Texture Enhanced Tissue Analysis

by

Elizabeth Vargas Vargas

Supervised by

Keith Goatman

Toshiba Medical Visualization Systems

Abstract

Radiomics is the relatively new term used to describe the collection of multiple features from objects, or regions of interest, within a medical image. Usually these features are subsequently used to classify these regions, for example for organ segmentation or to differentiate benign and malignant tumours. Radiomic features are anything that can be measured about the regions of interest from the image, for instance shape metrics (such as volume and aspect ratio), first-order greylevel statistics (such as the mean and standard deviation), and texture metrics.

Textures are all around us. It is an intuitive concept, described qualitatively with words such as rough, smooth, fine, coarse, regular, irregular, etc. Many ways have been proposed to model textures mathematically in order to derive quantitative descriptions of it, the most popular methods including greylevel co-occurrence matrices, Laws' filters, Gabor filters, and fractal dimension.

This study explores the use of Markov Random Fields Markov Random Field (MRF) in texture analysis in Computer Tomography (CT) medical images. Specifically, a non-parametric MRF approach is adopted to classify tissues based on their texture. Two applications are investigated: firstly, the classification of entire regions of interest, and secondly, the classification of individual pixels.

However, recent studies indicate that texture features can have poor between-centre repeatability, being sensitive to different scanners and acquisition parameters. In CT imaging there is a continual compulsion to reduce the radiation dose, which usually results in increased image noise. This in turn has encouraged the application of image noise reduction algorithms. If texture features are to be useful in practice, the effects of image noise and noise reduction must be less than the differences between tissues of interest. Hence, this study also assessed the robustness of the MRF texture features to both image noise and noise reduction.

In this study experiments were performed to assess the performance of the MRF in differentiating textures from kidneys, liver, lungs and spleens in CT datasets. Each dataset was tested at five simulated dose settings, both with and without noise reduction to assess the robustness of the metrics to these parameters. The results showed that the Region of Interest (ROI) classification accuracy ranged from 75.1% to 87.8% without Adaptive Iterative Dose Reduction 3D (AIDR3D) noise reduction, and between 84.5% to 85.8% for the AIDR3D corrected volumes. The classification of individual pixels was more challenging. The results were better when differentiating the liver and lung pixels, than for differentiating the spleen and kidneys. The main problem was the misclassification of background pixels as an organ of interest, due to insufficiently representative training data from other organs and anatomical structures. Future work could include the use of 3D information to define textural features, as well as convex optimization to build a more robust dictionary.

Contents

List of Figures

List of Tables

List of Acronyms

AIDR3D Adaptive Iterative Dose Reduction 3D

CT Computer Tomography

FN False Negatives

FP False Positives

FT Fourier Transform

GLCM Gray-Level Co-ocurrence Matrix

GLRLM Gray-Level Run-Length Matrix

KNN K-Nearest Neighbour

LoG Laplacian of Gaussian

MAP Maximum a posterior probability

MRF Markov Random Field

PDF Probability Density Function

ROI Region of Interest

TMVS Toshiba Medical Visualization Systems

TN True Negatives

TP True Positives

UdG Universitat de Girona

Acknowledgments

First and foremost I would like to thank my family for their constant support. A mis padres, por su apoyo constante en todas las decisiones que he tomado en mi vida y por enseñarme con su ejemplo el valor del trabajo arduo y la perseverancia, ustedes son mi más grande inspiración y son las personas que más admiro en el mundo. A mi hermanita, por hacerme sentir siempre acompañada aunque estemos lejos y por ser siempre mi amiga y consejera. A mi abuelita, por haberme cuidado desde que era niña y por tenerme siempre presente en sus oraciones. Todos ustedes son mi más grande apoyo y es gracias a ustedes que todo esto ha sido posible. Mil gracias.

I also would like to express my gratitude to the Image Analysis Research Team at Toshiba for their help and support during this time. Keith, for sharing his expertise, kindness and overall patience during these four months. Catalina, for her priceless support since the beginning of the project until the last moment. Corne, for sharing his expertise about the evaluation metrics, as well as some LaTeXtips. Erin, for her help with the ground truth data that I needed to evaluate the segmentation. Andy, for sharing his knowledge about the statistical methods and for his help with technical problems. Alison, for sharing her work experience and made sure that I have all I needed. To James, Daniel, Carrie and Chengjia, thank you very much for sharing this time with me and for the moral support during these months. I learned a lot from each of you and I feel very lucky to have the opportunity to work with so talented people.

Last but not least, I would like to thank all the people involved in the VIBOT programme for allowing me to be part of this Master and fulfil my dreams. Thank you very much, you changed my life.

Chapter 1

Introduction

1.1 Motivation

Radiomics [26] is rather a new term, used to describe a set of multiple features derived from a Region of Interest (ROI) within a medical image. Normally these features are then used to classify these regions; the features could include shape, first-order grey-level statistics, and textural features. The most intuitive use of texture is to characterize different type of tissues. Since intensity values among different organs are very similar, the use of these types of features should help better differentiate tissues of interest.

(a) Thorax Computer Tomography (CT) scan

(b) Thorax CT scan with Adaptive Iterative Dose Reduction 3D (AIDR3D) correction

Figure 1.1: Effect of AIDR3D application

This study focuses on the use of Markov Random Field (MRF) in texture analysis, specifically applying them to CT imaging. CT imaging involves ionising radiation, which carries a risk

of inducing cancer. Although this risk is extremely small, there is consequently a continual compulsion to reduce the radiation dose, which usually results in increased image noise. Visually, the increase in image noise has a large effect on appearance and qualitative texture assessment. For this reason, one of the objectives of this study is to evaluate the robustness of the algorithm under different noise conditions. This was done using real CT datasets that have been processed, using realistic physical models of the noise, to simulate lower dose acquisitions. This approach was taken so that the patient only had to be scanned once. A noise reduction algorithm, known as AIDR3D, was also evaluated. Fig. 1.1 shows a CT coronal section through an abdomen, both before and after the application of the AIDR3D noise reduction.

There are a large number of potential applications of textural features in CT images. Probably the most obvious is segmentation. This can be easily solved for some organs, such as differentiating liver and lungs. However, it is more challenging to differentiate, for example, the liver and spleen, because their tissues are very similar. Another interesting application is to use texture to describe changes in tissues, whether natural or due to pathology. Applications could include oncology (cancer) treatments such as radiotherapy, where it is helpful to know that the therapy is harming the tumour and not the surrounding healthy tissue. Furthermore, such software could be of benefit to pharmaceutical companies by giving feedback on drug performance faster and more quantitatively. Last, but not least, different tumour genotypes may have different visible textures. This problem is approached by studying the correlation between cancer imaging and gene expression, which is known as *radiogenomics*.

1.2   Radiomics
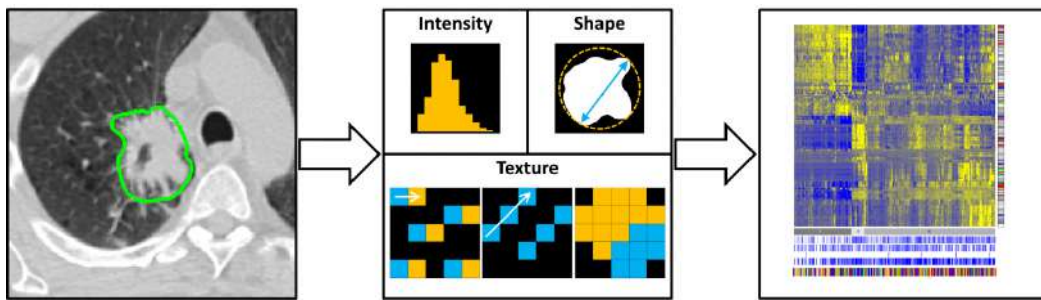
Figure 1.2: Schematic representation of a typical radiomics workflow [26]

Radiomics is a concept based on the use of ROI to compute a set of features that allows it characterization. Fig. 1.2 illustrates the workflow, which consists in three steps. First of all, a high quality standardized image is acquired. After that, the ROI is defined and finally a large

number of quantitative features is exacted from the previously defined ROI. These features describe, amongst others, image intensity, texture and shape and size of the ROI. The final step is the analysis of the features [26].

Radiomics provides complementary information to traditional sources of patient information, such as demographics, blood biomarkers, or genomics. It aims to improve individualized treatment selection and monitoring. If it is effective it has the potential to improve patient treatment (by selecting the most appropriate and effective treatments), and patient experience (by replacing invasive investigations with non-invasive imaging) [26].

1.3   Computer Tomography (CT) Imaging

Figure 1.3: Toshiba Aquilion One CT Scanner

Since its invention in the 1970s, CT has revolutionized medical imaging. It allows physicians to obtain high-quality, cross-sectional images of the internal structures of the body. Fig. 1.3 illustrates a Toshiba CT Scanner.

CT images are reconstructed from "projection data": a large number of measurements of x-ray transmission through the patient. The resulting images are tomographic "maps" of the x-ray linear attenuation coefficient. The fundamental task of a CT system is to rapidly make

a large number of highly accurate measurements of x-ray attenuation through the patient. A basic system consists of a patient table, a control console, a computer and a gantry that itself contains the rotating x-ray source and detectors, and the data-acquisition system (DAS) [20].

1.3.1   Adaptive Iterative Dose Reduction 3D (AIDR3D)

One of the most important disadvantages of Computer Tomography (CT) imaging is the amount of radiation that the patient receives, in contrast to other technologies. The dose of radiation applied to the patient can be reduced, however the result will be a very noisy image. This problem has lead to a lot research oriented to noise reduction, preserving spatial resolution. In this context, Toshiba has been committed to reduce the patient dose, as illustrated by Fig. 1.4. In 2011, a technology that reduces noise and improves spatial resolution was developed and it was called AIDR3D.
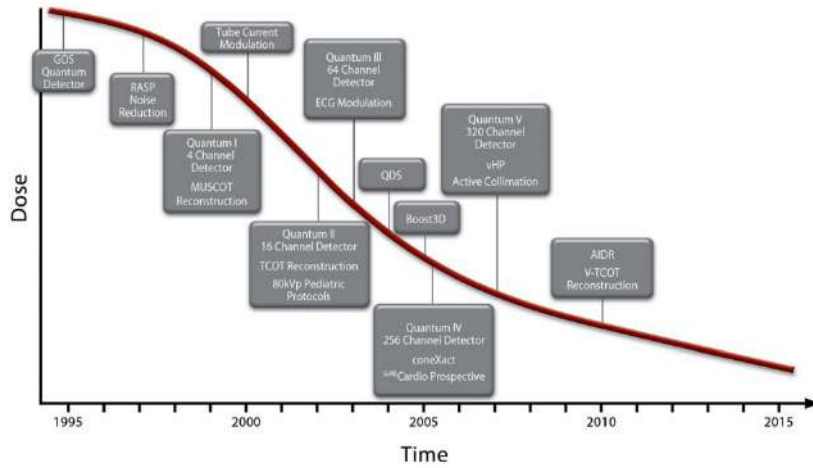
Figure 1.4: Toshiba Commitment to Dose Reduction

The AIDR3D algorithm is designed to work in the raw data, as well as in the reconstruction domain. It uses a scanner model and a statistical noise model considering photon and electronic noise to eliminate noise due to a photon starvation in the projection data. The statistical and scanner models are used with projection noise estimation for electronic noise reduction processing which takes place in the raw data domain. The first one analyse the physical properties of the CT scanner, while the second one is in charge of modelling the electronic and quantum noise. The projection noise estimation takes care of noise and artifacts reduction. The initial image is used as an input image in every iteration to be compared with the output image. A sophisticated iterative technique is then performed to optimize reconstructions by performing sharpening and

smoothing at the same time. Finally, a weighted blending is applied to the original reconstruction and the output of this iterative process to maintain the noise granularity. The result is an image with an increased signal to noise ratio and a improved spatial resolution [12].

1.4   Aims

The main goal of this project is to implement and evaluate a texture analysis metric based on Markov Random Field (MRF) and compare it with other texture metrics. The evaluation will be made on both simulated and real CT datasets, with the aim of finding texture measures which are able to differentiate tissue types while being insensitive to the image scanner configuration (i.e. features such as image noise level, resolution, etc.). In order to achieve this goal, the objectives to be accomplished are presented below.

- Implement a region classification algorithm using Markov Random Fields to model image texture.

- Implement a voxel classification algorithm that differentiate between different organ tissue

- Compare the proposed method with several different texture analysis metrics in real and simulated CT datasets.

- Evaluate the ability of the texture metrics to differentiate tissue types and their sensitivity to image noise.

1.5   Thesis Outline

This thesis is structured as follows: Chapter 2 presents the problem of texture classification and states the main reasons that makes it challenging and worth to be studied. Chapter 3 summarizes the state of the art on texture classification and dedicates a section to the use of texture in CT Imaging. In Chapter 4 the methodology that lead to the implementation of a MRF algorithm is explained, while Chapter 5 illustrates the results obtained in two different datasets. Finally, Chapter 6 summarizes the conclusions and the future work.

Chapter 2

Problem Definition

2.1 Medical Motivation

Discriminating one texture from another can be a challenge even for the human eye in some situations. In Computer Vision, textures can be described by a different amount of features studied through the literature [33]. The main challenge consist in determining which features are relevant enough to successfully distinguish one texture from another.

(a) Kidneys

(b) Liver

(c) Lungs

(d) Spleen
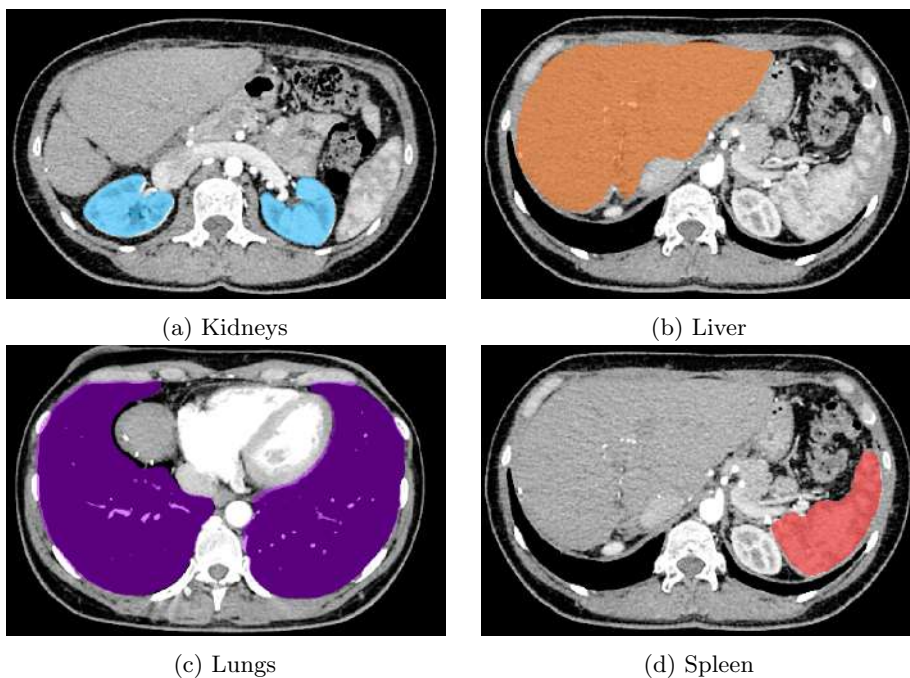
Figure 2.1: Regions of Interest

In medical image analysis, texture is one of the most useful features, since it can be applied to a wide variety of problems. One example is the classification of organ tissue, which is hard to classify using shape or gray level information because shape is not consistent and the intensities overlap for the case of soft tissues. The use of textures might be useful in this context, considering that it is homogeneous among the slices forming the 3D image [43].

This thesis deals with the problem of *texture classification*, applied to Computer Tomography (CT) imaging. Our goal is to differentiate between four different tissues, belonging to Region of Interest (ROI) previously defined: kidneys, liver, lungs and spleen. Fig. 2.1 illustrates the organs.

2.2   Technical Motivation

The problem of differentiating one texture from another depends not only on the features used, but also on different considerations about the data itself, as well as the conditions in which it was acquired. This section illustrates the main practical technical considerations that need to be taken into account when working with textures.

Scale

Texture features depend on the scale in which they are analysed. Fig. 2.2 illustrates an example of a texture where the zooming factor started at 16 (left) and is divided by two each time until the zooming factor is 1 (right). It is clear from this example that the way we perceive textures is different in each case, even though the image is the same.

Figure 2.2: Texture at different scales [39]

Wide dynamic range

The high range of intensities in the medical images affects the texture features specially since some of the high values are noise. This high range of intensities also affects computational time. In the datasets used in this work, CT images are quantized in 16 bits.

Noise

The parameters of a CT scanner need to be configured according to the x-ray dose that can be applied to the patient. In theory, a lower dose is desired, however the lower the dose, the noisier the image. This selection of parameters affects the resulting image and therefore the appearance of the textures.

Object texture inhomogeneity

An object might have several textures. One example of this are the lungs, which present seven main textures. Fig. 2.3 shows how diffuse lung disease changes the texture of the lungs. Six different patterns of texture were identified by [38].
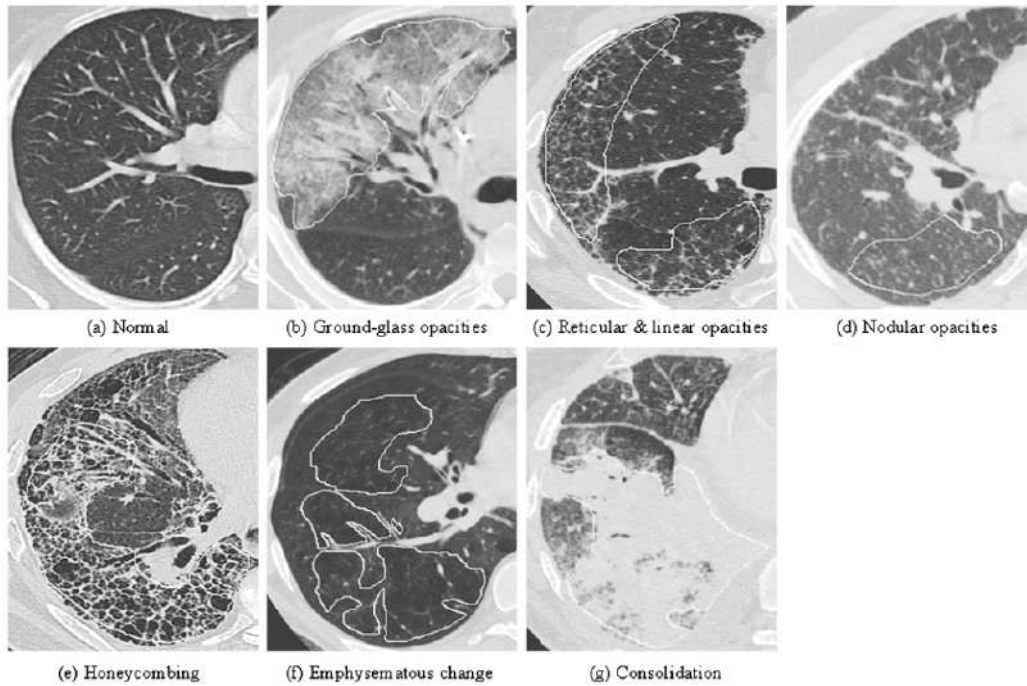
Figure 2.3: Seven typical textures seen in the lung. As these images show, a single lung may contain more than one texture type [38].

Algorithm parameters

Poor results are not always caused by a bad algorithm. Normally, even though the algorithm can achieve very good performance, the bad choice of parameters reduce the accuracy of the

texture classifications.

Similarity metric

The definition of a robust distance/similarity metric that measures how similar or how different two textures are from each other is also a fundamental issue in texture classification [33].

2.3 Proposed method

Taken into consideration these technical challenges, we chose a non-parametric Markov Random Field (MRF) as a strong descriptor for our application. The use of a dictionary to represent the textons of a particular class aims to overcome the problems of noise and texture inhomogeneity. Moreover, the approach uses a histogram that is oriented to beat the problem of wide dynamic range in CT images. Different parameters and similarity metrics were tested as well, in order to find the ones that leads to a more accurate classification. Further details can be found in Chapter 4, in which we also compare them with other state-of-the-art descriptors and classification techniques.

Chapter 3

State of the Art

This chapter reviews the main published work related to classification using textural features. Section 3.1 presents a classification of the most common texture metric algorithms, while section 3.3 focuses on the use of Markov Random Field (MRF) for texture classification. Last but not least, section 3.4 review different methods that use textures to classify voxels in Computer Tomography (CT) images.

3.1 Texture Features Extraction Methods

This section presents some of the most important techniques to extract textural features divided into four different categories, according a classification presented in [36].

3.1.1 Statistical Methods

Figure 3.1: Gray-Level Co-ocurrence Matrix (GLCM) [34]. The intensity 1 occurs with the intensity 1 one time (blue cells) and the intensity 2 occurs with the intensity 1 two times (pink cells).

GLCM

The most common method to extract textural features extraction is the use of GLCM, which characterize the texture of an image by calculating how often pairs of pixel with specific values and in a specified spatial relationship occur in an image. After its calculation, several statistics measures are extracted from this matrix [33].

Fig. 3.1 illustrates the computation of a GLCM (right) from a 2D image (left). The GLCM has size n by n, where n represents the maximum intensity of the image. Each position i, j represents the number of times that the intensity i occurs with the intensity j. In this example, the intensity 1 occurs with the intensity 1 one time (blue cells) and the intensity 2 occurs with the intensity 1 two times (pink cells). In this particular case, the distance, d, between intensities is 1 and the angle, θ is $0°$, however both parameters can be modified depending on the implementation. Fig. 3.2 illustrates some possible angles for a 2D image: $0°$ (yellow), $45°$ (green) and $90°$ (orange).

GLCM has probed to be successful to discriminate textures, in particular when the ratios or element values are used [24]. The main drawback of this technique is the complexity in time of its calculation.

Figure 3.2: GLCM Orientations. $0°$ (yellow), $45°$ (green) and $90°$ (orange).

Gray-Level Run-Length Matrix (GLRLM)

Run-length statistics represents the coarseness of a texture in specific directions [43], the same ones defined for the GLCM. Typically, fine textures contain more short runs with similar intensities and coarse textures have more long runs with different gray level intensities.

Figure 3.3: GLRLM. The number of pixels equal to zero for run-length 1 is 2 (blue), while the number of pixels equal to two for run-length 2 is 1 (pink)

A run-length matrix P is defined as a n by k matrix, where n is the maximum gray level in the image and k is equal to the possible maximum run length in the image [43]. Element $P(i,j)$ represents the number of runs with pixels of gray level intensity equal to i and length of run equal to j along a specific orientation. Fig. 3.3 illustrates an example of the computation of a GLRLM for 0° orientation: the number of pixels equal to zero for run-length 1 is 2 (blue), while the number of pixels equal to two for run-length 2 is 1 (pink).

Once the run-length matrices are calculated along different directions, several texture descriptors are computed. Typically, some common descriptors extracted from GLRLM are: short run emphasis (SRE), long run emphasis (LRE), high gray-level run emphasis (HGRE), low gray-level run emphasis (LGRE), pair-wise combinations of the length and gray level emphasis (SRLGE, SRHGE, LRLGE, LRHGE), run-length non- uniformity (RLNU), grey-level non-uniformity (GLNU), and run percentage (RPC) [15]. The main drawback of this approach, similarly to GLCM is the expensive computational time.

3.1.2 Texture Filter Methods

Gabor Filters

A Gabor function in 2D is defined by Eq.3.1.

$$G(x,y;x_0,\omega_{x_0},\sigma_x,y_0,\omega_{y_0},\sigma_y) \equiv e^{-\frac{(x-x_0)^2}{2\sigma_x^2}-\frac{(y-y_0)^2}{2\sigma_y^2}}e^{j\omega_{x_0}x+j\omega_{y_0}y} \qquad (3.1)$$

The Fourier Transformation of this function is presented in Eq. 3.2.

$$\hat{G}(\omega_x,\omega_y;x_0,\omega_{x_0},\sigma_x,y_0,\omega_{y_0},\sigma_y) = 2\pi\sigma_x\sigma_y e^{-\frac{\sigma_x^2(\omega_x-\omega_{x_0})^2}{2}-\frac{\sigma_y^2(\omega_y-\omega_{y_0})^2}{2}} \qquad (3.2)$$

Gabor filters are used to extract local image features by convolving an input image with a 2-D Gabor function. The filter acts as a local band-pass filter with certain optimal joint localization properties in the spatial domain and in the frequency domain [10].
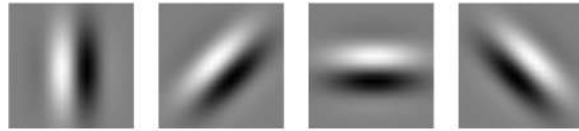
Figure 3.4: Gabor filters bank at different orientations [44]

Fig. 3.4 illustrates an example of a Bank of Gabor Filters that contains different orientations and spatial frequencies, whose objective is to cover appropriately the spatial frequency domain.

The images are convolved with the filter bank and the resulting features can be used directly for classification or segmentation or they can be transformed into a post-processed feature vector.

Despite Gabor filters has probed to work very well for texture classification, their main drawback is that a good result depends entirely on the election of the right type of filters. Moreover, Gabor filters have not zero mean, which produces a non-uniform coverage of the Fourier domain [22].

Laplacian of Gaussian (LoG) Filters

The 2-D LoG function centered on zero and with Gaussian standard deviation σ is presented in Eq. 3.3.

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2 + y^2}{2\sigma^2}\right]e^{\frac{x^2+y^2}{2\sigma^2}} \quad (3.3)$$

The LoG operator calculates the second spatial derivative of an image. That is to say, in areas where the image has a constant intensity the LoG response will be zero, while in the vicinity of a change in intensity, however, the LoG response will be positive on the darker side, and negative on the lighter side. This response produces the effect of edge sharpening in an image [25]. Fig. 3.5 illustrates the LoG filter in 2D.
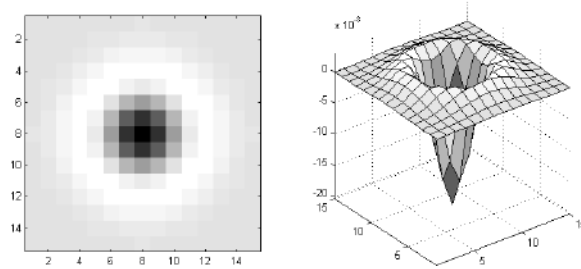
Figure 3.5: Laplacian of Gaussian function in 2D [1]

In [22] is proposed an image retrieval approach based on log-Gabor filter scheme. They made emphasis on the filter design to preserver the relationship with receptive fields and take advantage of their strong orientation selectivity. The experimental results shows that this approach has better results than the use of Gabor filters by themselves.

3.1.3 Model-based Methods

Fractal Dimension

Fractal dimension is an essential parameter of Fractal geometry, as well as an important characteristic of fractals, because it contains information about their geometric structure. Fig. 3.6 illustrates the first iterations to build a fractal.

In a bounded set X considered in Euclidean $n-$space, the set X is said to be self-similar when X is the of N_r distinct non-overlapping copies of itself, each of which is similar to X scaled down by a ratio r. In Eq. 3.4 can be seen how the the Fractal Dimension (D) can be derived from these variables [29].

$$D = \frac{log(N_r)}{log(\frac{1}{r})} \tag{3.4}$$

In [17] is presented an overview of the algorithms to calculate the fractal dimension of a signal, including box-counting methods, fractional Brownian motion (fBm) methods and area measurement methods.
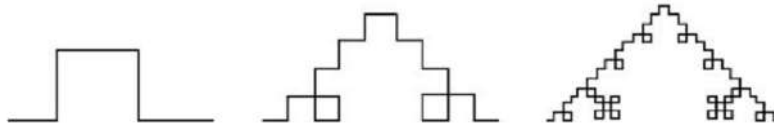
Figure 3.6: First iterations to build a fractal [17]

Even thought fractal dimension has been widely used in different texture applications, in some cases, fractal analysis does not perform correct image segmentation. In fact, some images are complex to study because they present irregularities and more regular zones at all scales, without following a clear law [17].

Autoregressive Model

Autoregressive models are used in image analysis to express spacial dependency between neighbouring pixels. Eq. 3.5 defines a polynomial with lag operators $\mathcal{L}_0^k y_{ij} = y_{i-kj}, \mathcal{L}_1^k y_{ij} = y_{ij-k}$.

$$\theta(z_0, z_1) := 1 - \sum_{ij=-n}^{n} \theta_{ij} z_0^i z_1^j, \theta_{00} = 0 \tag{3.5}$$

The autoregressive random field y_{ij} is defined by the stochastic difference, as illustrated by

Eq. 3.6, where η_{ij} represents white noise.

$$\theta(\mathcal{L}_0, \mathcal{L}_1)y_{ij} = \eta_{ij} \tag{3.6}$$

In [23] is presented an approach that uses rotation invariant autoregressive models for texture classification. The model affords rotation invariant parameters, however it becomes overcomplete and ultimately ill-conditioned as the number of nodes are increased (to achieve invariance to smaller rotations).

MRF

MRF are one of the most common *Model Based Methods* used for texture classification. The basic principle, is the assumption that the intensity at each pixel in the image depends on the intensities of only the neighbouring pixels [37]. Since MRF are the main focus of this work, Section 3.3 describes them in more detail.

3.1.4 Frequency Domain and Wavelet Methods

Fourier Transform (FT)

The Fourier Transform corresponds to a change of basis, to represent a signal as a infinite weighted sum of an infinite number of sinusoids. Eq. 3.7 defines the Fourier Transform of a signal $g(x, y)$ [8]. Fig. 3.7 illustrates an example of the FT of a 2D function.

$$\mathcal{F}(g(x,y))(u,v) = \int \int_{\inf}^{\inf} g(x,y)e^{-i2\pi(ux+vy)}dxdy \tag{3.7}$$
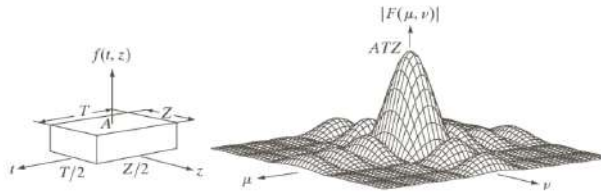
Figure 3.7: Fourier Transform in 2D [9]

Frequency-based texture descriptions are usually based on transforms that provide measurements of the spatial frequency spectrum. The most common method in this context is the FT. The features are obtained by taking the transform of an area of interest a calculate the FT, producing a 2D spectrum. It is converted from a Cartesian to a polar coordinate

system, where the radius, r, indicates frequency and the angle, θ, its direction. In case an orientation-independent texture measurement is desired, the two-dimensional polar spectrum can be averaged over, producing a one-dimensional, directionally invariant, frequency spectrum [3].

Wavelets

A wavelet ψ is a function produced by shifting by b and scaling by a a basic function $\psi(t)$ called mother wavelet, as illustrated by Eq. 3.8 [24].

$$\psi_{ab}(t) \equiv \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \tag{3.8}$$

Fig. 3.8 illustrates an example of an image decomposition using the Wavelet transform, which extract directional details that capture horizontal, vertical and diagonal activity. This leads actually to the basic idea of multi-resolution analysis, which allows for the zooming in and out on the underlying texture structure [4].
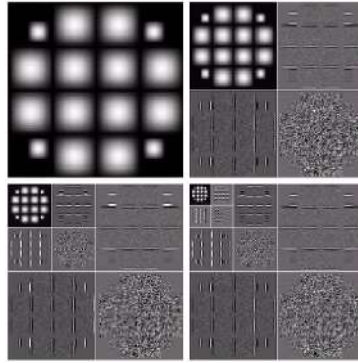
Figure 3.8: Wavelets [9]

Dettori and Semler [4] presented a paper focused on multi-resolution texture analysis for the classification of tissues from normal chest and abdomen CT scans. They used different texture features from distinct multi-resolution transforms: the Haar wavelet, Daubechies, Coiflet wavelet, ridgelet and curvelet transforms. Their results were compared with classification using coocurrence and run-length matrices, showing that the multiresolution features outperform the statistical methods.

3.2 Combined Methods

As could be seen on the previous sections, different approaches presents advantages and disadvantages, depending mostly on the type of dataset they were tested on. For this reason, some

authors have tried to combine the information provided by each method, in order to improve the way in which texture in characterized.

In [28], an approach that combines statistical and spectral methods for classification of texture images is presented and compared to other combined textural feature extraction methods: Wavelet Statistical Features (WSF), Combination of wavelet and second order statistical features and Spatial frequency features. The proposed method is called Multi Resolution Combined Statistical and Spatial Frequency (MRCSF) and combines first order statistical properties like mean, energy, variance and entropy, second order statistical properties like Markov Random Field Matrix, GLCM combined with spatial frequency of Multi resolution analysis. The results showed that the Classification rate for MRCSF technique is 92.6% for 112 images.

Barley and Town [2] presents a similar approach, in which the efficacy of different descriptors, e.g. GLCM, Gabor wavelets, steerable pyramids and Scale-invariant feature transform (SIFT), is evaluated by combining several of them by means of machine learning approaches such as Bayesian networks, support vector machines, and nearest-neighbour approaches. A comparison of the average and best success rates demonstrates the importance of selecting the correct machine learning technique for each classification task.

3.3 Markov Random Fields (MRF)

3.3.1 Parametric Models

In order to build the model, the image is represented by an M by N lattice $\mathcal{S}$ and a neighbourhood, $\mathcal{N}$ is defined. Fig. 3.9 illustrates the different type of neighbourhoods that could be defined around a pixel, establishing the order of the MRF model. Typically, the size of the neighbourhood is determined according to the data and the problem that will be solved.

The random variables are the grey values of the image, $X(i,j)$ at site (i,j). A label represents a site distinction in the image defined on $\mathcal{S}$. Let $\mathcal{L}$ be a set of k discrete labels, $\{l_1, l_2, ..., l_k\}$. A discrete MRF is a random field that satisfies the properties described below [37]:

1. *Positivity:* $P(X) > 0$ for all X.

2. *Markovianity:* $P(X(i,j)|all\,point\,in\,\mathcal{L}\,except\,(i,j)) = P(X(i,j)|all\,point\,in\,\mathcal{N})$

3. *Homogeneity:* $P(X(i,j)|all\,point\,in\,\mathcal{N})$ depends only on the configuration of neighbours and is translation invariant.

The task in classification is then to assign X_s the right label from the set $\mathcal{L}$. The objective is the *joint posterior probability* of the MRF labels [16], determined according to Eq. 3.9 by
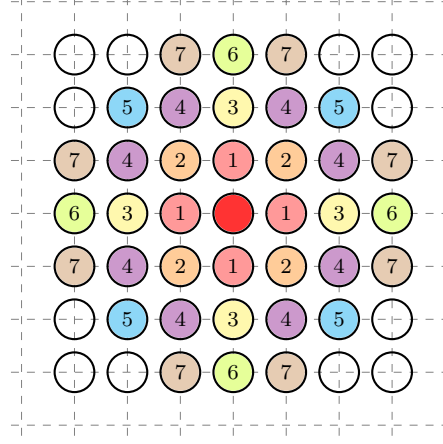
Figure 3.9: Markov Random Field model orders

means of the Bayes rule. The statistical criteria for optimality is the Maximum a posterior probability (MAP), which consist in assigning labels, such that $P(l|x)$ is maximum.

$$P(l|x) = P(x|l)P(l) \tag{3.9}$$

Generally, MRF characterize mutual influences among pixels using conditional MRF distributions, that is to say, they derive the *joint distribution* from *conditional distributions*, which is indeed a very difficult task. Hammersley and Clifford (1971) found out a solution to this problem by stating the equivalence between MRF and **Gibbs distributions**. The MRF-Gibbs equivalence theorem indicates that the *joint distribution* of a MRF is a Gibbs distribution, with respect to a specific neighbourhood system [16].

A Gibbs random field is completely characterized by its energy function. In order to use this property, the neighbourhood, $\mathcal{N}$, needs to be decomposed in a set of small neighbourhoods that form cliques, defined as a set of vertices, such that every par of vertices is connected by an edge [18]. Therefore, the cliques could be single site, pairs, triples and quadruples, as illustrated by Fig. 3.10, where different clique types for the second-order neighbourhood are presented. In the same way as the neighbourhood size, cliques are determines according to the problem and are usually used in the definition of the conditional probabilities.
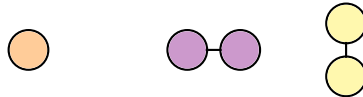
Figure 3.10: Clique types

Eq. 3.10 illustrates the probability representation according to Gibbs distributions, where $E(x)$ is a energy or cost function, presented by Eq. 3.11 and Z is the partition function, exhibited in Eq. 3.12.

$$P(x) = \frac{1}{Z} \exp -E(x) \tag{3.10}$$

The elements of the sum are known ar *clique potencials*, and depends, as their name says, on the cliques defined for the problem.

$$E(x) = \sum_{c \in \mathcal{C}} \Psi_c(x_c) \tag{3.11}$$

$$Z = \sum_x \exp(-E(x)) \tag{3.12}$$

In conclusion, we are interested in finding the most probable labelling,

$$l* = \arg \max_l P(l|x) = \arg \min_l E(l; x) \tag{3.13}$$

Which is known ad MAP inference or *energy minimization* and could be solved using a large amount of techniques including dynamic programming, graph-cuts, belief propagation among others.

In [32] is presented a survey about energy minimization algorithms for pixel-labelling tasks. The authors compare the performance and different trade-offs among graphs cuts, loopy belief propagation (LBP), tree-reweighted message passing (TRW) and iterated conditional model (ICM). The algorithms were applied to a binary segmentation problem, which led to conclude that graph cuts are guaranteed to compute the global minimum, as is TRW, while LBP implementations come extremely close but never actually attain the global minimum.

In the literature, there is a large amount of articles that use this framework to model different type of labelling problems. In general, the classification problem is related to assign a class to each image, composed by only one type of texture, while the segmentation problem focuses on assigning labels to pixels. For this reason, the literature review includes methods for classification and segmentation.

Kato and Pong [13] presents an approach for image segmentation, in which they combine color and texture features. Their framework uses combinatorial optimization (simulated annealing) to maximize the posterior probability. The misclassification rate obtained is however very big for 256 by 256 images.

Parallel to the traditional approaches, different authors had tried to come up with new ideas to solve this discrete optimization problem. Hochbaum [11] proposes an efficient polynomial time

algorithm to solve the MRF problem and apply it for image segmentation. The experiments are carried out in simulated medical images and the results are visually satisfactory, however, no quantitative result is presented and no relevant dataset is used to evaluate the algorithm.

3.3.2 Non-Parametric Models

In contrast with the approaches previously described, non-parametric models do not have a fix number of parameters, but instead the number of parameters grow with the amount of training data [21], since the complexity of the model increase upon the sample. For example, Varma and Zisserman [40] proposed an approach based on a two dimensional histogram, that represents the joint pdf between the central pixels and the neighbourhood texton representation, using a dictionary. They later extended their approach to be used on image patches [41], obtaining 97.9% accuracy on the San Francisco Dataset [14], where hand-segmented regions are classified, after using only a single image for training.

The following chapters investigate the use of these non-parametric methods in CT imaging. The non-parametric approach was chosen based on the high accuracies attained by Varma and Zisserman, and that the method relies on fewer prior assumptions, such as the form of the distribution. Section 4.1 describes the non-parametric method used in more detail.

3.4 Texture Algorithms in CT Imaging

Texture classification is, as could be seen along this chapter, a well known problem in computer vision. However, it has also been widely studied in the medical imaging field. One common application is the use of textures to characterize tissues in CT images, in order to perform segmentation, diagnosis, etc.

In [27] texture analysis is used to characterise oral cancer involving mucosa and to assess its effectiveness in differentiating between the various grades of the tumour on CT images. The texture features used were fractal dimension, lacunarity and GLCM, which were computed for each Region of Interest (ROI): lesions and contralateral normal side. The results showed that the difference between the mean fractal dimension and GLCM parameters of the lesion compared with the normal ROI were statistically significant, while there were no difference observe between different tumor types.

A similar approach can be found in [19], except that this time the objective is to define an optimally performing computer-aided diagnosis (CAD) architecture for the classification of liver tissue from non-enhanced CT. Similarly as in the previous case, the ROI are already defined by radiologist and for each of them a set of textural features are extracted, including first order statistics, spatial gray level dependence matrix, gray level difference method, Law's

texture energy measures and fractal dimension measurements. Two different classifiers were constructed and compared: a five multilayer perceptron neural networks (NNs) and a second one that comprises five different primary classifiers: one multilayer perceptron NN, one probabilistic NN, and three k-nearest neighbour classifiers. The best results are obtained using the last classifier with selected features from a fused feature set.

At last, Susomboon and Raicu proposed an approach for automatic single-organ segmentation in 2-D CT images in [30]. They approach consisted in three steps: probability image of the organ of interest computation using texture features, split-and-merge segmentation and region growing for refinement. To extract texture features, a grey-level co-ocurrence matrix was computed and nine texture descriptors were extracted: entropy, energy, contrast, sum average, variance, correlation, maximum probability, inverse different moment, and cluster tendency. The results presented 88% sensitivity and 96% specificity. The authors extended this approach to 3-D CT in [31].

Chapter 4

Methodology

4.1 The MRF Classifier

In [40], Varma and Zisserman introduced a classifier that uses the intensity values of a local neighbourhood as texture features, in order to perform single image classification on different texture class instances.

The algorithm models the Probability Density Function (PDF) of the central pixels conditioned on their neighbours. According to the information presented in Chapter 3, it can be clearly seen that it represents the principle of a Markov Random Field (MRF), formalised by Eq. 4.1, where x_c is a pixel location in the 2D integer lattice on which the image I has been defined and $\mathcal{N}(x_c)$ is the neighbourhood around x_c. This implies that the full conditional distribution of x_c depends only on the neighbours $\mathcal{N}(x_c)$.

$$p(I(x_c)|I(x), \forall x \neq x_c) = p(I(x_c)|I(x), x \in \mathcal{N}(x_c)) \tag{4.1}$$

The proposed approach is divided into two main stages: a *learning stage* where statistical distribution models of texture classes are learned from training examples, and a *classification stage* where novel images are classified by comparing their distributions to the learnt models.

Extract Textons → Agregate Textons → Kmeans → Label Pixels →
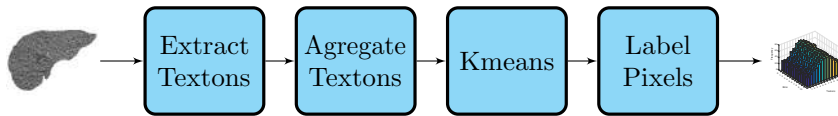
Figure 4.1: Learning Pipeline. A Region of Interest (ROI) slice is used as input. Textons are extracted from the slice and aggregated to be subsequently clustered using K-means algorithm to create a dictionary. The pixels of the original ROI are labelled using their distance to the textons dictionary and a 2D histogram is generated, as a model of the ROI.

4.1.1 Learning stage

Fig. 4.1 illustrates the pipeline of the learning stage, which is described in detail below.

1. A texton is represented by a set of $N^2 - 1$ pixels, in the neighbourhood, $\mathcal{N}$, of a pixel. Fig. 4.2 illustrates this concept: the yellow pixels form a texton in a 3 by 3 neighbourhood. It is important to note that the central pixel is not included in the texton.
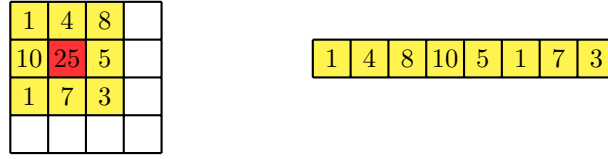
Figure 4.2: Texton Representation. The texton associated to the central pixel (red) is formed by the pixels of its neighborhood (yellow).

2. The textons calculated in step 1 from all the training images are then aggregated by the known texton class (i.e. lung, liver etc.). In the original implementation of Varma & Zisserman, only a subset of the training set is selected in this stage, for performance reasons, however in our implementation we used all the training images.
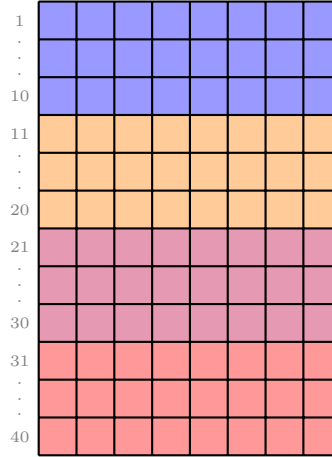
Figure 4.3: Textons Dictionary: This is an example of a dictionary of 10 textons per class, represented by a different color in the illustration.

3. The aggregated textons from all the training images are then clustered using the K-means algorithm [5]. The resultant cluster centres form a dictionary of textons. The most important parameter required by the K-means algorithm is the number of cluster centres.

Varma and Zisserman found that 10 cluster centres performed best on their datasets, and therefore we used this as the starting point in our investigation. Fig. 4.3 presents an example of a dictionary for the case of a 3 by 3 neighbourhood. Each row represents a texton, a vector of eight pixels. Since there are four texture classes in our experiments (see chapter 2), when the number of K-means cluster centres is 10, the dictionary will contain 40 rows in total (i.e. 10 textons per class, one for each K-mean cluster centre).

4. The pixels of each image in the training set are labelled to the index of the closest texton in the dictionary. The distance from a texton to the dictionary of textons is calculated by computing the euclidean distance of each texton. Fig. 4.4 illustrates an example of the calculation of the distance from a single texton to the dictionary. In this case, the minimum distance corresponds to the $10th$ texton in the dictionary, which means that the pixel that is represented by this texton, will be labelled as 10.
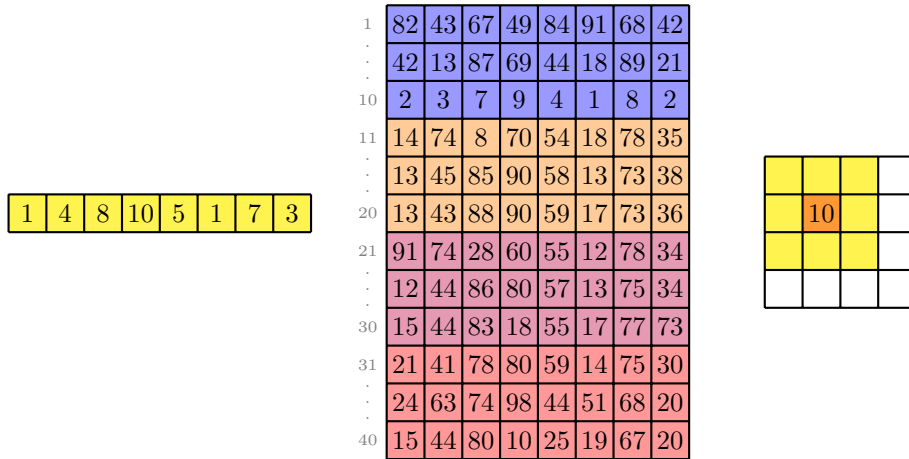
Figure 4.4: Distance to Dictionary. The texton of the pixel that is intended to be labelled is compared against each texton of the dictionary by using the euclidean distance. The label corresponds to the index of the texton in the dictionary with the minimum distance.

5. Once the label of each pixel has been assigned, we retrieve the original intensity value of that pixel. We accumulate this value as a frequency count of the histogram. After repeating this process for all labels, we have one histogram per label, so we concatenate the histograms of all labels to generate the 2D histogram (Fig. 4.5). In this case, since the Computer Tomography (CT) images are in 16-bits, the total range of values can not be covered. For this reason, the values between the percentile 25 and 75 of the total range of values of the training set was used to build the histogram.
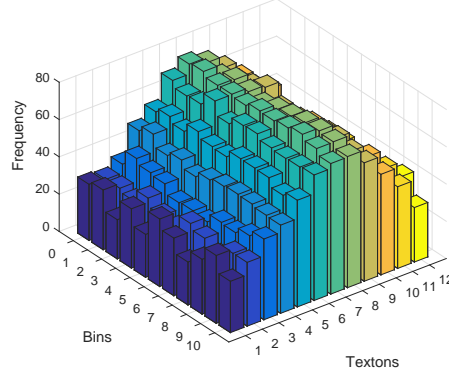
Figure 4.5: 2D Histogram. Each row represents an intensity histogram of grey levels from pixels belonging to the same label.

4.1.2 Classification stage
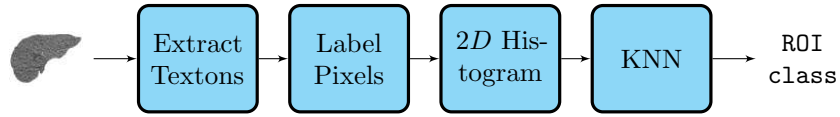
Figure 4.6: Classification Pipeline. An ROI slice is used as input. The pixels of the original ROI are labelled using their distance to the textons dictionary and a 2D histogram is generated, as a model of the ROI. Using the K-Nearest Neighbour (KNN) classifier, the ROI is assigned to the same class as his closest model.

1. A texton is represented by a set of $N^2 - 1$ pixels, in the neighbourhood, $\mathcal{N}$, of a pixel, in the same way as in the previous stage.

2. The pixels of each image on the training set are labelled to the index of the closest texton in the dictionary build on the learning stage.

3. For each of the textons in the dictionary, a one dimensional distribution of the central pixels' intensity is learnt and represented by a 2D histogram, using the same parameters as for the training set.

4. Each model on the testing set is compared against each model on the training set and following the KNN approach, the test image is classified into the same class as it's closest train image. The authors proposed the use of the χ^2 distance, illustrated by Eq. 4.2 as a metric to evaluate the distance between models. We went beyond that and proposed

to use the euclidean distance, presented in Eq. 4.3, given that it characterizes better the degree of proximity between histograms.

$$C(p_i, q_j) = \frac{1}{2} \sum_{k=1}^{K} \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)} \tag{4.2}$$

$$\mathrm{d}(\mathbf{p}, \mathbf{q}) = \mathrm{d}(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} \tag{4.3}$$

4.2 MRF vs. Statistical Methods

In order to compare the proposed method with state of the art algorithms, a series of features were calculated using a Toshiba Medical Visualization Systems (TMVS) command line tool that calculates a wide range of features within a given set of ROI, including those from Gray-Level Co-ocurrence Matrix (GLCM) and Gray-Level Run-Length Matrix (GLRLM) presented in Sections 3.1.1 and 3.1.1 respectively.

Features for Denoised Volumes	
Mean	Percentile 90
Mean Absolute	Percentile 99
Root Mean Square	Interquartile Range
Standard Deviation	GLCM bins32 (1_0_0) SumVariance
Skew	GLCM bins32 (1_1_0) SumVariance
Kurtosis	GLCM bins32 (0_1_0) SumVariance
Median	GLCM bins32 (-1_1_0) SumVariance
Minimum	GLCM bins32 (1_0_0) ClusterTendency
Maximum	GLCM bins32 (1_1_0) ClusterTendency
Range	GLCM bins32 (0_1_0) ClusterTendency
Percentile 1	GLCM bins32 (-1_1_0) ClusterTendency
Percentile 10	RLE bins32 Mean GrayLevelNonUniformity
Percentile 25	RLE bins32 Mean RunLengthNonUniformity
Percentile 75	

Table 4.1: Set of features used for volumes with Adaptive Iterative Dose Reduction 3D (AIDR3D) correction

Table 4.1 summarizes the features used to classify denoised datasets. Among others, it includes statistics of the ROI, such as mean, median and standard deviation. Table 4.2 illustrates the set of features used to classify the noisy datasets. Since the intensity values have a large variation in those volumes, the statistics of the region are not included, instead the Gray Level Non Uniformity calculated from the GLRLM is calculated in different directions.

Features for Noisy Volumes	
RLE bins32 (0_0_1)	RLE bins32 (0_1_1)
RLE bins32 (1_0_1)	RLE bins32 (0_1_0)
RLE bins32 (1_0_0)	RLE bins32 (0_1_-1)
RLE bins32 (1_0_-1)	RLE bins32 (-1_1_1)
RLE bins32 (1_1_1)	RLE bins32 (-1_1_0)
RLE bins32 (1_1_0)	RLE bins32 (-1_1_1)
RLE bins32 (1_1_-1)	RLE bins32 Mean

Table 4.2: Set of features used for volumes without AIDR3D

For the purpose of classification, a *bagged decision tree* was used. It bags an ensemble of decision trees for classification. Every tree in the collection is grown on an independently drawn bootstrap replica of input data. Observations not included in this replica are "out of bag" for this tree. To compute predictions for unseen data, the *bagged decision tree* takes an average of predictions from individual trees [35].

4.3 Segmentation

The problem of classifying images that represents only a single texture is extrapolated to images in which more than one texture is present and it's necessary to identify the pixels that belong to each class.

The initial approach to solve this problem was to label each of the pixels according to the distance of each texton to the dictionary and assign the label according to the texton. To mask the background of the image, we define a distance threshold above which a pixel is considered to be too far from the training set. Any pixel above this threshold is labeled as background. This threshold is initially set to 1.000.

In order to improve the segmentation, different strategies were intended. This section summarizes their main principles.

4.3.1 Parameter Configuration

The first step considered the parameters of the algorithm that could be altered to enhance the result. We considered different neighborhood sizes (3, 5 and 7), as well as different dictionary sizes (10, 20 and 30). All of these settings were tested while keeping the background threshold constant (1000).

4.3.2 Majority Voting Scheme

A Majority Voting Scheme was implemented by comparing the three closest element of the dictionary of textons.

4.3.3 Morphological Operations

The anatomy of human organs are continuous in the space, which means that it is not possible that one organ has holes in the middle of it or that it has some small portions separated. Following this principle, we applied morphological operations, such as erosion and closing to improve the segmentation results by filling the gaps and removing isolated pixels [9].

4.3.4 Adaptive Threshold

The background threshold value (1000) was arbitrarily selected. A better way to determine this value might be selecting it from the training set by using the distance between the textons and the dictionary on each slice and finding different statistics to find the best value for the threshold.

4.3.5 Background Information

Since the background include different organs, such as the heart, the stomach, as well as bones and air, some information about the background was added to the dictionary. In order to achieve this, some slices were randomly selected from the training volumes and the textons of non-class pixels were extracted and clustered with the K-means algorithm to be added to the dictionary.

4.4 Datasets

Two different datasets were used to perform experiments. The first datasets consisted of single textured images, while the other one consisted in CT images.

4.4.1 Universitat de Girona (UdG) Dataset

The single-textured dataset [42] consisted of real textures commonly found in our every day life, such as beans, lentils, pasta, paper, etc., as illustrated in Appendix A.1. In total, 20 different classes compose this dataset and for each class, six different samples are available. They were captured at two different distances, using 6 illuminant tilt angles: 0, 60, 120, 180, 240, 300.

4.4.2 TMVS Datasets

The dataset is composed by 5 patients, identified by sequence 04635 to 04649. For each patient, there are 10 different volumes, which corresponds to different noise levels, and with/without noise reduction (AIDR3D). Table 4.3 summarizes the percentage of dose reduction.

AIDR3D	No AIDR3D
0	0
25	25
50	50
75	75
88	88

Table 4.3: Percentage of dose reduction.

The DICOM volumes corresponds to CT Images in high resolution, of size 512 by 512. The number of slices varied depending on the patients from 572 to 954. Appendix A.2 illustrates an example of one slice of each image.

4.5   Evaluation

In order to evaluate the performance of both implementations, different metrics are used, based on the number of hits and failures in the classification.

		True diagnosis	
		Positive	Negative
Screening test	Positive	TP	FP
	Negative	FN	TN

Figure 4.7: Confusion Matrix

Fig. 4.7 illustrates a Confusion Matrix, elaborated from the statistics presented below [21].

- **True Positives (TP):** Number of elements correctly classified as Positive by the Screening test.

- **False Positives (FP):** Number of elements classified as Positive by the Screening test, but the true diagnosis is that they are Negative.

- **True Negatives (TN):** Number of elements correctly classified as Negative.

- **False Negatives (FN):** Number of elements classified as Negative by the Screening test, when they are actually Positive in the True diagnosis.

4.5.1 Accuracy

Corresponds to the percentage of elements correctly classified, that is to say, the sum of the True Positives and True Negatives, divided by the total number of elements, as represented by Eq. 4.4.

$$ACC = \frac{TP + TN}{P + N} \tag{4.4}$$

4.5.2 Precision

Represents the percentage of elements that are correctly labelled as member of the class, from the total number of elements identified as member of the class. Eq. 4.5 presents the formula to calculate this value.

$$PPV = \frac{TP}{TP + FP} \tag{4.5}$$

4.5.3 Recall

Represents the percentage of elements correctly classified as member of the class, from the actual total number of member of the class. Eq. 4.6 illustrates the formula to compute this value.

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN} \tag{4.6}$$

4.5.4 Dice Coefficient

Also called F1-score, is the harmonic mean of precision and recall [21]. Eq. 4.7 illustrates the formula to calculate. The most important characteristic of this metric is its independence of the true negatives, which made it quite representative in cases when the background is too big, as occurs with CT images.

$$F1 = \frac{2\,TP}{2\,TP + FP + FN} \tag{4.7}$$

Chapter 5

Results

This chapter presents the results obtained when performing different experiments that use the methodology illustrated in Chapter 4.

5.1 Texture Classification

Using the Universitat de Girona (UdG) dataset, the images were divided into training set (4 per class) and testing set (2 per class) and classified using the implementation of Varma & Zisserman presented in Section 4.1. Three different parameters were considered in the experiments, in order to validate their influence in the algorithm's performance. The evaluation used the accuracy metric defined in Section 4.5.1.

Neighbours	Bins	Centroids	Accuracy
3	256	10	95
3	128	10	83.75
3	256	20	95
5	256	10	**97.5**
5	128	10	82.5
5	256	20	96.25
7	256	10	96.25
7	128	10	82.5
7	256	20	**97.5**

Table 5.1: Experiments on UdG Dataset

Table 5.1 summarizes the parameters used for each experiments, as well as the accuracy obtained, which is graphically presented in Fig. 5.1. The best results are obtained when the neighbourhood size is 5 with a dictionary of 10 textons per class, with a **97.5%** accuracy, which is actually the same as when the neighbourhood size is 7 and the number of centroids is 20.
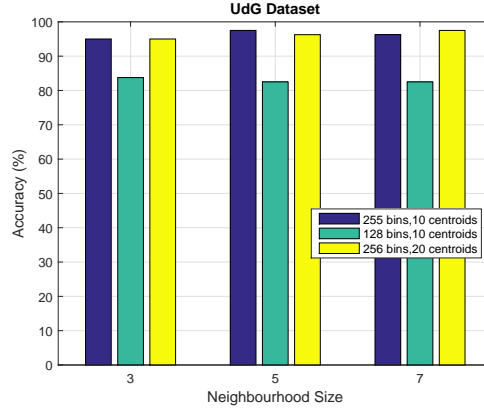
Figure 5.1: Accuracy for the experiments on UdG Dataset

5.2 Regions of Interest Classification

In [41], an experiment on the San Francisco Database was performed using segmentations made by hand in [14]. The segmentations corresponded to six different classes: Air, Building, Car, Road, Vegetation, and Trunk. The objective was to classify each region into one of the classes.

The same experiment was proposed in this work, using the Markov Random Field (MRF) classifier introduced in section 4.1. In this case, the Region of Interest (ROI), were segmented by a clinician into four different classes: Kidney, Liver, Lung and Spleen. Table 5.2 presents the number of slices of each organ for each patient.

Patient	Kidney	Liver	Lung	Spleen
04635	164	415	302	337
04636	155	370	167	-
04637	106	290	167	253
04638	88	299	175	182
04639	96	191	195	139

Table 5.2: Slices per organ for each patient

5.2.1 Experiments on Denoised Volumes

This section presents the results of a set of experiments, in which 3 volumes corresponding to the *0 dose without Adaptive Iterative Dose Reduction 3D (AIDR3D) reconstruction* were used as a training, while 2 volumes *with AIDR3D reconstruction*, with noise intensities 0, 25, 50, 75 and 88 were used for testing.

χ^2 **Distance**

Following the same methodology as in the original paper by Varma & Zisserman, the χ^2 distance was used as a metric to compare 2D histograms and proceed to classify the ROIs. Fig. 5.2 illustrates the results for a neighbourhood of size 3. Fig. 5.2a corresponds to the instance when the dictionary has 10 textons per class, while Fig. 5.2b illustrates the case for 20 textons. Appendix B.1.1 presents the values corresponding to this graphs in Table B.1 and Table B.2.

From the graphs can be clearly seen that the use of more centroids decreases the accuracy of the algorithm, specially for the kidney. One reason for this behaviour might be the introduction of noisy textons to the dictionary, that leads to miss-classifications. It is also clearly seen that the classification of lung tissue has the best results, scoring an accuracy of 100% on each case. Moreover, kidney and spleen are also fairly differentiated from other tissues. The main problem is then, to classify liver tissue, which is in most of the cases confused with spleen.
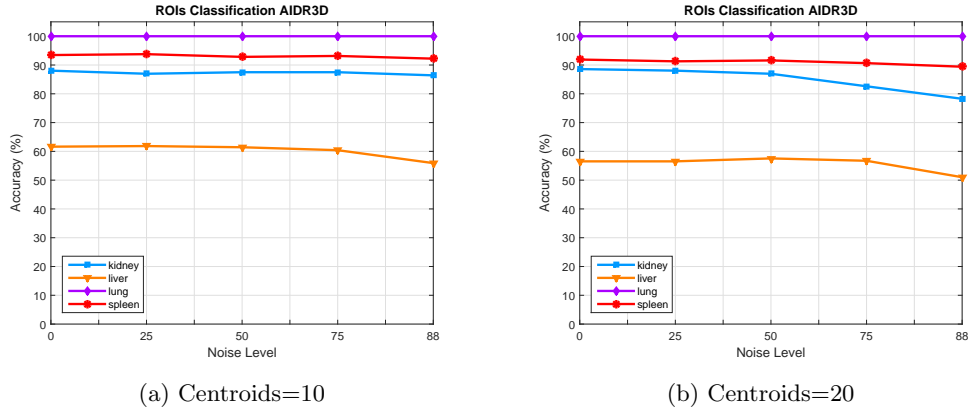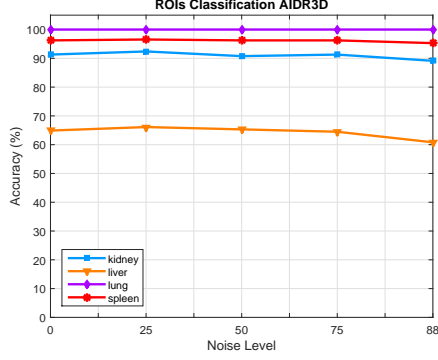

(a) Centroids=10   (b) Centroids=20

Figure 5.2: **Experiments on Denoised Volumes:** χ^2 *Distance*, Neighbourhood=3

Fig. 5.3 illustrates the same experiment, but using neighbourhoods of size 5. Table B.3 and Table B.4, included in Appendix B.1.1, show the results for these experiments. Fig. 5.3a exhibits the diagram for 10 textons per class, while Fig. 5.3b does the same for 20 centroids. Similarly as in the previous case, better results are obtained when the dictionary has 10 centroids, as when it has 20.

Comparing the size of the neighbourhood, from the experiments results can be established that the use of a larger neighbourhood improves slightly the accuracy of the algorithm. This might be due to the fact that a smaller neighbourhood is more sensitive to noise than a bigger one, which leads to misclassification. In some cases however, results are very consistent, such as in the case of the lungs, which always score 100% accuracy, while the liver always presents the poorest performance, achieving no more than 65% in the best case.
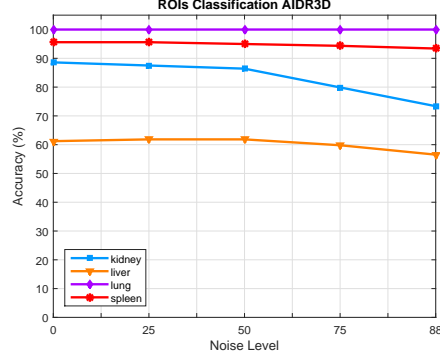
(a) Centroids=10 (b) Centroids=20

Figure 5.3: **Experiments on Denoised Volumes:** χ^2 *Distance*, Neighbourhood=5

Euclidean Distance

Euclidean distance was proposed as an alternative metric to compare the distance between 2D histograms. Fig. 5.4 illustrates the results for a neighbourhood of size 3, which corresponds to the values presented in Table B.5 and Table B.6 in Appendix B.1.2.
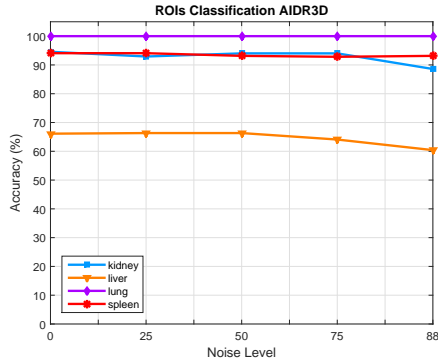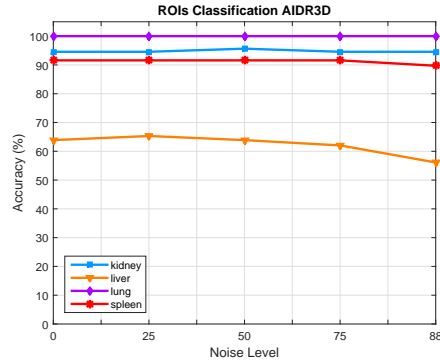
(a) Centroids=10 (b) Centroids=20

Figure 5.4: **Experiments on Denoised Volumes:** *Euclidean Distance*, Neighbourhood=3

The use of the Euclidean distance enhance the results for all the organs. The use of more neighbours has a positive impact on the results as well, as can be seen in Fig. 5.5. When 10 centroids are used, the accuracy of all the organs (except liver) is above 90%, as illustrated in Fig. 5.5a. The use of more centroids has proved to have a negative impact on the results, as it happen when using the χ^2 distance. Appendix B.1.2 shows this results in Table B.7 and Table B.8.

(a) Centroids=10

(b) Centroids=20
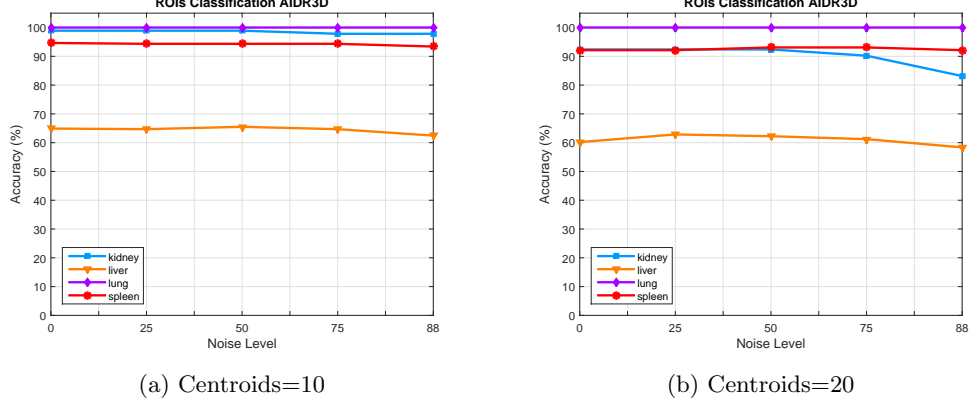
Figure 5.5: **Experiments on Denoised Volumes:** *Euclidean Distance*, Neighbourhood=5

5.2.2 Experiments on Noisy Volumes

This set of experiments, in contrast to the Experiments on Denoised Volumes presented in Section 5.2.1, uses as a training 3 different volumes corresponding to the *0 dose with AIDR3D reconstruction*, while *2 volumes of doses 0, 25, 50, 75 and 88 without AIDR3D reconstruction* were used for testing.

χ^2 Distance

Following the approach of the Experiments on Denoised Volumes, Fig. 5.6 illustrates the accuracy obtained per organ for a neighbourhood of size 3. Appendix B.2.1 presents the values obtained in Table B.9 and Table B.10, corresponding to Fig. 5.6a and Fig. 5.6b respectively.

(a) Centroids=10

(b) Centroids=20

Figure 5.6: **Experiments on Noisy Volumes:** χ^2 *Distance*, Neighbourhood=3

Similarly as what occurred in the previous experiments, the use of more textons on each class of the dictionary decreases the accuracy of the algorithm, except in the last experiment, 88 noise, where the use of a larger dictionary improves the accuracy. Even though most of the organs accuracy decreases significantly when the noise increases, the lung classification remains scoring 100%.

Fig. 5.7a presents the results for a neighbourhood of size 5, using 10 centroids 5.7a and 20 5.7b. The use of a larger neighbourhood increases the accuracy notably, specially for the noisiest images. In this case, the use of more textons in the dictionary has only a positive impact on the noisiest dataset (88). The accuracy values of these experiments can be found in Appendix B.2.1, illustrated by Table B.11 and Table B.12.

(a) Centroids=10   (b) Centroids=20
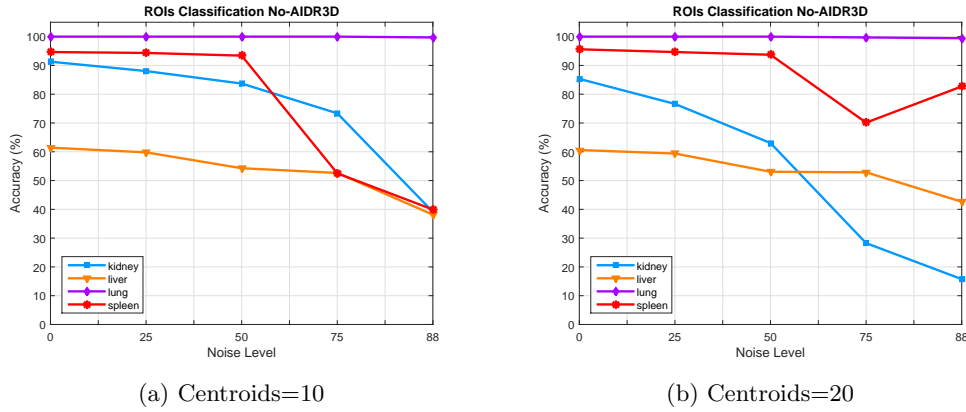
Figure 5.7: **Experiments on Noisy Volumes:** χ^2 *Distance*, Neighbourhood=5

Euclidean Distance

The Euclidean distance was also included in this set of experiments. Fig. 5.8 illustrates the results for a neighbourhood of size 3, using 10 (Fig. 5.8a) and 20 (Fig. 5.8b) textons per class on the dictionary. Appendix B.2.2 shows the corresponding values in Table B.13 and Table B.14.

Similarly as what occurred in the experiments on Denoised Volumes, the use of the euclidean distance enhanced the results, however, in this case the difference between both accuracies is larger, achieving in some cases more than 30% difference, such as in the case of the liver with 88 noise. On the other hand, the accuracy of classifying the spleen tissue decreases considerably using the euclidean distance, compared with the accuracy obtained using χ^2 distance. The lungs are consistent and achieve 100% in every experiment.

Fig. 5.9 shows the results obtained for a neighbourhood of size 5. Fig. 5.9a presents the accuracy obtained using 10 textons per class in the dictionary and Fig. 5.9b illustrates the
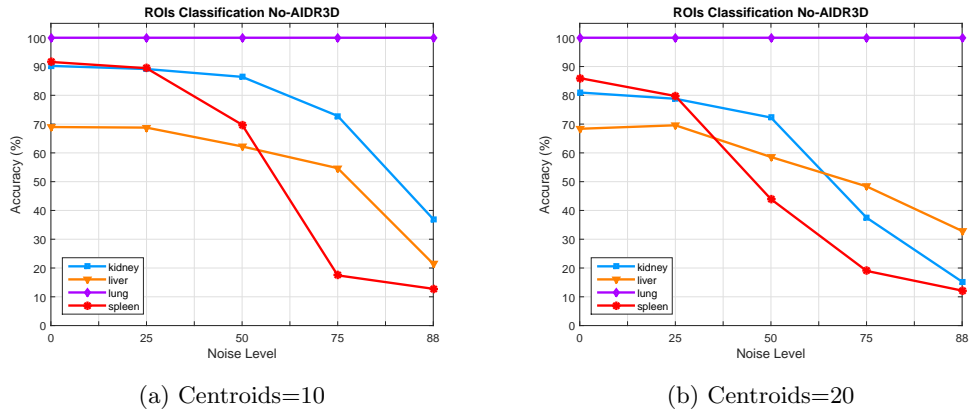
(a) Centroids=10   (b) Centroids=20
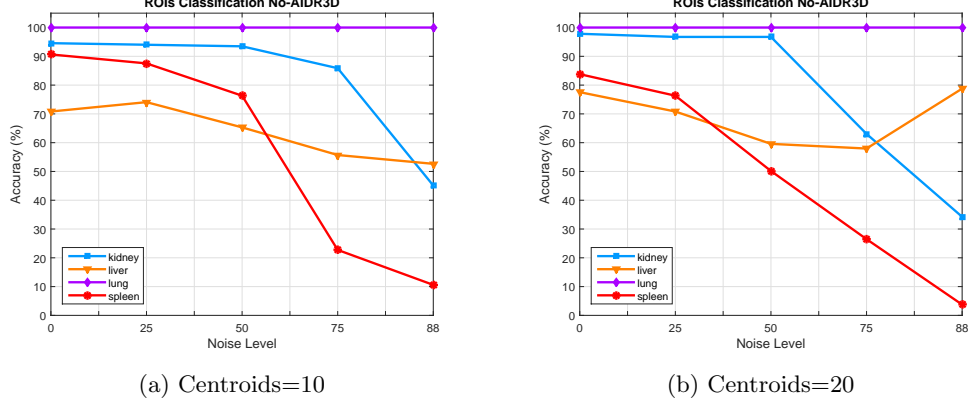
Figure 5.8: **Experiments on Noisy Volumes:** *Euclidean Distance*, Neighbourhood=3

results for 20 textons. The use of a larger neighbourhood improves the accuracy for all the organs. Using 10 centroids, all the results are above 50%, which is an improvement with respect to the χ^2 results, specially for the noisiest datasets.

(a) Centroids=10   (b) Centroids=20
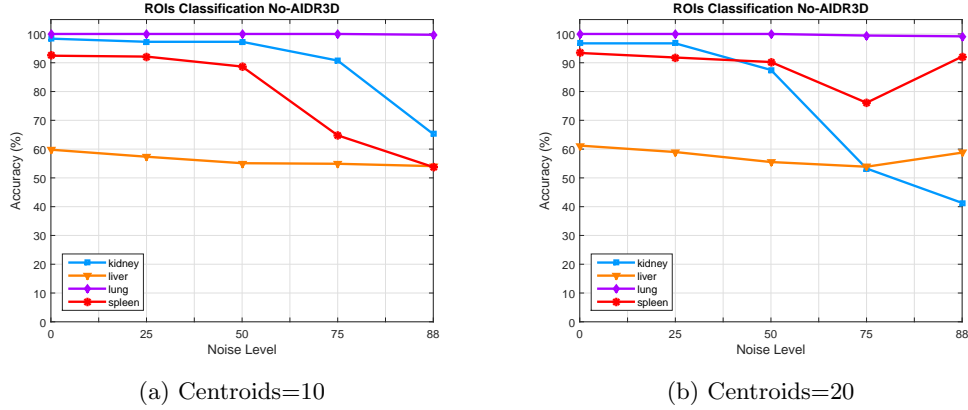
Figure 5.9: **Experiments on Noisy Volumes:** *Euclidean Distance*, Neighbourhood=5

5.3   MRF vs. Statistical Methods

This section presents the results of the classification using statistical methods, following the methodology described in section 4.2. Fig. 5.10 presents the results per organ, using statistical methods, compared to the best results obtained using MRF with χ^2 and euclidean distance, in the Experiments of on Denoised Volumes. In Fig. 5.10a the results for the kidney are presented,

where can be clearly seen that the MRF with euclidean distance presents better results than the other methods. In Fig. 5.10b are illustrated the results for the liver, which presents similar performance in the noisiest datasets. Similarly, the lungs are perfectly discriminated in the three cases, as presented by Fig. 5.10c. Finally, the spleen presents similar results for the three classifiers, with a slightly superiority of the χ^2 distance, as can be seen in Fig. 5.10d. In conclusion, when performing experiments of on Denoised Volumes, the three different approaches presents similar performance, with the exception of the kidney, in which the use of the MRF classifier outperforms the other results.

(a) Kidneys   (b) Liver

(c) Lungs   (d) Spleen
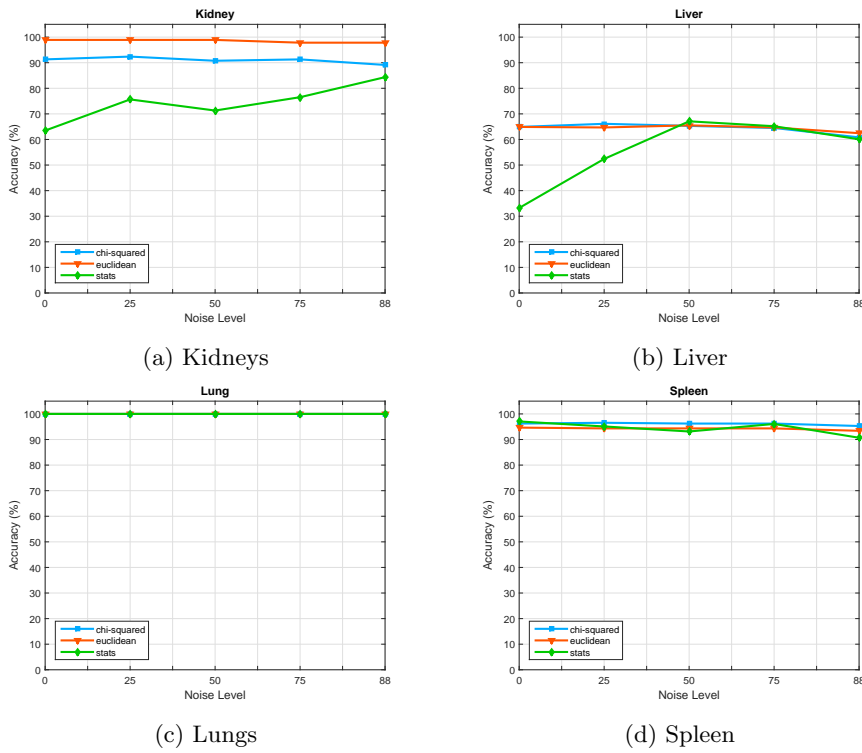
Figure 5.10: **Experiments on Denoised Volumes:** Comparison

Fig. 5.11 presents the comparison for the Experiments of on Noisy Volumes. Fig. 5.11a presents the results for the kidney, which shows clearly the superiority of the MRF methods over the statistical ones. In the case of the liver, Fig. 5.11b illustrates how similar is the behaviour of the three methods, however for the noisiest datasets, the euclidean distance overcomes the results of the other methods. The lungs are very consistent for the MRF methods, in comparison with the Experiments of on Denoised Volumes, however for the statistical methods, the performance is very inferior, compared to the one obtained on the previous experiments, as Fig. 5.11c shows.

Something similar happen with the spleen, which also decreases its performance, as illustrated by Fig. 5.11d. In conclusion, we can see that in this type of experiments, the performance of the statistical features decreases and it's outperformed by the MRF methods.

(a) Kidneys

(b) Liver

(c) Lungs

(d) Spleen
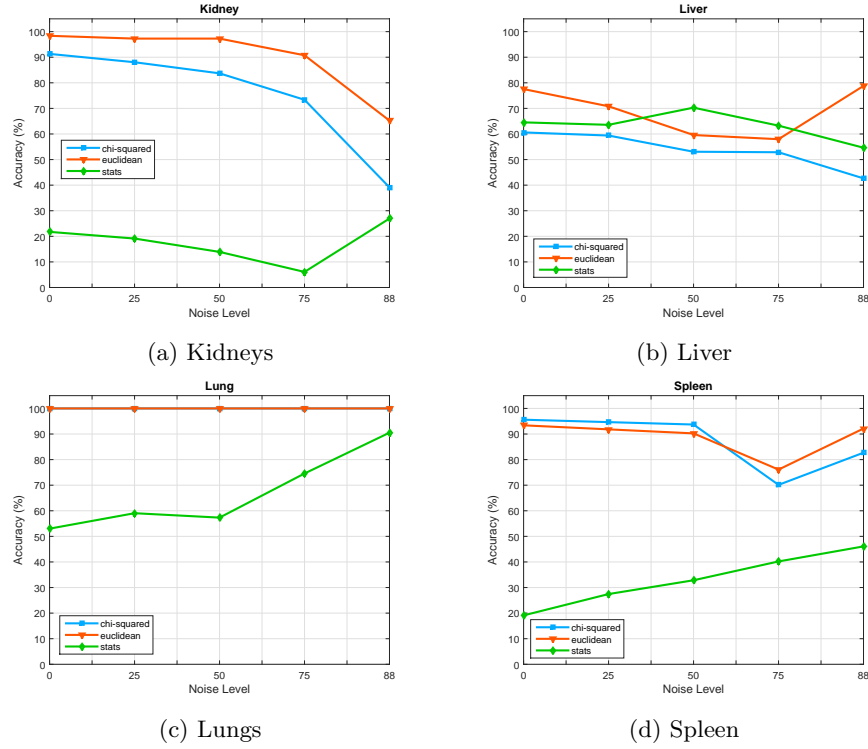
Figure 5.11: **Experiments on Noisy Volumes:** Comparison

5.4 Segmentation

This section presents the results obtained when carrying out the segmentation methodology proposed in Section 4.3. Fig. 5.14, 5.15 and 5.16 illustrates the results of the experiments described in Section 4.3.1. They demonstrated that the use of a larger neighbourhood with a smaller dictionary size increases the dice coefficient obtained per class. It affects some organs more drastically than others, as in the case of the kidneys, which performs very poorly when the size of the neighbourhood is small. One reason for this behaviour might the influence that the noise have in a small neighbourhood, which is reduced when a larger neighbourhood is used, as there are more pixels to take into account.

(a) Liver   (b) Lung
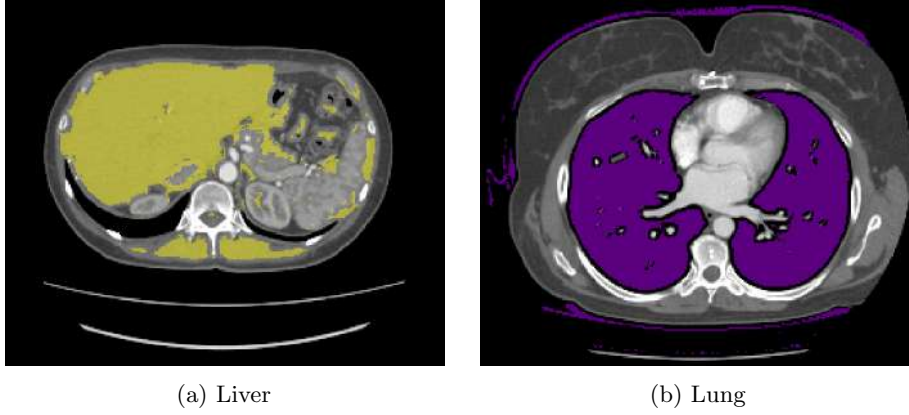
Figure 5.12: Segmentations of large ROI

Fig. 5.17 presents the results corresponding to the approach described in Section 4.3.2, using two different configurations: a neighbourhood of size 5 and one of size 7, with 10 textons per class on the dictionary. The box plots shows better results for the larger neighbourhood, since larger dice coefficients are achieved, however the difference is not very large and for some organs such as lungs, the use of a neighbourhood of size 5 improves the results.

(a) Kidney   (b) Spleen
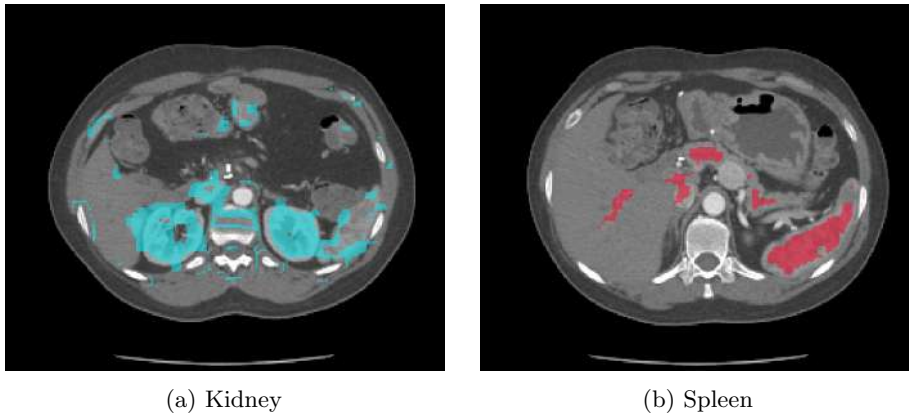
Figure 5.13: Segmentations of small ROI

Fig. 5.18 presents the results obtained when performing the morphological operations introduced in Section 4.3.3. The results shows that performing an erosion followed by a closing improves the results, specially by increasing the minimum value in the kidneys and the spleen.

Fig. 5.19 illustrates the results for the experiments described in Section 4.3.4. The use of an adaptive threshold improved the results in general, specially for the liver. The addition of

morphological operations to those results however deteriorated drastically the results for the lungs, but at the same time improved them for liver.

Fig. 5.20 and 5.21 exhibits the results when applying the approach proposed in Section 4.3.5. The use of textons that belongs to the background in the dictionary was a right guess, however it did not have the expected impact, since the background is too large and heterogeneous. The use of a majority voting scheme had a negative effect on the dice coefficient obtained.

Fig. 5.12 illustrates an example of the segmentation of the largest organs, lungs and liver. It can be seen from the figures that even though most of the area covered by the ROI is correctly recognized, there also a large number of false positives that reduce the dice coefficient. The main problem with lungs is the misclassification with background, while the liver is being confused with other soft tissues, such as stomach. Fig. 5.13 on the other hand, presents the segmentation results for the smallest ROI, kidney and spleen. In this case, there are also a considerable number of false positives for the kidney image, but a large number of false negatives in the case of the spleen, from voxels that are normally classified as liver.
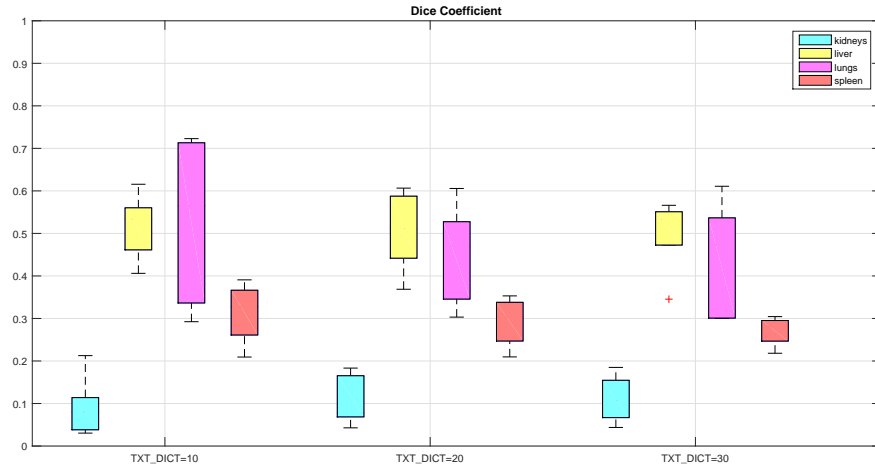

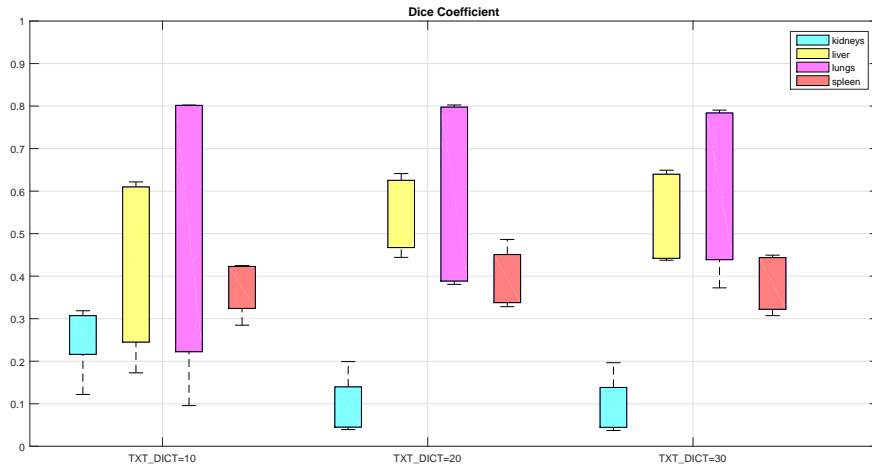Figure 5.14: Neighbourhood=3, Threshold=1000
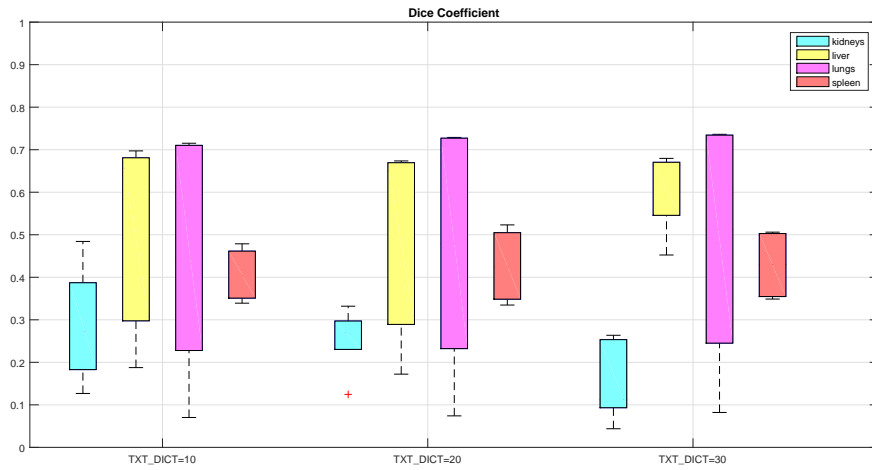
Figure 5.15: Neighbourhood=5, Threshold=1000
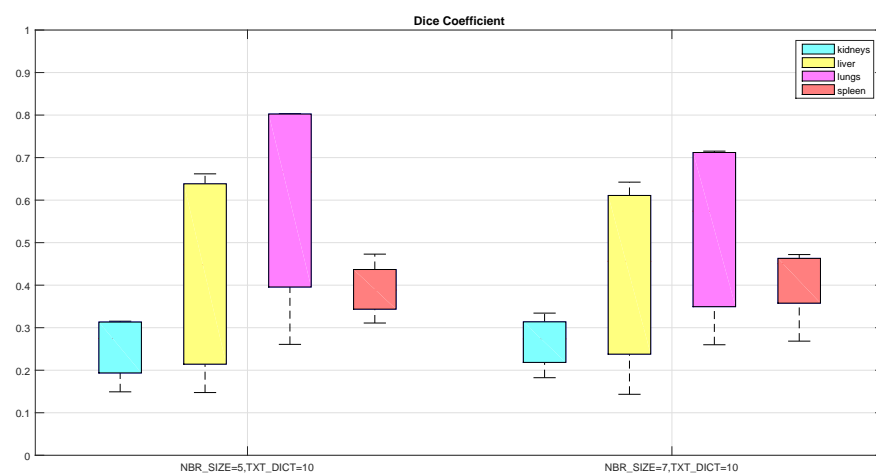

Figure 5.16: Neighbourhood=7, Threshold=1000
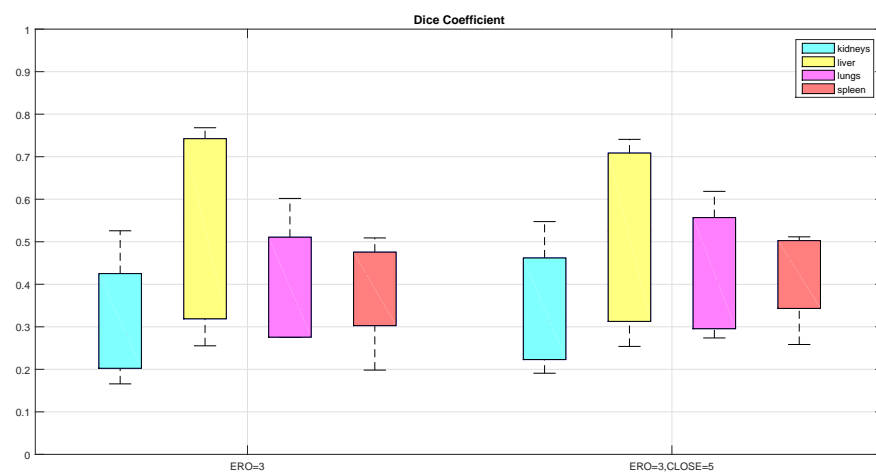
Figure 5.17: Majority Voting Scheme

Figure 5.18: Morphological Operations

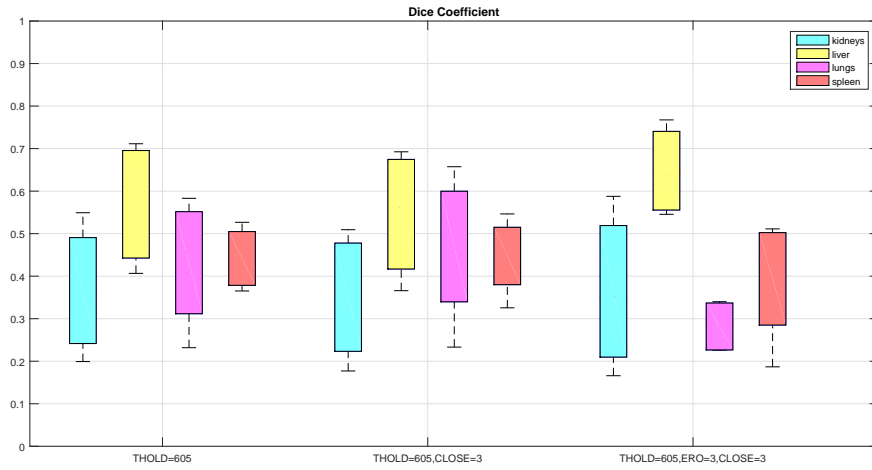Figure 5.19: Adaptive Threshold
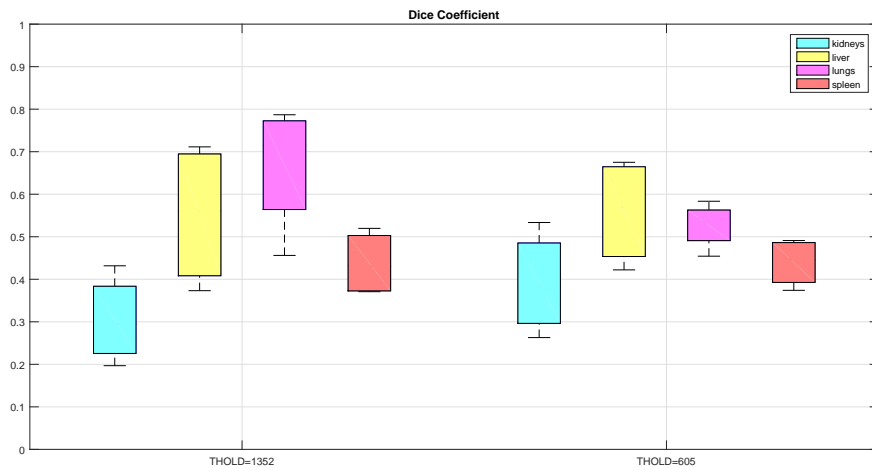
Figure 5.20: Background Textons
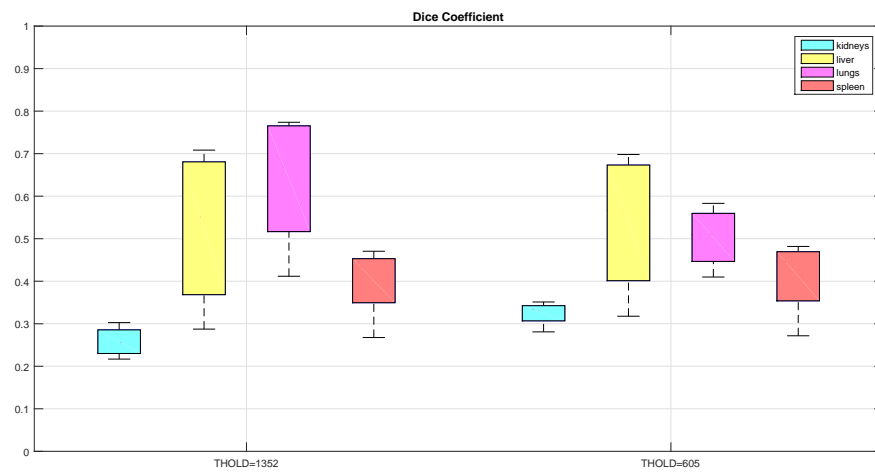
Figure 5.21: Background Textons using Majority Voting Scheme

Chapter 6

Conclusions

6.1 Discussion

This study implemented and evaluated Markov Random Field (MRF) texture analysis for use in Computer Tomography (CT) images. As presented in the literature review, there have been quite a number of studies using texture features to diagnose diseases from medical imaging. However, there has been some recent concern about applying the results from one institution to another, due to variations in imaging techniques and feature measurement [7]. In this study we focused on the effect on the texture measures of both the image noise itself and the noise reduction algorithm. The results showed that the texture metrics were reassuringly robust to the noise level and Adaptive Iterative Dose Reduction 3D (AIDR3D) noise reduction. In high quality images the noise reduction had little effect on the detection accuracy, reducing it slightly from 87.8% to 85.8%. In contrast, in noisy, lower dose datasets the noise reduction had a more noticeable effect, improving accuracy from 75.1% to 84.5%. This results were compared to the accuracy obtained when using statistical methods and the results showed that our method is more robust when is used in noisy datasets.

The results also demonstrated the importance of the metric used to compare different histograms. Varma & Zisserman proposed using the χ^2 distance. The χ^2 metric is used to determine whether two histograms are samples from the same statistical distribution. However, this is subtly different from a metric, such as the Euclidean distance, which indicates how different two histogram distributions are. For this project, the latter type of metric seems more appropriate, and indeed was shown to perform consistently better on these datasets.

Classification of individual pixels had poorer performance than region classification. In general, the use of larger neighbourhoods, smaller dictionary sizes, morphological closing and background textons presented the best results, however the segmented images still presented a high rate of false positives. There are a number of reasons for this. The first is that calculating

the class of a large region is much less prone to noise than for a single pixel. The second is that pixel classification has to also account for background regions. The wide range of anatomical variation outside the organs of interest was hard to represent sufficiently in a small texton dictionary, and consequently there were a lot of false positives in the background region that adversely affected the dice coefficient results.

6.2   Future Work

Further work can be undertaken to improve the classification and segmentation results, as well as to evaluate more potential applications of this project.

The first and most obvious step is the use of 3D texton neighbourhoods. Studies involving CT image and texture algorithms, such as the greylevel co-occurrence matrix, have shown a small but significant improvement in accuracy using 3D regions. However, it is likely to be more computationally expensive. Care also needs to be taken when analysing anisotropic volumes (i.e. where the voxels are not cubic).

Another possible improvement is the inclusion of rotationally invariant features. This could be implemented quite simply by including circularly shifted versions of the measured textons in the dictionary. However, this was not a priority in this project, as the careful alignment of patients for a CT scan means that textures can reasonably be expected to be consistently orientated.

A more complex addition might be the use of a sparse modelling approach to find representative objects [6]. In summary, this approach consists in finding a few representative for a dataset to form a dictionary, assuming that each data point can be expressed as a linear combination of the representatives. The problem is formulated as a sparse multiple measurement vector problem, solved via complex optimization.

Lastly, future work should evaluate the use of these features in differentiating healthy and unhealthy tissue. This would have a lot of potential applications. However, it is also likely to be much harder than differentiating organs, as the changes are likely to be more subtle.

Appendix A

Datasets

A.1 Universitat de Girona (UdG) Dataset

Figure A.1: UdG Dataset: 20 different classes composed by 6 different samples each, captured at 2 distances using 6 illuminant tilt angles.

A.2 Toshiba Medical Visualization Systems (TMVS) Datasets

(a) 0 No AIDR3D

(b) 25 No AIDR3D

(c) 50 No AIDR3D

(d) 75 No AIDR3D

(e) 88 No AIDR3D
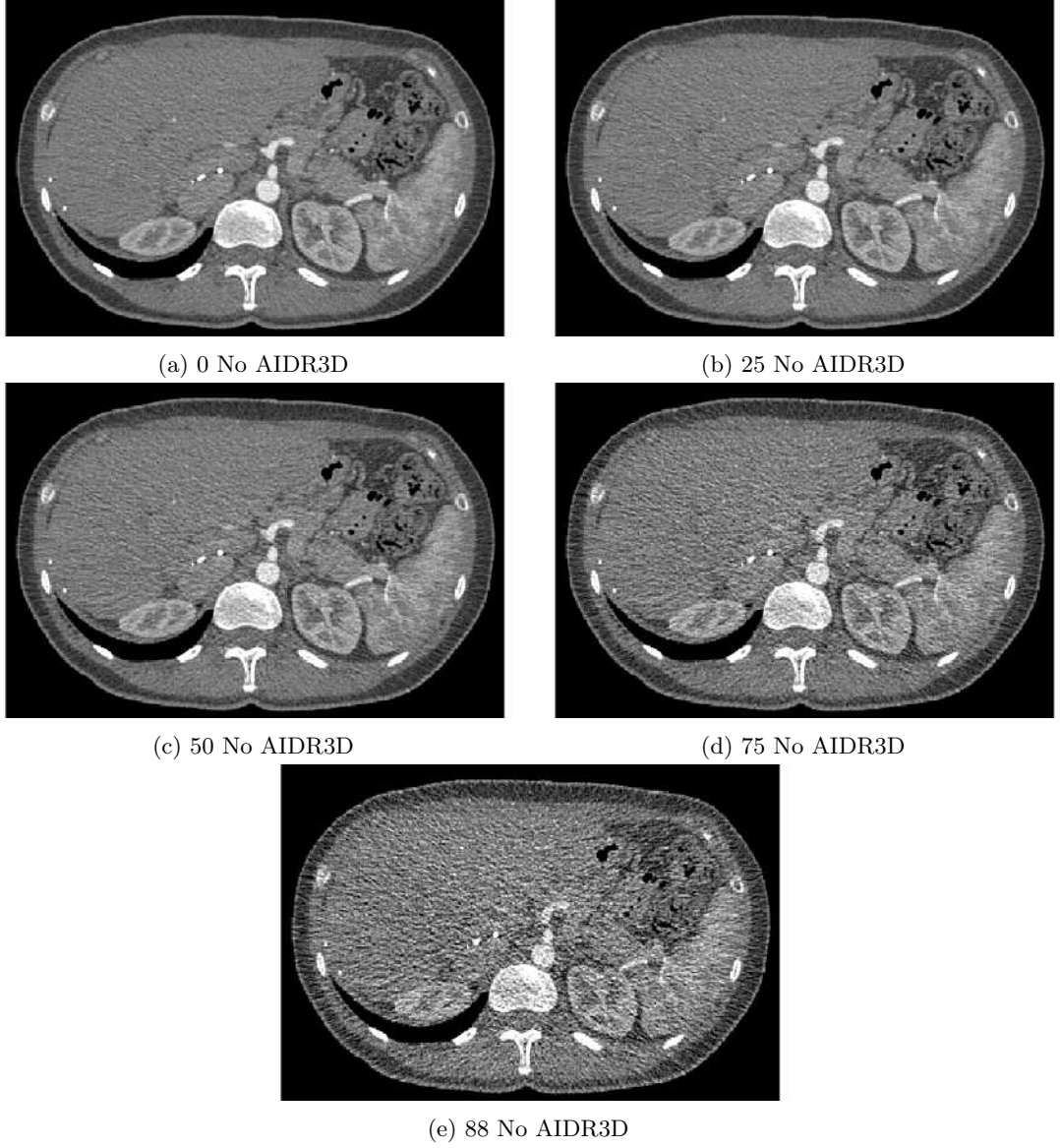
Figure A.2: TMVS Dataset without Adaptive Iterative Dose Reduction 3D (AIDR3D) reconstruction: Five subjects, each with five simulated low doses: 0, 25, 50, 75 and 88.

(a) 0 AIDR3D

(b) 25 AIDR3D

(c) 50 AIDR3D

(d) 75 AIDR3D

(e) 88 AIDR3D
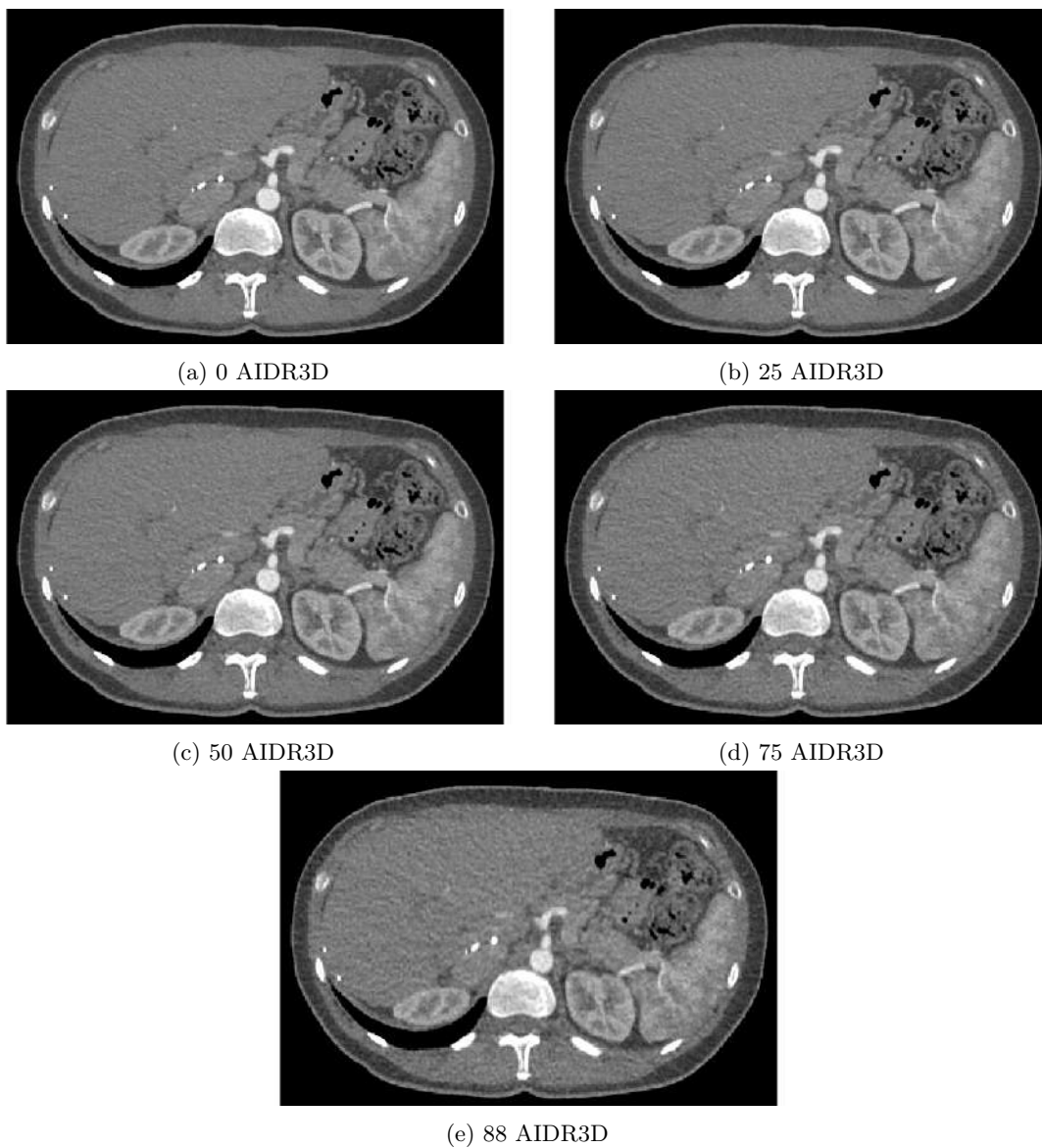
Figure A.3: TMVS Dataset with AIDR3D reconstruction: Five subjects, each with five simulated low doses: 0, 25, 50, 75 and 88.

Appendix B

Regions of Interest Classification

B.1 Experiments on Denoised Volumes

B.1.1 χ^2 Distance

Organ	0	25	50	75	88
kidney	88.04	86.96	87.5	87.5	86.41
liver	61.63	61.84	61.43	60.41	55.92
lung	100	100	100	100	100
spleen	93.46	93.77	92.83	93.15	92.21

Table B.1: χ^2 *Distance:* Neighbourhood=3,Centroids=10

Organ	0	25	50	75	88
kidney	88.59	88.04	86.96	82.61	78.26
liver	56.53	56.53	57.55	56.73	51.02
lung	100	100	100	100	100
spleen	91.9	91.28	91.59	90.65	89.41

Table B.2: χ^2 *Distance:* Neighbourhood=3,Centroids=20

Organ	0	25	50	75	88
kidney	91.3	92.39	90.76	91.3	89.13
liver	64.9	66.12	65.31	64.49	60.82
lung	100	100	100	100	100
spleen	96.23	96.54	96.23	96.23	95.28

Table B.3: χ^2 *Distance:* Neighbourhood=5,Centroids=10

Organ	0	25	50	75	88
kidney	88.59	87.5	86.41	79.89	73.37
liver	61.22	61.84	61.84	59.8	56.53
lung	100	100	100	100	100
spleen	95.6	95.6	94.97	94.34	93.4

Table B.4: χ^2 *Distance:* Neighbourhood=5,Centroids=20

B.1.2   Euclidean Distance

Organ	0	25	50	75	88
kidney	94.57	92.93	94.02	94.02	88.59
liver	66.12	66.33	66.33	64.08	60.41
lung	100	100	100	100	100
spleen	94.08	94.08	93.15	92.83	93.15

Table B.5: *Euclidean Distance:* Neighbourhood=3,Centroids=10

Organ	0	25	50	75	88
kidney	94.57	94.57	95.65	94.57	94.57
liver	63.88	65.31	63.88	62.04	56.12
lung	100	100	100	100	100
spleen	91.59	91.59	91.59	91.59	89.72

Table B.6: *Euclidean Distance:* Neighbourhood=3,Centroids=20

Organ	0	25	50	75	88
kidney	98.91	98.91	98.91	97.83	97.83
liver	64.9	64.69	65.51	64.69	62.45
lung	100	100	100	100	100
spleen	94.65	94.34	94.34	94.34	93.4

Table B.7: *Euclidean Distance:* Neighbourhood=5,Centroids=10

Organ	0	25	50	75	88
kidney	92.39	92.39	92.39	90.22	83.15
liver	60.2	62.86	62.24	61.22	58.37
lung	100	100	100	100	100
spleen	92.14	92.14	93.08	93.08	92.14

Table B.8: *Euclidean Distance:* Neighbourhood=5,Centroids=20

B.2 Experiments on Noisy Volumes

B.2.1 χ^2 Distance

Organ	0	25	50	75	88
kidney	90.22	89.13	86.41	72.83	36.96
liver	68.98	68.78	62.24	54.69	21.43
lung	100	100	100	100	100
spleen	91.59	89.41	69.78	17.45	12.77

Table B.9: χ^2 *Distance:* Neighbourhood=3,Centroids=10

Organ	0	25	50	75	88
kidney	80.98	78.8	72.28	37.5	15.22
liver	68.37	69.59	58.57	48.37	32.86
lung	100	100	100	100	100
spleen	85.98	79.75	43.93	19	12.15

Table B.10: χ^2 *Distance:* Neighbourhood=3,Centroids=20

Organ	0	25	50	75	88
kidney	91.3	88.04	83.7	73.37	39.13
liver	61.43	59.8	54.29	52.65	38.16
lung	100	100	100	100	99.72
spleen	94.65	94.34	93.4	52.52	39.94

Table B.11: χ^2 *Distance:* Neighbourhood=5,Centroids=10

Organ	0	25	50	75	88
kidney	85.33	76.63	63.04	28.26	15.76
liver	60.61	59.39	53.06	52.86	42.65
lung	100	100	100	99.72	99.45
spleen	95.6	94.65	93.71	70.13	82.7

Table B.12: χ^2 *Distance:* Neighbourhood=5,Centroids=20

B.2.2   Euclidean Distance

Organ	0	25	50	75	88
kidney	94.57	94.02	93.48	85.87	45.11
liver	70.82	74.08	65.31	55.71	52.65
lung	100	100	100	100	100
spleen	90.65	87.54	76.32	22.74	10.59

Table B.13: *Euclidean Distance:* Neighbourhood=3,Centroids=10

Organ	0	25	50	75	88
kidney	97.83	96.74	96.74	63.04	34.24
liver	77.55	70.82	59.59	57.96	78.78
lung	100	100	100	100	100
spleen	83.8	76.32	50.16	26.48	3.738

Table B.14: *Euclidean Distance:* Neighbourhood=3,Centroids=20

Organ	0	25	50	75	88
kidney	98.37	97.28	97.28	90.76	65.22
liver	59.8	57.35	55.1	54.9	54.08
lung	100	100	100	100	99.72
spleen	92.45	92.14	88.68	64.78	53.77

Table B.15: *Euclidean Distance:* Neighbourhood=5,Centroids=10

Organ	0	25	50	75	88
kidney	96.74	96.74	87.5	53.26	41.3
liver	61.22	58.98	55.51	53.88	58.78
lung	100	100	100	99.45	99.18
spleen	93.4	91.82	90.25	76.1	92.14

Table B.16: *Euclidean Distance:* Neighbourhood=5,Centroids=20

Bibliography

[1] Generic multivariate gaussian kernel in any derivative order. `https://suinotes.wordpress.com/2010/05/27/generic-multivariate-gaussian-kernel-in-any-derivative-order/`, 2010. Accessed: 2015-05-19.

[2] Alexander Barley and Christopher Town. Combinations of feature descriptors for texture image classification. *Journal of Data Analysis and Information Processing*, 2014(August):67–76, 2014.

[3] Robert A Brown and Richard Frayne. A comparison of texture quantification techniques based on the fourier and s transforms. *Medical physics*, 35(11):4998–5008, 2008.

[4] Lucia Dettori and Lindsay Semler. A comparison of wavelet, ridgelet, and curvelet-based texture classification algorithms in computed tomography. *Computers in Biology and Medicine*, 37(4):486–498, 2007.

[5] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification.* John Wiley & Sons, 2012.

[6] Ehsan Elhamifar, Guillermo Sapiro, and Rene Vidal. See all by looking at a few: Sparse modeling for finding representative objects. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1600–1607. IEEE, 2012.

[7] Xenia Fave, Molly Cook, Amy Frederick, Lifei Zhang, Jinzhong Yang, David Fried, Francesco Stingo, and Laurence Court. Preliminary investigation into sources of uncertainty in quantitative imaging features. *Computerized Medical Imaging and Graphics*, 2015.

[8] David A Forsyth and Jean Ponce. A modern approach. *Computer Vision: A Modern Approach*, 2003.

[9] Rafael C Gonzalez, Richard Eugene Woods, and Steven L Eddins. *Digital image processing using MATLAB*. Pearson Education India, 2004.

[10] Simona E Grigorescu, Nicolai Petkov, and Peter Kruizinga. Comparison of texture features based on gabor filters. *Image Processing, IEEE Transactions on*, 11(10):1160–1167, 2002.

[11] Dorit S Hochbaum. Multi-label markov random fields as an efficient and effective tool for image segmentation, total variations and regularization. *Numerical Mathematics: Theory, Methods and Applications*, 6(1):169–198, 2013.

[12] R Irwan, S Nakanishi, and Alain Blum. AIDR 3D - Reduces Dose and Simultaneously Improves Image Quality. *Toshiba Medical Systems*, pages 1–8, 2011.

[13] Zoltan Kato and Ting-Chuen Pong. A markov random field image segmentation model for color textured images. *Image and Vision Computing*, 24(10):1103–1114, 2006.

[14] Scott Konishi and Alan L Yuille. Statistical cues for domain specific image segmentation with performance analysis. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 125–132. IEEE, 2000.

[15] Hegadi R. S. Kulkarni U. P. Kulkarni S. B., Shirodkar S. Iris recognition using fusion of gray level co-occurrence matrix and gray level run length matrix. pages 241–246, 2013.

[16] Stan Z Li. Markov random field modeling in image analysis. 2009.

[17] Renaud Lopes and Nacim Betrouni. Fractal and multifractal analysis: a review. *Medical image analysis*, 13(4):634–649, 2009.

[18] R Duncan Luce and Albert D Perry. A method of matrix analysis of group structure. *Psychometrika*, 14(2):95–116, 1949.

[19] Stavroula G Mougiakakou, Ioannis K Valavanis, Alexandra Nikita, and Konstantina S Nikita. Differential diagnosis of ct focal liver lesions using texture features, feature selection and ensemble driven classifiers. *Artificial Intelligence in Medicine*, 41(1):25–37, 2007.

[20] Karen M Mudry, Robert Plonsey, and Joseph D Bronzino. *Biomedical Imaging*, volume 10. Crc Press, 2003.

[21] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[22] Rodrigo Nava, Boris Escalante-Ramírez, and Gabriel Cristóbal. Texture image retrieval based on log-gabor features. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 414–421. Springer, 2012.

[23] JDB Nelson. Fused lasso and rotation invariant autoregressive models for texture classification. *Pattern Recognition Letters*, 34(16):2166–2172, 2013.

[24] Maria Petrou and Pedro Garcia Sevilla. *Image Processing: Dealing with Texture*. John Wiley and Sons, 2006.

[25] A. Walker R. Fisher, S. Perkins and E. Wolfart. Laplacian/laplacian of gaussian. `http://homepages.inf.ed.ac.uk/rbf/HIPR2/log.htm`, 2003. Accessed: 2015-05-16.

[26] Radiomics.org. About radiomics. `http://www.radiomics.org/?q=node/4`, 2015. Accessed: 2015-05-27.

[27] JV Raja, M Khan, VK Ramachandra, and O Al-Kadi. Texture analysis of ct images in the characterization of oral cancers involving buccal mucosa. 2014.

[28] RS Sabeenian and V Palanisamy. Texture image classification using multi resolution combined statistical and spatial frequency method. *International Journal of Technology And Engineering System (IJTES)*, 2(2):167–171, 2011.

[29] P Shanmugavadivu and V Sivakumar. Fractal dimension based texture analysis of digital images. *Procedia Engineering*, 38:2981–2986, 2012.

[30] Ruchaneewan Susomboon, Daniela Raicu, Jacob Furst, and David Channin. Automatic single-organ segmentation in computed tomography images. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pages 1081–1086. IEEE, 2006.

[31] Ruchaneewan Susomboon, Daniela Stan Raicu, and Jacob Furst. A hybrid approach for liver segmentation. In *Proceedings of MICCAI workshop on 3D segmentation in the clinic: a grand challenge*, pages 151–160, 2007.

[32] Richard Szeliski, Ramin Zabih, Daniel Scharstein, Olga Veksler, Vladimir Kolmogorov, Aseem Agarwala, Marshall Tappen, and Carsten Rother. A comparative study of energy minimization methods for markov random fields. In *Computer Vision–ECCV 2006*, pages 16–29. Springer, 2006.

[33] Vishal Sharad Thakare. Survey on image texture classification techniques. *International Journal of Advancements in Technology*, 4(1):97–104, 2013.

[34] Inc. The MathWorks. Gray-level co-occurrence matrix (glcm). `http://uk.mathworks.com/help/images/gray-level-co-occurrence-matrix-glcm.html?refresh=true`, 2015. Accessed: 2015-05-14.

[35] Inc. The MathWorks. Treebagger class. `http://uk.mathworks.com/help/stats/treebagger-class.html?refresh=true`, 2015. Accessed: 2015-05-31.

[36] Toshiba Medical Visualization Systems Europe. Texture analysis. Technical report, 2012.

[37] Mihran Tuceryan and Anil K Jain. Texture analysis. *The handbook of pattern recognition and computer vision*, 2:207–248, 1998.

[38] Yoshikazu Uchiyama, Shigehiko Katsuragawa, Hiroyuki Abe, Junji Shiraishi, Feng Li, Qiang Li, Chao-Tong Zhang, Kenji Suzuki, and Kunio Doi. Quantitative computerized analysis of diffuse lung disease in high-resolution computed tomography. *Medical Physics*, 30(9), 2003.

[39] L. Vandevenne. Lode's computer graphics tutorial. `http://lodev.org/cgtutor/randomnoise.html`, 2004. Accessed: 2015-05-27.

[40] Manik Varma and Andrew Zisserman. Texture classification: Are filter banks necessary? In *Computer vision and pattern recognition, 2003. Proceedings. 2003 IEEE computer society conference on*, volume 2, pages II–691. IEEE, 2003.

[41] Manik Varma and Andrew Zisserman. A statistical approach to material classification using image patch exemplars. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(11):2032–2047, 2009.

[42] M. Petrou X. Lladó and J. Martí. Surface texture recognition by surface rendering. *Journal of Optical Engineering*, 44(3):037001–037001, 2005.

[43] Dong-Hui Xu, Arati S Kurani, Jacob D Furst, and Daniela S Raicu. Run-length encoding for volumetric texture. *Heart*, 27:25, 2004.

[44] Wilbert N. Wiskott L. Berkes P. Zito, T. Modular toolkit for data processing (mdp): a python data processing frame work, 2008.