	(56, 56, 128)
conv2d_5	(56, 56, 256)
conv2d_6	(56, 56, 256)
conv2d_7	(56, 56, 256)
MaxPooling2d_3	(28, 28, 256)
conv2d_8	(28, 28, 512)
conv2d_9	(28, 28, 512)
conv2d_10	(28, 28, 512)
MaxPooling2d_4	(14, 14, 512)
conv2d_11	(14, 14, 512)
conv2d_12	(14, 14, 512)
conv2d_13	(14, 14, 512)
MaxPooling2d_5	(7, 7, 512)
Flatten_1	25088
dense_1	4096
dense_2	4096
dense_3	1000

On the right column (Output shape) the last dimension of the convolutionnal layers is the number of filters used for the convolution. Each output from a convolutionnal layer has a 3-dimensional shape. At every layer, filters are applied on the output of the previous convolutionnal

layer. Each filter is there to capture patterns. For example, the first filter layer *Conv2d_1* (see table above) captures patterns such as edges, corners, dots, etc. Subsequent layers combine these patterns to make larger ones (such as combining edges to make squares, circles, etc.). As we move through the layers, the patterns become more complex, so there are larger combinations of patterns to capture. Therefore, we increase the filter size in subsequent layers to capture as many combinations as possible [77].

Let's finish by taking the concrete example of the convolutional layer conv2d_10. The input of shape $28 \times 28 \times 512$ is convolved by **512 filters** whith shape $2 \times 2 \times 512$. The result is 512 outputs with a shape of 14×14 (see fig. E.1).

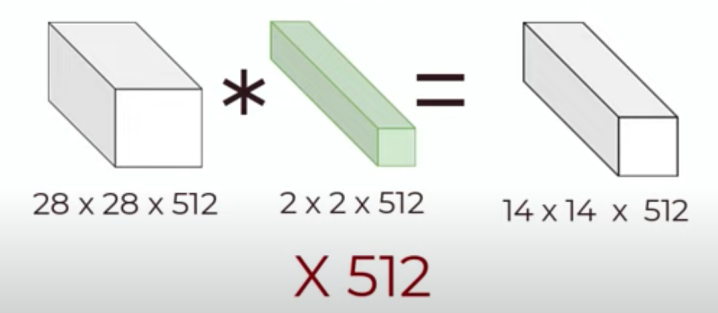


Figure E.1: calculation details of the convolutional layer conv2d_10 [78]

Bibliography

[1] Fabio Gelsomino, Monica Barbolini, Andrea Spallanzani, Giuseppe Pugliese, and Stefano Cascinu. The evolving role of microsatellite instability in colorectal cancer: a review. *Cancer treatment reviews*, 51:19–26, 2016.

[2] Thierry Lecomte, T André, F Bibeau, B Blanc, R Cohen, JP Lagasse, P Laurent-Puig, J Martin-Babau, Y Panis, F Portales, et al. Cancer du côlon non métastatique. *Thésaurus national de Cancérologie digestive*, 2019.

[3] Mehdi Karoui and Julien Taieb. La chimiothérapie néoadjuvante dans le cancer du côlon: un nouveau concept? *Hépato-gastro & oncologie digestive*, 19(7):498–505, 2012.

[4] Jingjun Wu, Qinhe Zhang, Ying Zhao, Yijun Liu, Anliang Chen, Xin Li, Tingfan Wu, Jianying Li, Yan Guo, and Ailian Liu. Radiomics analysis of iodine-based material decomposition images with dual-energy computed tomography imaging for preoperatively predicting microsatellite instability status in colorectal cancer. *Frontiers in oncology*, 9:1250, 2019.

[5] Carol E DeSantis, Chun Chieh Lin, Angela B Mariotto, Rebecca L Siegel, Kevin D Stein, Joan L Kramer, Rick Alteri, Anthony S Robbins, and Ahmedin Jemal. Cancer treatment and survivorship statistics, 2014. *CA: a cancer journal for clinicians*, 64(4):252–271, 2014.

[6] H Ueno, J Murphy, JR Jass, H Mochizuki, and IC Talbot. Tumourbudding'as an index to estimate the potential of aggressiveness in rectal cancer. *Histopathology*, 40(2):127–132, 2002.

[7] E Van Cutsem, Heinz-Josef Lenz, CH Kohne, Volker Heinemann, Sabine Tejpar, Ivan Melezínek, Frank Beier, Christopher Stroh, Philippe Rougier, JHJM van Krieken, et al. Fluorouracil, leucovorin, and irinotecan plus cetuximab treatment and ras mutations in colorectal cancer. 2015.

[8] Josep Tabernero, Heinz-Josef Lenz, Salvatore Siena, Alberto Sobrero, Alfredo Falcone, Marc Ychou, Yves Humblet, Olivier Bouché, Laurent Mineur, Carlo Barone, et al. Analysis of circulating dna and protein biomarkers to predict the clinical activity of regorafenib and assess prognosis in patients with metastatic colorectal cancer: a retrospective, exploratory analysis of the correct trial. *The Lancet Oncology*, 16(8):937–948, 2015.

[9] Clare Fiala and Eleftherios P Diamandis. Utility of circulating tumor dna in cancer diagnostics with emphasis on early detection. *BMC medicine*, 16(1):1–10, 2018.

[10] Robert J Gillies, Paul E Kinahan, and Hedvig Hricak. Radiomics: images are more than pictures, they are data. *Radiology*, 278(2):563–577, 2016.

[11] Andrew J Wong, Aasheesh Kanwar, Abdallah S Mohamed, and Clifton D Fuller. Radiomics in head and neck cancer: from exploration to application. *Translational cancer research*, 5(4):371, 2016.

[12] Rajat Thawani, Michael McLane, Niha Beig, Soumya Ghose, Prateek Prasanna, Vamsidhar Velcheti, and Anant Madabhushi. Radiomics and radiogenomics in lung cancer: a review for the clinician. *Lung cancer*, 115:34–41, 2018.

[13] Vishwa S Parekh and Michael A Jacobs. Deep learning and radiomics in precision medicine. *Expert review of precision medicine and drug development*, 4(2):59–72, 2019.

[14] Michele Avanzo, Lise Wei, Joseph Stancanello, Martin Vallieres, Arvind Rao, Olivier Morin, Sarah A Mattonen, and Issam El Naqa. Machine and deep learning methods for radiomics. *Medical physics*, 47(5):e185–e202, 2020.

[15] Albert Monjallon. *Introduction à la méthode statistique: par Albert Monjallon*. Vuibert, 1954.

[16] Albert Comelli, Alessandro Stefano, Claudia Coronnello, Giorgio Russo, Federica Vernuccio, Roberto Cannella, Giuseppe Salvaggio, Roberto Lagalla, and Stefano Barone. Radiomics: A new biomedical workflow to create a predictive model. In *Annual Conference on Medical Image Understanding and Analysis*, pages 280–293. Springer, 2020.

[17] Burak Koçak, Emine Şebnem Durmaz, Ece Ateş, and Özgür Kılıçkesmez. Radiomics with artificial intelligence: a practical guide for beginners. *Diagnostic and Interventional Radiology*, 25(6):485, 2019.

[18] Greet Kerckhofs. Lgbio2050 – single photon emission computed tomography (spect), 2020.

[19] Stefania Rizzo, Francesca Botta, Sara Raimondi, Daniela Origgi, Cristiana Fanciullo, Alessio Giuseppe Morganti, and Massimo Bellomi. Radiomics: the facts and the challenges of image analysis. *European radiology experimental*, 2(1):1–8, 2018.

[20] Dennis Mackin, Xenia Fave, Lifei Zhang, David Fried, Jinzhong Yang, Brian Taylor, Edgardo Rodriguez-Rivera, Cristina Dodge, A Kyle Jones, and Laurence Court. Measuring ct scanner variability of radiomics features. *Investigative radiology*, 50(11):757, 2015.

[21] Michela Lecchi, Piero Fossati, Federica Elisei, Roberto Orecchia, and Giovanni Lucignani. Current concepts on imaging in radiotherapy. *European journal of nuclear medicine and molecular imaging*, 35(4):821–837, 2008.

[22] Hyun Woo Goo and Jin Mo Goo. Dual-energy ct: new horizon in medical imaging. *Korean journal of radiology*, 18(4):555–569, 2017.

[23] David Bolus, Desiree Morgan, and Lincoln Berland. Effective use of the hounsfield unit in the age of variable energy ct. *Abdominal Radiology*, 42(3):766–771, 2017.

[24] Fabian Isensee, Philipp Kickingereder, Wolfgang Wick, Martin Bendszus, and Klaus H Maier-Hein. Brain tumor segmentation and radiomics survival prediction: Contribution to the brats 2017 challenge. In *International MICCAI Brainlesion Workshop*, pages 287–297. Springer, 2017.

[25] Nicolas de Vaucleroy, Benoît Macq, and Christophe De Vleeschouwer. " mathematical morphology applied to radiomics. 2019.

[26] Paul Desbordes, Benoit Macq, et al. Prognostic power of texture based morphological operations in a radiomics study for lung cancer. *arXiv preprint arXiv:2012.12652*, 2020.

[27] Erchan Aptoula and Sébastien Lefèvre. Morphological texture description of grey-scale and color images. *Advances in imaging and electron physics*, 169:1–74, 2011.

[28] Jiangwei Lao, Yinsheng Chen, Zhi-Cheng Li, Qihua Li, Ji Zhang, Jing Liu, and Guangtao Zhai. A deep learning-based radiomics model for prediction of survival in glioblastoma multiforme. *Scientific reports*, 7(1):1–8, 2017.

[29] Dmitrii Bychkov, Nina Linder, Riku Turkki, Stig Nordling, Panu E Kovanen, Clare Verrill, Margarita Walliander, Mikael Lundin, Caj Haglund, and Johan Lundin. Deep learning based tissue analysis predicts outcome in colorectal cancer. *Scientific reports*, 8(1):1–11, 2018.

[30] Chengliang Yang, Anand Rangarajan, and Sanjay Ranka. Visual explanations from deep 3d convolutional neural networks for alzheimer's disease classification. In *AMIA annual symposium proceedings*, volume 2018, page 1571. American Medical Informatics Association, 2018.

[31] Rahul Paul, Samuel Hawkins, Matthew B Schabath, Robert J Gillies, Lawrence O Hall, and Dmitry B Goldgof. Predicting malignant nodules by fusing deep features with classical radiomics features. *Journal of Medical Imaging*, 5(1):011021, 2018.

[32] Jennifer S Golia Pernicka, Johan Gagniere, Jayasree Chakraborty, Rikiya Yamashita, Lorenzo Nardo, John M Creasy, Iva Petkovska, Richard RK Do, David DB Bates, Viktoriya Paroder, et al. Radiomics-based prediction of microsatellite instability in colorectal cancer at initial computed tomography evaluation. *Abdominal Radiology*, 44(11):3755–3763, 2019.

[33] Clara Ousset-Masquelier. A quel âge devient-on senior ? – santé magazine, 2016.

[34] Pauline Fréour. Notre cerveau commence à décliner dès 45 ans, 2012.

[35] Dennis Mackin, Xenia Fave, Lifei Zhang, Jinzhong Yang, A Kyle Jones, Chaan S Ng, and Laurence Court. Harmonizing the pixel size in retrospective computed tomography radiomics studies. *PloS one*, 12(9):e0178524, 2017.

[36] Paul Bourke. Cross correlation. *Cross Correlation", Auto Correlation—2D Pattern Identification*, 1996.

[37] Roger Grosse. Topic 4: Local analysis of image patches. submitted, 2019.

[38] Maria Lyra, Agapi Ploussi, Maritina Rouchota, and Stella Synefia. Filters in 2d and 3d cardiac spect image processing. *Cardiology research and practice*, 2014, 2014.

[39] Nicolas Thome. Bases du traitement des images: Filtrage d'image. submitted, 2016.

[40] LO Iheme. Frequency domain bandpass filtering for image processing. *Electrical and Electron Eng Depart Digit Imag Proc Eastern Medit Univ*, 2011.

[41] Barry Van Veen. Introduction to circular convolution and filtering with the dft. submitted, 2013.

[42] Guillaume Thibault, B. Fertil, C. Navarro, S. Pereira, P. Cau, N. Lévy, J. Sequeira, and J. Mari. Texture indexes and gray level size zone matrix. application to cell nuclei classification. 2009.

[43] Jean Serra. Image analysis and mathematical morphology. 1982.

[44] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019.

[45] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[46] Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4918–4927, 2019.

[47] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[48] National Lung Screening Trial Research Team. Reduced lung-cancer mortality with low-dose computed tomographic screening. *New England Journal of Medicine*, 365(5):395–409, 2011.

[49] Gary King and Langche Zeng. Logistic regression in rare events data. *Political analysis*, 9(2):137–163, 2001.

[50] Tensorflow. Classification on imbalanced data. `https://www.tensorflow.org/tutorials/structured_data/imbalanced_data?hl=en`, 2021.

[51] Byeongho Heo, Sanghyuk Chun, Seong Joon Oh, Dongyoon Han, Sangdoo Yun, Youngjung Uh, and Jung-Woo Ha. Slowing down the weight norm increase in momentum-based optimizers. *arXiv preprint arXiv:2006.08217*, 2020.

[52] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8):2, 2012.

[53] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[54] Francesc J Ferri, Pavel Pudil, Mohamad Hatef, and Josef Kittler. Comparative study of techniques for large-scale feature selection. In *Machine Intelligence and Pattern Recognition*, volume 16, pages 403–413. Elsevier, 1994.

[55] Max Kuhn, Kjell Johnson, et al. *Applied predictive modeling*, volume 26. Springer, 2013.

[56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[57] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[58] Kashvi Taunk, Sanjukta De, Srishti Verma, and Aleena Swetapadma. A brief review of nearest neighbor algorithm for learning and classification. In *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pages 1255–1260. IEEE, 2019.

[59] Akhil Kumar, Vithala R Rao, and Harsh Soni. An empirical comparison of neural network and logistic regression models. *Marketing letters*, 6(4):251–263, 1995.

[60] Vinod Kumar Chauhan, Kalpana Dahiya, and Anuj Sharma. Problem formulations and solvers in linear svm: a review. *Artificial Intelligence Review*, 52(2):803–855, 2019.

[61] Wikistat. Machines à vecteurs supports — wikistat. `https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-m-app-svm.pdf`, 2016.

[62] B Chandra, Sati Mazumdar, Vincent C Arena, and Nagender Parimi. Elegant decision tree algorithm for classification in data mining. In *WISE Workshops*, pages 160–169. Citeseer, 2002.

[63] Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[64] Andrius Vabalas, Emma Gowen, Ellen Poliakoff, and Alexander J Casson. Machine learning algorithm validation with a limited sample size. *PloS one*, 14(11):e0224365, 2019.

[65] Konstantinos Sechidis. Comparison of different preprocessing techniques and feature selection algorithms in cancer datasets.

[66] Ryan A. Mardani. Practical_ml_tutorial_facies_examp. `https://github.com/mardani72/Practical_ML_Tutorial_Facies_examp`, 2020.

[67] Karimollah Hajian-Tilaki. Receiver operating characteristic (roc) curve analysis for medical diagnostic test evaluation. *Caspian journal of internal medicine*, 4(2):627, 2013.

[68] Lara Lusa et al. Evaluation of smote for high-dimensional class-imbalanced microarray data. In *2012 11th international conference on machine learning and applications*, volume 2, pages 89–94. IEEE, 2012.

[69] Amine Chemchem, François Alin, and Michaël Krajecki. Combining smote sampling and machine learning for forecasting wheat yields in france. In *2019 IEEE second international conference on artificial intelligence and knowledge engineering (AIKE)*, pages 9–14. IEEE, 2019.

[70] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

[71] Lei Yang, Di Dong, Mengjie Fang, Yongbei Zhu, Yali Zang, Zhenyu Liu, Hongmei Zhang, Jianming Ying, Xinming Zhao, and Jie Tian. Can ct-based radiomics signature predict kras/nras/braf mutations in colorectal cancer? *European radiology*, 28(5):2058–2067, 2018.

[72] Cuishan Liang, Yanqi Huang, Lan He, Xin Chen, Zelan Ma, Di Dong, Jie Tian, Changhong Liang, and Zaiyi Liu. The development and validation of a ct-based radiomics signature for the preoperative discrimination of stage i-ii and stage iii-iv colorectal cancer. *Oncotarget*, 7(21):31401, 2016.

[73] Kavitha Mukund, Natalia Syulyukina, Sonia Ramamoorthy, and Shankar Subramaniam. Right and left-sided colon cancers-specificity of molecular mechanisms in tumorigenesis and progression. *BMC cancer*, 20(1):1–15, 2020.

[74] Canadian Cancer Society. Stade et grade. `https://www.cancer.ca/fr-ca/cancer-information/cancer-101/what-is-cancer/stage-and-grade/?region=on`, 2018.

[75] Jean Marc Phelip. Cancer du colon : classifications moléculaires et anatomiques nécessaires à la décision thérapeutique. `https://www.fmcgastro.org/texte-postu/postu-2018-paris/cancer-du-colon-classifications-moleculaires-et-anatomiques-necessaires-a-la-decision-therapeutique/#:~:text=Il%20s'agit%20de%20tumeurs,m%C3%A9sappariement%20de%20l'ADN).`", 2018.

[76] Joyce Alvin Bearden. X-ray wavelengths. *Reviews of Modern Physics*, 39(1):78, 1967.

[77] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078, Sep 2020.

[78] Ajay Halthor. What do filters of convolution neural network learn? `https://www.youtube.com/watch?v=eL80Im8Hq0k`.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**