# Multiresolution Volumetric Texture Segmentation

Constantino Carlos Reyes-Aldasoro

Thesis submitted to
The University of Warwick
for the degree of
Doctor of Philosophy

Department of Computer Science
November 30, 2004

THE UNIVERSITY OF
WARWICK

# Contents

# List of Figures

# List of Tables

# Acknowledgements

This work was conducted within the Signal and Image Processing Research Group in the Department of Computer Science at the University of Warwick with the support of *The University of Warwick*, *Consejo Nacional de Ciencia y Tecnología CONACYT* and *Instituto Tecnológico Autónomo de México ITAM*, in particular Dr. Federico Kuhlmann. I am grateful to all of them.

I am very grateful to my supervisor Dr. Abhir Bhalerao for the support and guidance through the years at Warwick; he was always ready to help anyone who asked his advice, he always had time for a chat and always challenged our ideas to help us produce better results. Thank you.

I would like to thank Dr. Trygve Randen from *Schlumberger Stavanger Research* in Norway for providing the natural texture images and Dr. Simon Warfield from *Brigham and Women's Hospital* in Boston, USA who provided some of the MRI data sets that were used in this thesis and also kindly revised some of the results.

I would also like to thank Dr. Nasir Rajpoot, Prof. Roland Wilson and Dr. Chang-Tsun Li for the helpful comments I received from them.

I shared several labs with different people during these years: Xiaoran Mo, Vincent Ng, Denis Fan and Li Wang. I enjoyed sharing the process of a PhD with you all.

Finally, I am most grateful to my wife Brigitte Garcia for her love and support, reading my manuscript with a critical statistical eye, and embracing her very own 'PhD' (days and nights!) *our son Santiago.* And of course, to my parents Carolina and Constantino; without them I would not be here.

# Declaration

I declare that, except where acknowledged, the material contained in this thesis is my own work and that it has neither been previously published nor submitted elsewhere for the purpose of obtaining an academic degree.

Constantino Carlos Reyes-Aldasoro

# Summary


## Multiresolution Volumetric Texture Segmentation

Constantino Carlos Reyes-Aldasoro

Thesis submitted to The University of Warwick
for the degree of Doctor of Philosophy
30 November, 2004

This thesis investigates the segmentation of data in 2D and 3D by texture analysis using Fourier domain filtering. The field of texture analysis is a well-trodden one in 2D, but many applications, such as Medical Imaging, Stratigraphy or Crystallography, would benefit from 3D analysis instead of the traditional, slice-by-slice approach. With the intention of contributing to texture analysis and segmentation in 3D, a *multiresolution volumetric texture segmentation* (M-VTS) algorithm is presented.

The method extracts textural measurements from the Fourier domain of the data via sub-band filtering using a Second Orientation Pyramid. A novel *Bhattacharyya space*, based on the Bhattacharyya distance is proposed for selecting of the most discriminant measurements and produces a compact feature space. Each dimension of the feature space is used to form a Quad Tree. At the highest level of the tree, new positional features are added to improve the contiguity of the classification. The classified space is then projected to lower levels of the tree where a boundary refinement procedure is performed with a 3D equivalent of butterfly filters.

The performance of M-VTS is tested in 2D by classifying a set of standard texture images. The figures contain different textures that are visually stationary. M-VTS yields lower misclassification rates than reported elsewhere ([104, 111, 124]).

The algorithm was tested in 3D with artificial isotropic data and three Magnetic Resonance Imaging sets of human knees with satisfactory results. The regions segmented from the knees correspond to anatomical structures that could be used as a starting point for other measurements. By way of example, we demonstrate successful cartilage extraction using our approach.

# Acronyms and Notation

| | |
|---|---|
| $M-VTS$ | Multiresolution Volumetric Texture Segmentation |
| $SOP$ | Second Order Pyramid |
| $LBG$ | Linde Buzo Gray Vector Quantising Algorithm |
| $CNS$ | Central Nervous System |
| $WM$ | Brain White Matter |
| $GM$ | Brain Grey Matter |
| $MRI$ | Magnetic Resonance Imaging |
| $SPGR$ | Gradient Echo Pulse Sequences |
| $Tr$ | Repetition Time |
| $Te$ | Echo Time |
| $PCA$ | Principal Components Analysis |
| $LEF$ | Local Energy Function |
| $TU$ | Texture Unit |
| $N_{TU}$ | Texture Unit Number |
| $LBP$ | Local Binary Pattern |
| $LVQ$ | Learning Vector Quantisation |
| $SOM$ | Self-Organising Feature Maps |
| $kNN$ | k Nearest Neighbours |
| $PCE$ | Positional Contiguity Enhancing (Features) |
| $MRF$ | Markov Random Fields |
| $MMRF$ | Multiresolution Markov Random Fields |
| $ROI/VOI$ | Region/Volume of Interest |
| $BS$ | Bhattacharyya Space |
| $BF$ | Butterfly Filter |
| $QT$ | Quad Tree |
| $TLE$ | Temporal Lobe Epilepsy |
| $FCD$ | Focal Cortical Dysplasia |
| $MFT$ | Multiresolution Fourier Transform |

| | |
|---|---|
| $\mathcal{VD}$ | Volumetric data |
| $\mathcal{I}$ | Image data |
| $f(r)$ | A 1D function of $r$ |
| $r, c, d$ | Co-ordinates of the spatial domain: rows, columns, slices |
| $N_r, N_c, N_d$ | Number of rows, columns and slices |
| $L_r = \{1, 2, \ldots, r, \ldots, N_r\}$ | Rows domain |
| $L_c = \{1, 2, \ldots, c, \ldots, N_c\}$ | Columns domain |
| $L_d = \{1, 2, \ldots, d, \ldots, N_d\}$ | Slices domain |
| $L_r \times L_c \times L_d$ | Dimensions of the volumetric data |
| $x = (r, c, d) \in L_r \times L_c \times L_d$ | An element of the volumetric data |
| $\mathcal{VD}(x) = g, \quad g \in G$ | Grey level/ Intensity associated with element $x$ |
| $N_g$ | Number of grey levels |
| $G = \{1, 2, \ldots, g, \ldots, N_g\}$ | Quantised grey level domain |
| $h(g)$ | Histogram of the data |
| $\mathcal{N} \subset (L_r \times L_c \times L_d)$ | A neighbourhood of an element of the space |
| $L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}}$ | Dimensions of the neighbourhood |
| $\tilde{\mathcal{VD}}(x) = T[\mathcal{VD}(\mathcal{N})]$ | An operator on the volumetric data around the neighbourhood $\mathcal{N}$ |
| $T^i$ | Operator associated to measurement $i$ |
| $\mathcal{VD}_\omega = \mathscr{F}[\mathcal{VD}]$ | Fourier transform of $\mathcal{VD}$ |
| $\rho, \kappa, \delta$ | Co-ordinates for the Fourier domain |
| $\mathcal{F}_n$ | Fourier coefficients |
| $F^i$ / $F^i_\omega$ | A filter associated to measurement $i$ in the spatial / Fourier domains |
| $F^{\mathcal{G}}/F^{\mathcal{G}}_\omega$ | A Gabor filter in the spatial / Fourier domain |
| $\tilde{\mathcal{VD}}^i_\omega = F^i_\omega \mathcal{VD}_\omega$ | Volumetric data $\mathcal{VD}$ filtered by filter $F^i_\omega$ (Fourier domain) |
| $\tilde{\mathcal{VD}}^i = \mathscr{F}^{-1}[\mathcal{VD}^i_\omega] = F^i * \mathcal{VD}^i$ | Volumetric data $\mathcal{VD}$ filtered by filter $F^i$ (Spatial domain) |
| $\nabla\mathcal{VD} = \frac{\partial \mathcal{VD}}{\partial r}\hat{r} + \frac{\partial \mathcal{VD}}{\partial c}\hat{c} + \frac{\partial \mathcal{VD}}{\partial d}\hat{d}$ | Gradient operator over the volumetric data |
| $\hat{r}, \hat{c}, \hat{d}$ | Unitary vectors in the direction of the axes |

| | |
|---|---|
| $S$ | Space of the measurements over the data |
| $S^i = \tilde{\mathcal{V}\mathcal{D}}^i$ | Measurement $i$ over the data |
| $S^i(x) = g,$ | The value of element $x$ at measurement $i$ |
| $S(x) = \{S^1(x), \ldots, S^{N_i}(x)\}$ | A vector (pattern) of the values of element $x$ at every measurement |
| $S(x) \in \underbrace{G \times G\times, \ldots, G}_{N_i}$ | Domain of the measurement vector |
| $\mathscr{G}_a$ | A Gaussian function |
| $\tilde{S^i} = S^i * \mathscr{G}_a$ | A smoothed version of $S^i$ |
| $S^i_\omega$ | Fourier transform of measurement $i$ |
| $N_i$ | Number of measurements over the data |
| $L_i = \{1, 2, \ldots, i, \ldots, N_i\}$ | Measurement domain |
| $L_r \times L_c \times L_d \times L_i$ | Dimensions of the measurement space |
| $L_j = \{\ldots, j, \ldots,\}$ | A permutation of the measurement domain |
| $S_F \subset S$ | Feature space extracted from the measurement space |
| $N_f \leq N_i$ | Number of features selected from the measurement space |
| $L_f \subset L_i$ | Feature domain |
| $L_r \times L_c \times L_d \times L_f$ | Dimensions of the feature space |
| $S^f \in S_F,\ S^f \in S$ | A feature |
| $\mu$ | Mean |
| $\sigma$ | Standard deviation |
| $s$ | Skewness |
| $ku$ | Kurtosis |
| $\langle \cdot, \cdot \rangle$ | Inner product |
| $\|\| \cdot \|\|_2$ | Norm |
| $\mathbb{Z}$ | The set of integers numbers |
| $\mathbb{R}$ | The set of real numbers |
| $\mathbb{C}$ | The set of complex numbers |

| | |
|---|---|
| $\lambda$ | Labelling function |
| $\lambda_a$ | Approximation of labelling function (Classifier) |
| $a_k \in S$ | Description of a class through a single point; the points $a_k$ define hyperplanes perpendicular to the chords that connect them |
| $a_{kn}$ | Description of a class with $n$ points per class |
| $\hat{a}_k \in S$ | An estimate of the mean / prototype values for a certain class $k$ |
| $N_k$ | Number of classes |
| $N_{kNN}$ | Number of Nearest Neighbours |
| $m_i \in S$ | A reference vector (neuron) associated with class $i$ |
| $m_w$ | A *winner* neuron (closest to input signal $x$) in a competitive process |
| $\alpha(t)$ | A monotonically decreasing scalar gain factor |
| $\mathcal{N}_w$ | A neighbourhood around the winner neuron |
| $R_k \subset S$ | Regions corresponding to a partition of $S$ |
| $L_k = \{1, 2, \ldots, N_k\}$ | Classes domain |
| $(k_1, k_2);\ k_1, k_2 \in L_k$ | A pair of classes |
| $\epsilon$ | Misclassification error |
| $N_p$ | Number of pairs of classes |
| $L_p = \{(1,2), \ldots, (k_1, k_2), \ldots, (N_k - 1, N_k)\}$ | Domain of pairs of classes |
| $D$ | A distance measure |
| $D_2$ | Euclidean distance measure |
| $D_8$ | Chess-board distance |
| $D_B(k_1, k_2)$ | Bhattacharyya distance measure |
| $BS_{IP}$ | Bhattacharyya space |
| $BS_I, BS_P$ | Marginal distributions of the Bhattacharyya space |
| $BS_{(I)}, BS_{(P)}$ | Order statistics of the marginal distributions of the $BS$ |

| | |
|---|---|
| $v = \{0, 0, \ldots, 0\}$ | Initial state for feature selection |
| $v' = \{1, 1, \ldots, 1\}$ | Final state for feature selection |
| $(QT)^{\mathcal{L}}$ | Quad Tree ($QT$) at level $\mathcal{L}$ |
| $(gp)^{\mathcal{L}}$ | Gaussian Pyramid ($gp$) at level $\mathcal{L}$ |
| $(gp)^{\mathcal{L},n}$ | $n$ Expansions of $(gp)^{\mathcal{L}}$ |
| $(G)^{\mathcal{L}}$ | Grey level/ Intensity associated with the data of a Quad Tree at level $\mathcal{L}$ |
| $CM(g_1, g_2, D_8, \theta)$ | Un-normalised co-occurrence matrix, as a function of two grey levels, distance and orientation |
| $\theta, \phi$ | Orientation measures |
| $\varphi_1, \varphi_2, \ldots$ | Textural features of the co-occurrence matrix |
| $\mathcal{R}, \mathcal{C}, \mathcal{D}$ | *Row, Column* and *Slice* co-ordinates, (PCE features) |
| $\psi(r)$ | A mother Wavelet |
| $\psi_{k,l}(r)$ | Scaled and dilated Wavelet |
| $\Psi(k, l)$ | Wavelet transform |
| $c_{k,l}$ | Coefficients of Wavelet transform |
| $\mathcal{T}_r, P, \Phi$ | Trace, diametrical and circus functionals of the Trace transform |
| $ls, rw$ | The set that describes the elements of each wing (left/right) of the butterfly filters |
| $N_w$ | Number of elements in $lw/rw$ |
| $\tilde{S}^i_{lw}(x)/\tilde{S}^i_{rw}(x)$ | Weighted average of the grey levels of $lw/rw$ |
| $w_q$ | Weight function for the elements of the filters |
| $N_{lw}/N_{rw}$ | Number of elements that share class with the pivot element $x$. |
| $\tilde{S}_{x-lw}/\tilde{S}_{x-rw}$ | The weighted combination of the element $x$ and $\tilde{S}^i_{lw}/\tilde{S}^i_{rw}$ |

# Chapter 1

# Introduction

## 1.1 Volumetric Texture Analysis

The study of *Volumetric Texture* is a very challenging subject. To start with, there is not a single accepted definition of texture in two dimensions (2D). Then, the extra third dimension that is included in texture in three dimensions (3D) increases considerably the computational complexity. In some cases, the extension of texture analysis to three dimensions can be easily achieved, but in others, for example those using orientation or phase, careful consideration is needed.

Volumetric texture has received much less attention than its spatial 2D counterpart which has seen the publication of numerous and differing approaches for texture analysis and feature extraction (for example [8, 15, 29, 55, 56, 144, 146]), and classification and segmentation ([14, 71, 75, 79, 147, 156]).

The considerable computational complexity that is introduced with the extra dimension is partly responsible for lack of research in volumetric texture. But also there is an important number of applications for 2D texture analysis. Yet, there is a growing number of problems where a study of volumetric texture is of interest. The analysis of crystallographic texture - the organisation of grains in polycrystalline materials - is of interest in relation with certain characteristics of ceramic materials such as ferro- or piezoelectricity [143]. In Stratigraphy, also known as Seismic Facies Analysis [18, 125],

the volumetric texture of the patterns of seismic waves within sedimentary rock bodies can be used to locate potential hydrocarbon reservoirs. In Medical Imaging, the data provided by the scanners of several acquisition techniques such as Magnetic Resonance Imaging (MRI) [88, 128], Ultrasound [166] or Computed Tomography (CT) [65, 136] deliver grey level data in three dimensions. Different textures in these data sets can allow the discrimination of anatomical structures. The importance of Texture in MRI has been the focus of researchers, such as Lerski [94] and Schad [134], and a COST European group was established for this purpose [28].

Texture analysis has been used with mixed success in medical imaging: for detection of micro-calcification and lesions in breast imaging [72, 137, 142], for knee segmentation [78, 100], for the delineation of cerebellar volumes [131], for quantifying contralateral differences in epilepsy subjects [164, 165], to diagnose Alzheimer's disease [133] and brain atrophy [136], and to characterise spinal cord pathology in Multiple Sclerosis [106]. Most of this reported work, however, has employed solely 2D measures, usually co-occurrence matrices that are limited by computational cost. Furthermore, feature selection is often performed in an empirical way with little regard to training data, which are usually available.

The most common technique to deal with volumetric data is to slice the volume in 2D cross-sections. The individual slices can be used in a 2D texture analysis [12]. A simple extension of the 2D slices is to use orthogonal 2D planes in the different axes, and then proceed with a 2D technique, Gabor filters for instance [166]. However, high frequency oriented textures could easily be missed by these filter planes. In those cases it is important to conduct a volumetric analysis. Thus, the analysis of volumetric texture has many potential applications and the intention of this thesis is to contribute to this developing field.

## 1.2 The Spatial and Fourier Domains

The Fourier transform [16] is a well-known mathematical operation that translates a signal from the spatial or time domain, into the Fourier or frequency domain. The Fourier domain will be widely used through this thesis since some of the signals that appear as texture have strong energy concentration at different frequencies and orientations. Consequently, different textures can be discriminated by the amount of energy that the signal displays in different frequency bands. This will be explained in detail in section 2.4.2. For example, figure 1.1 (a) presents a 1D signal in the spatial domain and its transformation into the Fourier domain. It can be observed that the signal appears fairly repetitive in the spatial domain, which translates into three main *spikes* in the Fourier domain. The basis of the Fourier transform is the analysis of exponential Fourier series, that is, the representation of a signal by the sum of the exponential signals that are orthogonal to each other. The exponential function $e^{jr}$ forms a family of orthogonal functions within a certain interval $[r_0, r_0 + R]$ with the functions $e^{jn2\pi r\rho_0},\ n = 0, \pm 1, \pm 2, ...$ where $e$ is the complex Euler identity: $e^{jn2\pi r\rho_0} = cos(n2\pi r\rho_0) + jsin(n2\pi r\rho_0)$, and $\rho_0$ will determine the frequency of the sinusoidals. Any signal can be expressed by a series of these functions:

$$\begin{aligned} f(r) &= \mathcal{F}_0 + \mathcal{F}_1 e^{j2\pi r\rho_0} + \mathcal{F}_2 e^{j4\pi r\rho_0} + \mathcal{F}_3 e^{j6\pi r\rho_0} + \dots \\ &\quad \mathcal{F}_{-1} e^{-j2\pi r\rho_0} + \mathcal{F}_{-2} e^{-j4\pi r\rho_0} + \mathcal{F}_{-3} e^{-j6\pi r\rho_0} + \dots \\ f(r) &= \sum_{n=-\infty}^{\infty} \mathcal{F}_n e^{jn2\pi r\rho_0} \end{aligned} \tag{1.1}$$

where $r_0 < r < r_0 + R$ and $R = \frac{2\pi}{\rho_0}$. The constants $\mathcal{F}_n$, called the Fourier coefficients of $f$ are defined by the inner product ($\langle \cdot, \cdot \rangle$) between the original signal and the exponential:

$$\mathcal{F}_n = \frac{\int_{r_0}^{r_0+R} f(r)(e^{jn2\pi r\rho_0})^* dr}{\int_{r_0}^{r_0+R} e^{jn2\pi r\rho_0}(e^{jn2\pi r\rho_0})^* dr} = \frac{1}{R}\int_{r_0}^{r_0+R} f(r)e^{-jn2\pi r\rho_0} dr \tag{1.2}$$

where $(\cdot)^*$ represents the complex conjugate of a signal. Two important observations follow. First, the orthogonality of the functions $\mathcal{F}_n e^{jn2\pi r\rho_0}$ mean that their inner product is zero:

$$\langle \mathcal{F}_n e^{jnr}, \mathcal{F}_m e^{jmr} \rangle = \frac{1}{R} \int_{r_0}^{r_0+R} \mathcal{F}_n e^{jnr} \mathcal{F}_m e^{jmr} dr = 0, \quad m \neq n. \tag{1.3}$$

The inner product of a signal with itself is called the *norm*:

$$\| f \|_2 = \langle f, f \rangle^{1/2} \tag{1.4}$$

and it can be used as a measure of the energy of the signal.

Second, it is important to notice that the previous analysis is limited to a certain interval in time or space. To consider the range $(-\infty, +\infty)$ the period has to be evaluated in the limit $R \to \infty$. By taking the limit (and after some manipulation) the Fourier transform pair is reached [16]:

$$\begin{aligned} f(r) &= \int_{-\infty}^{\infty} f_\omega(\rho) e^{j2\pi r\rho} d\rho \\ f_\omega(\rho) &= \int_{-\infty}^{\infty} f(r) e^{-j2\pi r\rho} dr \end{aligned} \tag{1.5}$$

which is sometimes represented by:

$$f \quad \overset{\mathscr{F}}{\longleftrightarrow} \quad f_\omega \tag{1.6}$$

The previous expressions can be complex and therefore two planes can be used to show it: (real/imaginary) or (magnitude/phase). The Fourier transform can be applied to signals of more than one dimension, in the case of a 3D signal, the corresponding Fourier transforms are:

$$f(r,c,d) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_\omega(\rho,\kappa,\delta) e^{j2\pi(r\rho+c\kappa+d\delta)} d\rho \; d\kappa \; d\delta$$

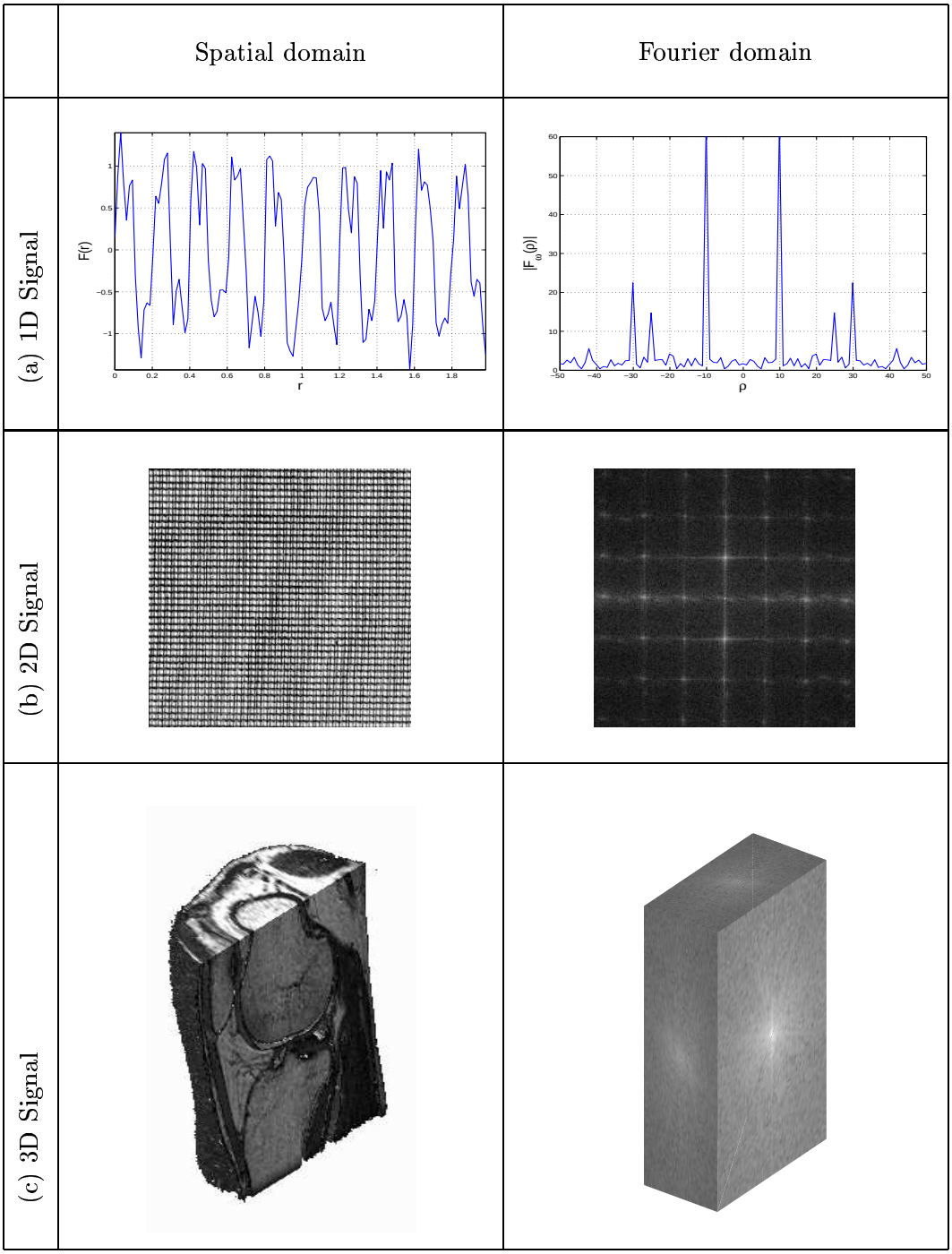



**Figure 1.1: Signals in spatial and Fourier domains: (a) 1D, (b) 2D (image) and (c) 3D (volume). Periodic signals tend to concentrate its energy in spikes in the Fourier domain, this is easier to visualise in 1D and 2D. (c) is a counter example, there are no visible spikes to be seen in the Fourier domain.**

$$f_\omega(\rho,\kappa,\delta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(r,c,d)e^{-j2\pi(r\rho+c\kappa+d\delta)}dr\ dc\ dd \tag{1.7}$$

where $(r,c,d)$ are the co-ordinates in the spatial domain and $(\rho,\kappa,\delta)$ are the co-ordinates in the Fourier domain.

Figure 1.1 (b) shows one image with a textured pattern and its corresponding 2D Fourier transform. It can be seen that some of the ingredients or characteristics of the image are noticeable in its Fourier counterparts.

Figure 1.1 (c) presents a case of a 3D signal in the spatial and Fourier domains. The spatial data correspond to the Magnetic Resonance Imaging (MRI) of a human knee that has been sliced in the sagittal and axial planes. The Fourier domain is also sliced in the 3 axes. A description of the Magnetic Resonance Imaging technology is presented in appendix C, while a brief description of the human knee is presented in appendix D.

Analysing volumetric data in this way is harder than 2D images (and often the analysis is performed slice-by-slice). However, the 3D Fourier transform allows us to work with the whole volumetric set and any operation like filtering, thresholding or convolution can be performed in 3D.

Figure 1.3 shows further examples of images with different textures. Periodicity appears as bright spots (equivalent to the spikes in 1D), orientation appears perpendicular in opposite domains and the randomness appears as a signal that increases its intensity towards the centre.

## 1.3 Sub-band Filtering

Certain characteristics of signals in the spatial domain such as periodicity are quite distinctive in the Fourier domain, as shown in figure 1.1. If the data (for example the image in figure 1.2) contain textures that vary in orientation and frequency, then certain filter sub-bands will contain more energy than others, and 'roughness' will be characterised by more or less energy in certain bands or regions. In the example presented, the low pass filter (which corresponds to the centre of the Fourier domain) returns a blurred

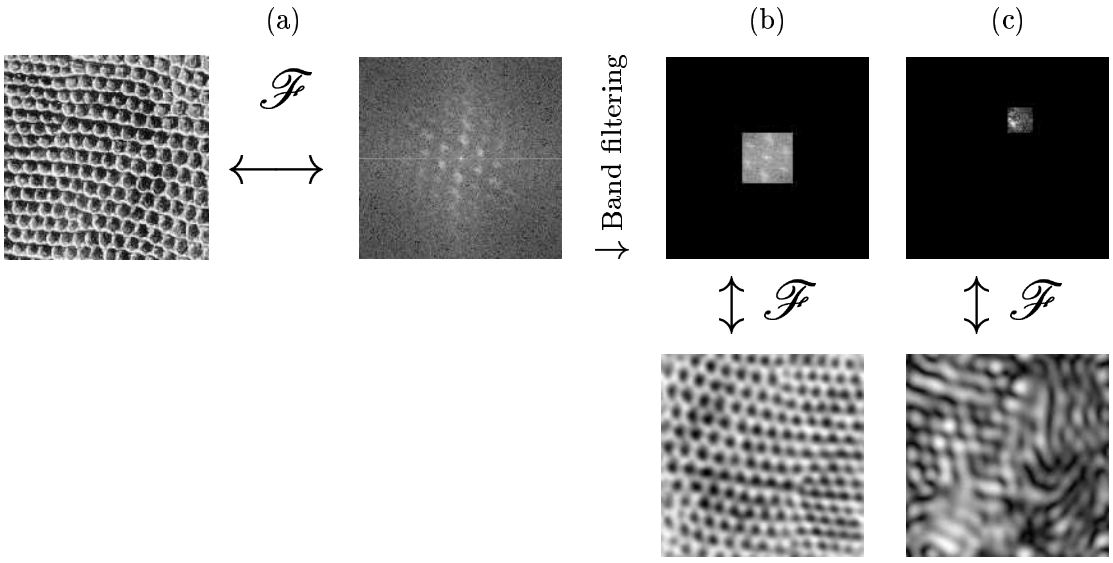



**Figure 1.2: Sub-band filtering in the Fourier domain: (a) An image (reptile skin) in the spatial and Fourier domains, (b) A low pass filtered version of the image, (c) A band pass version of the image.**

version of the image, while a band pass filter captures other details of the texture.

The principle of sub-band filtering can equally be applied to volumetric data (section 2.4.2).

## 1.4 Definition of Volumetric Texture

A single definition of texture does not exist, but most of the numerous definitions that are present in the literature, have some common elements that emerge from the etymology of the word. Texture comes from the Latin *textura*, the past participle of the verb *texere*, to weave [108]. From here, it is expected that a texture will exhibit a certain structure created by common elements, repeated in a certain regular way, as in the threads that form a fabric. Hawkins [58] identifies three *ingredients* of texture:

- some local 'order' is repeated over a region which is large in comparison to the order's size,
- the order consists on the non-random arrangement of elementary parts, and,
- the parts are roughly uniform entities having approximately the same dimensions everywhere within the textured region.

**Table 1.1: Psycho-visual properties of Texture.**

| Author | Properties |
|---|---|
| Ravishankar [127] | Granular, marble-like, lace-like, random, random nongranular and somewhat repetitive, directional locally oriented, repetitive. |
| Tamura [144] | Coarseness, contrast, directionality, linelikeness, regularity, roughness. |

While these ingredients can describe some textures, as those of figure 1.3 (a), there are some other cases that are not so uniform or deterministically arranged. For instance, the pebbles of figure 1.3 (b) appear to be in a random placement. The term *visual texture* is some times used in an attempt to distinguish it from the tactile concept of texture. From this visual context, *image texture* is defined by Tuceryan and Jain [146] as:

- A function of the spatial variation in pixel intensities.

Gonzalez [50] relates certain properties of texture with the approaches to texture analysis:

- Statistical: smooth, coarse, grainy,...
- Structural: arrangement of feature primitives (sometimes called *textons*) according to certain rules,
- Spectral: global periodicity based on the Fourier spectrum.

Some of these properties are visually meaningful and are helpful to describe textures. In fact, studies have analysed texture from a psycho-visual slant [127, 144] and have identified the properties presented in table 1.1.

It is important to notice that these *properties* are different from the *features* or *measurements* (although some other works refer to the properties as features of the data) that can be extracted from the textured regions.

Two more ingredients of texture should be mentioned:

- Texture is inherently scale dependent [14, 67, 139]. The texture of a brick wall would change completely is we get close enough to observe the texture of a single brick.

- The texture of an element (pixel or voxel) is implicitly related to its neighbours. It is not possible to describe the texture of a single element, as it will always depend on the neighbours to create the texture. This can be exploited through: Fourier methods, which extract frequency components according to the relation of elements; a Markovian approach in which the attention is restricted to a small neighbourhood; or a co-occurrence matrix where occurrence of the grey levels of neighbouring elements is recorded.

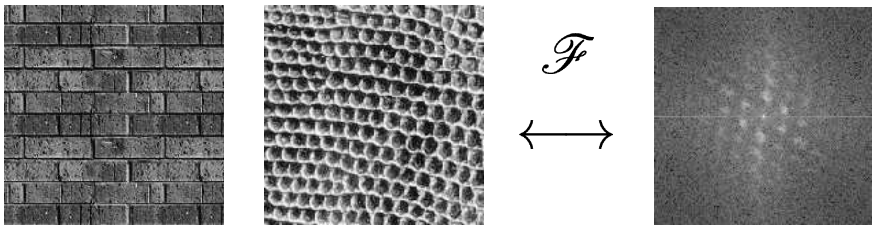
All the properties and ingredients that were previously mentioned about texture, or more specifically, *visual*, or *2D* texture, can be applied to volumetric texture.

In this thesis, *Volumetric Texture* is considered as the texture that can be found in volumetric data (this is sometimes called *solid texture* [12]). Figure 1.4 shows four examples of volumetric data with some textured regions. Volumetric Texture is different from 3D Texture, Volumetric Texturing or Texture Correlation.

*3D Texture* [20, 31, 32, 95, 99] refers to the observed 2D texture of a 3D object that is being viewed from a particular angle and whose lighting conditions can alter the shadings that create the visual texture. This analysis is particularly important when the direction of view or lighting can vary from the training process to the classification of the images. Our volumetric study is considered as volume-based (or image-based for 2D); that is, we consider no change in the observation conditions. In Computer Graphics the rendering of repetitive geometries and reflectance into voxels is called *Volumetric Texturing* [110]. A different application of texture in Magnetic Resonance is the one described by the term *Texture Correlation* proposed by Bay [3] and now widely used [4, 49, 107, 119] which refers to a method that measures the strain on trabecular bone under loading conditions by comparing loaded and unloaded digital images of the same specimen.

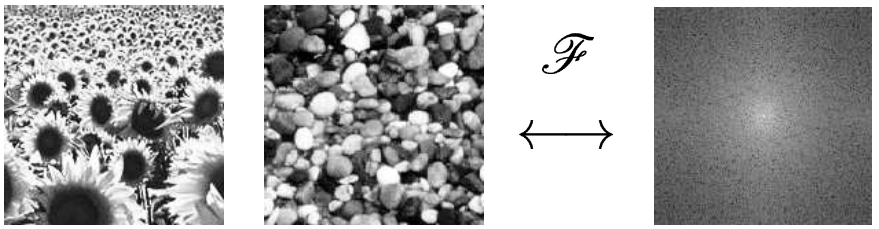
Throughout this work we will consider that volumetric data, $\mathcal{VD}$, will have dimensions for rows, columns and slices $N_r \times N_c \times N_d$ and is quantised to $N_g$ grey levels. Let $L_r = \{1, 2, \ldots, r, \ldots, N_r\}$, $L_c = \{1, 2, \ldots, c, \ldots, N_c\}$ and $L_d = \{1, 2, \ldots, d, \ldots, N_d\}$ be the spatial domains of the data (for an image $L_r, L_c$ would be horizontal and vertical), and $G = \{1, 2, \ldots, g, \ldots N_g\}$ the set of grey tones. The volumetric data $\mathcal{VD}$ can be

(a) Non-random arrangement of elementary parts, parts are roughly uniform



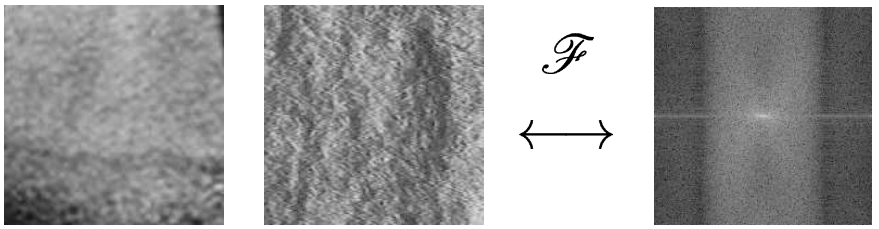
Bricks Reptile skin Reptile skin in Fourier domain

(b) Roughly regular parts, random arrangement



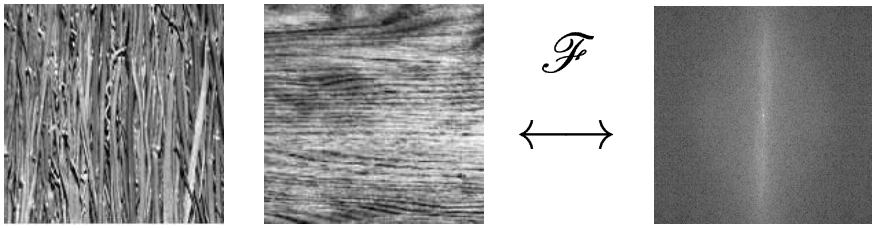
Sunflowers Pebbles Pebbles in Fourier domain

(c) Regular parts are not identifiable, nor there is a non-random arrangement



Bone in a T2 MRI Wool Wool in Fourier domain

(d) Dominant orientation present, regular parts may or may not be identifiable



Straw Wood Wood in Fourier domain

**Figure 1.3: Ingredients that can be identified in spatial textures. Examples in spatial and Fourier domains.**

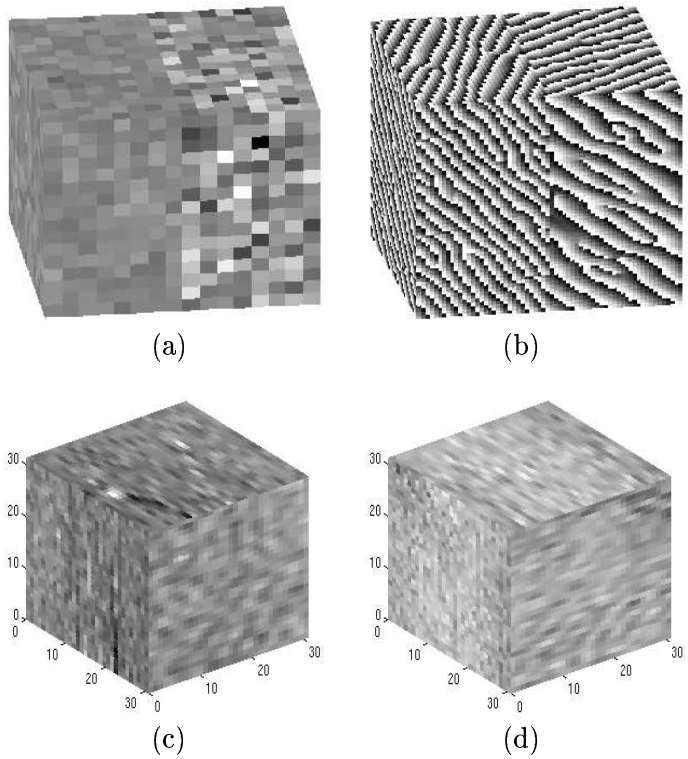



**Figure 1.4: Volumetric texture examples: (a) A cube divided into two regions with Gaussian noise, (b) A cube divided into two regions with oriented patterns of different frequencies and orientations, (c) A sample of muscle from MRI, (d) A sample of bone from MRI.**

represented then as a function that assigns a grey tone to each triplet of co-ordinates:

$$L_r \times L_c \times L_d; \mathcal{VD} : L_r \times L_c \times L_d \rightarrow G \tag{1.8}$$

An image then is a special case of volumetric data when $L_d = \{1\}$, that is [56]:

$$L_r \times L_c; \mathcal{I} : L_r \times L_c \rightarrow G \tag{1.9}$$

## 1.5 Multiple Resolution in Texture

Multiresolution methods have been widely used in image analysis problems for some time now. The fundamental aspect is the expansion of the image into a set of images

at different scales or resolutions. Images at lower spatial resolutions are created by a recursive process of filtering and sub-sampling. The use of multiresolution techniques has been motivated by different reasons: reducing the computational complexity of the task; enabling the application of filters with increasing larger support; optimisation problems when the solution at a coarse scale can be used to initialise the solution at the next finer resolution. The reduction of image size results in a decreased complexity, opening up solutions to otherwise difficult problems. Filtering at various scales or channels can extract different measurements or features of an image, as done with Gabor or Lognormal filters [71, 81] of various sizes and orientations or the Second Orientation Pyramid tessellation [162] (section 2.4.2).

Another implication of multiresolution techniques is a trade-off between spatial resolution or positional accuracy, and measurement uncertainty or the measurement class membership. Multiresolution pyramids [17] or trees [132, 140, 162] can be constructed by a weighted average of neighbouring elements, from lower, fine levels up to a coarse higher level where each node is considered as the parent of several child nodes. At the higher levels these trees have, on one hand, reduced uncertainty in the measurement values (grey level for instance) but inherently lower spatial resolution. Appendix B describes the construction of pyramids and trees that will be used through this thesis.

Figure 1.5 (a) shows an example of a tree of an image with two noisy regions with normal distributions that have an overlapping grey scale. At the image level, the two regions are not distinguishable in the histogram (b), but at lower spatial resolutions of a tree, the smoothing makes the classes separate from each other, as the histograms in (c,d) show. (The use of the histograms at different resolutions is sometimes called *multiresolution histograms* [53], a term that could be confusing since it is the images and not the histograms themselves which are in a multiresolution space.)

Uncertainty in image processing was recognised and discussed by Granlund and Wilson [158] and motivated the use of pyramidal representations for segmentation [162] and later, for similar reasons, the development of joint spatial spatial-frequency representa-

tions such as the Multiresolution Fourier Transform (MFT), which is an over-complete, Fourier Wavelet basis [157]. In this work, we have adapted the feature estimation ideas from [162] into a new classification framework based on the work of Schroeter and Bigun [135].

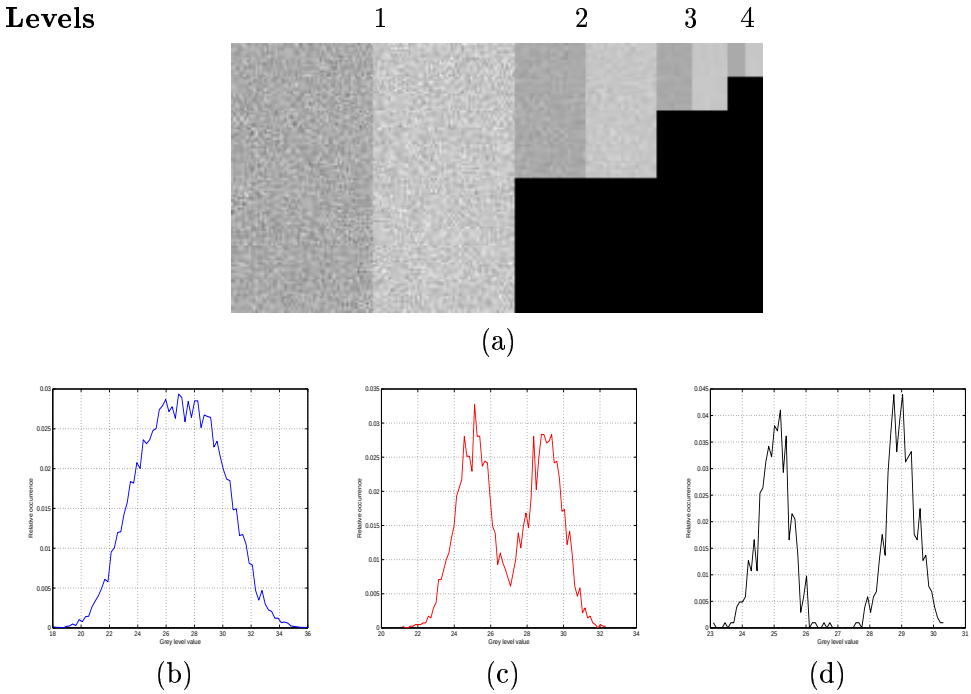



**Figure 1.5: A tree to reduce the uncertainty in the distribution, the coarsest image contains two regions with Gaussian noise ($\mu_1 = 25, \sigma_1 = 5, \mu_2 = 29, \sigma_2 = 5$). (a) Quad Tree of 4 levels placed side by side. (b) Histogram at lowest level (level 1). (c) Histogram at level 2. (d) Histogram at level 3.**

While a tree is effective in reducing the uncertainty in the distribution at the expense of the spatial resolution, it should be noted that it is useful for the discrimination of two classes only if the mean values of the measurements differ. If the distributions possess similar mean values, then, as the higher levels are constructed, the two regions and their distributions will merge into one. This is illustrated in figure 1.6 where two visually distinctive regions are not separable from the histogram. It is therefore very important to select the correct measurement before constructing a multiresolution tree. In the previous example if a different measurement were extracted from a local window; variance for instance, the classification would be improved.

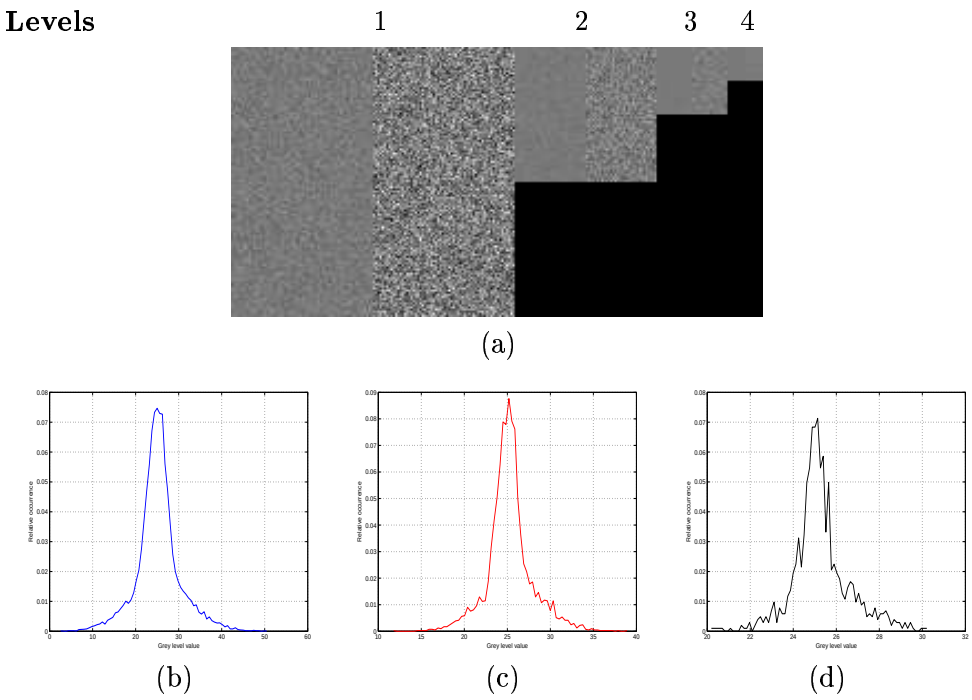
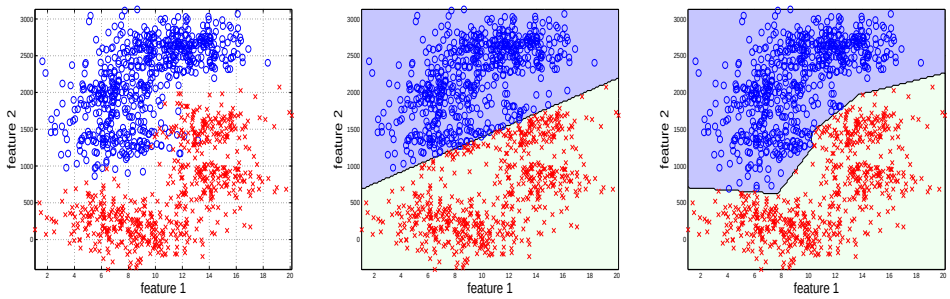



**Figure 1.6: A tree to reduce the uncertainty in the distribution. When the distributions have similar mean values, the regions tend to merge into one ($\mu_1 = 25, \sigma_1 = 2, \mu_2 = 25.5, \sigma_2 = 5$). (a) Quad Tree of 4 levels placed side by side. (b) Histogram at lowest level (level 1). (c) Histogram at level 2. (d) Histogram at level 3.**

## 1.6 Classification of the Space

The classification or labelling problem is that of assigning every element of the data, or the measurements extracted from the data, into one of several possible classes [54]. For medical data the classes can represent a unique anatomical structure such as a bone, cartilage or muscle. The classifying process corresponds to a partitioning of the space of the measurements into regions. For example, figure 1.7 presents two different populations in a two-measurement space and the possible partitioning into two classes with different classification schemes. The classifiers used were: thresholding that result in a straight boundary; and LVQ1, which adapts better to the original distribution. The choice of a classification algorithm can therefore be important. When the original classes of the populations are known then the accuracy of the classification scheme can




(a) (b) (c)

**Figure 1.7: A classification problem. Two different populations are distributed over a two measurement space, (a) represents the ground truth. The populations are classified with two different strategies, first with thresholding which results in a straight boundary (b), and with LVQ1 a more sophisticated classifier which results in a piecewise linear boundary (c). The colours in (b,c) represent the classes into which the elements have been assigned; (c) is closer to the ground truth.**

be measured, but this is not generally the case, sometimes only estimates of the accuracy can be achieved.

There is a vast amount of literature concerned with these problems [6, 11, 36, 54]. In chapter 3 we will analyse some of the techniques that may be of use for texture classification. Two important aspects related to the classification process are: the effect of the Local Energy Function (LEF) and the context or positional information of the elements and not only their measurement values. This will be treated in sections 3.2 and 5.5.1 respectively.

## 1.7 Objectives of the Thesis

This thesis intends to contribute to the relatively new problem of volumetric texture analysis and classification by proposing a multiresolution volumetric texture segmentation (M-VTS) methodology. The main goal is to provide a method that receives volumetric data (we have concentrated on MRI) and returns a series of classes with homogeneous characteristics, like anatomical structures, based on textural properties. To achieve this objective, several problems must be solved.

1. A set of measurements that can extract textural features in 3D is required. An

effective *measurement space* needs to be generated.

2. The number of measurements that can be extracted from the data can be considerable, and when working in three dimensions, it will be even greater. The reduction of dimensions of the measurement space will not only reduce the computational complexity but may also achieve better results through selection of the *best features*.

3. The measurement space needs to be *classified*. We require a suitable classifier that discriminates classes properly and in a reasonable time. In the MRI data, classes will be ascribed to anatomical features.

4. The measurements extracted and the classification algorithm need to be compared against other existing techniques. The benchmark comparison is essential for all the steps of the classification process.

5. Multiresolution techniques have been proposed in other applications and it is possible that texture segmentation will benefit from multiresolution algorithms. This proposition will be investigated.

## 1.8 Contributions of the Thesis

The original contributions of this thesis are:

1. A survey of the volumetric texture extraction methods is presented; some advantages and disadvantages of the techniques are highlighted.

2. We propose an extension of the *Second Order Pyramid* (SOP) (based on Spann and Wilson [162]), a 2D Method, into 3D.

3. For validation, the measurements proposed are compared against other texture extraction techniques and different aspects of classification algorithms are analysed. It is found that the SOP measurements are equivalent to others.

4. Based on the Bhattacharyya distance, a novel *Bhattacharyya Space* [128] is proposed as a methodology to select the most discriminant features of the measurement space.

5. A *multiresolution classification algorithm* (based on [135, 162]) is extended from 2D to 3D. Some extensions of the algorithm from 2D to 3D are required; in particular the use of boundary smoothing filters. The use of a Markov Random Field (MRF) approach is compared with pyramidal butterfly filters, the latter providing superior results.

6. Context is included in the classifier to improve the results. A *contiguity enhancing* strategy is proposed through the use of new features (Positional Contiguity Enhancing features) that are added to the feature space.

7. A series of 2D and 3D data sets are classified and compared with other methodologies and the proposed algorithm is shown to give excellent results. Three MRIs of human knees are classified. For validation and reproducibility, the data sets generated and the code is made available on the web page:
http://www.dcs.warwick.ac.uk/~creyes/m-vts.

8. As an application of the methodology, it is shown how a segmentation of an anatomical structure through the proposed algorithm can be used for the extraction of other structures like the cartilage.

## 1.9 Outline of the Thesis

An overview of the thesis is represented graphically in figure 1.8. In chapter 2, a literature review of the related work on volumetric texture, or spatial techniques that can be extended to 3D is presented. Next, chapter 3 discusses classifiers with emphasis in the Local Energy Function. Feature selection is investigated in chapter 4. A new Bhattacharyya space is proposed as a way to select the most discriminant features. Chapter 5 describes a multiresolution segmentation algorithm and an extension of butterfly filters

is proposed as a hierarchical methodology for the refinement of the boundaries. Also, a new set of positional features that added to a feature space can yield better contiguity is also proposed. Chapter 6 compares the results of the proposed methodology with others present in the literature for 2D data. Also, volumetric data sets are presented and classified with a special emphasis given to human knee MRIs. The MRI knees are first classified into anatomical regions and the results are used to segment the cartilage. Finally, chapter 7 presents conclusions and proposals for further work.

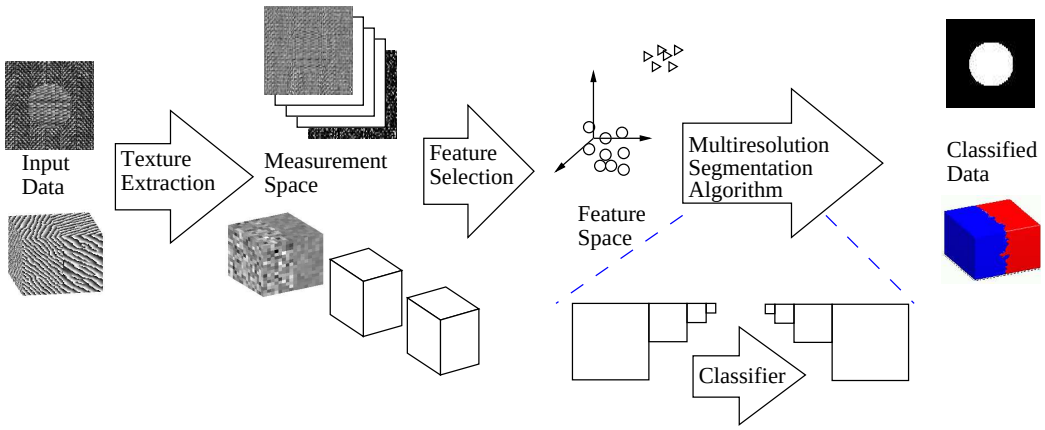


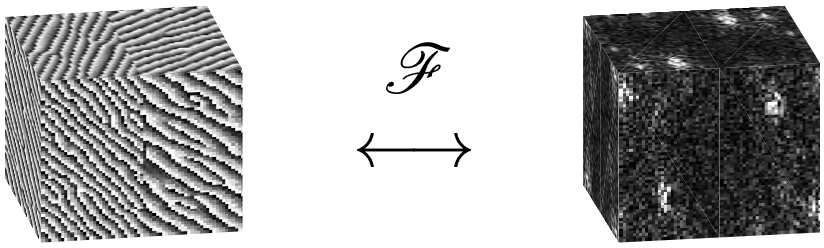

**Figure 1.8: Graphical outline of the thesis. Textured data sets, both in 2D and 3D, are processed in three main steps: measurement extraction, feature selection and multiresolution segmentation.**

# Chapter 2

# Texture Analysis: the Measurement Space

In this chapter, a review of texture measurement extraction methods will be presented. A special emphasis will be placed on the use of these techniques in 3D. Some of the techniques have been widely used in 2D but not in 3D, and some others have already been extended. We will analyse and compare them in order to select the most adequate measurement for the data used in this thesis. We conclude that sub-band filtering with SOP in the Fourier domain is the most appropriate measurement for its simplicity and power. For visualisation purposes, two data sets will be presented: an artificial set with two oriented patterns (figure 2.1) and one volumetric set of a human knee MRI set (figure 1.1 (c)). These sets along with several other 2D and 3D will be classified and the performance compared with different techniques in chapter 6.

All the measurements extracted from the data, either 2D or 3D will form a multivariate space, regardless of the method used. The space will have as many dimensions or variables as measurements extracted. Of course, numerous measurements can be extracted from the data, but a higher number of measurements does not always imply a better space for classification purposes. In some cases, having more measurements can yield lower classification accuracy and in others a single measurement can provide the




(a) (b)

**Figure 2.1: Artificial data set with two oriented patterns of [$64 \times 32 \times 64$] elements each, with different frequency and orientation: (a) Data in the spatial domain, (b) in the Fourier domain.**

discrimination of a certain class. Therefore the selection of a proper set of measurements is a difficult task. In this chapter we will concentrate on the generation of the measurements and in chapter 4, feature selection and extraction will be investigated. In terminology of Hand [54], we will call the first set of all dimensions the *measurement space* and the reduced set the *feature space*.

## 2.1 Spatial Domain Measurements

The spatial domain methods operate directly with the values of the pixels or voxels of the data:

$$\tilde{\mathcal{VD}}(x) = T[\mathcal{VD}(\mathcal{N})], \quad \mathcal{N} \subset (L_r \times L_c \times L_d), \quad x \in (L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}}) \tag{2.1}$$

where $T$ is an operator defined over a neighbourhood $\mathcal{N}$ (relative to the element $x$ that belongs to the region $(L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}})$). The result of any particular operator $T^i$ will become one dimension of the multivariate measurement space $S$:

$$S^i = \tilde{\mathcal{VD}}^i \tag{2.2}$$

and the measurement space will contain as many dimensions $i$ as the operations performed on the data.

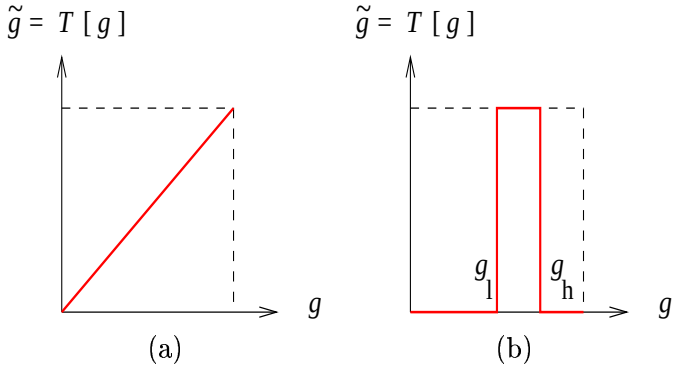



**Figure 2.2: Mapping functions of the grey level: (a) Grey levels remain unchanged, (b) Thresholding between values $g_l$ and $g_h$.**

### 2.1.1 Single Element Mappings

The simplest case arises when the neighbourhood $\mathcal{N}$ is restricted to a single element $x$. $T$ then becomes a mapping $T : G \rightarrow G$ on the grey level of the element: $\tilde{g} = T[g]$ and is sometimes called a *mapping function* [50]. Figure 2.2 shows two cases of these mappings; the first involves no change of the grey level, while the second case thresholds the grey levels between certain arbitrary low and high values $g_l, g_h$.

This technique is simple and popular and is known as *grey level thresholding*, which can be based either on global (all the image or volume) or local information. In each scheme, single or multiple thresholds for the grey levels can be assigned. The philosophy is that pixels with a grey level below a threshold belong to one region and the remaining pixels to another region. In any case the idea is to partition into regions, $object/background$, or $object_a/object_b/\ldots background$. Thresholding methods rely on the assumption that the objects to segment are distinct in their grey levels and use the histogram information, thus ignoring spatial arrangement of pixels. Although in many cases good results can be obtained, in MRI, the intensities of certain structures are often not uniform, sometimes due to inhomogeneities of the magnets, and therefore simple thresholding can divide a single structure into different regions. Another matter to consider is the noise intrinsic to the images that can lead to a misclassification. In many cases the optimal selection of the threshold is not a trivial matter.

The *histogram* [163] of an image measures the relative occurrence of elements at

certain grey levels. The histogram is defined as:

$$h(g) = \frac{\#\{x \in (L_r \times L_c \times L_d) : \mathcal{VD}(x) = g\}}{\#\{L_r \times L_c \times L_d\}}, \quad g \in G \tag{2.3}$$

where # denotes the number of elements in the set. This approach involves only the first-order measurements of a pixel [27] since the surrounding pixels (or voxels) are not considered to obtain higher order measurements. Figure 2.3 presents the two data sets and their corresponding histograms. The histogram of the human knee (b) is quite dense and although two local minima or valleys can be identified around the values of 300 and 900, using these thresholds may not be enough for segmenting the anatomical structures of the image. It can be observed that the lower grey levels, those below the threshold of 300 correspond mainly to background, which is highly noisy. The pixels with intensities between 301 and 900 roughly correspond to the region of muscle, but include parts of the skin, and the borders of other structures like bones and tissue. Many of the pixels in this grey level region correspond to transitions from one region to another. (The muscles of the thigh; Semimembranosus and Biceps Femoris in the hamstring region, do not appear as uniform as those in the calf; the Gastrocnemius, and Soleus.) The third class of pixels with intensities between 901-2787 roughly correspond to bones – femur, tibia and patella – and some tissue – Infrapatellar Fat Pad, and Suprapatellar Bursa. These tissues consist of fat and serous material, which have similar grey levels as the bones. The most important problem is that bone and tissue share the same range of grey levels in this MRI and using just thresholding it would not be possible to distinguish successfully between them.

Some other important observations about the image need to be made. Some anatomical structures like tendons, ligaments and menisci appear dark in the T1 MRI images, and therefore will be considered as background. In the upper right hand side of the image, there is a discontinuity of the shape of the leg, and the very top of the image is a solid region of pixels with intensity levels lower than 300. Although these regions

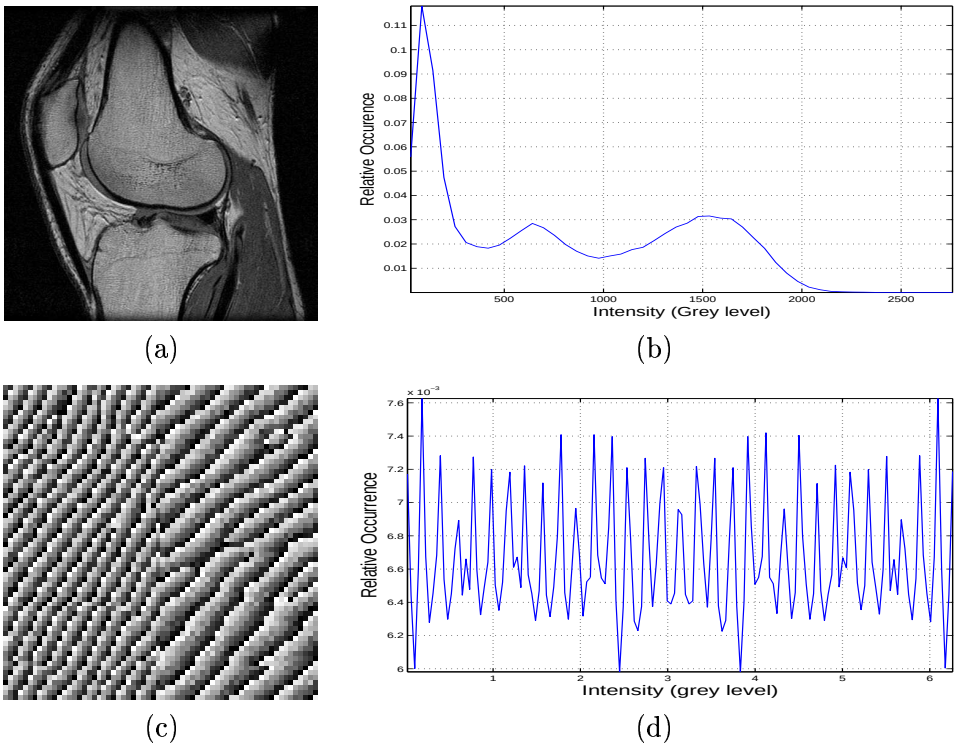
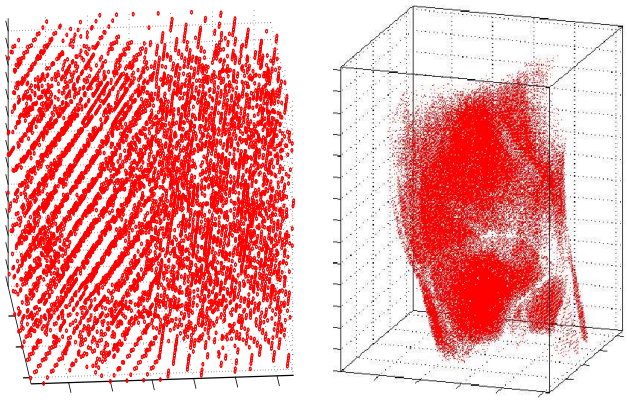



**Figure 2.3: Two images (one slice of the data sets) and their histograms: (a, b) Human Knee MRI, and (c, d) Oriented textures.**

are not background, the inhomogeneity of the magnetic field with which the image was acquired causes the effect of background. This problem will not be addressed in this work.

The histogram of the oriented textures (figure 2.3 (d)) is not as dense and smooth as the one corresponding to the human knee and it spreads through the whole grey level region without showing any valleys or hills that could point out that a thresholding could help in discrimination the textures involved.

Figure 2.4 shows the result of thresholding over the data sets previously presented. The human knee was thresholded at the $g = 1500$ and the oriented data at $g = 5.9$. For the knee some structure of the leg is visible (like the Tibia and Fibula in the lower part) but this thresholding is far from useful. For the oriented data both regions contain pixels above the threshold.



(b) (c)

**Figure 2.4: Thresholding effect on 3D sets: (a) Oriented textures thresholded at $g = 5.9$, (b) Human knee thresholded at $g = 1500$.**

### 2.1.2 Neighbourhood Filters

When $T$ comprises a neighbourhood bigger than a single element, several important measurements arise. When a convolution with kernels is performed, this can be considered as a filtering operation, which is described below. If the relative position is not taken into account, the most common measurements that can be extracted are statistical moments. These moments can describe the distributions of the sample, that is the elements of the neighbourhood, and in some cases these can help to distinguish different textures. Yet, since they do not take into account the particular position of any pixel, two very different textures could have the same distribution and therefore the same moments. Even with this limitation, some researchers use these measurements as descriptors for texture. Acha [1] performed burn diagnosis to distinguish healthy skin from burn wounds. Their textural measurements were a set of parameters; mean, standard deviation, and skewness, of the colour components of their images. Their feature selection invariably selected the mean values of lightness, hue, and chroma, so perhaps the discrimination power resides in the amplitude levels more than the texture of the images. Kapur [78] studied 3D medical data in a model-based analysis where a spatial relationship, measuring distance and orientation, between bone and cartilage is modelled from a set of manually segmented images and is later used in model-based segmentation.

Lorigo [100] performed segmentation of the bone in MRI using active contours. Both used local variance as a measure of texture.

For a neighbourhood related to the element $x = (r, c, d)$ a neighbourhood can be seen as a subset $\mathcal{N}$ of the data with the domains $L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}}$ (of size $N_r^{\mathcal{N}}, N_c^{\mathcal{N}}, N_d^{\mathcal{N}}$) related to the data in the following relations:

$$L_r^{\mathcal{N}} \subset L_r \quad L_r^{\mathcal{N}} = \{r, r+1, \ldots, r+N_r^{\mathcal{N}}\},\ 1 \leq r \leq N_r - N_r^{\mathcal{N}}, \tag{2.4}$$

$$L_c^{\mathcal{N}} \subset L_c \quad L_c^{\mathcal{N}} = \{c, c+1, \ldots, c+N_c^{\mathcal{N}}\},\ 1 \leq c \leq N_c - N_c^{\mathcal{N}}, \tag{2.5}$$

$$L_d^{\mathcal{N}} \subset L_d \quad L_d^{\mathcal{N}} = \{d, d+1, \ldots, d+N_d^{\mathcal{N}}\},\ 1 \leq d \leq N_d - N_d^{\mathcal{N}}, \tag{2.6}$$

$$(r, c, d) \in (L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}}) \tag{2.7}$$

The first four moments of the distribution; mean $\mu$, standard deviation $\sigma$, skewness $s$ and kurtosis $ku$, are obtained by:

$$\mu_{\mathcal{VD}} = \frac{1}{N_r^{\mathcal{N}} N_c^{\mathcal{N}} N_d^{\mathcal{N}}} \sum_{L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}}} \mathcal{VD}(r, c, d) \tag{2.8}$$

$$\sigma_{\mathcal{VD}} = + \sqrt{\frac{1}{N_r^{\mathcal{N}} N_c^{\mathcal{N}} N_d^{\mathcal{N}} - 1} \sum_{L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}}} \left(\mathcal{VD}(r, c, d) - \mu_{\mathcal{VD}}\right)^2} \tag{2.9}$$

$$s_{\mathcal{VD}} = \frac{1}{N_r^{\mathcal{N}} N_c^{\mathcal{N}} N_d^{\mathcal{N}} - 1} \sum_{L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}}} \left(\frac{\mathcal{VD}(r, c, d) - \mu_{\mathcal{VD}}}{\sigma}\right)^3 \tag{2.10}$$

$$ku_{\mathcal{VD}} = \frac{1}{N_r^{\mathcal{N}} N_c^{\mathcal{N}} N_d^{\mathcal{N}} - 1} \sum_{L_r^{\mathcal{N}} \times L_c^{\mathcal{N}} \times L_d^{\mathcal{N}}} \left(\frac{\mathcal{VD}(r, c, d) - \mu_{\mathcal{VD}}}{\sigma}\right)^4 \tag{2.11}$$

Figure 2.5 shows the results of calculating the four moments over a neighbourhood of size $16 \times 16$ in a sliding way (overlapping) for the two previous data sets. It is important to mention that for higher moments the accuracy of the estimation will depend on the number of points. With only 256 points, the estimation is not very accurate.

For the knee data, it is interesting to observe that the higher values of the standard deviation correspond to the transition regions, roughly close to the edges of bones tissue and skin. The lower values correspond to more homogeneous regions, and account for

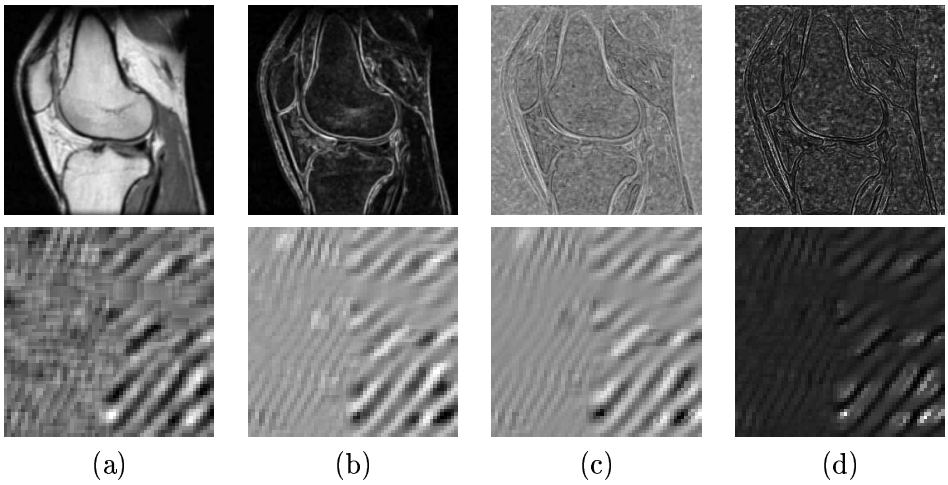



**Figure 2.5: Four moments for one slice of the examples: (a) Mean, (b) Standard Deviation, (c) Skewness, (d) Kurtosis.**

more than 92% of the total elements.

While for the knee data, some of the results can be of interest, for the oriented textures the moments resemble a blurred version of the original image. From here we can observe that if these moments are of interest it will imply that a significant difference on the grey levels is present.

### 2.1.3 Convolutional Filters

The most important characteristic of the measurements that were presented in the previous section was that the relative position of the elements inside the neighbourhood is not considered. In contrast to this, there are many methods in the literature that use a template to perform an operation among the elements inside a neighbourhood. If the template is not isotropic then the relative position is taken into account. The template or filter is used as a sliding window over the data to extract the desired measurements. The operators respond differently to vertical, horizontal, or diagonal edges, corners, lines or isolated points.

The templates or filters will be arrays of different size: $2 \times 2$, $3 \times 3$, etc. To use these filters in 3D is just necessary to extend one extra dimension and have filters of sizes: $2 \times 2 \times 2$, $3 \times 3 \times 3$, etc. The design and filtering effect will depend on the coefficients

assigned to each element of the template: $z_1, z_2, z_3, \ldots$ that will interact with the voxels $x_1, x_2, x_3, \ldots$ of the data. If the coefficients of these filters are related to the values of the data through an equation $R = z_1x_1 + z_2x_3 + z_3x_3, \ldots$, this is considered as a *linear filter*. Other operations such as the *median, maximum, minimum, etc.* are possible. In those cases the filter is considered as a *non-linear filter*. The simplest case of these filters would be when all the elements $z_i$ have equal values and the effect of the convolution is an averaging of neighbouring elements. A very common set of filters is the one proposed by Laws [93] that emerge from the combination of three basic vectors: [1 2 1] used for averaging, [−1 0 1] used for edges and [−1 2 −1] used for detecting spots. The outer product of two of these vectors can create many masks used for filtering.

These filters can easily be extended into 3D by using 3 vectors and have been used to analyse muscle fibre structures from confocal microscopic images by Lang [92]. The problem of Laws mask remains in the selection of the vectors; a great number of combinations can be generated in 3D and not all of them would be useful.

Differential filters are of particular importance for texture analysis. Applying a gradient operator $\nabla$ to the data will result in a vector:

$$\nabla\mathcal{VD} = \frac{\partial\mathcal{VD}}{\partial r}\hat{r} + \frac{\partial\mathcal{VD}}{\partial c}\hat{c} + \frac{\partial\mathcal{VD}}{\partial d}\hat{d} \tag{2.12}$$

where $\hat{r}, \hat{c}, \hat{d}$ represent unitary vectors in the direction of each dimension. In practice the partial derivatives are obtained by the difference of elements, and while a simple template like $\boxed{1}\boxed{-1}$, $\begin{array}{|c|}\hline 1 \\ \hline -1 \\ \hline \end{array}$ would perform the difference of neighbouring pixels in each direction, $3 \times 3$ operators like Sobel: $\begin{array}{|c|c|c|}\hline 1 & 0 & -1 \\ \hline 2 & 0 & -2 \\ \hline 1 & 0 & -1 \\ \hline \end{array}$, $\begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array}$ or Prewitt: $\begin{array}{|c|c|c|}\hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline \end{array}$, $\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$ are commonly used. Roberts operator $\begin{array}{|c|c|}\hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array}$, $\begin{array}{|c|c|}\hline 0 & 1 \\ \hline -1 & 0 \\ \hline \end{array}$ is used to obtain differences in the diagonals. The differences between elements will visually sharpen the data; contrary to the smoothing that is created by averaging.

The use of the magnitude of the gradient ($MG$)

$$MG = \left( \left( \frac{\partial \mathcal{VD}}{\partial r} \right)^2 + \left( \frac{\partial \mathcal{VD}}{\partial c} \right)^2 + \left( \frac{\partial \mathcal{VD}}{\partial d} \right)^2 \right)^{\frac{1}{2}} \tag{2.13}$$

has been reported by Bernasconi [5] as a texture measurement to analyse the transition between grey matter (GM) and white matter (WM) in brain MRIs. A blurred transition between GM and WM, (lower magnitude values) could be linked to Focal cortical dysplasia (FCD), a neuronal disorder. Bernasconi proposes a ratio map of GM thickness multiplied by the relative intensity of voxel values with respect to a histogram-based threshold that divides GM and WM and then divide this product by the grey level intensity gradient. Their results enhance the visual detection of lesions.

In seismic applications, Randen [125] used the gradient to detect two attributes of texture: dip and azimuth. Instead of the magnitude, they are interested in the direction, which in turn poses the problem of unwrapping in presence of noise; a non-trivial problem. They first obtain the gradient of the data and then calculate a local covariance matrix whose eigenvalues are said to describe dip and azimuth. These measures are said to be adequate for seismic data where parallel planes run along the data, but when other seismic objects are present, like faults, other processing is required.

The Zucker-Hummel filter [167]:

$$\begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{2}} & 1 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{3}} \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \begin{bmatrix} \frac{-1}{\sqrt{3}} & \frac{-1}{\sqrt{2}} & \frac{-1}{\sqrt{3}} \\ \frac{-1}{\sqrt{2}} & -1 & \frac{-1}{\sqrt{2}} \\ \frac{-1}{\sqrt{3}} & \frac{-1}{\sqrt{2}} & \frac{-1}{\sqrt{3}} \end{bmatrix} \tag{2.14}$$

has also been used as a gradient operator ([85, 86, 88, 89, 136]). Once the filter is convolved in each of the axis, either the magnitude or the orientation of the gradient at each voxel can be used to calculate three-dimensional histograms. If orientation is considered, the values are grouped into *bins* of solid angles. These histograms can be visualised with an extended Gaussian image (3D orientation indicatrix) [88]. The histograms and the

metrics that can be extracted from them - anisotropy coefficient, integral anisotropy measure or local mean curvature - can reveal important characteristics of the original data, like the anisotropy, which can be linked to different brain conditions. The measure of anisotropy in brains has shown that there is some indication of higher degree of anisotropy in brains with brain atrophy than in normal brains [136].

This filter is also used as a step of the 3D co-occurrence matrix proposed by Kovalev and Petrou [87, 88] and will be further discussed in section 2.3.2.

## 2.2 Wavelets

Wavelet decomposition and Wavelet Packet are two common techniques used to extract measurements from textured data [19, 40, 42, 91, 121, 147] since they provide a tractable way of decomposing images (or volumes) into different frequency components sub-bands at different scales.

Wavelet analysis is based on mathematical functions, the *Wavelets*, which present certain advantages over Fourier analysis when discontinuities appear in the data, since the *analysing* or *mother* Wavelet $\psi$ is a localised function limited in space (or time) and does not assume a function that stretches infinitely as the sinusoidals of the Fourier analysis. The Wavelets or *small waves* should decay to zero at $\pm\infty$ (in practice they decay very fast) so in order to cover the space of interest (which can be the real line $\mathbb{R}$) they need to be shifted along $\mathbb{R}$. This could be done with integral shifts:

$$\psi(r-k), \qquad k \in \mathbb{Z}, \tag{2.15}$$

where $\mathbb{Z} = \{\ldots, -1, 0, 1, \ldots\}$ is the set of integers. To consider different frequencies, the Wavelet needs to be dilated, one way of doing it is with a binary dilation in integral powers of 2:

$$\psi(2^l r-k), \qquad k, l \in \mathbb{Z}. \tag{2.16}$$

The signal $\psi(2^l r - k)$ is obtained from the mother Wavelet $\psi(r)$ by a binary dilation $2^j$ and a dyadic translation $k/2^l$. The function $\psi_{k,l}$ is defined as:

$$\psi_{k,l}(r) = 2^{l/2}\psi\left(2^l r - k\right), \qquad k, l \in \mathbb{Z}. \tag{2.17}$$

The scaled and translated Wavelets need to be orthogonal to each other in the same way that sines and cosines are orthogonal, i.e.:

$$\langle \psi_{k,l}(r), \psi_{m,n}(r) \rangle = 0, \quad \text{for} \quad (k,l) \neq (m,n) \qquad k, l, m, n \in \mathbb{Z}. \tag{2.18}$$

Next, for the basis to be orthonormal, the functions need to have unit length, if $\psi$ has unit length, then all of the functions $\psi_{k,l}$ will also have unit length:

$$\| \psi_{k,l}(r) \|_2 = 1. \tag{2.19}$$

Then, any function $f$ can be written as:

$$f(r) = \sum_{k,l=-\infty}^{\infty} c_{k,l}\psi_{k,l}(r), \tag{2.20}$$

where $c_{k,l}$ are called the *Wavelet coefficients*, analogous to the notion of the Fourier coefficients and are given by the inner product of the function and the Wavelet:

$$c_{k,l} = \langle f, \psi_{k,l} \rangle = \int_{-\infty}^{\infty} f(r)\psi^*_{k,l}(r)dr. \tag{2.21}$$

The Wavelet transform of a function $f(r)$ is defined as [22]:

$$\Psi(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} f(r)\psi^*\left(\frac{r-b}{a}\right) dr. \tag{2.22}$$

The Wavelets must satisfy certain conditions, of which perhaps the most important

one is the *admissibility condition* which states that:

$$\int_{-\infty}^{\infty} \frac{|\psi_\omega(\rho)|^2}{|\omega|} d\rho < \infty. \tag{2.23}$$

where $\psi_\omega(\rho)$ is the Fourier transform of the mother Wavelet $\psi(r)$. This condition implies that the function has zero mean:

$$\int_{-\infty}^{\infty} \psi(r) dr = 0, \tag{2.24}$$

which implies that the Fourier transform of the Wavelet $\psi$ vanishes at zero frequency [1].

From a signal processing point of view, it may be useful to think of the coefficients and the Wavelets as filters, in other words, we are dealing with band pass filters, not low pass filters. To cover the low pass regions, Mallat [103] introduced a *scaling function*, which does not satisfy the previous admissibility condition (and therefore it is not a Wavelet) but covers the low pass regions. In fact this function should integrate to 1. So, by a combination of a Wavelet and a scaling function it is possible to split the spectrum of a signal into a low pass region and a high pass region. This combination of filters is sometimes called a *quadrature mirror filter pair* [51]. The decomposition can continue by splitting the low pass region into another low pass and a band pass. This process is represented in figure 2.6. To prevent the dimensions of the decomposition expanding at every step, a down-sampling step is performed in order to keep the dimensions constant. In many cases the down-sampling presents no degradation of the signals but it may not always be the case. If the down-sampling step is eliminated, the decomposition will provide an over-complete representation called *Wavelet Frames* [147].

It is important to remember that $e^{jx} = cos(x) + jsin(x)$ is the only function necessary to generate the orthogonal space in Fourier analysis. While through the use of Wavelets, it is possible to extract information that can be obscured by the sinusoidals, there is a large number of Wavelets to choose from: *Haar, Daubechies, Symlets, Coiflets,*

[1] This condition prevents a Gabor filter (section 2.4.1) from being a Wavelet since it is possible that a Gabor filter will have a value different from zero at the origin of the Fourier domain.

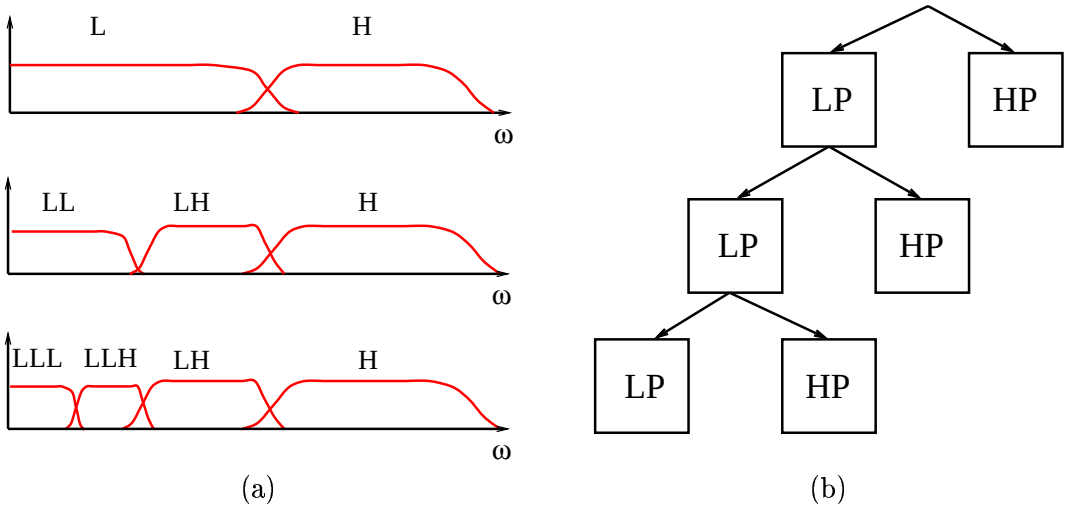



**Figure 2.6: (a) Wavelet decomposition by successively splitting the spectrum. (b) Schematic representation of the decomposition.**

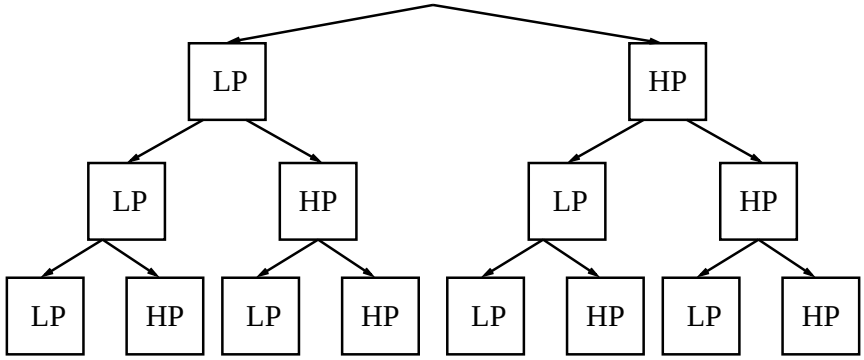



**Figure 2.7: Wavelet packet decomposition. Both high pass and low pass are further split.**

*Biorthogonal, Meyer, etc.*, and some of them have variations according to the moments of the function. The nature of the data and the application can determine which family to use, but even with this knowledge, it is not always clear how to select a certain family of Wavelets.

The decomposition does not have to be restricted to the low pass region of course. When the high pass region (or band pass) is decomposed, an *adaptive Wavelet decomposition* or *Wavelet packet* is used. Figure 2.7 shows a schematic representation of a Wavelet packet.

The previous description of the Wavelet decomposition was based on a 1D function $f(r)$. When dealing with more than one dimension, the extension is usually performed

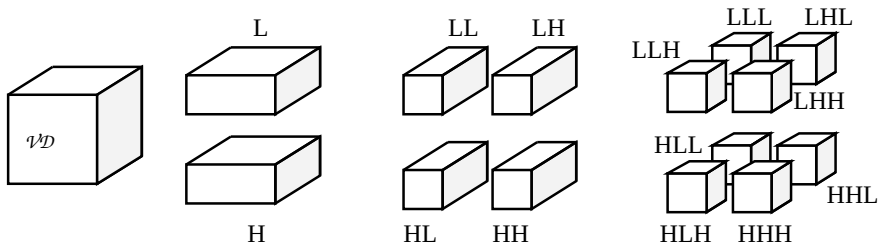
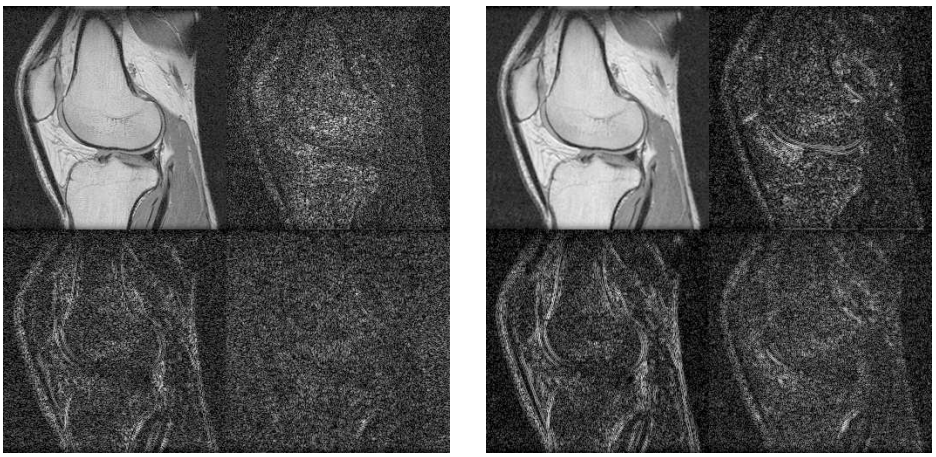



**Figure 2.8: A schematic representation of a 3D Wavelet decomposition.**



(a) (b)

**Figure 2.9: Two levels of a 2D Wavelet decomposition of one slice of the human knee MRI, (a) Level 1, (b) Level 2.**

by separable Wavelet and scaling functions applied in each dimension. In 2D, 4 options are obtained in one level of decomposition: **LL, LH, HL, HH**, that is low pass in both dimensions (LL), one low pass and one high pass in opposite dimensions (LH, HL) and high pass in both dimensions (HH). Figure 2.8 represents schematically a 3D separable Wavelet decomposition, eight different combinations of the filters (**LLL, LLH, LHL, LHH, HLL, HLH, HHL, HHH**) can be achieved in the first level of a 3D decomposition.

Figure 2.9 presents the first two levels of a 2D Wavelet decomposition (*Coiflet 1* used) of one slice of the human knee MRI.

Unser [147] has analysed texture with Wavelets and concludes that Wavelet transform is an attractive tool for characterising textures due to its properties of multiresolution and orthogonality. He mentions that having more than one level led to better

results than a single resolution analysis.

Jafari [69] studied Wavelets in 2D and 3D for the study of temporal lobe epilepsy (TLE) and concluded that the extracted features are linearly separable and the energy features derived from the 2D Wavelet transform provide higher separability compared with 3D Wavelet decomposition of the hippocampus.

## 2.3 Joint Statistics: The Co-occurrence Matrix

The co-occurrence matrix defines the joint occurrences of grey tones (or ranges of tones) and is constructed by analysing the grey levels of neighbouring pixels. The co-occurrence matrix is a widely used technique over 2D images and some extensions to three dimensions have been proposed. We begin with a description of 2D co-occurrence.

### 2.3.1 2D Co-occurrence

Let the original image $\mathcal{I}$ with dimensions for rows and columns $N_r \times N_c$ be quantised to $N_g$ grey levels. The co-occurrence matrix will be a symmetric $N_g \times N_g$ matrix that will describe the number of co-occurrences of grey levels in a certain orientation and a certain element distance. The distance can be understood as a *chess-board* distance ($D_8$) [50]. The un-normalised co-occurrence matrix entry $CM(g_1, g_2, D_8, \theta)$ records the number of times that grey levels $g_1$ and $g_2$ jointly occur at a neighbouring distance $D_8$, in the orientation $\theta$. For example, if $L_c = \{1, 2, \ldots, N_c\}$ and $L_r = \{1, 2, \ldots, N_r\}$ are the horizontal and vertical co-ordinates of an image $\mathcal{I}$, and $G = \{1, \ldots, g_1, \ldots, g_2, \ldots, N_g\}$ the set of quantised grey levels, then the values of the un-normalised co-occurrence matrix $CM(g_1, g_2)$ within a distance $D_8 = 1$ and $\theta = \frac{3\pi}{4}$ is given by:

$$\begin{aligned} CM(g_1, g_2, 1, 135^o) &= \#\{((r_1, c_1), (r_2, c_2)) \in (L_r \times L_c) \times (L_r \times L_c) \mid \\ &\quad ((r_1 - r_2 = 1, c_1 - c_2 = 1),\ \mathcal{I}(r_1, c_1) = g_1, \mathcal{I}(r_2, c_2) = g_2\}, \end{aligned} \tag{2.25}$$

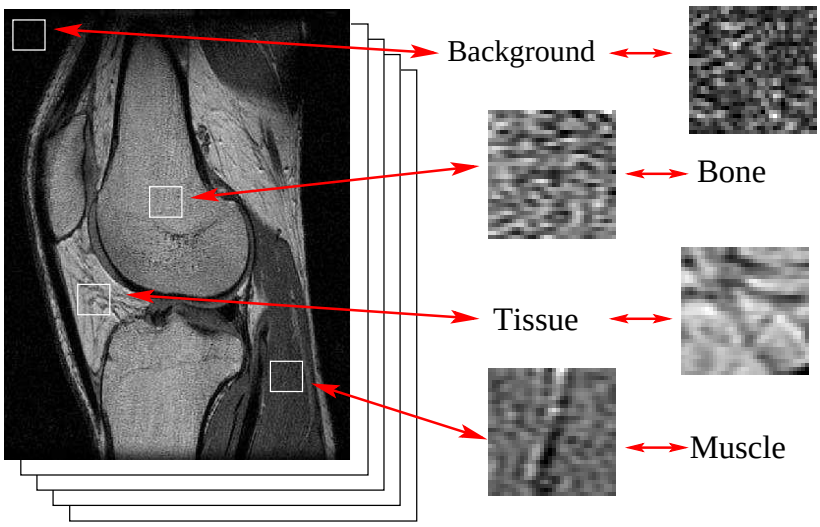



**Figure 2.10: Human knee MRI and four selected regions.**

where # denotes the number of elements in the set. In other words, the matrix will be formed counting the number of times that two pixels with values $g_1, g_2$ appear contiguous in the direction down and to the right (south-east). In this way, a co-occurrence matrix is able to measure local grey level dependence: textural coarseness and directionality. For example, in coarse images, the grey level of the pixels change slightly with distance, while for fine textures the levels change rapidly. From this matrix, different features like *entropy, uniformity, maximum probability, contrast, correlation, difference moment, inverse difference moment, correlation* can be calculated [56]. It is assumed that all the texture information is contained in this matrix.

As an example to illustrate the properties of the co-occurrence matrix, 4 training regions, namely, *background, muscle, bone, tissue*, were selected from the MRI. Figure 2.10 shows the training samples location in the original image. For every texture, the co-occurrence matrix was calculated for 4 different orientations $\theta = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$ and three distances $D_8 = \{1, 2, 3\}$. The results are presented in the figures 2.11, 2.12, 2.13, 2.14. Here are some brief observations about the co-occurrence matrices and their distributions:

**Table 2.1: Characteristics of the co-occurrence matrix**

(a) (b) (c) (d) (e) (f)

(a) High values in the main diagonal imply uniformity in the image. That is, most transitions occur between similar levels of grey.
(b) High values outside the main diagonal imply abrupt changes in the grey level, from very dark to very bright.
(c) High values in the upper part imply a darker image.
(d) High values in the lower part imply a brighter image.
(e) High values in the lower central region part imply an image whose transitions occur mainly between similar grey levels and whose histogram is roughly of Gaussian shape.
(f) In a noise image, the transitions between different grey levels should be balanced and it should be invariant to $D_8$ and $\theta$. The reverse, a balanced matrix, does not imply a noisy image.

- **Background**. The distribution of the co-occurrence matrix suggests a highly noisy nature with a skew towards the darker regions. There is a certain tendency to be more uniform in the horizontal direction ($\theta = 0$) at a distance of $D_8 = 1$ which is the only matrix that is significantly different from the rest of the set.

- **Muscle**. The co-occurrence matrix is highly concentrated in the central region, the middle grey levels, and there is a lower spread compared with the background. A vertical structure can be observed, this in turn gives a certain vertical and horizontal ($\theta = 0, \frac{\pi}{2}$) uniformity, only at distance $D_8 = 1$.

- **Bone**. The nature of the bone is highly noisy as it can be observed from the matrices, but, compared with the background, there is no skew towards dark or bright. As in the case of the muscle there is certain horizontal uniformity, but not vertical.

- **Tissue**. The distribution is skewed towards the brighter levels. The tissue presents several major structures with a $135^o$ orientation, this makes the $\theta = \frac{\pi}{4}$ matrix to be more dispersed than the other orientations for distance $D_8 = 1$. As the distance increases the matrices spread towards a noisy configuration.

When observing the matrices, it is important to note which textures are invariant to distance or angle. Some characteristics of the co-occurrence matrix are presented in table 2.1.

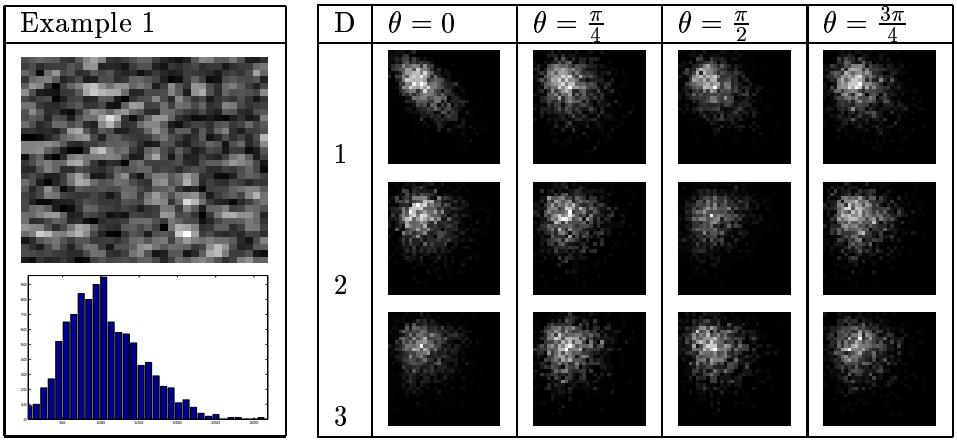



**Figure 2.11: A sample of background, its histogram and co-occurrence matrices.**

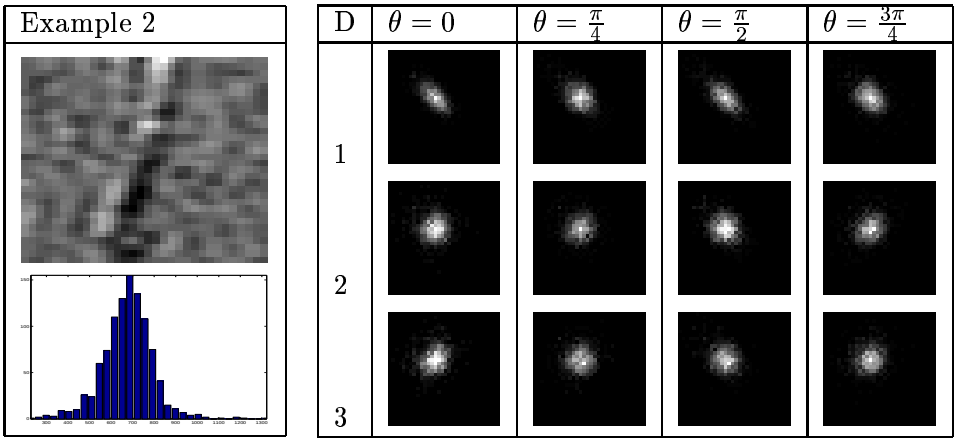



**Figure 2.12: A sample of muscle, its histogram and co-occurrence matrices.**

Some of the features determine the presence of a certain degree of organisation, but others measure the complexity of the grey level transitions, and therefore are more difficult to identify. The textural features[2] as defined in [55] and [56] are presented in tables 2.2 and 2.3.

[2]There are slight differences in the features presented in both texts, notice for instance that Contrast and Correlation are sometimes equivalently displayed as:
$\varphi_2 = \sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} |g_1 - g_2|^k (cm(g_1, g_2))$, $\varphi_3 = \sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} \frac{(g_1-\mu)(g_2-\mu)cm(g_1,g_2)}{\sigma^2}$

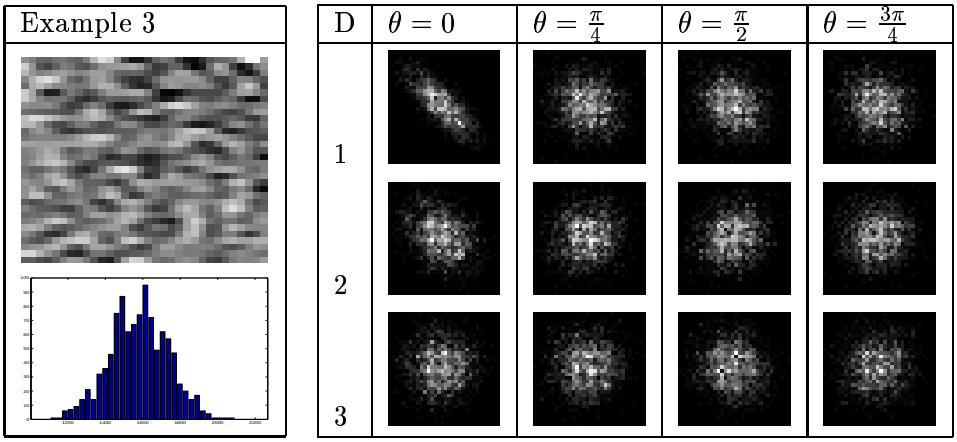



**Figure 2.13: A sample of bone, its histogram and co-occurrence matrices.**

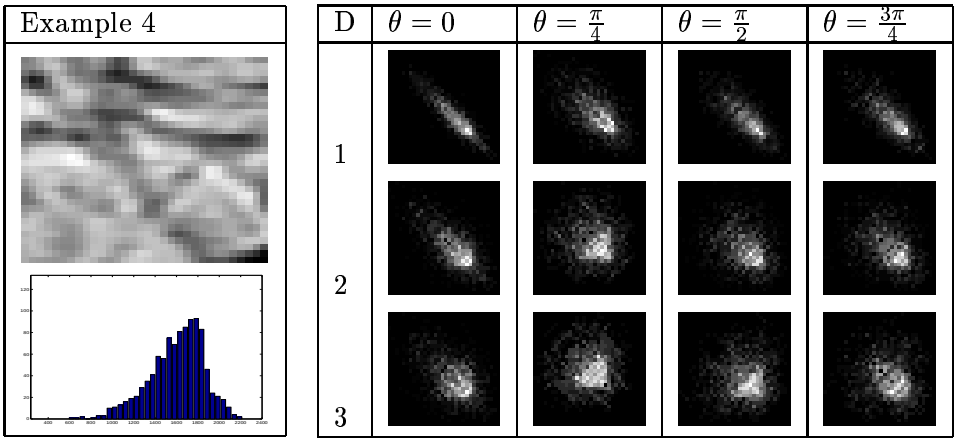



**Figure 2.14: A sample of tissue, its histogram and co-occurrence matrices.**

Any single matrix feature or combination can be used to represent the local regional properties but it can be difficult to predict which combination will help discriminate regions without some experimentation.

The major disadvantage of the co-occurrence matrix is that its dimensions will depend on the number of grey levels. In many cases, the grey levels are quantised to reduce the computational cost and information is lost inevitably. Otherwise, the computational burden is huge. To keep computation tractable, the grey levels are quantised, $D_8$ is restricted to a small neighbourhood and a limited number of angles $\theta$ are cho-

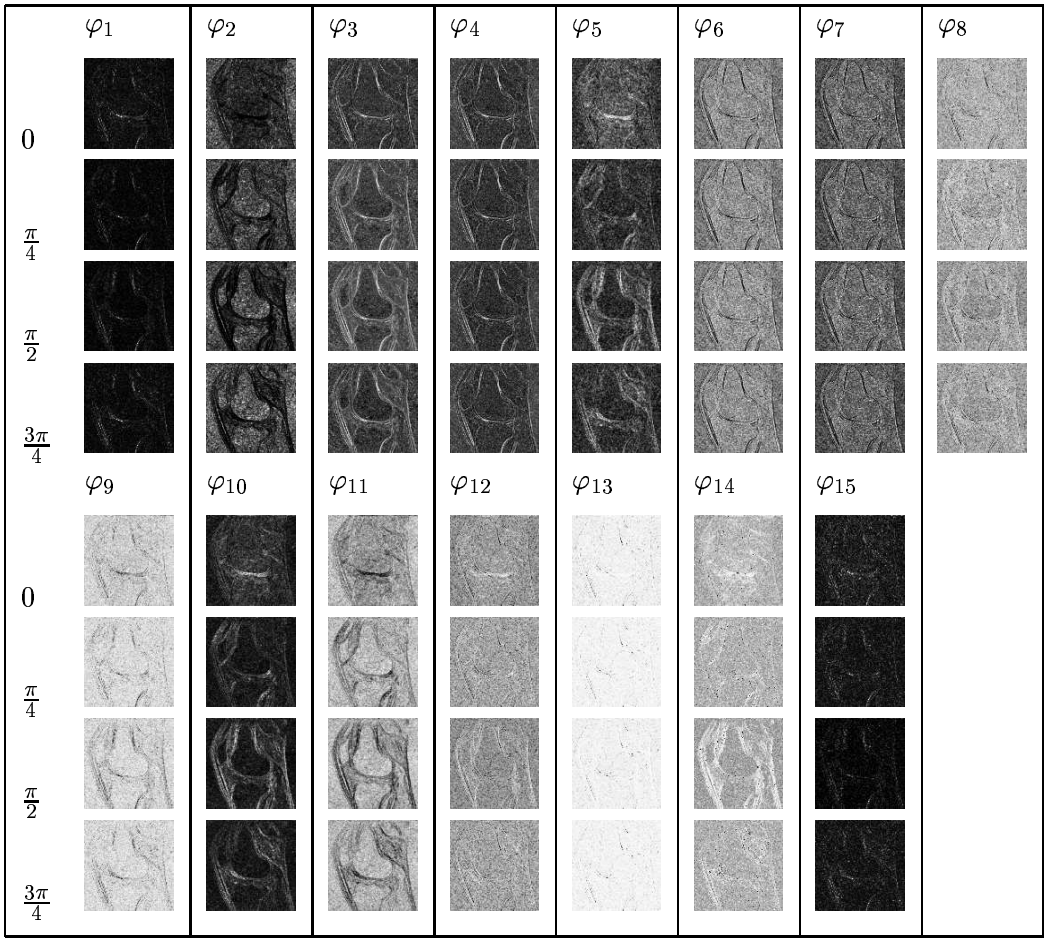



**Figure 2.15: 15 Features of the co-occurrence matrix for 4 different angles and distance $D_8 = 1$.**

sen. An important implication of quantising the grey levels is that the sparsity of the co-occurrence matrix is reduced.

The images are normally processed by blocks of a certain size $4 \times 4$, $8 \times 8$, $16 \times 16$, etc., and they have an overlap which allows for rapid computation of the matrix [2, 24]. Even with these restrictions, the number of features can be very high and a selection method is required. If 4 angles are selected, with 15 textural features, the space will be of 60 features for every distance $D_8$; if $D_8 = 1, 2, 3$, the feature space will have 180 dimensions. Figure 2.15 presents 60 features calculated on one slice of the MRI.

**Table 2.2: Textural features of the co-occurrence matrix [55] [56]:**

| | | |
|---|---|---|
| *Angular Second Moment (Uniformity)* | $\varphi_1 = \sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} \{cm(g_1, g_2)\}^2$ | (2.26) |
| *Element Difference Moment (Contrast)* | $\varphi_2 = \sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{\substack{g_1=1 \\ \lvert g_1-g_2 \rvert = n}}^{N_g} \sum_{g_2=1}^{N_g} cm(g_1, g_2) \right\}$ | (2.27) |
| *Correlation* | $\varphi_3 = \frac{\sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} (g_1 g_2) cm(g_1, g_2) - \mu_x \mu_y}{\sigma_x \sigma_y}$ | (2.28) |
| *Sum of Squares (Variance)* | $\varphi_4 = \sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} (g_1 - \mu)^2 cm(g_1, g_2)$ | (2.29) |
| *Inverse Difference Moment* | $\varphi_5 = \sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} \frac{cm(g_1, g_2)}{1 + \mid g_1 - g_2 \mid^k}$ | (2.30) |
| *Sum Average* | $\varphi_6 = \sum_{g_1=2}^{2N_g} g_1 p_{x+y}(g_1)$ | (2.31) |
| *Sum Variance* | $\varphi_7 = \sum_{g_1=2}^{2N_g} (g_1 - \varphi_8)^2 p_{x+y}(g_1)$ | (2.32) |
| *Sum Entropy* | $\varphi_8 = -\sum_{g_1=2}^{2N_g} p_{x+y}(g_1) \log\{p_{x+y}(g_1)\}$ | (2.33) |
| *Entropy* | $\varphi_9 = -\sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} cm(g_1, g_2) \log\{cm(g_1, g_2)\}$ | (2.34) |
| *Difference Variance* | $\varphi_{10} = Var\{p_{x-y}(g_3)\}$ | (2.35) |
| *Difference Entropy* | $\varphi_{11} = -\sum_{g_1=0}^{N_g-1} p_{x-y}(g_1) \log\{p_{x-y}(g_1)\}$ | (2.36) |
| *Information Measures of Correlation* | $\varphi_{12} = -\frac{HXY - HXY1}{max\{HX, HY\}}$ | (2.37) |
| | $\varphi_{13} = (1 - e^{-2(HXY2 - HXY)})^{\frac{1}{2}}$ | (2.38) |
| *Maximal Correlation Coefficient* | $\varphi_{14} = (\text{Sec. Larg. Eigenvalue of } Q)^{\frac{1}{2}}$ | (2.39) |
| *Maximum Probability* | $\varphi_{15} = max\{cm(g_1, g_2)\}$ | (2.40) |

**Table 2.3: Notation used for the co-occurrence matrix and its features [55] [56]**

| | |
|---|---|
| $cm(g_1, g_2) = \frac{CM(g_1,g_2)}{\sum CM}$, | $(g_1, g_2)th$ entry in a normalised matrix. |
| $N_g$ | Number of grey levels of the quantised image. |
| $p_x(g_1) = \sum_{g_2=1}^{N_g} cm(g_1, g_2)$ | $g_1$th entry in the marginal-probability matrix by summing the rows. |
| $p_y(g_2) = \sum_{g_1=1}^{N_g} cm(g_1, g_2)$ | $g_2$th entry in the marginal-probability matrix by summing the columns. |
| $p_{x+y}(g_3) = \sum_{\substack{g_1=1 \\ g_1+g_2=g_3}}^{N_g} \sum_{g_2=1}^{N_g} cm(g_1, g_2)$, | $g_3 = 2, 3, \ldots, 2N_g$. |
| $p_{x-y}(g_3) = \sum_{\substack{g_1=1 \\ \lvert g_1-g_2\rvert=g_3}}^{N_g} \sum_{g_2=1}^{N_g} cm(g_1, g_2)$, | $g_3 = 0, 1, \ldots, N_g - 1$. |
| $HXY = -\sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} cm(g_1, g_2) \log\{cm(g_1, g_2)\}$ | Entropy of $cm(g_1, g_2)$ |
| $HX = -\sum_{g_1=1}^{N_g} p_x(g_1) \log\{p_x(g_1)\}$ | Entropy of $p_x(g_1)$ |
| $HY = -\sum_{g_2=1}^{N_g} p_y(g_2) \log\{p_y(g_2)\}$ | Entropy of $p_y(g_2)$ |
| $HXY1 = -\sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} cm(g_1, g_2) \log\{p_x(g_1)p_y(g_2)\}$ | |
| $HXY2 = -\sum_{g_1=1}^{N_g} \sum_{g_2=1}^{N_g} p_x(g_1)p_y(g_2) \log\{p_x(g_1)p_y(g_2)\}$ | |
| $Q(g_1, g_2) = \sum_{g_3} \frac{cm(g_1,g_3)cm(g_2,g_3)}{p_x(g_1)p_y(g_2)}$ | |

### 2.3.2 3D Co-occurrence

When the co-occurrence of volumetric data sets $\mathcal{VD}$ is to be analysed, the un-normalised co-occurrence matrix will become a five dimensional matrix:

$$CM(g_1, g_2, D_8, \theta, d), \tag{2.41}$$

where $d$ will represent the slice separation of the voxels. Alternatively, two directions: $\theta, \phi$ could be used. The computational complexity of this technique will grow considerably with this extension, the co-occurrence matrix could also be a very sparse matrix. The sparsity of the matrix could imply that quantising could improve the complexity,

or special techniques of sparse matrix could also be used [24].

Early use of co-occurrence matrices with 3D data was reported by Ip and Lam [68] who partitioned the data, calculated the co-occurrence matrix and three features for each partition and then classified the data into homogeneous regions. The homogeneity is based on the features of the sub-partitions.

A generalised multidimensional co-occurrence matrix was presented by Kovalev and Petrou [87]. They propose $M$-dimensional matrices, which measure the occurrence of *attributes* or *relations* of the elements of the data; grey level is one attribute but others (such as magnitude of a local gradient) are possible. Rather than using the matrices themselves, features are extracted and used in several applications: to discriminate between normal brains and brains with pathologies, to detect defects on textures, and to recognise shapes. These matrices have been used in several publications: to measure texture anisotropy [88], for the analysis of MRI brain data sets [85], to detect age and gender effects in structural brain asymmetry [86], and the detection of schizophrenic patients [89].

In [89] Kovalev, Petrou and Suckling use the magnitude of the gradient calculated with the Zucker-Hummel filter over the data. The authors use 3D co-occurrence to discriminate between schizophrenic patients and controls. The data of T1-MRIs are filtered spatially and then the co-occurrence matrix will be a function of

$$CM(MG_1, MG_2, D_8, \theta, d), \tag{2.42}$$

where $MG$ is the magnitude of the gradient at a certain voxel. This method requires an empirical threshold to discard gradient vectors with magnitudes lower than 75 units (from a range 0-600) as a noise removal of the data. The authors reported that not all the slices of the brain were suitable for discrimination between schizophrenics and normal control subjects. It is the most inferior part of the brain, in particular the tissue close to the sulci, which gives the slices whose features provide discrimination between the populations.

Another variation of the co-occurrence matrix is reported by Kovalev and Kruggel [85] where the matrix will include both grey level intensity and gradient magnitude:

$$CM(g_1, g_2, MG_1, MG_2, D_8, \theta, d). \tag{2.43}$$

This matrix is called by the authors IIGGAD (Intensity, Gradient, Angle, Distance) who also consider a reduced version like IID, GGD, GAD. The traditional co-occurrence matrix will be a particular case of the IIGGAD matrix. This matrix is used in the discrimination of brain data sets of patients with mild cognitive disturbance. This technique is also used to segment brain lesions but the method is not straightforward. First, a representative descriptor of the lesion is required. For this, a VOI that contains the lesion is required, which leads to a training set manually determined. Then, a mapping function is used to determine the probability of a voxel being in the lesion or not, based on the distances between the current VOI and the representative VOI of the lesion. Again this step needs tuning. The segmentation is carried out with a sliding-window analysing the data for each VOI. Post-processing with knowledge of the WM area is required to discard false positives. The method segments the lesions of the WM but there is no clinical validation of the results.

Kovalev and Kruggel [86] have also studied the brain asymmetry in 3D MRI with co-occurrence matrices. Gradients as well as intensities are included in the matrix, which is then used to analyse the brain asymmetry. It was found that male brains are more asymmetric than females and that changes of asymmetry with age are less prominent. The results reported correspond closely to other techniques and they propose the use of texture-based methods for digital morphometry in neuroscience.

Mahmoud-Ghoneim [101] has reported using 3D co-occurrence matrices in MRI data for evaluation of gliomas. An experienced neuroradiologist selected homogeneous volumes of interest (VOI) corresponding to a particular tissue: WM, active tumour, necrosis, oedema, etc. Co-occurrence matrices were obtained from these VOIs and their parameters were used in a pair-wise discrimination between the classes. The results are

compared against 2D co-occurrence matrices, which are outperformed by the 3D approach. Herlidou has used a similar methodology for the evaluation of osteoporosis [61], diseased skeletal muscle [62] and intracranial tumours [63]. Regions of interest (ROI) were manually selected from the data and then measurements were extracted with different techniques with the objective of discriminating between the classes of the ROIs. As with other techniques, the partitioning of the data presents a problem: if the region or volume of interest (VOI/ROI) selected is too small, it will not capture the structure of a texture, if it is too big, it will not be good for segmentation. It is important to mention that the objective of this thesis is not to discriminate between two separate volumes, but from a single region (the whole MRI set) distinguish the presence of two or more different structures characterised by their texture.

In summary, co-occurrence matrices can be extended without much trouble into 3D and are a good way of characterising textural properties of the data. The segmentation or classification with these matrices presents several disadvantages: first, the computational complexity to calculate the matrix is considerable for even small ranges of grey levels (even with fast methods [2, 24]), many times the ranges are quantised into a lower range with the possible loss of information. Second, the parameters on which the matrix depends: distance, orientation and number of bins in the 3D cases can yield a huge number of possible different matrices. If this dimensionality problem is not enough, there is a large number of features that can be extracted from the matrix and choosing the adequate features can depend on the data analysed and the specific analysis to be performed.

## 2.4 Frequency Filtering

In this section we will discuss some filters that can be applied to textured data. A filter is defined by Webster as *a device or material for suppressing or minimising waves or oscillations of certain frequencies (as of electricity, light, or sound)* [108]. In the context of images or volumes, these filters can be understood as a technique that will modify the

frequential content of the data. As mentioned before, textures can vary in their spectral distribution in the frequency domain, and therefore a set of sub-band filters can help in their discrimination.

The classification process goes beyond the filtering stage, and according to Randen [124] the process consists of the following steps: *filtering, non-linearity, smoothing, normalising non-linearity* and *classification*. The first step corresponds to the output of the filters, then, a local energy function (LEF) is obtained with the non-linearity and the smoothing. The normalising is an optional step before feeding everything to a classifier. Figure 2.16 demonstrates some of these steps. In this section we will concentrate on the filter responses and for all of them we will use the magnitude as the non-linearity. The smoothing step is quite an important part of the classification as it may influence considerably the results. Section 3.2 will discuss in more detail the impact of the size of the LEF in the classification.

The spatial and frequency domains are related through the use of the Fourier transform, such that a filter $F$ in the spatial domain (that is the mask or template) will be used through a convolution with the data:

$$\tilde{\mathcal{VD}} = F * \mathcal{VD} \tag{2.44}$$

where $\tilde{\mathcal{VD}}$ is the filtered data. From the convolution theorem [50] the same effect can be obtained in the frequency domain:

$$\tilde{\mathcal{VD}}_\omega = F_\omega \mathcal{VD}_\omega \tag{2.45}$$

where the $\tilde{\mathcal{VD}}_\omega = \mathscr{F}[\tilde{\mathcal{VD}}]$, $\mathcal{VD}_\omega = \mathscr{F}[\mathcal{VD}]$, and $F_\omega = \mathscr{F}[F]$ are the corresponding Fourier transforms. The filters in the Fourier domain are named after the frequencies that are to be allowed to pass through them: *low pass, band pass* and *high pass* filters. Figure 2.17 shows the filter impulse response and the resulting filtered human knee for low pass, high pass and band pass filters.

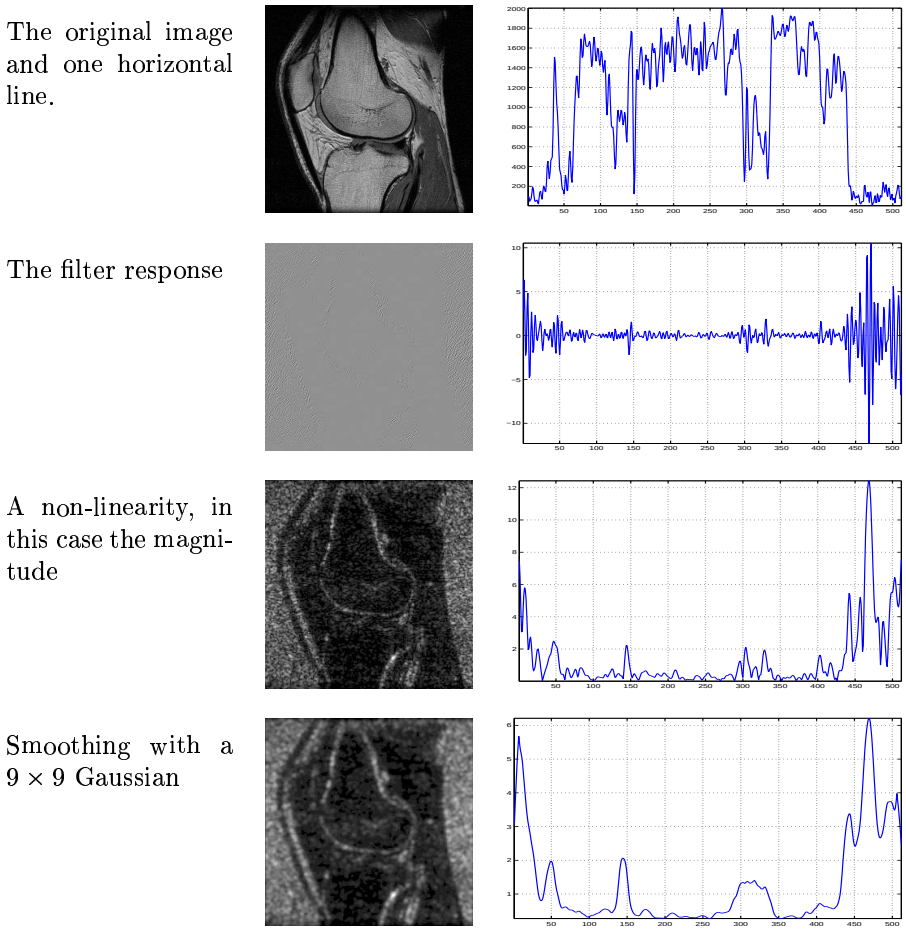



**Figure 2.16: A filtering measurement extraction process.**

The filters just presented have a very simple formulation and combinations through different frequencies will form a *filter bank*, which is an array of band pass filters that span the whole frequency domain spectrum. The idea behind the bank is to select and isolate individual frequency components. Besides the frequency, in 2D and 3D there is another important element of the filters, the *orientation*. The filters previously presented vary only in their frequency but remain isotropic with respect to the orientation of the filter; these filters are considered *ring* filters for the shape of the magnitude in the frequency domain. In contrast, the *wedge* filters will span the whole frequencies but only in certain orientations. Figure 2.18 presents some examples of these filters. Of course, the filters can be combined to concentrate only on a certain frequency and a certain orientation,

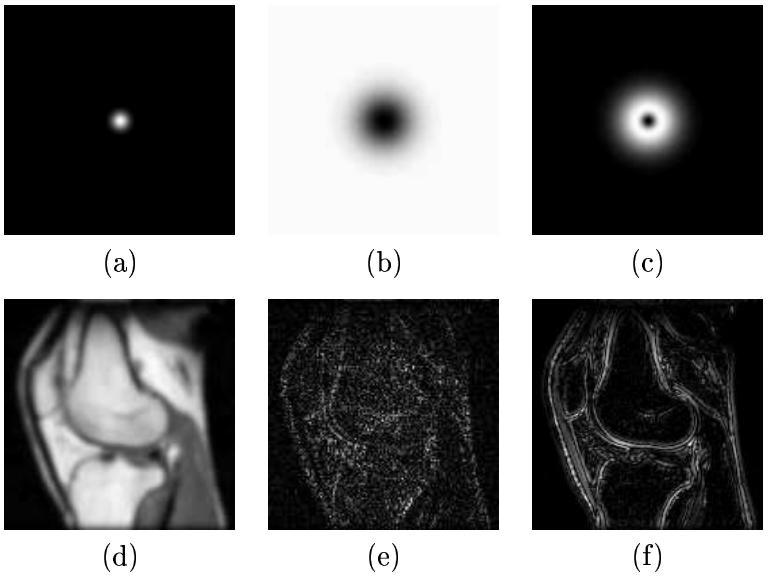



**Figure 2.17: Frequency filtering of the Human knee MR: Top Row (a) Low pass filter, (b) High pass filter, (c) Band pass filter. Bottom Row (d,e,f) Corresponding filtered images.**

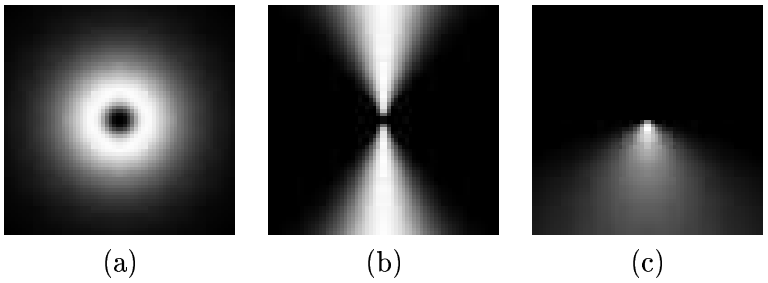



**Figure 2.18: Different frequency filters: (a) Ring filter, (b) Wedge filter (c) Lognormal filter [155].**

so called *sub-band filtering*. Many varieties of sub-band filtering schemes exist, perhaps the most common is *Gabor filters* which will be described in the next section.

### 2.4.1 Sub-band Filtering with Gabor Filters

The multichannel filtering approach to texture is inspired by the human visual system that can segment textures preattentively [102]. Experiments in psychophysical and neurophysiological data have led us to think that the human visual system performs some local spatial-frequency analysis on the retinal image by a bank of tuned band pass filters [38]. In the context of communication systems, Gabor [46] presented the

concept of local frequency, which has been used in computer vision by many researchers in the form of a multichannel filter bank [15, 71, 81, 122]. One of the advantages of this approach is the use of simple statistics of grey values of the filtered images as features or measurements of the textures.

Jain and Farrokhnia [71] present the Gabor filter as an even-symmetric function whose impulse response is the product of a Gaussian function $\mathcal{G}_a$ of parameters $(\mu, \sigma^2)$ and a modulating cosine. In 3D, the function is [130]:

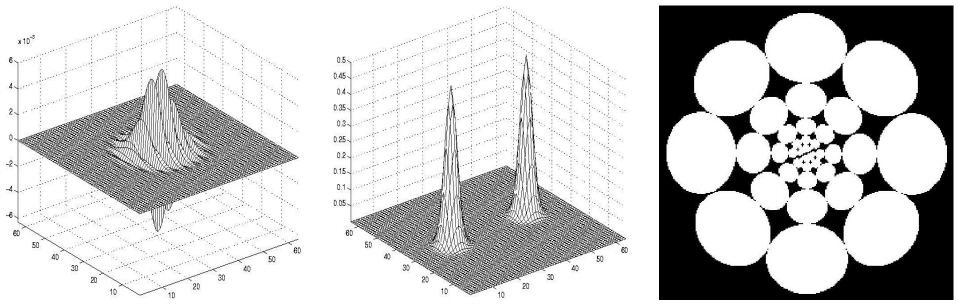
$$F^{\mathcal{G}} = exp\left\{-\frac{1}{2}\left(\frac{r^2}{\sigma_r^2} + \frac{c^2}{\sigma_c^2} + \frac{d^2}{\sigma_d^2}\right)\right\} cos(2\pi(r\rho_0 + c\kappa_0 + d\delta_0)), \tag{2.46}$$

where $\rho_0, \kappa_0, \delta_0$ are the frequencies corresponding to the centre of the filter, and $\sigma_r^2, \sigma_c^2, \sigma_d^2$ are the constants that define the Gaussian envelope. The Fourier transform of equation 2.46 is:

$$\begin{aligned} F_\omega^{\mathcal{G}} &= A\ exp\left\{-\frac{1}{2}\left(\frac{(\rho-\rho_0)^2}{\sigma_\rho^2} + \frac{(\kappa-\kappa_0)^2}{\sigma_\kappa^2} + \frac{(\delta-\delta_0)^2}{\sigma_\delta^2}\right)\right\} \\ &+ A\ exp\left\{-\frac{1}{2}\left(\frac{(\rho+\rho_0)^2}{\sigma_\rho^2} + \frac{(\kappa+\kappa_0)^2}{\sigma_\kappa^2} + \frac{(\delta+\delta_0)^2}{\sigma_\delta^2}\right)\right\}, \end{aligned} \tag{2.47}$$

where $\sigma_\rho = \frac{1}{2\pi\sigma_r}, \sigma_\kappa = \frac{1}{2\pi\sigma_c}, \sigma_\delta = \frac{1}{2\pi\sigma_d}$, and $A = 2\pi\sigma_r\sigma_c\sigma_d$. The filter has two real-valued lobes, of Gaussian shape that have been shifted $\pm(\rho_0, \kappa_0, \delta_0)$ frequency units along the frequency axes $\pm(\rho, \kappa, \delta)$ and rotated by a certain angle $(\theta, \phi)$ with respect to the positive $\rho$ axis. Figure 2.19 (a,b) presents a 2D filter in the spatial and Fourier domains.

The filter-bank is typically arranged in a *rosette* (figure 2.19 (c)) with several radial frequencies and orientations. The rosette is designed to cover the 2D frequency plane by overlapping filters whose centres lie in concentric circles with respect to the origin. The orientation and bandwidths are designed such that filters with the same radial frequency will overlap at the 50% of their amplitudes. In [71] it is recommended to use four orientations $\theta = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$, and radial frequencies at octaves. The use of Gabor filters for the extraction of texture measurements has been widely used in 2D



(a) (b) (c)

**Figure 2.19: 2D even symmetric Gabor filters in: (a) Spatial domain, (b) Fourier domain. (c) A filter bank arranged in a *rosette*, 5 frequencies, 4 orientations.**

[10, 15, 37, 38, 71, 124, 154].

As an example, the oriented pattern data of figure 2.1 is filtered with a 3D Gabor filter. Figure 2.20 (a) shows the envelope of the filter in the Fourier domain, this filter was multiplied with the data in the Fourier domain and one slice of the result in the spatial domain is presented in figure 2.20 (c). The presence of two classes appears clearly. By thresholding at the midpoint between the grey levels of the filtered data, two classes can be roughly segmented 2.20 (d,e).

The use of Gabor filters in 3D is not as common as in 2D. Zhan [166] approximates the complete set of 3D Gabor features with two banks of 2D filters located at the orthogonal coronal and axial planes. In their application of Ultrasound prostate images, they claim that this approximation is sufficient to characterise the texture of the prostate. This approach is clearly limited since it is only analysing 2 planes of a whole 3D volume. If the texture were of high frequencies that do not lie in either plane, then the characterisation would fail.

Other cases of 3D Gabor filters have been reported by Bigun [9] who describes the design of 3D filters for image sequences, Fernandez [42] who combines wavelets and Gabor filters for segmentation of 3D seismic sections and Rousseau [130] who applied the filters to clinical ultrasound volumes of carotid. For the design of a filter bank in 3D the radial frequencies are also taken in octaves and the orientation of azimuth and elevation can be both restricted to 4 angles: $(\theta, \phi) = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$ which yield a total of

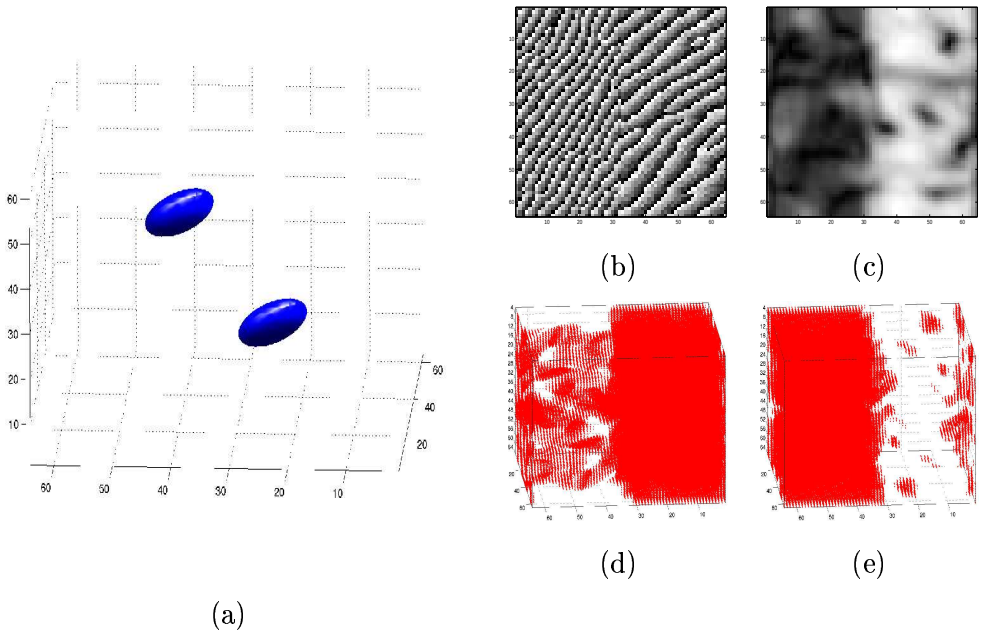



**Figure 2.20: (a) An even symmetric 3D Gabor in the Fourier domain, (b) One slice of the Oriented pattern data and (c) its filtered version with the filter from (a). (c,d) Two classes obtained from thresholding the filtered data.**

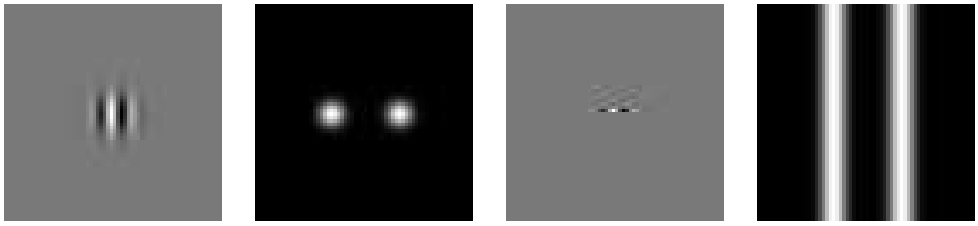
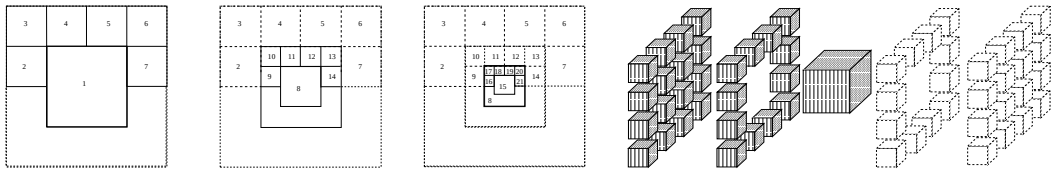



**Figure 2.21: Comparison of 2D and 1D Gabor filters. An even symmetric 2D Gabor in: (a) spatial domain, and (b) Fourier domain. A 1D Gabor filter in: (c) spatial domain, and (d) Fourier domain.**

13 orientations.

In some cases, 1D Gabor filters have been used over data of more than one dimension ([90, 126]). The essence of the Gabor filter remains, in the sense that a cosine modulates a Gaussian function, but the filters change notoriously. Consider the following comparison between 1D and 2D filters presented in figure 2.21. The 2D filter is localised in frequency while the 1D filter spans through the Fourier domain in one dimension allowing a range of radial frequencies and orientations to be covered by the filter.



(a) (b) (c) (d)

**Figure 2.22: 2D and 3D Second Orientation Pyramid (SOP) tessellation. Solid lines indicate the filters added at the present order while dotted lines indicate filters added in higher orders, as the central region is sub-divided. (a) 2D order 1, (b) 2D order 2, (c) 2D order 3, and (d) 3D order 1.**

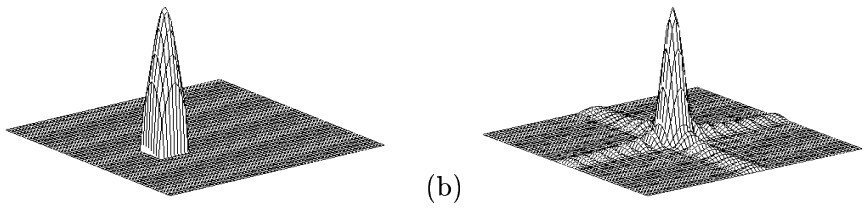
### 2.4.2 Sub-band Filtering with Second Orientation Pyramid

Wilson and Spann [161] proposed a set of operations that subdivide the frequency domain of an image into smaller regions by the use of two operators *quadrature* and *centre-surround*. By the combination of these operations, it is possible to construct different tessellations of the space, one of which is the Second Order Pyramid (SOP) (figure 2.22). In this work, a band-limited filter based on truncated Gaussians (figure 2.23) has been used to approximate the finite prolate spheroidal sequences (FPSS). The filters are real, band-limited functions which cover the Fourier half-plane. Since the Fourier transform is symmetric, it is possible to use only half-plane or half-volume and still keep the frequency information. A description of the sub-band filtering with SOP process will follow.

Any given volume $\mathcal{VD}$ whose centred Fourier transform is $\mathcal{VD}_\omega = \mathscr{F}[\mathcal{VD}]$ can be subdivided into a set of $i$ regions $L_r^i \times L_c^i \times L_d^i$: $L_r^i = \{r, r+1, \ldots, r+N_r^i\}$, $1 \leq r \leq N_r - N_r^i$, $L_c^i = \{c, c+1, \ldots, c+N_c^i\}$, $1 \leq c \leq N_c - N_c^i$, $L_d^i = \{d, d+1, \ldots, d+N_d^i\}$, $1 \leq d \leq N_d - N_d^i$, that follow the conditions:

$$L_r^i \subset L_r, \sum_i N_r^i = N_r, \quad L_c^i \subset L_c, \sum_i N_c^i = N_c, \quad L_d^i \subset L_d, \sum_i N_d^i = N_d, \quad (2.48)$$
$$(L_r^i \times L_c^i \times L_d^i) \cap (L_r^j \times L_c^j \times L_d^j) = \{\phi\}, \quad i \neq j.$$

In 2D, the SOP tessellation involves a set of 7 filters, one for the low pass region and six for the high pass (figure 2.22 (a)). In 3D, the tessellation will consist of 28 filters for



(a) (b)

**Figure 2.23: Band-limited 2D Gaussian filter (a) Frequency domain $F^i_\omega$, (b) Magnitude of spatial domain $|F^i|$.**

the high pass region and one for the low pass (figure 2.22 (d)). The $i$-th filter $F^i_\omega$ in the Fourier domain ($F^i_\omega = \mathscr{F}[F^i]$) is related to the $i$-th subdivision of the frequency domain as:

$$L_r \times L_c \times L_d; F^i_\omega : \begin{cases} L^i_r \times L^i_c \times L^i_d & \rightarrow \quad \mathscr{G}_a(\mu^i, \Sigma^i) \\ (L^i_r \times L^i_c \times L^i_d)^c & \rightarrow \quad 0 \end{cases} \quad \forall i \in SOP \tag{2.49}$$

where $\mathscr{G}_a$ describes a Gaussian function, with parameters $\mu^i$, the centre of the region $i$, and $\Sigma^i$ is the co-variance matrix that will provide a cut-off of 0.5 at the limit of the band (for 2D figure 2.23). In 3D, the filters will again be formed by truncated 3D Gaussians in an octave-wise tessellation that resembles a regular Oct-tree configuration. In the case of MRI data, these filters can be applied directly to the K-space. (The image that is presented as an MRI is actually the inverse Fourier transform of signals detected in the MRI process, thus the K-space looks like the Fourier transform of the image that is being filtered.)

The measurement space $S$ in its frequency and spatial domains will be defined as:

$$S^i_w(\rho, \kappa, \delta) = F^i_\omega(\rho, \kappa, \delta)\ \mathcal{VD}_\omega(\rho, \kappa, \delta) \quad \forall(\rho, \kappa, \delta) \in (L_r \times L_c \times L_d), \tag{2.50}$$

$$S^i = |\mathscr{F}^{-1}[S^i_\omega]| \tag{2.51}$$

The same methodology for the first order of the SOP can be extended to the next orders. At every step, one of the filters will contain the low pass (i.e. the centre) of the region analysed, $\mathcal{VD}_\omega$ for the first order, and the six (2D) or 28 (3D) remaining will subdivide the high pass bands of the surround of the region.

For simplicity we only detail the co-ordinate systems in 2D:

$$\begin{aligned}
\textbf{Centre}: F^1: \quad & L_r^1 = \{\frac{N_r}{4}+1,\ldots,\frac{3N_r}{4}\}, L_c^1 = \{\frac{N_c}{4}+1,\ldots,\frac{3N_c}{4}\},\\
\textbf{Surround}: F^{2-7}: \quad & L_r^{3,4,5,6} = \{1,\ldots,\frac{N_r}{4}\}, L_r^{2,7} = \{\frac{N_r}{4}+1,\ldots,\frac{N_r}{2}\},\\
& L_c^{2,3} = \{1,\ldots,\frac{N_c}{4}\}, L_c^4 = \{\frac{N_c}{4}+1,\ldots,\frac{N_c}{2}\},\\
& L_c^5 = \{\frac{N_c}{2}+1,\ldots,\frac{3N_c}{4}\}, L_c^{6,7} = \{\frac{3N_c}{4}+1,\ldots,N_c\}.
\end{aligned}$$

For a pyramid of order 2, the region to be subdivided will be the central region (of order 1) described by $(L_r^1(1) \times L_c^1(1))$ which will become $(L_r(2) \times L_c(2))$ with dimensions $N_r(2) = \frac{N_r(1)}{2}, N_c(2) = \frac{N_c(1)}{2}$, (or in general $N_{r,c}(o+1) = \frac{N_{r,c}(o)}{2}$, for any order $o$). It is assumed that $N_r(1) = 2^a, N_c(1) = 2^b, N_d(1) = 2^c$ so that the results of the divisions are always integer values. The horizontal and vertical frequency domains are expressed by: $L_r(2) = \{\frac{N_r(1)}{4}+1,\ldots,\frac{3N_r(1)}{4}\}$, $L_c(2) = \{\frac{N_c(1)}{4}+1,\ldots,\frac{3N_c(1)}{4}\}$ and the next filters can be calculated recursively: $L_r^8(1) = L_r^1(2)$, $L_c^8(1) = L_c^1(2)$, $L_r^9(1) = L_r^2(2)$, etc. To visualise the SOP on a textured image, an example is presented in figure 2.24.

Figure 2.25 (b) shows the corresponding measurement space $S$ for the first two orders of the SOP: $S^{2-14}$. The effect of the filtering should be clear now, as some regions (corresponding to a particular texture) are highlighted by some filters and blocked by others. $S^8$ is a low pass filter and keeps a blurred resemblance to the original image. The background is highlighted in the high frequency filters, as should be expected of a noisy nature. Bone is also highlighted in $S^{4,5}$.

It should be mentioned that we have worked with the magnitude of the inverse Fourier transform. The phase information ([13, 23, 82]) has not been thoroughly studied, partly because of the problem of unwrapping in the presence of noise, but it deserves more attention in the future. The phase unwrapping in the presence of noise is a difficult problem since the errors that are introduced by noise accumulate as the phase is unwrapped. If the K-space is available, it should be used and this problem would be avoided since the K-space is real space.

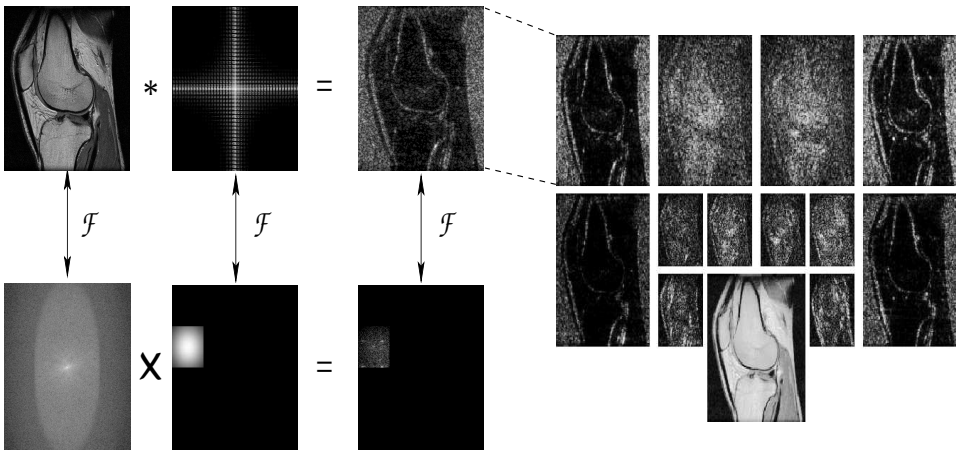



**Figure 2.24: A graphic example of the sub-band filtering. The top row corresponds to the spatial domain and the bottom row to the Fourier domain. A textured image is filtered with a sub-band filter with a particular frequency and orientation by a product in the Fourier domain, which is equivalent to a convolution in the spatial domain. The filtered image becomes one measurement of the space $S$, $S^2$ in this case.**

## 2.5 Local Binary Patterns and Texture Spectra

Wang and He [59, 150] and Ojala [111] have proposed two similar methods that try to explore the relations between neighbouring pixels. These methods concentrate in the relative intensity relations between the pixels in a small neighbourhood and not in their absolute intensity values or the spatial relationship of the whole data. The underneath assumption is that texture is not properly described by the Fourier spectrum [150] and traditional low pass / band pass / high pass filters.

To overcome the problem of characterising texture, Wang and He proposed a texture filter based in the relationship of the pixels of a $3 \times 3$ neighbourhood. A *Texture Unit* ($TU$) is first calculated by differentiating the grey level of a central pixel $x_0$ with the grey level of its 8 neighbours $x_i$. The difference is measured as 0 if the neighbour, $x_i$ has a lower grey level, 1 if they are equal and 2 if the neighbour has a bigger grey level. It is possible to quantise $G$ by introducing a small positive value $\Delta$. Thus the $TU$ is

(a)

(b)

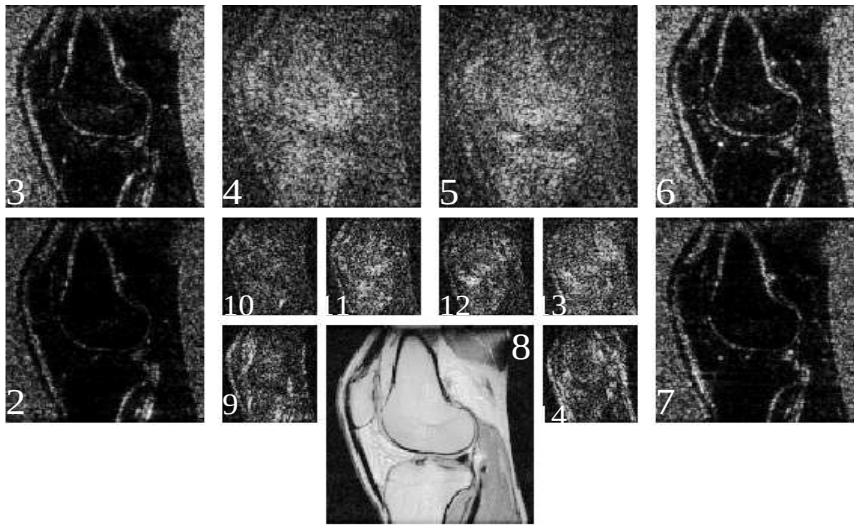



**Figure 2.25: (a) One slice of a human knee MRI and (b) Measurements 2 to 14 of the textured image. (Note how different textures are highlighted in different measurements. In each set, the measurement $S^i$ is placed in the position corresponding to the filter $F^i_\omega$ in the frequency domain).**

defined as:

$$TU = \begin{array}{|c|c|c|}\hline E_1 & E_8 & E_7 \\ \hline E_2 & & E_6 \\ \hline E_3 & E_4 & E_5 \\ \hline\end{array} = \{E_1, \ldots, E_8\}, \quad E_i = \begin{cases} 0 & \text{if} \quad g_i \leq (g_0 - \Delta) \\ 1 & \text{if} \quad (g_0 - \Delta) < g_i \leq (g_0 + \Delta) \\ 2 & \text{if} \quad g_i > (g_0 + \Delta) \end{cases}$$

After the $TU$ has been obtained, a *texture unit number* ($N_{TU}$) is obtained by weighting each element of the $TU$ vector:

$$N_{TU} = \sum_{i=1}^{8} E_i \times 3^{i-1}, \quad N_{TU} \in \{0, 1, \ldots, 6560\} \tag{2.52}$$

The sum of all $N_{TU}$ elements for a given image will span from 0 to 6560 ($3^8 = 6561$) and

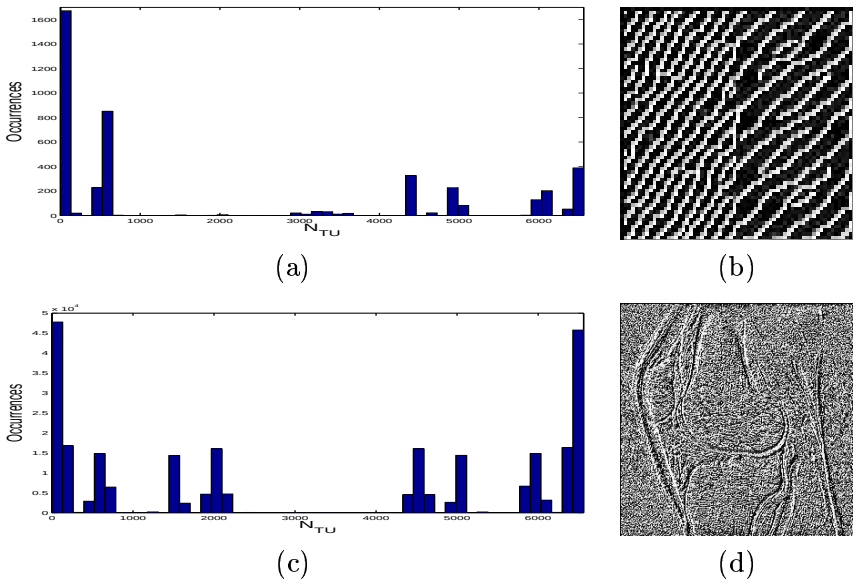


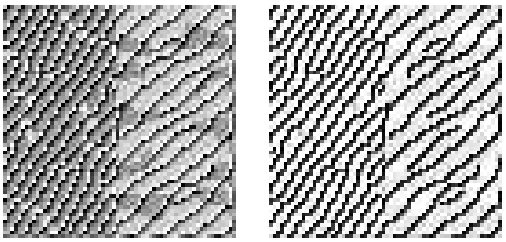

**Figure 2.26: The texture spectrum and its corresponding filtered image of (a,b) Oriented data, (c,d) Human knee MRI.**

it is called the *texture spectrum*, which is a histogram of the filtered image. Since there is no unique way of labelling and ordering the texture units, the results of a texture spectrum are difficult to compare. For example, two slices of our example data were processed with the following configuration:

| | | |
|---|---|---|
| 1 | 27 | 243 |
| 3 | | 729 |
| 9 | 81 | 2187 |

with $\Delta = 0$. Figure 2.26 shows the spectra and the filtered images.

The first striking observation of the texture spectrum comes in the filtered images. The oriented data seem to be filtered by an edge detection filter, the human knee also shows this characteristic around the edges of the bones and the skin. He and Wang claim that the filtering effect of the texture spectrum *enhances subjectively the textural perception*. This may well be an edge enhancement, which for certain textures could be an advantage; as an example they cite lithological units in a geological study. However, not every texture would benefit from this filtering. Another serious disadvantage is that this filtering is presented as a pre-processing step for a co-occurrence analysis. Co-occurrence by itself can provide many features, if this Texture spectrum filter is added as a pre-processing step, a huge amount of combinations are possible, just the labelling



(a) (b)

| | | |
|---|---|---|
| 3 | 9 | 27 |
| 1 | | 81 |
| 2187 | 729 | 243 |

(c)

| | | |
|---|---|---|
| 2187 | 81 | 9 |
| 729 | | 3 |
| 243 | 27 | 1 |

(d)

**Figure 2.27: Filtered versions of the oriented data with different labellings (a,b) Filtered data, (c,d) arrangements of $E_i$.**

to obtain the $N_{TU}$ could alter significantly the results. Figure 2.27 shows the result of using a different order for the $N_{TU}$.

To the best of our knowledge, this technique has not been extended to 3D yet. To do so, a neighbourhood of size $3 \times 3 \times 3$ should be used as a $TU$, and then the $N_{TU}$ of $3^{26} = 2.5 \times 10^{12}$ combinations would appear. The texture spectrum would not be very dense with that many possible texture units. In our opinion, this would not be a good method in 3D.

Ojala [111] presented a variation of the previous algorithm, which they call *Local Binary Pattern* (LBP). They limit the pixel difference to two options: $g_i \geq g_0$, $g_i < g_0$, and instead of $3^8 = 6561$ there are only $2^8 = 256$ possible texture units, which simplifies considerably the spectrum. If this method is to be extended into 3D the texture units would drop to $2^{26} = 67,108,864$, perhaps still too large. Two main advantages of texture spectrum and LBP is that there is no need of quantising the feature space and there is a certain immunity to low frequency artefacts such as those presented by inhomogeneities of the MRI process. In the same paper, another measure is presented; the grey level difference method (DIFFX DIFFY) where a histogram of the absolute grey-level differences between neighbouring pixels is computed in vertical and horizontal directions. This measure is a variation of a co-occurrence matrix but instead of registering the pairs of grey levels, the method registers the absolute differences. The

distance is restricted to 1 and the directions are restricted to 2. They report that the results of this method are better than other texture measures, such as Laws masks or Gaussian Markov Random Fields.

In a more recent paper, Ojala [112] presents another variation to the LBP by considering the sign of the difference of the grey-level differences histograms. Under the new consideration, LBP is a particular case of the new operator called $p_8$. This operator is considered as a probability distribution of grey levels, when $p(g_0, g_1)$ denotes the co-occurrence probabilities, they use $p(g_0, g_1 - g_0)$ as a joint distribution. Then, a strong assumption is taken in the form of the independence of the distribution which they manipulate to $p(g_0, g_1 - g_0) = p(g_0)p(g_1 - g_0)$. The authors present an error graph, which does not fall to zero, yet they consider this average error to be small and therefore independence to be a reasonable assumption. This comparison was made for only 16 grey levels, it would be very interesting to report for 256 or 4096 grey levels.

Besides the texture measurement extraction with the signed grey-level differences, a discrimination process is presented. The authors present their segmentation results on the images arranged by Randen [124] which allows comparison of their method, but they do not present a comparison for the measurements separated from their segmentation method, which could influence considerably. This method quantises the difference space to reduce the dimensionality, then uses a sampling disk to obtain a sample histogram, uses a small number of bins, lower than their own reliability criterion, and also uses a local adjustment of the grey scales for certain images. With all these particular adjustments their results outperform those reported by Randen, we will compare the method presented in this thesis against LBP and $p_8$ in chapter 6, table 6.2.

## 2.6 The Trace Transform

The *Trace Transform* proposed by Kadyrov and Petrou [74, 116] is a generalised version of the Radon transform and has seen recent applications in texture classification [75, 141]. As some other transforms, the Trace transform measures certain image characteristics

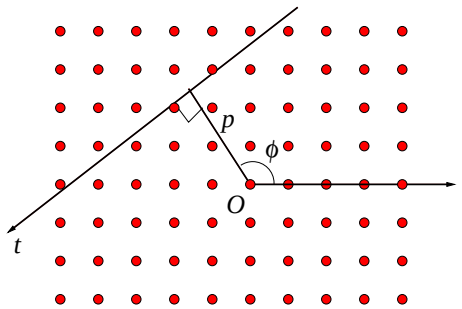



**Figure 2.28: The Trace transform parameters.**

in a space that is non-intuitive. One of the main advantages of the Trace transform is its invariance to Affine transformations, that is, translation, rotation and scaling.

The basis of the transformation is to scan an image with a series of lines or traces defined by two parameters: an orientation $\phi$ and a radius $p$, relative to an origin $O$ shown in figure 2.28. The Trace transform calculates a functional $\mathcal{T}_r$ over the line $t$ defined by $(\phi, p)$; with the functional, the variable $t$ is eliminated. If the integral is used as the functional, we will be calculating the Radon transform, but one is not restricted to the integral as a functional. Some of the functionals proposed in [75] are shown in table 2.4, but many other options are possible. The Trace transform results in a 2D function of the variables $(\phi, p)$. As an example, figure 2.29 show the Trace transform of one slice of the oriented data with three different functionals.

With the use of two more functionals over each of the variables, a single number called the *triple feature* can be obtained: $\Phi[P[\mathcal{T}_r[\mathcal{I}]]]$. These features are called *diametrical* functional $P$, and the *circus* functional $\Phi$. Again there are many options for each of the functionals, (table 2.4). The combinations of different functionals can easily lead to thousand of features. The relevance of the features has to be evaluated in a training phase and then a set of weighted features can be used to form a similarity measure between images. In [75] it is reported that the Trace transform is much more powerful than the co-occurrence matrix to distinguish between pairs of Brodatz textures.

In order to extend this method into 3D, the trace along the data will be defined by three parameters, $(\phi, p)$ and another angle of orientation, say $\theta$, so the Trace transform

**Table 2.4: Some functionals for Trace ($\mathcal{T}_r$), diametrical $P$ and circus $\Phi$.**

| | $\mathcal{T}_r$ | $P$ | $\Phi$ |
|---|---|---|---|
| 1 | $\sum_{i=0}^{N} t_i$ | $max(t_i)$ | $\sum_{i=0}^{N-1} \lvert t_{i+1} - t_i \rvert^2$ |
| 2 | $\sum_{i=0}^{N} i t_i$ | $min(t_i)$ | $\sum_{i=0}^{N-1} \lvert t_{i+1} - t_i \rvert$ |
| 3 | $\sqrt{\sum_{i=0}^{N} t_i^2}$ | $\sqrt{\sum_{i=0}^{N} t_i^2}$ | $\sqrt{\sum_{i=0}^{N} t_i^2}$ |
| 4 | $max(t_i)$ | $\frac{\sum_{i=0}^{N} i t_i}{\sum_{i=0}^{N} t_i}$ | $\sum_{i=0}^{N} t_i$ |
| 5 | $\sum_{i=0}^{N-1} \lvert t_{i+1} - t_i \rvert$ | $\sum_{i=0}^{N} i t_i$ | $max(t_i)$ |
| 6 | $\sum_{i=0}^{N-1} \lvert t_{i+1} - t_i \rvert^2$ | $\frac{1}{N} \sum_{i=0}^{N} (t_i - \hat{t})^2$ | $max(t_i) - min(t_i)$ |
| 7 | $\sum_{i=0}^{N-2} \lvert t_{i-2} + t_{i-1} - t_{i+1} - t_{i+2} \rvert$ | $c$ so that $\sum_{i=0}^{c} t_i = \sum_{i=c}^{N} t_i$ | $i$ so that $t_i = max(t_i)$ |

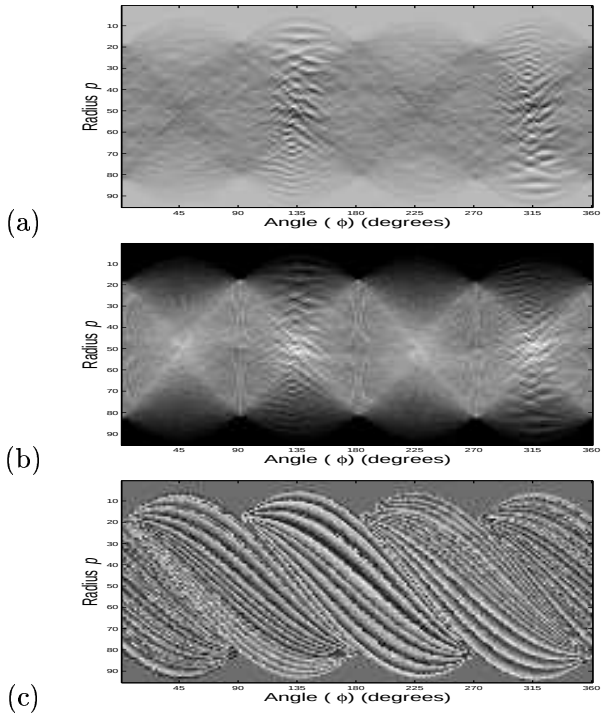



**Figure 2.29: Three examples of the Trace transform of the Oriented Pattern: (a) Functional 1, (b) Functional 4 and (c) Functional 5.**

would produce a 3D set. This can be reduced again by a series of functionals into a single value without complications, but the computational complexity is increased considerably.

Still, one important observation remains, the Trace transform shows to be power-

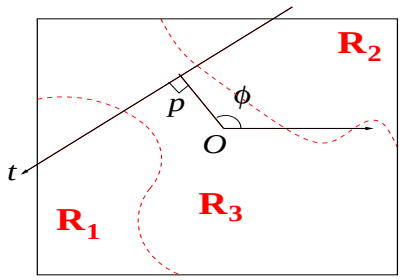



**Figure 2.30: A multitextured Image.**

ful in characterising a textured region into a series of triple features, and then uses these to distinguish from other textures. Used in this way, as a global technique it has been reported to have applications on detection of Alzheimer's disease [133]. As it was mentioned before, the objective of the thesis is to partition a region rather than discriminating between two separate sets. When tracing over an image with several textures (as figure 2.30 for example), the line $t$ would be passing through different regions and thus capturing the characteristics of different textures.

## 2.7 Summary

The fact that texture is not properly defined and that textured images or volumes can vary widely has led to a great number of texture extraction methods. We concentrated on techniques that can be used in 3D and thus only relevant techniques for measurement extraction were presented in this chapter.

Spatial domain methods such as local mean and standard deviation can be used for their simplicity, and for some applications this can be good enough to extract textural differences between regions. Also, these methods can be used as a pre-processing step for other extraction techniques.

Although easy to implement, co-occurrence measures are outperformed by filtering techniques, the computational complexity is high and can have prohibitive costs when extended to 3D. Another strong drawback of co-occurrence is that the range of grey levels can increase the computational complexity of the technique. In most cases, the

number of grey levels is quantised to a small number, this can be done either directly on the data or in the measurements that are extracted, but inevitably they result in a loss of information. When the range of grey levels exceeds the typical $0 - 255$ that is used in images, this issue is even more critical. The range of grey levels in MRI can easily be $0 - 4095$. Some of the extensions proposed for 3D have been used as global techniques, that is, that the features are obtained from a whole region of interest. This is a different approach to what this thesis is trying to achieve, that is, from a single region, discriminate two or more different textures.

Wavelets are a popular and powerful technique for feature extraction. By the use of separable functions a 3D volume can be easily decomposed. A disadvantage of Wavelet techniques is that there is not an easy way to select a Wavelet family, which can lead to many different options and therefore a great number of measurements. When classification is performed, having a larger number of measurements does not imply a better classification result, nor does it ease the computational complexity. Pichler [118] notes that the main problem in Wavelet analysis is determining the decomposition level that yields the best results. He also reports that since the channel parameters cannot be freely selected, the Wavelet transform is sub-optimal for feature extraction purposes.

The extraction of textural measurements with Gabor filters is a powerful and versatile method that has been widely used. While the *rosette* configuration is good for many textures in some cases will not be able to distinguish some textures. Also, the origin of the filters in the Fourier domain has to be set to zero to avoid that the filters react to regions with constant intensity [71]. Another disadvantage of the Gabor filters is their non-orthogonality due to their overlapping nature that leads to redundant features in different channels [96].

Pichler compared Wavelets and Gabor filters and overall Gabor filtering had better results than Wavelets but with a higher computational effort. In another comparison of Wavelets and Gabor filters, Chang [19] mentions that Wavelets are more natural and effective for textures with dominant middle frequency channels and Gabor is suitable

for images with energy in the low frequency region.

A technique that escapes the problem of the range of grey levels is the Texture Spectrum and Local Binary Pattern (LBP) [57, 111] by taking the sign of the difference between grey level values of neighbouring pixels and weighting the orientation by a power of 2. Both LBP and signed grey level differences provide good segmentation results in 2D but the extension to 3D will imply having a very large number of combinations. The possibility of different labellings of the elements in a neighbourhood can lead to many different measurements.

The Trace transform provides a way of characterising textures with invariance to rotation, translation and scaling, that enables this relatively new technique to discriminate between pairs of texture with success over co-occurrence matrices. In this thesis, the purpose is not to achieve discrimination between separate regions, but to partition one region that contains more than one texture.

We conjecture that the sub-band filtering with a SOP tessellation is a powerful technique to extract measurements of textured data. It is easy to program and extend to 3D, and exploits the computational advantages of working in the Fourier domain through a fast Fourier transform. Along with Wavelets and Gabor filters, this method is not restricted to the analysis of spatial interactions over relatively small neighbourhoods. Contrary to Gabor filters, the SOP tessellation is non-overlapping and the zero frequency values do not have to be set to zero. The number of measurements is not as large as Texture Spectrum, or Trace transform, yet it extracts properly noisy regions such as the background of the MRI data, which are not easily highlighted by Wavelet filtering.

A comparison between Wavelets and sub-band filtering in figures 2.9 and 2.25 shows that the low pass filtering of both techniques seems to give similar results; it is the high frequencies that highlight some differences. The background of the image, which is mainly noise, is clearly separated in the SOP filtering process in the $S^{2,3,6,7}$ measurements, but not so in the Wavelet decomposition. This may well be due to the selected Wavelet family or the fact that the SOP filter uses more bands.

In order to assess the selection of SOP sub-band filtering as a measurement for texture extraction, it will have to be tested against other extraction techniques and the results need to be compared. This will be done in section 3.1 against a set of 9 images with different textures. These images were proposed by Randen [124] and are used as a benchmark for texture methods. The results of the LBP will not be compared here since the measurements and the classification method have to be separated. A comparison against the results of [112], [104] as well as those of [124] will be presented in chapter 6.

# Chapter 3

# Classification of the Measurement Space

This chapter will examine the classification of the multivariate space that is formed by the measurements extracted from the data with a Second Orientation Pyramid sub-band filtering. Since classification is in itself a broad area of research, we will concentrate here in the evaluation of several important aspects that can affect the classification results.

First of all, it is important to evaluate the discrimination power of the measurements extracted with the sub-band filtering. To do this, it is necessary to have a set of textured images to be used as a test set. To the best of our knowledge, no volumetric test data sets are available, so the best option is to use a 2D set. In [124] Randen proposed a set of images with different natural textures that were captured under different illumination conditions and with different equipment, but were selected to be visually stationary. Each texture has been globally histogram equalised and they have the same mean value so that spread the same range of grey levels. (Several publications related to texture segmentation present regions that could be easily discriminated by a simple grey level thresholding, thus they are not exploiting textural properties but rather intensity differences). These images, which are fairly hard to classify even by eye, are becoming a benchmark for assessing different segmentation algorithms, [98, 104, 111].

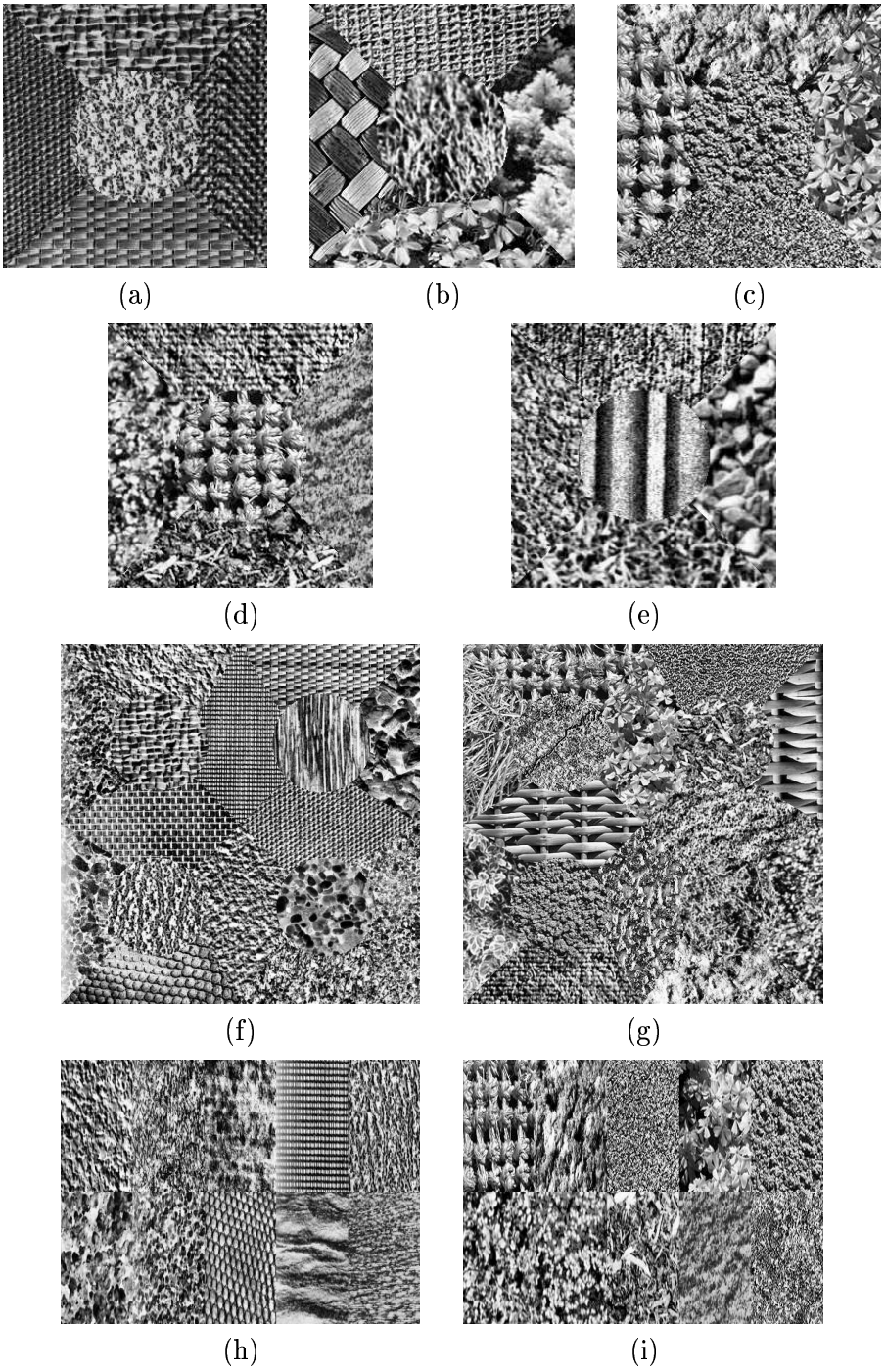



**Figure 3.1: Composite texture images arranged by Randen and Husøy [124].**

The nine texture images segmented in this work correspond to figure 11 in [124] and are presented in figure 3.1. Figures (a) to (e) consist of 5 different textures in images with size $256 \times 256$ pixels, (f) and (g) have 16 textures and are $512 \times 512$ pixels, (h) and (i) have 10 textures and $640 \times 256$ pixels.

After the measurements have been assessed in their ability to provide class separation, the classification process needs to be examined. Four aspects will be analysed in this chapter: *normalising* the measurement space; the use of a *local energy function* and different *classification techniques*; and their *computational complexity*. It is important that these steps are analysed separately in order to assess them individually, otherwise a bad measurement could be obscured by a sophisticated classification technique.

## 3.1 Classification

The classification problem is that of assigning every element of the data, or the measurements extracted from the data, into one of several possible classes [54].

Let us consider the measurement space to be classified as a set $S$ to be partitioned into subsets (sometimes called clusters). Partitioning is considered as a mapping operator $\lambda : S \rightarrow \{1, 2, \ldots, N_k\}$, where the clusters or classes are $\lambda^{-1}(1)$, $\lambda^{-1}(2)$, etc., and these are unknown. Then, for every element $x \in S$, $\lambda_a$ will be an estimator for $\lambda$ where, for every class, there is a point $\{a_1,\ a_2, \ldots\} \in S$ such that these points define hyperplanes perpendicular to the chords connecting them, and split the space into regions $\{R_1,\ R_2, \ldots\}$. These regions define the mapping function $\lambda_a : S \rightarrow \{1, 2 \ldots, N_k\}$ by $\lambda_a(x) = 1$ if $x \in R_1$, $\lambda_a(x) = 2$ if $x \in R_2$, etc. This partitioning minimises the Euclidean distance from the elements of the space to the points $a$, expressed by [35]:

$$\rho(a_1, a_2, \ldots) = \sum_{x \in (L_r \times L_c \times L_d)} \min_{1 \leq j \leq N_k} ||S(x) - a_j|| \tag{3.1}$$

This process is represented in figure 3.2.

The measure of closeness of the estimator $\lambda_a$ to $\lambda$ defines a misclassification error by

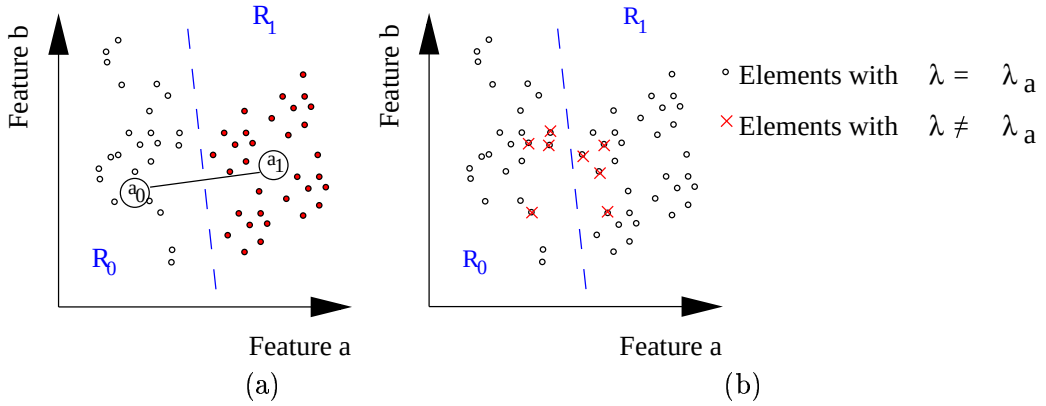



**Figure 3.2: Classification of elements in a set $S$: (a) points $a_1, a_2$ and hyperplane, (b) Elements that will be misclassified by the hyperplane.**

$\epsilon[\lambda_a] = P(\lambda_a(x) \neq \lambda(x))$ for an arbitrary point $x \in S$ in the space:

$$\epsilon[\lambda_a] = \frac{\#\{x \in (L_r \times L_c \times L_d) : \lambda(S(x)) \neq \lambda_a(S(x))\}}{\#\{L_r \times L_c \times L_d\}} \tag{3.2}$$

If the values of the points $a_k$ are known, or there is a way of estimating these from training data (it is important that the training and test sets are disjoint), the classification procedure will be a *supervised* methodology, otherwise it will be an *unsupervised* methodology. For this thesis, the points in the measurement space $a_k$ were obtained by filtering separate training data with the SOP. Once the measurement space $S$ was calculated for every training image, the average was used as an estimate of the mean of the class: $\hat{a}_k$.

It is important to observe that if one measurement is several orders of magnitude greater than others (as it may be the case between low pass and high pass measurements), the result will be dominated by the former. It may help thus to normalise or scale the measurements in order for the measurements to be comparable. This whitening of the distributions presents another problem that is exemplified in figure 3.3. By scaling the measurements, the elements can change their distribution and different structures can appear. A common normalising technique is to divide by the standard deviation to equalise the variance of the measurements. This transformation implies that both

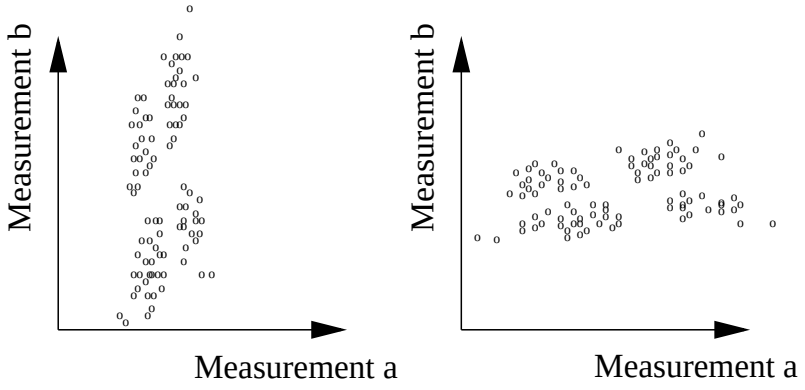



**Figure 3.3: The scaling of the measurements can yield different structures.**

**Table 3.1: Comparative misclassification results (%) of the natural textures (figure 3.1) with and without normalising the measurement space.**

| *Misclassification (%)* | Figures | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | a | b | c | d | e | f | g | h | i | Average |
| Normalised | 14.2 | 34.6 | 28.1 | 32.7 | 23.2 | 42.2 | 59.9 | 39.8 | 50.4 | 36.1 |
| Un-normalised | 17.7 | 34.9 | 28.5 | 33.0 | 27.4 | 45.6 | 61.3 | 35.0 | 51.1 | 37.2 |

measurements are equally important, which may not always be the case.

In order to determine the effect of scaling in the classification of images of figure 3.1, they were sub-band filtered with the SOP, and the corresponding measurement spaces were classified. The results are presented in table 3.1. It is very interesting to notice that for all but one of the images the scaling proved helpful in the classification. It is only figure 3.1 (h) for which the normalising gives a higher misclassification.

The results obtained will be compared with those presented by Randen [124]. In order to maintain a consistent method, we will normalise the measurement space for all the images. Table 3.2 compares the results of the SOP sub-band filtering with different measurement extraction techniques. From the 10 techniques compared, only 3 had a higher misclassification rate than the SOP; among these ones were the popular Gabor banks, and co-occurrence matrices. Seven other techniques had, on average, a lower misclassification rate.

While analysing the results presented in table 3.2, it is important to remember two things. First, if the elements of the space were assigned to a class at random, the

**Table 3.2: Comparative misclassification results (%) of the natural textures (figure 3.1) (Table 3 in [124]) and SOP sub-band filtering without local energy function (LEF). Best results are in bold.**

| *Misclassification (%)* | Figures | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Measurement | a | b | c | d | e | f | g | h | i | Average |
| Laws | 9.7 | 25.7 | 32.4 | 27.3 | 25.7 | 48.3 | 54.3 | 41.9 | 37.8 | 33.68 |
| Ring/Wedge | 14.6 | 35.5 | 28.9 | 35.5 | 22.4 | 43.8 | 67.8 | 44.5 | 48.3 | 37.92 |
| Dyadic Gabor | 10.7 | 34.8 | 22.6 | 25.2 | 24.6 | 60.1 | 58.2 | **32.3** | 47.9 | 35.16 |
| Gabor Banks | **8.2** | 34.0 | 25.8 | 36.9 | 28.4 | 54.8 | 71.5 | 39.7 | 54.8 | 39.34 |
| DCT | 13.2 | 27.0 | 25.5 | 37.8 | 22.6 | 40.9 | 49.0 | 38.2 | 33.0 | 31.91 |
| Daubechies 4 | 8.7 | 22.8 | 25.0 | 23.4 | 21.8 | 38.2 | 45.2 | 40.9 | 30.1 | 28.46 |
| f16b | 8.7 | **18.9** | 23.3 | **18.4** | **17.2** | **36.4** | **41.7** | 39.8 | 28.5 | **25.88** |
| Co-occurrence | 9.9 | 27.0 | 26.1 | 51.1 | 35.7 | 49.6 | 55.4 | 35.3 | 49.1 | 37.69 |
| AR | 19.6 | 19.4 | **23.0** | 23.9 | 34.0 | 58.0 | 46.4 | 56.7 | **28.7** | 34.41 |
| Average | 11.5 | 27.2 | 25.9 | 31.1 | 24.7 | 47.8 | 54.4 | 41.0 | 39.8 | 33.71 |
| SOP | 14.2 | 34.6 | 28.1 | 32.7 | 23.2 | 42.2 | 59.9 | 39.8 | 50.4 | 36.17 |

probability of correct classification would be: one in five, one in 16 or one in 10, that is a misclassification of 80% for images (a,b,c,d,e), 93.75% for images (f,g) and 90% for images (h,i). Second, it is very important to notice that while the measurements presented by Randen have been filtered with a local energy function, the SOP has not been smoothed. We will now analyse the use of a local energy function.

## 3.2 Impact of the local energy function (LEF) in the classification

The choice of a local energy function (LEF) has been separated from the measurement extraction in order to assess the power of the measurements on its own. Nevertheless, the LEF can influence considerably the classification process. The simplest, and perhaps most common way to use a LEF is to smooth the space with a convolution of a kernel, either Gaussian or uniform.

Another way of averaging the values of neighbours is through the construction of a pyramid or tree. These methods have the advantage of reducing the dimensions of the space at higher levels. Yet another averaging can be performed in which an anisotropic operator, namely butterfly filters. This option can be used to improve the classification,

**Table 3.3: Comparative misclassification results (%) of SOP sub-band filtering with Local Energy Function (LEF) of different sizes and best of table 3.2. Where SOP results outperform Randen's results they are in bold.**

| *Misclassification (%)* | Figures | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Size of LEF | 11a | 11b | 11c | 11d | 11e | 11f | 11g | 11h | 11i | Average |
| No LEF | 14.2 | 34.6 | 28.1 | 32.7 | 23.2 | 42.2 | 59.9 | 39.8 | 50.4 | 36.1 |
| $5 \times 5$ | 11.4 | 33.4 | 25.3 | 27.9 | 21.0 | 39.2 | 56.7 | 36.3 | 46.5 | 33.1 |
| $9 \times 9$ | 9.3 | 32.6 | **22.4** | 23.4 | 18.6 | **35.5** | 52.9 | **31.8** | 42.5 | 29.9 |
| $13 \times 13$ | 9.0 | 31.7 | **20.6** | 20.7 | **17.2** | **32.7** | 49.5 | **27.9** | 39.5 | 27.6 |
| $17 \times 17$ | 9.3 | 30.4 | **19.5** | 19.6 | **16.6** | **31.0** | 47.2 | **26.1** | 37.7 | 26.4 |
| $21 \times 21$ | 9.6 | 29.4 | **19.0** | 19.2 | **16.3** | **29.8** | 46.0 | **25.3** | 36.6 | 25.7 |
| Best of Table 3.2 | **8.2** | **18.9** | 23.0 | **18.4** | 17.2 | 36.4 | **41.7** | 32.3 | **28.7** | |

especially near the borders between textures. We will analyse these cases in the following chapters.

As it was mentioned in section 2.4, the smoothing of the filter responses can influence considerably the results of classification process. Previous research shows that a Gaussian smoothing [15, 71, 123, 124] is better than uniform, yet the issue of the size of the smoothing filter is quite important. We considered Gaussian functions $\mathscr{G}_a$ of different sizes $(N_r^{\mathscr{N}}, N_c^{\mathscr{N}}, N_d^{\mathscr{N}}) = 3 \times 3 \times 1, 5 \times 5 \times 1, 7 \times 7 \times 1, \ldots, 37 \times 37 \times 1$. These Gaussian functions were used to smooth each of the 35 filter responses corresponding to $S$ of order 5:

$$\tilde{S}^i = S^i * \mathscr{G}_a \tag{3.3}$$

Table 3.3 presents some of the results of classifying after using a LEF to smooth the measurement space. To provide a comparison, the best results of the table 3.2 were included. It is quite clear that the use of a LEF improves drastically the classification for all the images. With the use of the LEF on the spaces $S$ of images (c,e,f,h), the classification results outperform those of the texture extraction techniques presented by Randen, and the average can be as good as the best result. Thus, these results confirm that sub-band filtering with a SOP filtering can extract textural measurements that are as good as those presented by Randen.

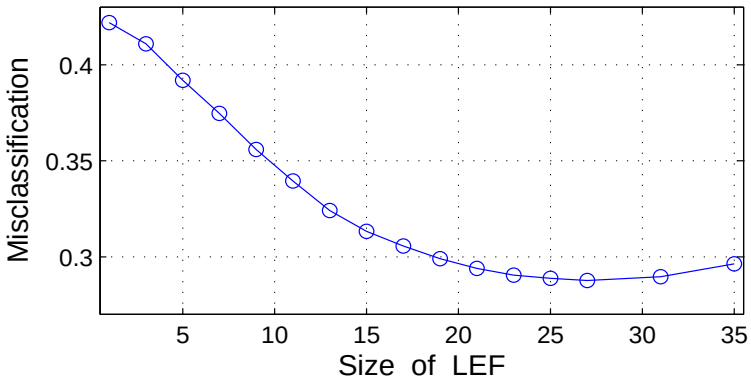



**Figure 3.4: Misclassification rates with different sizes of the LEF.**

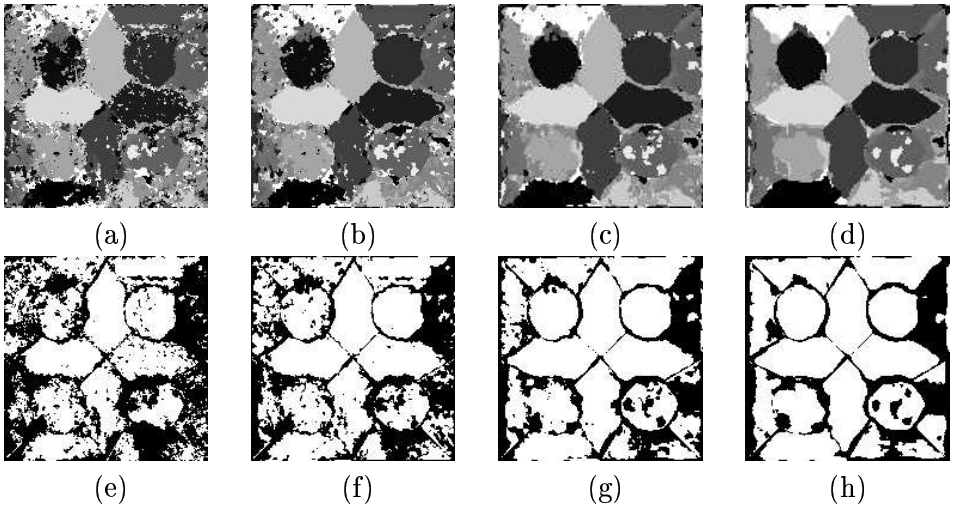



**Figure 3.5: Classification with different sizes of the Gaussian LEF for smoothing. Top row: (a)** $3 \times 3$**, (b)** $9 \times 9$**, (c)** $17 \times 17$**, (d)** $27 \times 27$**. Bottom row: corresponding correctly classified pixels (e,f,g,h) (correct labelled in white)**

The effect of the LEF on the misclassification of image (a) reveals an important aspect of the smoothing of $S$. As the size of the LEF grows, the misclassification reaches a minimum level and then increases. This shows that the size cannot be increased indefinitely. Figure 3.4 shows the misclassification for figure 3.1 (f) with LEF of different sizes. As the size of the LEF increases, the misclassification decreases until a minimum level for a LEF of size $27 \times 27$, and then the misclassification begins to increase. This is due to the blurring of the boundaries between adjoining textures that can be seen in figure 3.5. This problem can be reduced with butterfly filters which will be presented in section 5.3.

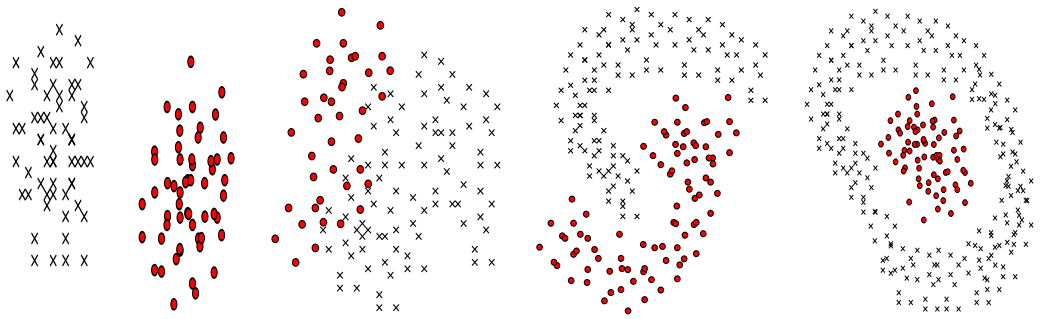
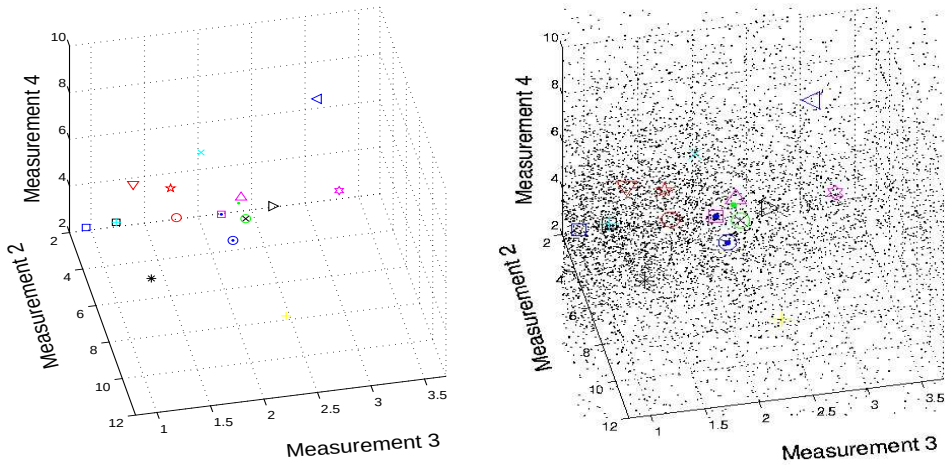



**Figure 3.6: Four cases of spaces in two dimensions with two classes and different distributions: (a) Well separated, (b) Overlapping, (c) Non-linearly separable, (d) Concentric.**

## 3.3 Different classification techniques

The results obtained so far are encouraging in the sense that the misclassification obtained for the images is comparable to those reported by Randen, but there is a big underlying assumption that needs to be examined, that is: the classification of the space by hyperplanes defined by an estimation of the mean value (sometimes called prototype) of each class, assumes that the classes are linearly separable. This may not always be the case. So far we have not observed whether the classes follow a certain distribution. In some cases, the distribution of the data can dictate the choice of classification techniques or parameters. Some examples of different distributions are presented in figure 3.6: (a) shows the ideal case where classes are well separated and easily partitioned (and with low misclassification) with a linear plane; (b) shows a more realistic situation in which the classes are overlapping; (c,d) present a rather particular situations, where the classes do not overlap, but a linear partition would fail.

To see how dense the measurement space is, a sample of elements from $S^i$ from figure 3.1 (f) is presented in figure 3.7. In (a), the estimates of the means, $\hat{a}_k$, for the 16 classes can be seen, and in (b), the same estimates are immersed in some of the elements of the unclassified measurement space $S^{2,3,4}$. The density of the space does not allow a simple recognition of the shapes of the classes. Furthermore, these represent only 3 measurements out of many more that belong to the space. It is therefore necessary to evaluate other classification techniques that can provide better results in cases where a




(a) (b)

**Figure 3.7: Visualising the measurement space: (a) The estimate of the mean $\hat{a}_k$ of 16 classes of image 3.1 (f), for 3 measurements, and (b) The estimates immersed in the unclassified measurement space $S^{2,3,4}$.**

single linear thresholding may not be the optimal choice.

There are many techniques for classification in the literature. In this thesis we will evaluate the performance of just two of them and determine whether these algorithms can improve the classification results.

### 3.3.1 K-Nearest Neighbours

Since we are using a set of training data in a supervised classification, it can be intuitively appealing to use not just one value as an estimate of the mean; $\hat{a}_k$, but a group of points $\{a_{11}, a_{12} \ldots, a_{21}, a_{22}, \ldots, , a_{31}, a_{32}, \ldots\}$ associated to the classes. The set of *neighbours* would determine the class of the elements to be labelled in the space by searching for the neighbour that is closest to the element, and assigning the class associated to this neighbour. By having more than one element per class, the boundaries that will separate the regions will not be linear, but piece-wise linear (figure 3.8). While the classification can be improved, the computational complexity increases as the number of selected neighbours grows.

Several implementation issues are related with this *K Nearest Neighbours* (kNN) classifier:

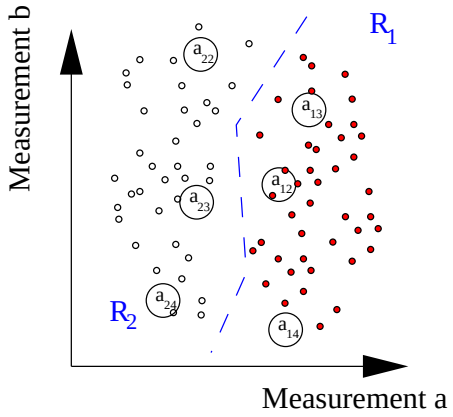
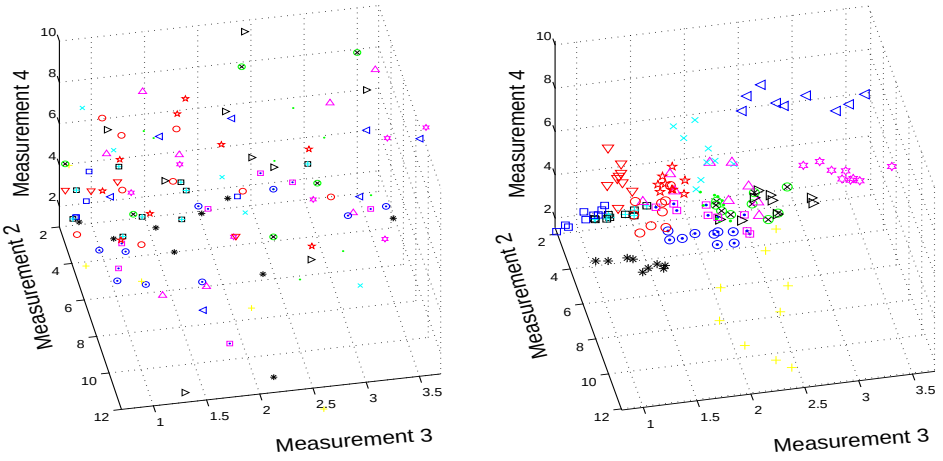



**Figure 3.8: Classification with several neighbours associated to each class.**

- The distance measure to be employed: we used a Euclidean distance after a normalisation of the measurement space.
- The number of neighbours selected ($N_{kNN}$) is important as it increases considerably the computational complexity. For our experiments we used 9 and 16 neighbours per class.
- The selection of the neighbours themselves. We used a simple (and perhaps naïve) approach of selecting at random a series of neighbours from the measurement space of the training data. (The next section will improve on this selection process).

Table 3.4 compares the classification of the kNN classification (kNN 9, kNN 16) with the previous classification schemes, which only use a single estimate of the mean value ($\hat{a}_k$). While the use of 16 neighbours improves the results over 9 neighbours, as perhaps expected, all these results are much higher that those previously obtained. These results indicate that the neighbours selected at random fail to capture the shapes of the classes, and, the boundaries defined by the estimate of the mean provide a better classification. One of the possible causes of this result is that the neighbours selected at random could be far away from the estimated mean, and then an overlap of the classes could lead to a higher misclassification. Figure 3.9 (a) shows 9 randomly selected neighbours for each class of the training data for figure 3.1 (f). These neighbours do not resemble the estimates presented previously.




(a) (b)

**Figure 3.9: Nine elements per class to be used as neighbours for classification: (a) Randomly selected, (b) Trained as self-organising feature maps (SOM).**

**Table 3.4: Comparative misclassification results (%) of the natural textures (figure 3.1) with different classification techniques. Best results are in bold.**

| *Misclassification (%)* | Figures | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Classifier | a | b | c | d | e | f | g | h | i | Average |
| $\hat{a}_k$ | **14.2** | 34.6 | **28.1** | 32.7 | **23.2** | **42.2** | 59.9 | **39.8** | **50.4** | **36.1** |
| kNN 9 | 23.0 | 42.1 | 47.8 | 52.1 | 36.8 | 57.6 | 75.0 | 49.4 | 65.2 | 49.9 |
| kNN 16 | 21.9 | 40.2 | 43.9 | 51.3 | 34.2 | 55.9 | 74.1 | 49.0 | 62.2 | 48.1 |
| LVQ 9 | 14.7 | 34.9 | 29.0 | 34.4 | 23.0 | 42.3 | 59.4 | 47.6 | 62.2 | 38.6 |
| LVQ 16 | 14.5 | **33.1** | **28.1** | **32.6** | 23.5 | 42.4 | **59.3** | 47.2 | 62.2 | 38.0 |

### 3.3.2 Learning Vector Quantisation (LVQ)

The higher misclassification obtained previously with the kNN method may be due to the random selection of the neighbours from the training data, as the elements selected might be outliers. It is thus important to improve the neighbour choice.

Learning Vector Quantisation (LVQ) [84] is a classification method where the smallest distance from an unclassified element (or vector) of the space to a set of reference vectors is sought to assign a class. This is different from kNN in that the reference vectors undergo a *learning stage* that approximates the distribution of the classes. This procedure of organising is a particular case of Kohonen's [84] Self-Organising Feature Maps (SOM).

The SOM consists of a series of nodes, $m_i \in S,\ i = \{1, 2, \ldots, N_{kNN}\}$, sometimes

called reference vectors or neurons. These neurons will act competitively upon a given input, $x \in S$ and have lateral interconnections to other neurons - different topologies of the interconnections like *grid* or *hexagonal*, are possible. The training is performed with a series of codebook vectors, each of them will be associated with a class. The self-organising algorithm proposed by Kohonen follows two basic steps: (a) matching and finding a winner node, $m_w$ determined by the minimum Euclidean distance $||x - m_w||$ from the reference vectors to the input vector $x$, and (b) the update of the position of winning neuron. The update will look to decrease the distance metric. If the input signal is used in a sequential form it generates a time process, mathematically:

$$\begin{aligned} m_w(t+1) &= m_w(t) + \alpha(t)(x(t) - m_w(t)), \\ m_i(t+1) &= m_i(t), \quad \text{for } i \neq w \end{aligned} \tag{3.4}$$

where $\alpha(t)$ is a monotonically decreasing scalar gain factor. It is possible to extend the updating of the neurons not just to the winner node, but to a neighbourhood around it:

$$\begin{aligned} m_i(t+1) &= m_i(t) + \alpha(t)(x(t) - m_i(t)), \text{ for } i \in \mathcal{N}_w \\ m_i(t+1) &= m_i(t), \text{ for } i \notin \mathcal{N}_w, \end{aligned} \tag{3.5}$$

where $\mathcal{N}_w$ represents a neighbourhood around the winner neuron which again should be shrinking with time.

For the classification of our images, a number of elements (500) were selected at random from the measurement spaces $S$ of the training data and used as the input signal for the SOMs to adapt topologically to the classes. Two sizes of SOM were used, $3 \times 3$ and $4 \times 4$, in order to maintain the same number of neurons as neighbours in the previous section. Figure 3.9 (b) shows the trained neurons corresponding to the 16 SOMs (one for each class) for image 3.1 (f). The trained SOMs appear more compact and closer to the mean estimates than the randomly selected neighbours. It is expected that the LVQ will outperform kNN.

The classification results of this method are presented in table 3.4. Several observations can be made: first, the classification of the kNN method is easily outperformed; second, an increase in the number of neurons improves the classification results; third, the results of the LVQ method are not much better than for the linear partitioning method with just one point per class.

This may come as a surprise, but Randen also reported using thresholding in some cases. It still can be the case that the trained SOM overlap in the space. Unfortunately, this is hard to visualise for a 35-dimension measurement space. There are several optimised methods that prevent the overlapping, or that penalise certain updating of the neurons, but are beyond the scope of this thesis since the linear partitioning provides a good classification results and relatively low computational cost. Table 3.5 presents the classification times as an indication of the computational complexity. It can be seen that as more neighbours are included in the classification the complexity increases. For the LVQ, the training stage further increases the classification time. The notable time difference between LVQ 9 and LVQ 16 is due to the training; for the case of 9 reference vectors the training was restricted to 100 epochs and for the 16 vectors 1000 epochs were used.

The size of the image is another aspect to be considered in complexity issues. Figure 3.10 shows the comparison between using 1, 9 and 16 points for classification in images of size $16^2$, $32^2$, $64^2$, $128^2$, $256^2$ and $512^2$ pixels. For small images the increased complexity is negligible, but if the image is $512 \times 512$ the difference could be prohibitively expensive.

### 3.3.3 Unsupervised methods

All the classification techniques previously presented are supervised methods, and while the presence of training data can easily justify using it to improve classification, it is not always possible to have these data sets available. In such cases, an *unsupervised* (or *clustering*) technique is necessary. There are many algorithms that perform the

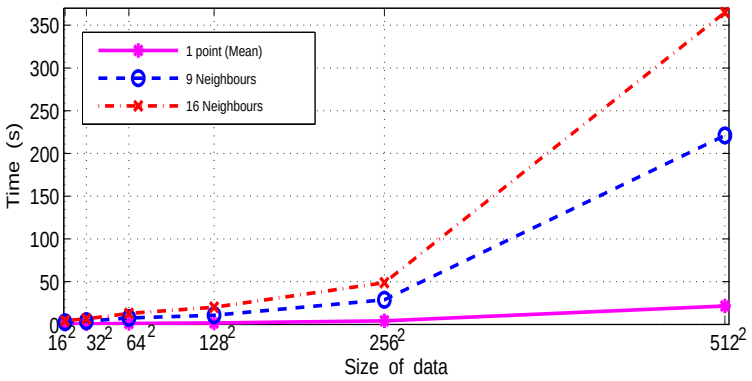



**Figure 3.10: Time comparison of classification to one point (mean) or several points (neighbours).**

**Table 3.5: Comparative time (s) results of the natural textures (figure 3.1) with different classification techniques.**

| *Time (s)* | Figures | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Classifier | a | b | c | d | e | f | g | h | i |
| $\hat{a}_k$ | 1.7 | 1.8 | 1.7 | 1.7 | 1.7 | 21.9 | 21.6 | 6.7 | 6.7 |
| kNN 9 | 10.7 | 9.4 | 11.3 | 10.7 | 10.9 | 221.6 | 223.0 | 134.9 | 134.1 |
| kNN 16 | 18.7 | 23.1 | 20.1 | 16.6 | 17.5 | 365.7 | 387.7 | 203.3 | 216.6 |
| LVQ 9 | 121.5 | 144.6 | 129.7 | 109.1 | 106.8 | 525.8 | 524.3 | 324.8 | 323.7 |
| LVQ 16 | 986.4 | 1008.8 | 1169.5 | 1029.8 | 1111.3 | 3815.5 | 3922.1 | 2380.7 | 2455.0 |
| Pixels | 256× 256 | 256× 256 | 256× 256 | 256× 256 | 256× 256 | 512× 512 | 512× 512 | 640× 256 | 640× 256 |

clustering in two basic categories [70]: *hierarchical* and *partitional* (or agglomerative and divisive [44]).

The agglomerative clustering imposes a hierarchical structure on the data as it forms clusters sequentially. The process begins with all the elements of the data being an individual cluster. At every step, the two most similar clusters are merged into one and the process continues until all elements belong to one single cluster. A hierarchical tree or dendogram that specifies which clusters are linked, and at what stage, is generated as the clusters are linked. The process can stop when a desired number of classes is reached. With these methods there is no need for training data and the linking of the elements can be done without the need of other information than the desired number of classes. The measure of similarity can yield different ways of grouping the elements, the two most common measures used are: *single linking* where the minimum distance between

**Table 3.6: Results for unsupervised classification with LBG algorithm.**

| Figures | a | b | c | d | e | f | g | h | i | Average |
|---|---|---|---|---|---|---|---|---|---|---|
| Misclassification (%) | 63.8 | 57.7 | 62.1 | 58.7 | 60.8 | 78.6 | 81.3 | 75.5 | 74.1 | 68.0 |
| Time (s) | 11.0 | 12.0 | 12.6 | 12.0 | 11.6 | 120.4 | 145.9 | 54.8 | 55.2 | |

the elements of the clusters is considered as the closest or most similar, and *complete linking* where the maximum distance between the elements of the clusters is used. Single linking generates loosely connected clusters, while complete linking generates compact clusters. As it was mentioned before, the distribution of the data can determine which method is most convenient [77].

The main disadvantage of these agglomerative clusters is the need of calculating the similarity between all the clusters, and in every step it is necessary to recalculate the measures for the elements that have been a merged. There are methods designed for large data sets [168] but the computational burden can still be huge. Even after a feature selection and dimensionality reduction that will be done in the next chapters the complexity is too high.

Partitional methods start with one single cluster and proceed to subdivide it several times until a desired number of clusters is reached. Some of these methods allow an element movement from cluster to cluster. Most of these methods will try to maximise a criterion (such as the square error). A common algorithm is the Linde-Buzo-Gray (LBG) [97] that starts with a single cluster and the average of its elements as a *codevector*. This codevector is split into two new codevectors, all the elements of the data look for the nearest of the codevectors and form a cluster accordingly. The squared error distortion measure is calculated and the value of each codevector is updated to reduce this distortion to a pre-established level. Once the level is reached, each codevector is split and the process continues until the desired number of clusters is reached.

Table 3.6 presents the results of classifying with the LBG algorithm. While the classification times are comparable to the kNN, the misclassification obtained is far worse than all the other methods.

## 3.4 Summary

We have analysed some important aspects related to the classification of textured data.

We can state them as:

- The measurement space generated by sub-band filtering with a SOP provides features that are equivalent to those present in the literature, in particular the ones assessed by Randen. Even without the use of a LEF to smooth the space, the SOP filtering outperforms the two popular methods of co-occurrence and Gabor filter banks.
- The use of LEFs for smoothing the measurement space prior to classification is of great help to the classification process. The size of the LEF should be chosen carefully, for a function that is too big can leak from one region to another and cause errors in the boundaries.
- The normalising of the space (by the standard deviation) improves the results in most of the analysed images, but it is not a general rule.
- Where there is training data available or it is possible to generate training samples, it is convenient to use the data and proceed with a supervised classification.
- The representation of a class by a single point in space is debatable, but for the data analysed, it was found to be acceptable. Zupan [168] and Randen [124] reported similar conclusions in this respect.
- If the data sets are of a high dimensionality (as it will be in the volumetric examples in chapter 6), it is better to use a simple thresholding that gives relatively good results at a fraction of the time of other methods. LVQ is a powerful method, but the computational complexity does not justify its use in our case.

# Chapter 4

# The Feature Space

In this chapter, a novel feature selection methodology based on the *Bhattacharyya Space* [128] is introduced. In the previous chapter, processing of the textured data produced a series of measurements that belong to a *Measurement Space* (following the notation of [54], it can also be called the Feature Space or Pattern Representation [80]). This space will consist of a number of dimensions, either the results of filters, features of the co-occurrence matrix or Wavelets, and not all the dimensions will contribute to the discrimination of the different textures that compose the original data. As an example of the importance of the measurement selection in order to improve the performance of a classifier in a posterior stage, four training classes of the human knee MRI have been manually segmented (figure 2.10) and each class has been sub-band filtered in 3D with an SOP of order 2 (the measurement space will consist of 58 dimensions). Figure 4.1 shows the scatter plot of three *bad* and three *good* measurements manually selected. While the classes in figure 4.1 (a) overlap completely, in figure 4.1 (b) they are fairly well separated: the small cluster right-most corresponds to the background, the middle cluster corresponds to the muscle and the two clusters that partly overlap, describe bone (horizontally spreading over measurement 5) and tissue (vertical over measurement 39). It becomes clear the necessity of selecting appropriate measurements from the space. Besides the discrimination power that some measurements have, there

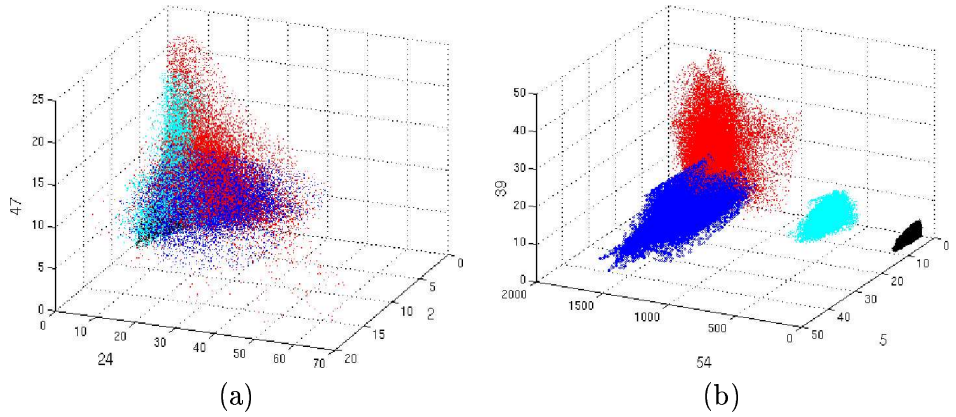



**Figure 4.1: Scatter plots of three features $S^i$ from human knee MRI (3D, order 2) (a) *bad* discriminating features $S^{2,24,47}$ (b) *good* discriminating features $S^{5,39,54}$. Note that each feature corresponds to a filtered version of the data, therefore the axis values correspond to the grey levels of each feature.**

is also a complexity issue related to the number of measurements selected.

Another advantage of selecting a subset of the space is that it can provide a better understanding of the underlying process that generated the data [52]. The selection of a particular orientation of the co-occurrence matrix can describe best a certain texture, or a filter can determine the frequency characteristics of the data.

## 4.1 Feature Selection and Extraction

The feature selection and extraction problem (as defined by Kittler [80]):

> is concerned with the mathematical tools for reducing the dimensionality of pattern representation. Pattern descriptors constituting the lower-dimensionality representation are referred to as features because of their fundamental role in characterising the distinguishing properties of pattern classes.

Therefore we face a problem of selecting a subset that will reduce the complexity and improve the performance of the classification. The reduced subset can be obtained in two different ways: *feature selection* or *feature extraction*. In feature selection, a set of the original measurements is discarded and the ones that are selected, which will be the most useful ones, will constitute the *Feature Space*. In contrast, the combination of a

series of measurements in a linear or non-linear mapping to a new reduced dimensionality is called feature extraction.

For feature selection the best way to obtain a reduced set in feature selection would be to test for every combination of measurements. For $N_i$ measurements, there would be $O(2^{N_i})$ different solutions, which yields computations impractical even for a small number of measurements. Branching techniques [80] can also obtain optimal solutions but they are still computationally intensive. It is necessary then to settle for sub-optimal solutions that will not analyse the whole space of combinations exhaustively. Most cases will belong to one of two groups: *forward section* or *backward elimination* (which are both particular cases of the *Plus l - take away r* algorithm). In forward selection, a search begins with an empty set; features are then included in the classifier one at a time. In backward elimination the starting state is the full set of features, and measurements are discarded one by one (figure 4.2). The process of selection or elimination continues up to a certain state where an evaluation criterion is satisfied and a final subset is reached. If each of the elements of the subset is included sequentially at a classifier, then we expect to improve the classification with every element, and if we would continue with any other element there would be degradation in the results.

The alternative to feature selection - feature extraction - will use all of the measurement space and map it to a lower dimensional space, where the new features will contain useful information through a projection that will ignore redundant and irrelevant information. Perhaps the most common feature extraction method is the *Principal Components Analysis* (PCA) where the new features are uncorrelated and these are the projections into new axes that maximise the variances of the data. As well as making each feature linearly independent, PCA allows the ranking of features according to the size of the variance in each principal axis from which a 'subspace' of features can be presented to a classifier. However, while this eigenspace method is very effective in many cases, it requires the computation of all the features for given data. In some of the applications presented in this thesis, the measurement space can be generated only

for a set of training samples, from which an adequate feature space can be determined. Then, only the required features are obtained for the whole data set therefore speeding the process.

One of the most common methods [34] of forward selection is the *wrapper approach* [83]. This approach uses the error rate of the classifier itself as the criterion to evaluate the features selected, it proposes a greedy selection, either as *hill climbing*, or *best first* as search algorithms and treats the measurements as a search space organisation, a representation where each state represents a measurement subset. For $N_i$ measurements, there are $N_i$ bits in each state indicating the presence (1) or absence (0) of the measurement. The state $\{0, 0, \ldots, 0\}$, the empty set will be the initial state for forward selection, and $\{1, 1, \ldots, 1\}$ will describe the whole measurement space (initial state for backward elimination). Figure 4.2 shows a 4 measurement state space for forward and backward processes. Each of the links will represent a single measurement added (continuous line) or deleted (dashed line).

The process of wrapper selection with a hill climbing search follows the sequence:

1. Start with an empty set of features $v \leftarrow \{0, 0, \ldots, 0\}$.
2. Expand $v$: generate new states by adding a single feature from $v$. In the example of figure 4.2 (a) the children of $v$ are $\{1, 0, 0, 0\}$, $\{0, 1, 0, 0\}$, $\{0, 0, 1, 0\}$, $\{0, 0, 0, 1\}$.
3. Apply the evaluation function $\lambda$ (that is, the classifier) to each child $w$ of $v$.
4. Let $v' =$ the child with the best evaluation $\lambda(w)$.
5. If $\epsilon[\lambda(v')] < \epsilon[\lambda(v)]$ then $v \leftarrow v'$ and go to 2, else finish with $v$ as a final subset.

The previous algorithm is in its most basic form and it can easily be varied, e.g. different ways of expanding $v$ rather than just considering every child can be used. It is important to bear in mind two issues: one is that hill climbing can lead to local sub-optimal solution, and the other is that the strength of the algorithm, the use of the classifier in the selection process instead of other evaluation functions, is at the same time its weakness, since the classification process can be slow.

One way to avoid the evaluation of each child of the current state will be proposed below. The *Bhattacharyya Space* is presented as a method that provides a ranking for

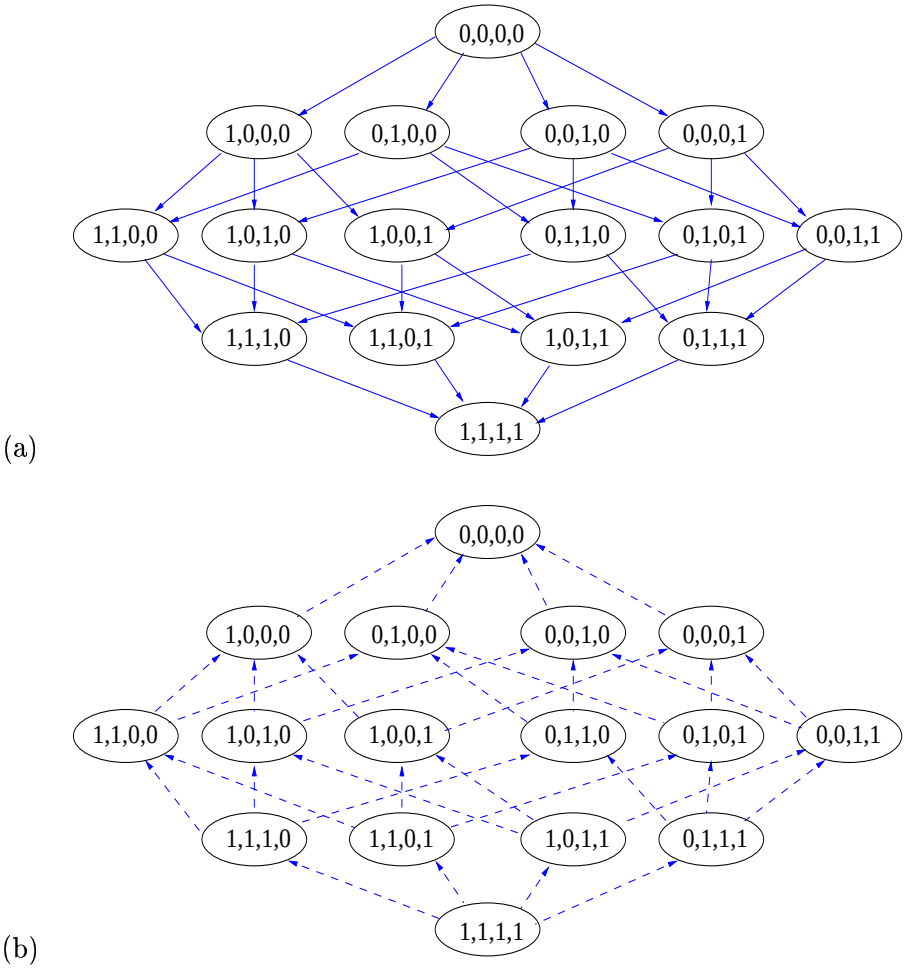



**Figure 4.2: State space for sequential processes. Each node is connected to nodes that have one measurement added or deleted: (a) Forward selection, (b) Backward elimination .**

the measurements based on the discrimination of a set of training data. This ranking process provides a single route to evaluate and therefore, the number of classifications, which will still be done for every feature to be added to the classifier, is significantly reduced. Since this method is a pre-processing step, and is done over a training data (of small dimensions compared to the whole data set), a heuristic solution to avoid being trapped at a sub-optimal solution is also proposed.

## 4.2 The Bhattacharyya distance

In order to obtain a quantitative measure of *how separable* two classes are, a distance measure is required. With the assumption of underlying distributions a *probabilistic distance* can be easily extracted from some parameters of the data. Kailath [76] compared the Bhattacharyya distance and the Divergence (Kullback-Leibler), and observed that Bhattacharyya yields better results in some cases while in other cases they are equivalent. In a recent study [7], a number of measures: Bhattacharyya, Euclidean, Kullback-Leibler, Fisher, have been studied for image discrimination and concluded that the Bhattacharyya distance is the most effective texture discrimination for subband filtering schemes.

In its simplest formulation, the Bhattacharyya distance between two classes can be calculated [27] from the variance and mean of each class in the following way:

$$D_B(k_1,k_2) = \frac{1}{4}\ln\left(\frac{1}{4}(\frac{\sigma_{k_1}^2}{\sigma_{k_2}^2}+\frac{\sigma_{k_2}^2}{\sigma_{k_1}^2}+2)\right)+\frac{1}{4}\left(\frac{(\mu_{k_1}-\mu_{k_2})^2}{\sigma_{k_1}^2+\sigma_{k_2}^2}\right) \tag{4.1}$$

where: $D_B(k_1,k_2)$ is the Bhattacharyya distance between $k_1-th$ and $k_2-th$ classes, $\sigma_{k_1}$ is the variance of the $k_1-th$ class, $\mu_{k_1}$ is the mean of the $k_1-th$ class, and $k_1,k_2$ are two different training classes.

For the multidimensional distance, the variances are replaced by co-variance matrices and the means become vectors [45]:

$$D_B(k_1,k_2) = \frac{1}{2}\ln\left(\frac{|\frac{1}{2}(\Sigma_{k_1}+\Sigma_{k_2})|}{\sqrt{|\Sigma_{k_2}||\Sigma_{k_1}|}}\right)+\frac{1}{4}(\mu_{k_1}-\mu_{k_2})^T(\Sigma_{k_1}+\Sigma_{k_2})^{-1}(\mu_{k_1}-\mu_{k_2}) \tag{4.2}$$

where $(\cdot)^T$ is the transpose of the matrix.

The Mahalanobis distance used in Fisher Linear Discriminant Analysis (LDA) is a particular case of the Bhattacharyya, when the variances of the two classes are equal; this would eliminate the first term of the distance. This term depends solely on the variances of the distribution. If the variances are equal this term will be zero, and it will grow as the variances differ. The second term, on the other hand, will be zero if

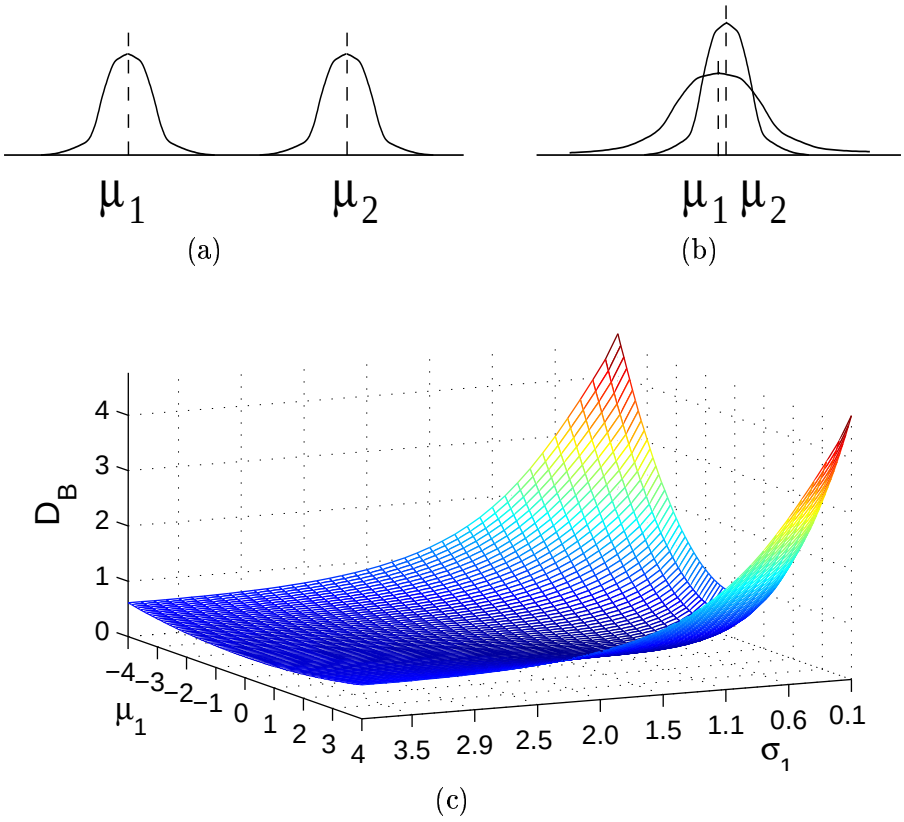



**Figure 4.3: The Bhattacharyya distance: (a) Different means with similar variances (b) Similar means, different variances, (c) Distribution of the distance relative to $\mu_1, \sigma_1$ when $\mu_2 = 0$ and $\sigma_2 = 1$.**

the means are equal and is inversely proportional to the variances. Figure 4.3 (a,b) represents these two cases and (c) shows the distribution of the distance in terms of $\mu_1, \sigma_1$ when $\mu_2 = 0$ and $\sigma_2 = 1$.

The assumption of normality can be a critical issue if there is no knowledge of the distributions. Nevertheless, the discrimination power can still be exploited. As an example, $D_B(k_1, k_2)$ was calculated for the four training classes (background, muscle, bone and tissue) of the human knee MRI (figures 2.10- 2.14) where some of the histograms present a certain degree of normality, but notably the tissue class does not. The results are presented in table 4.1. It should be noted the small Bhattacharyya distance between the tissue and the bone classes. These two classes have low discrimination power. The measurement space will be searched for some features that will have a large

| | $\mu$ | $\sigma$ | Background | Muscle | Bone | Tissue |
|---|---|---|---|---|---|---|
| Background | 91 | 49 | 0 | 4.36 | 12.51 | 11.70 |
| Muscle | 696 | 140 | 4.36 | 0 | 3.25 | 3.26 |
| Bone | 1605 | 212 | 12.51 | 3.25 | 0 | 0.0064 |
| Tissue | 1650 | 227 | 11.70 | 3.26 | 0.0064 | 0 |

**Table 4.1: Mean and variance values for four classes of a human knee MRI, and Bhattacharyya distance for each pair of classes; the higher the distance, the more separable the classes are.**

discrimination power between these two classes.

## 4.3 The Bhattacharyya Space

We define the *Bhattacharyya Space* as the space of Bhattacharyya distances of all possible pairs of classes for all dimensions of the measurement space. For measurement space $S$ generated with a 2D SOP filtering of order $o$, where the data consist of $N_k$ classes, each class pair $(p)$ between classes $k_1, k_2$ at measurement $i$ will have a Bhattacharyya distance $D_B(S^i_{k_1}, S^i_{k_2})$, and that will produce a Bhattacharyya space of dimensions $N_p = \binom{N_k}{2}$ and $N_i = 7o:\ N_p \times N_i$ (2D). The domains of the Bhattacharyya space are $L_i = \{1, 2, \ldots 7o\}$ and $L_p = \{(1,2),(1,3),\ldots (k_1,k_2),\ldots (N_k-1, N_k)\}$ where $o$ is the order of the SOP. In the volumetric case, $L_p$ remains the same (since it depends on the classes only), $N_i = 29o$ and $L_i = \{1, 2, \ldots 29o\}$.

The Bhattacharyya space, $BS_{IP}(i,p)$, is defined as:

$$L_p \times L_i; BS_{IP}(i,p) : L_p \times L_i \rightarrow D_B(S^i_{k_1}, S^i_{k_2}) \tag{4.3}$$

The Bhattacharyya space is a bivariate state whose marginal distributions are:

$$BS_I(i) = \sum_{p=1}^{N_p} BS_{IP}(i,p) = \sum_{p=1}^{N_p} D_B(S^i_{k_1}, S^i_{k_2}), \quad i = 1, \ldots, N_i \tag{4.4}$$

$$BS_P(p) = \sum_{i=1}^{N_i} BS_{IP}(i,p) = \sum_{i=1}^{N_i} D_B(S^i_{k_1}, S^i_{k_2}), \quad p = 1, \ldots, N_p \tag{4.5}$$

Both marginals can reveal important information about the feature selection process.

The marginal over the class pairs, $BS_I(i)$ sums the Bhattacharyya distance of every pair of a certain feature and thus will indicate how discriminant a certain sub-band SOP filter is over the whole combination of class pairs. On the other hand, $BS_P(p)$ sums the Bhattacharyya distance for a particular pair of classes over the whole measurement space. This is particularly revealing in when classifying for many different classes, since a particular pair can have a low discrimination potential.

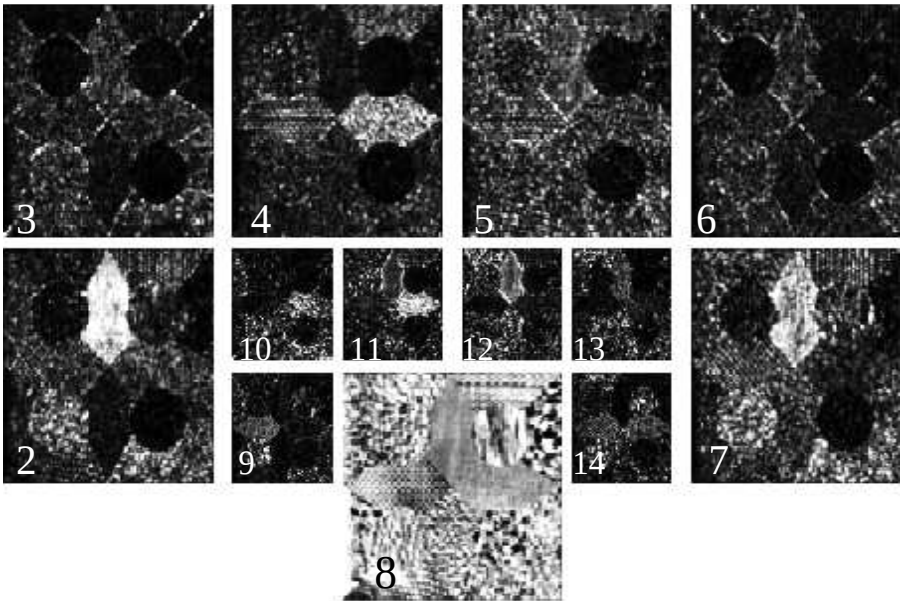
To visualise the previous distribution, the Bhattacharyya space and its two marginal distributions were obtained for the natural texture image with 16 classes (figure 3.1 (f)). Figure 4.4 (a) shows the Measurement space $S$ of order 2, and (b) presents the Bhattacharyya space. Figure 4.5 presents the corresponding marginals. These graphs present very interesting information towards the selection of the features for classification. In 4.5(a) a certain periodicity can be found (over the measurement space dimension): $BS^{1,7,14,21,28}$ have the lowest values (this is clearer in the marginal $BS_I(i)$). The measurements 1, 7, 14, 21, and 28 correspond to low pass filters of the Second Orientation Pyramid. Since the textures had been previously histogram equalised, the high pass features provided a better discrimination than the low pass features. The most discriminant features for the training data presented are $S^{19,18,11,\ldots}$.

The marginal $BS_P(p)$ can be useful to identify certain pairs of textures which are difficult to segment. The index of $p$ corresponds to the pairs $L_p = \{(1,2),(1,3),\ldots,(k_1,k_2),\ldots,(N_k-1,N_k)\}$.

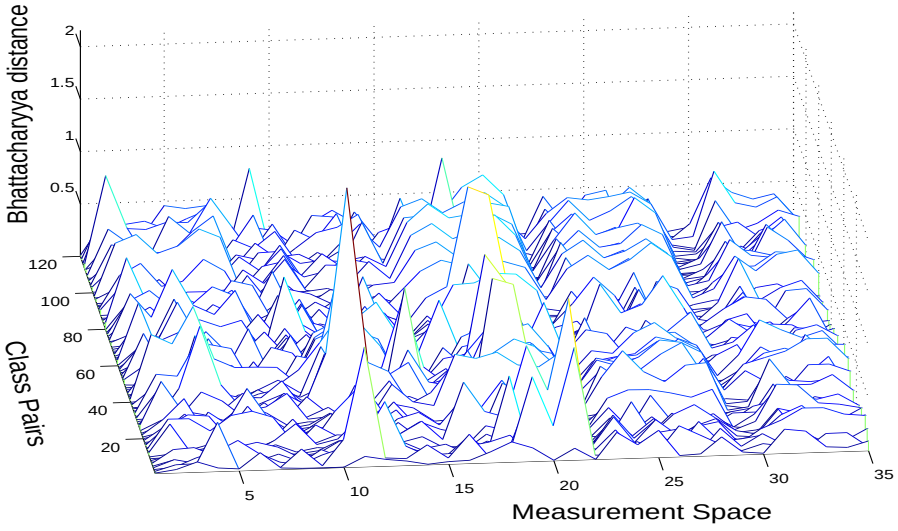
## 4.4 Order Statistics for Feature Ranking

The selection process of the most discriminant features that is proposed here uses the marginal of the Bhattacharyya space $BS_I(i)$ that indicates which sub-band filtered feature is the most discriminant. The marginal (for 2D) is a set:

$$BS_I(i) = \{BS_I(1), BS_I(2), \ldots BS_I(7o)\}, \tag{4.6}$$




(a)




(b)

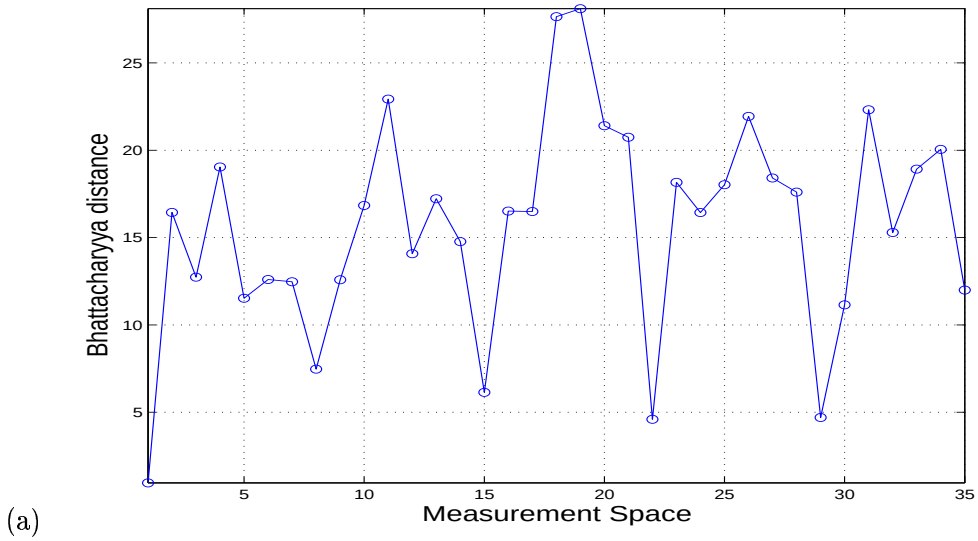
**Figure 4.4: (a) The Measurement space $S$ of order 2 for the 16-class image (b) The Bhattacharyya space $BS_{IP}$ for a measurement space $S$ of order 5.**

which can be sorted in increasing order and its order statistic will be:

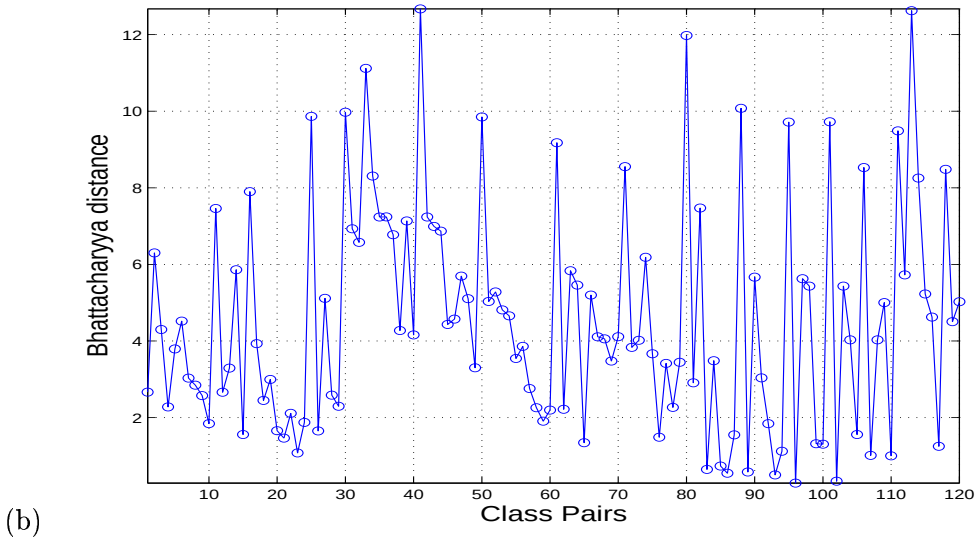
$$BS_{(I)}(i) = \{BS_{(I)}(1), BS_{(I)}(2), \ldots BS_{(I)}(7o)\}, \qquad \overset{min}{i}(BS_I(i)) = BS_{(I)}(1) \tag{4.7}$$

$$BS_{(I)}(1) \leq BS_{(I)}(2) \leq \ldots \leq BS_{(I)}(7o), \qquad \overset{max}{i}(BS_I(i)) = BS_{(I)}(7o)$$

$$BS_I(i) = BS_{(I)}(j).$$




(a)




(b)

**Figure 4.5: Marginals of the Bhattacharyya space of figure 4.4: (a) $BS_I(i)$, the index measurement space corresponds to space $S$, (b) $BS_P(p)$, The index of class pairs correspond to the pairs $L_p = \{(1,2),(1,3),\ldots(k_1,k_2),\ldots(N_k-1,N_k)\}$ .**

The domain $L_j = \{\ldots, j, \ldots\}$ provides a particular route for the state space search. In other words, a re-ordering of the elements of Measurement space $S$ is performed before being sequentially introduced to the classifier. The dimensions of the set remain the same as those of the measurement space: $N_j = N_i$.

Figure 4.6 exemplifies this route for a 4 measurement state space. It is important to mention two aspects of this selection process: the Bhattacharyya space is constructed on training data and the individual Bhattacharyya distances are calculated between pairs

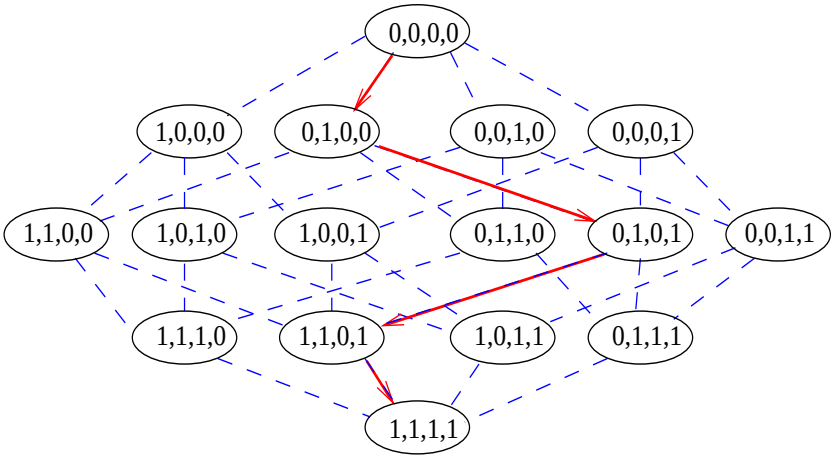



**Figure 4.6: State Space for sequential selection following the route determined from the Bhattacharyya space .**

of classes. Therefore, there is no guarantee that the feature selected will improve the classification of the whole data space, the features selected can be redundant to each other or can improve the classification for a pair of classes but not the final classification.

The conjecture to be tested is that the classification can be improved in a sequential selection defined by the Bhattacharyya space order statistics. To test this, the natural textures image was classified with several sequential selection strategies:

- Following the unsorted order of the measurement space: $S^1$,$S^2$,$S^3$ etc.
- Following the marginal $BS_I(i)$ of the Bhattacharyya space (figure 4.5 (a)) in decreasing order: $S^{19}$,$S^{18}$,$S^{11}$ etc.
- Following the marginal $BS_I(i)$ (figure 4.5 (a)) in increasing order: $S^{28}$,$S^{21}$,$S^{15}$ etc. The converse conjecture: the reverse order should provide the worst path for the classification.
- A couple of random permutations.

The sequential misclassification results of the previous strategies are presented in figure 4.7.

Although the Bhattacharyya space appears to provide the best route for classification, there are some features that when included increase the misclassification. A heuristic method is proposed to overcome this problem. If the whole state space is traversed up to the state $\{1, 1, \ldots, 1\}$, a misclassification graph will show the particular

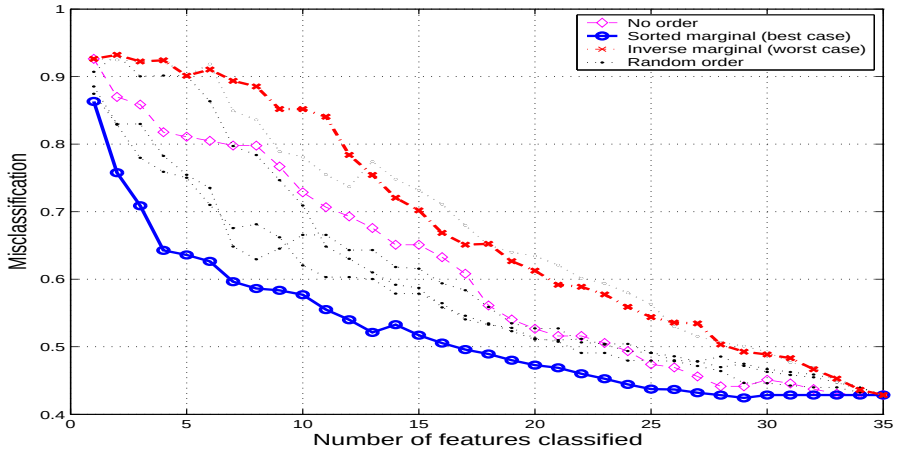



**Figure 4.7: Misclassification error for the sequential inclusion of features to the classifier for the 16-class natural textures image (figure 3.1 (f)).**

effect that each feature has on the misclassification (positive/negative). From the graph shown in figure 4.7, it can be seen that most of the features contribute positively to the classification with the exception of $BS_{(I)}(14, 30)$, and the last five features $BS_{(I)}(31-35)$ leave the classification unchanged. These features can be removed from the classification procedure:

$$S^j \in v \quad if \quad \lambda(S^1, S^2, \ldots, S^j) > \lambda(S^1, S^2, \ldots, S^{j-1}) \tag{4.8}$$

In the previous example, the set of features to be included in the classifier will be: $L_f = L_j \setminus \{14, 30 - 35\}$. $L_f$ is the domain of the *Feature Space* $S_F$; a reduced and ordered version of the measurement space $S$:

$$S_F^f \in S_F \subset S, \qquad N_f \leq N_i. \tag{4.9}$$

The dimensions of the Feature space are $L_r \times L_c \times L_d \times L_f$.

Another solution that is provided by the order statistics of the Bhattacharyya space marginal is the option to select a predetermined number of features as the *reduced set* or sub-space used for classification. This can be particularly useful in cases where it can be computationally expensive just to obtain the whole measurement space. Then,

**Table 4.2: Feature Selection through the Bhattacharyya Space.**

| | |
|---|---|
| 1 | Generate a set of training data samples for each class to be segmented. |
| 2 | Filter the data with the SOP and obtain a measurement space $S$ for each class. |
| 3 | Calculate the Bhattacharyya distance for each pair ($p$) for each measurement ($i$) and arrange them into the Bhattacharyya space $BS_{IP}(i,p)$. |
| 4 | Obtain the marginal $BS_I(i)$, and its order statistics $BS_{(I)}(j)$. |
| 5 | The measurements (filters from the SOP) will be submitted to a classifier in a wrapper approach in the order of the order statistic. This can continue up to the state $\{1,1,\ldots,1\}$ or just the first $n$ measurements can be selected as a subset when obtaining the measurements of the whole data can be expensive. |
| 6 | Once $\{1,1,\ldots,1\}$ has been reached, discard $S^j$ that do not improve the classification. |

based on the training data, just a few measurements are generated based on the first $n$ features provided by the Bhattacharyya space.

## 4.5 Summary

In this chapter we have presented a feature selection method through the use of a novel *Bhattacharyya Space* that is obtained through the Bhattacharyya distance of pairs of training classes. This method allows the selection of the most discriminant features of a measurement space $S$ through one of the marginals of the space. It can also be used for detecting which pairs of classes would be particularly hard to discriminate over all the space and in some cases, the individual use of one point of the space can be also of interest. The use of the Bhattacharyya Space implies that the number of classes is previously known, thus it is not presented as a method to determine the presence or absence of a number of clusters (one or more) in a certain space. If this is required, other methods like the *Two-point correlation function* or the *distance histogram* proposed by Fatemi-Ghomi [40] could be used.

The proposed Feature Selection methodology is summarised in the table 4.2.

# Chapter 5

# Multiresolution Classification

In the previous chapters several steps of a texture classification process have been presented. First, a sub-band filtering with a Second Orientation Pyramid (SOP), which provided textural measurements comparable to those present in the literature was described. Then, a feature selection method that allowed the selection of the most discriminant measurements was proposed. Some classification algorithms were tested and while the results obtained so far are encouraging, multiresolution techniques might further improve our results.

This chapter presents a multiresolution algorithm which we call *Multiresolution-Volumetric Texture Segmentation* (M-VTS) and compares its final results against three other sophisticated techniques: Local Binary Pattern (LBP) and the $p_8$ methods presented by Ojala [112], and Wavelet features with the Watershed transformation presented by Malpica [104]. All methods (table 6.2) easily outperform Randen's original results.

The multiresolution procedure will consist of three main stages which are annotated as follows:

↗ The process of *climbing* the levels or stages of a pyramid or tree.

- A decision or analysis at the highest level is performed, in the proposed method the space is *classified*.

↘ The process of *descending* from the highest level down to the original resolution.

The basic idea of the method is to reduce number of elements of the data, by climbing

a tree ([135, 162]), in order to reduce the uncertainty at the expense of spatial resolution. This climbing stage represents the decrease in dimensions of the data by means of averaging a set of neighbours on one level (which are called *children* elements or nodes) up to a *parent* element on the upper level. The decrease of elements should decrease the uncertainty in the elements values since they are tending to a mean. In contrast, the spatial position increases its uncertainty at every stage [162].

Classification is performed at the highest level of the tree and then propagated downwards; therefore it is crucial to obtain a good initial result before it is propagated to lower levels. A set of features that enhance the contiguity of the classification are presented in section 5.5.1. The classification is enhanced by these *Positional Contiguity Enhancing* (PCE) features, even without the new features, the multiresolution process reduces the misclassification considerably. In most of the cases, the PCE features improve the results.

Every parent node will inherit conditions (like the class it belongs to) to its children. Interaction of the neighbours can reduce the uncertainty in spatial position that is inherited from the parent node. This process is known as spatial restoration and boundary refining, which is repeated at every stage until the bottom of the tree or pyramid is reached. We tested two methods for the refining of the boundaries: *volumetric pyramidal butterfly filters*, were proposed as an extension to 3D of the 2D butterfly filters of Schroeter [135], these techniques outperform a *Markov Random Field* approach.

While the construction of a multiresolution tree of the data represents a considerable number of operations, it can actually simplify the whole process since some computationally intensive decisions (such as classification) can be performed at a lower resolution. Then, the spatial restoration can imply a lower number of operations since the areas classified will tend to be bigger and less speckled than if classified at a lower level. The main issues to be addressed when using a multiresolution technique are:

- Averaging procedure: size of neighbourhood, weighting values, etc. Two common methods are the Gaussian Pyramid [17] and the Quad/Oct Tree ($QT$) [47, 132, 140]. In our implementation we used the $QT$.

↗ Height of the tree. Although the construction is simple and the properties of the multiresolution are easy to perceive, there is no single way of determining the height of the tree. Schroeter and Bigun, who proposed a multiresolution algorithm in [135], left open the challenge of computing the optimal number of levels. We observed the variance of the space and the misclassification as indicators of the adequate height of a $QT$.

- Classification. It may be possible to use a different classifier at the top level of the tree. Unsupervised schemes could be used since the number of elements is reduced and the uncertainty is decreased.

↘ Spatial restoration. Once the data have been classified at the highest level, spatial restoration can be performed. This will typically mean the elimination of isolated elements (those with no neighbours in their own class), but it can be extended into eliminating isolated *areas* or regions, that is, small number of elements that share class but whose neighbours are in a different class. The question is then *how small is small?* In our implementation we restrict ourselves to isolated elements to avoid the arbitrary decision of the size.

↘ Boundary refinement. The boundary elements will be defined as any element with a neighbour in a class different to its own class. The refinement of the boundaries is closely related to the spatial restoration and can be successfully performed with butterfly filters.

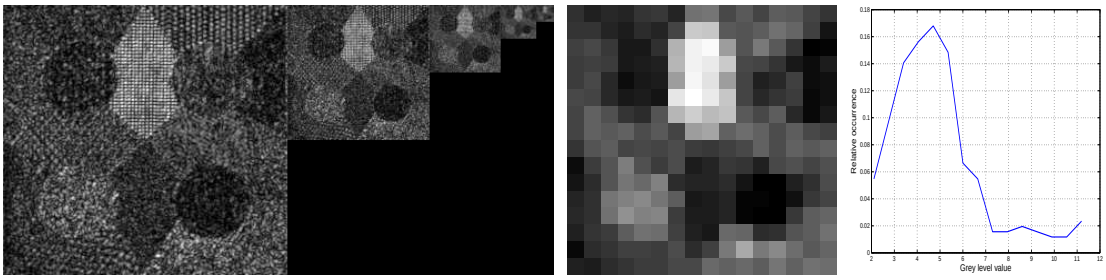
## 5.1 Climbing the Tree ↗

The measurement space $S$ that was obtained from the textured data through the SOP sub-band filters will constitute the lowest level of the tree. For each measurement $S^i$ of the space, a Quad Tree in 2D and an Oct Tree in 3D ($QT$ will be used for both cases) will be constructed. To climb a level in the $QT$, a $REDUCE$ operation (Appendix B details this operation for the construction of trees and pyramids) has to be performed:

$$(S^i)^{\mathcal{L}} = REDUCE(S^i)^{\mathcal{L}-1} \tag{5.1}$$

where $\mathcal{L}$ is the level of the tree. Accordingly, $(G)^{\mathcal{L}}$ will correspond to the range of grey levels of the tree level $\mathcal{L}$.

Every $REDUCE$ step averages four (in 2D) or eight (in 3D) contiguous elements to produce a single element in the upper level. Once a desired level is reached, the




(a) (b) (c)

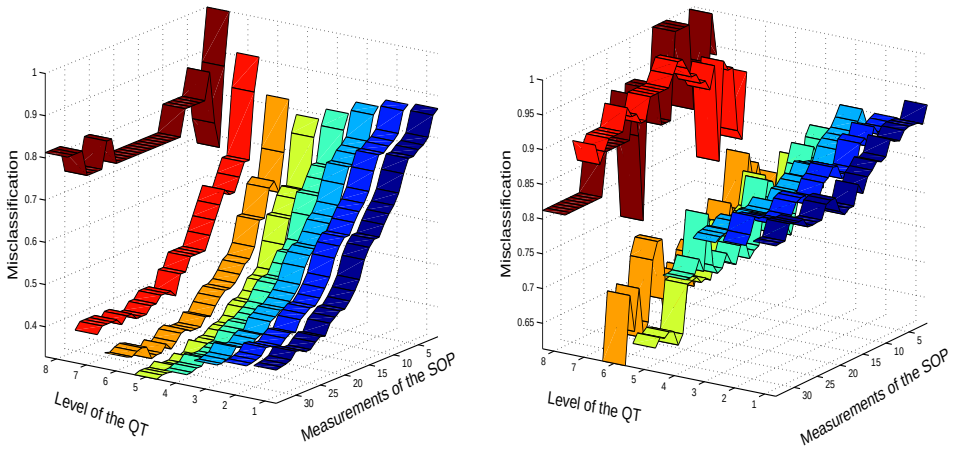
**Figure 5.1: A pyramid to reduce the uncertainty of a filtered texture image: (a) $QT$ constructed into a pyramid, (b) Level 6 of the pyramid, (c) Histogram at level 6.**

classification is performed.

Throughout this chapter we will illustrate the multiresolution procedure with the 16-texture image of figure 3.1 (f). At the sixth level of the tree $(S^i)^6$, each element is the average grey level from $32 \times 32$ neighbouring pixels at level 1. Figure 5.1 shows this process for $S^2$: (a) shows the first 5 levels of the $QT$, (b) zooms into the sixth level of the tree $(S^2)^6$ and its corresponding histogram is shown in (c). Even at this high level of the tree, the structure of the original image is preserved and some individual classes are distinguishable to the eye, in particular the one in the top-central position. The corresponding histogram shows how this texture could be discriminated on the grey level value itself because of its brightness, but the 15 remaining textures are mixed into one single (rough) bell-shaped distribution.

Two solutions will be proposed for determining the height of the tree: one will observe the variance and the other will use a classifier. First we will use a *wrapper* approach (chapter 4) which includes the classifier in the process and therefore determines the misclassification at every decision branch. Its main disadvantages are: the computational complexity and the requirement of a mask to determine the misclassification, yet, it provides a powerful methodology. We tried both supervised classification ($\hat{a}_k$) and unsupervised classification (LBG) and the results are presented in figure 5.2.

The sequential approach to the classification shows the individual participation of the measurements toward the final classification in the figure, the ribbons of figure 5.2 describe the contribution of each measurement in every level of the $QT$ towards the




(a) (b)

**Figure 5.2: Misclassification performed at every level of a Quad Tree, for every measurement of the SOP filtering Space: (a) Supervised Classification, (b) Unsupervised Classification.**

misclassification. In the supervised case (figure 5.2 (a)), two important observations can be made: first, most measurements contribute to improve the classification results; second, the classification improves as the $QT$ level is increased up to the fifth level, from there the misclassification increases. If the classification is performed unsupervised, that is, no training data, but the number of classes is supplied, (figure 5.2 (b)) a similar trend is noticed in the classification and the $QT$ level; the classification improves up to the sixth level and then it decreases. In the unsupervised case not every measurement contributes to the classification.

An alternative approach to establishing the height of the tree without the use of a classifier is through the variance of the spaces $S^i$ (figure 5.3). The variance of the central measurements (corresponding to the low pass filter) of five levels of the measurement space $S$: $S^1, S^8, S^{15}, S^{22}, \ldots$, was calculated for 8 levels of the $QT$. Two trends can be noticed. First, the higher the measurement from the SOP, the lower the variance. This should be expected since the filter is a low pass filter each time with a smaller bandwidth. Second, for the higher $\mathcal{L}$ values, the variance remains nearly constant for a number of levels of the $QT$, (which corresponds to the level of the SOP) and then it decreases and tends to a single value. A good choice of the height of the $QT$ would be the same level of the SOP or at most one level higher.

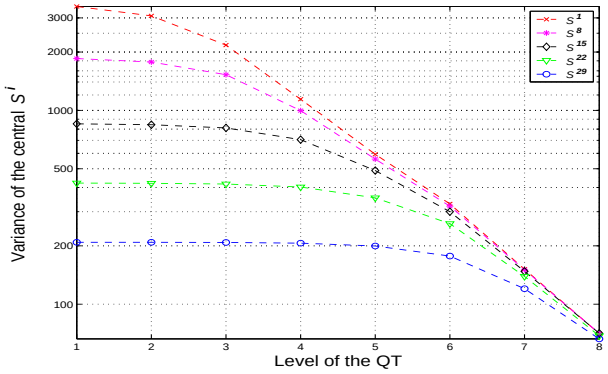



**Figure 5.3: Variance of the central SOP filters (low pass) measurements at every level of a $QT$. Notice how the variances tends to remain constant for the higher levels of the SOP filter at the lower levels of the $QT$ (see text).**

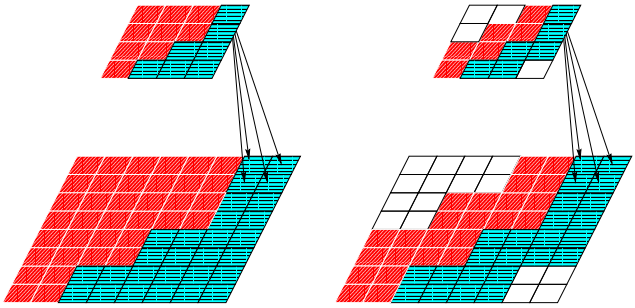



**Figure 5.4: Inheritance of properties to children elements (a) Class inheritance (b) Boundary Inheritance.**

## 5.2 Descending the Tree ↘

Once the height $\mathcal{L}$ of the tree has been decided, the tree $(S)^{\mathcal{L}}$ is constructed for every measurement of the space. At the highest level, the new reduced space can be classified, and the resulting classification has to be propagated downwards to regain full spatial resolution at the lowest level of the tree. The propagation implies that every parent inherits: (a) its class value to 4/8 children and, (b) the condition of being or not in a boundary (figure 5.4).

The descent from the highest level of the $QT$ requires spatial restoration and boundary refinement at every level; otherwise the result at the original level would look exactly the same as the highest level [135]. For this purpose, two conditions of the elements with respect to their neighbours are defined: *boundary element* and *isolated element*.

- An element will be considered as a boundary element when at least one of its neighbours (8 in 2D and 26 in 3D) belongs to a different class.
- An element will be considered as isolated when no contiguous neighbour belongs to its class assignment.

It is possible to define *isolated regions* as groups of elements of one class that are surrounded by elements of different classes. Since the number of elements that define how small the region is has to be arbitrary, we preferred not to use isolated regions in the spatial restoration process.

The first step in the spatial restoration process is to enhance the spatial contiguity of the elements by erasing isolated pixels. In this case, the mode class of the neighbours is assigned to the pixels. After this refinement, boundaries are re-estimated. The refinement of the boundaries is performed with the butterfly filters described in section 5.3.

## 5.3 Butterfly filters

In previous sections it was shown how smoothing can greatly improve the classification results, yet smoothing itself introduces some classification problems. When spatial neighbouring elements are averaged (either with Gaussian or uniform isotropic LEFs) there will be an error introduced, especially at the boundaries between classes. When the problem of classification consists only of 2 classes this error is not great. But when more than 2 classes are to be identified, the smoothing procedure can turn elements in the boundaries into a different class than that associated with both sides of the boundary. This problem is depicted graphically in figure 5.5, and is present in the results of figure 3.5, where, as the size of the LEF increases, a region of misclassification between regions grows bigger.

Yet, the effect of the local energy function can greatly improve the results by eliminating the variation of neighbouring elements. The problem then is to smooth selectively: in regions that are far away of the class boundaries an isotropic smoothing could be adequate, but near to the boundaries, careful treatment of the data is needed. This in turn poses the problem of determining class boundaries within the data. If the results of

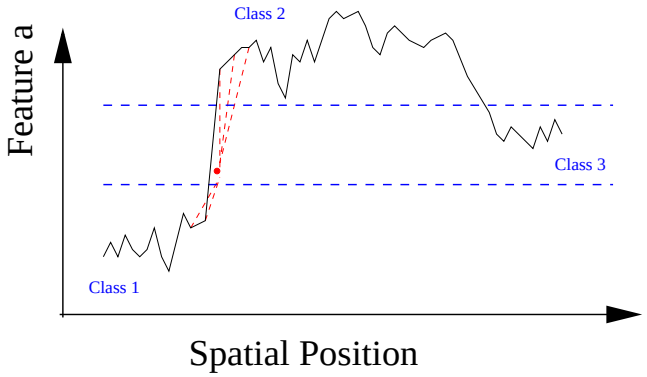



**Figure 5.5: Classification errors due to averaging of neighbouring elements.**

the filtering process have not been smoothed, then the classification results can be very noisy and isolated elements can be very common (even with a $3 \times 3$ Gaussian LEF as it is shown in figure 3.5 (a) a very speckled result is obtained). Determining boundaries at this level would be considerably difficult.

A compromise has to be reached between the smoothing and the class boundaries. The use of butterfly filters in a multiresolution classification can successfully smooth close to boundaries reducing the class mix. A solution to this problem is to apply a multiresolution procedure of smoothing several levels with a $QT$ and applying these filters in the descent to the lowest resolution.

Butterfly filters ($BF$) [135] are orientation-adaptive filters, that consist of two separate sets or *wings* with a pivot element between them. It is the pivot element $x = (r, c, d)$ which can be modified as a result of the filtering. Each of the wings will have a roughly triangular or pyramidal shape, which resembles a butterfly (figure 5.6 (a)) and they can be regarded as two separate sets of *anisotropic cliques*, arranged in a steerable orientation. We propose the extension of these $BF$ filters into 3D, and two possible shapes can be used: pyramidal or conic (figure 5.6 (b,c)), for ease of implementation we used pyramidal. The boundary determined by the classification process will define the orientation of the filter in order to place each of the wings of the butterfly at the sides of the boundary. The pivot element $x$ at which the filter is centred, is not included in the wings. The elements covered by each of the wings will be included in the filtering

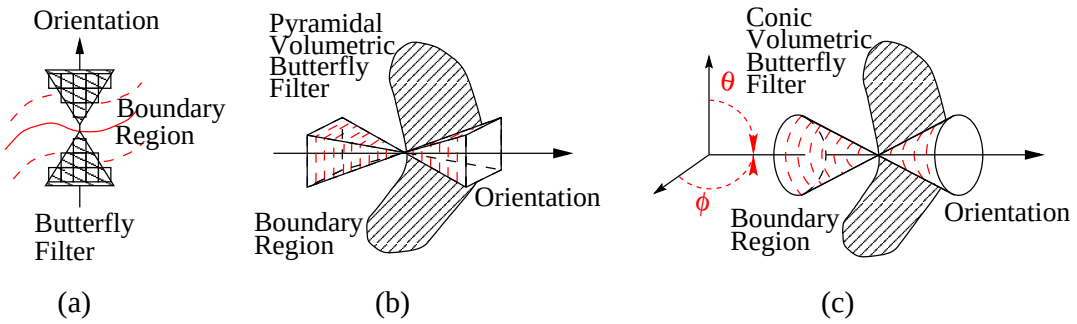



**Figure 5.6: (a) 2D Butterfly filter, (b) Pyramidal volumetric butterfly filters, (c) Conic volumetric butterfly filters. Orientation of $\phi$ and $\theta$ indicated in (c).**

process while the values of the elements along the boundary (which presumably have greater uncertainty) are not included in the smoothing process.

The $BF$ will consist of two sides, which we call left wing $lw$ and right wing $rw$, each of which will comprise $N_w$ elements:

$$\begin{aligned} lw &= \{lw_1, lw_2, \ldots, lw_{N_w}\} \\ rw &= \{rw_1, rw_2, \ldots, rw_{N_w}\} \end{aligned} \qquad lw, rw \in S. \tag{5.2}$$

For each wing, a weighted average of the values of the elements in each dimension is calculated:

$$\tilde{S}^i_{lw}(x) = \sum_{q=1}^{N_w} S^i(lw_q) w_q, \qquad \tilde{S}^i_{rw}(x) = \sum_{q=1}^{N_w} S^i(rw_q) w_q. \tag{5.3}$$

In the case that all the elements of the wing $lw/rw$ are to be included with equal weight, the weighting function will simply be $w_q = \frac{1}{N_w}, \forall q$. Once the average values of the butterfly wings $\tilde{S}_{lw}$, $\tilde{S}_{rw}$ have been calculated, the actual pivot element $x = (r, c, d)$ value (which was not included in the wings) is weighted averaged with them:

$$\tilde{S}_{x-lw} = (1-\alpha)S(x) + \alpha\tilde{S}_{lw}, \tag{5.4}$$

$$\tilde{S}_{x-rw} = (1-\alpha)S(x) + \alpha\tilde{S}_{rw}, \tag{5.5}$$

The weighted combination is controlled by a parameter $\alpha$ (figure 5.7) which is a scalar

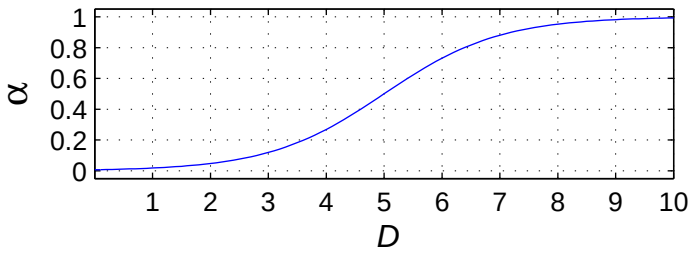



**Figure 5.7: Scalar gain $\alpha$.**

gain measure that depends on the dissimilarity of the distribution of the elements within the two butterflies:

$$\alpha = \frac{1}{1 + e^{(5-D)}}, \qquad D = \frac{|\tilde{S}_{lw} - \tilde{S}_{rw}|}{\sqrt{\sigma_{lw}^2 + \sigma_{rw}^2}} \tag{5.6}$$

where $\sigma_{lw}^2/\sigma_{rw}^2$ is the variance of the elements in each butterfly wing. Equation 5.6 replaces the experimental relation previously used [135, 162]. (The value of 5 was obtained after numerous simulations, similar results were obtained for values of 4 to 6.) The parameter $\alpha$ acts as a weighting factor derived from the closeness to a boundary or the distance between the distributions covered by the two sides of the butterfly filter, and provides a balance between the current position of the element and a new one calculated from its neighbours. It is interesting to note that this balancing procedure is quite similar to the update rule of the Kohonen SOM [84]. The distance measure between the updated pivot element and the prototype values of each class will determine to which class it will be reassigned. Figure 5.8 shows the process graphically. At the classification stage, the new positions $\tilde{S}_{x-lw}, \tilde{S}_{x-rw}$ replace the original position of element $x$. Instead of looking for a class based on $\lambda_a(S(x))$, the new values $\lambda_a(\tilde{S}_{x-lw})$ / $\lambda_a(\tilde{S}_{x-rw})$ will determine the class according to the closer distance to the prototypes of the classes. When dealing with volumes and not images, the boundaries between classes are not single lines but planes, and therefore the orientation of the butterflies requires two parameters $\theta$ and $\phi$. If the analysis is restricted to 2D, a complex number field can be used to simplify and improve speed [8]. In our implementation, we quantised the

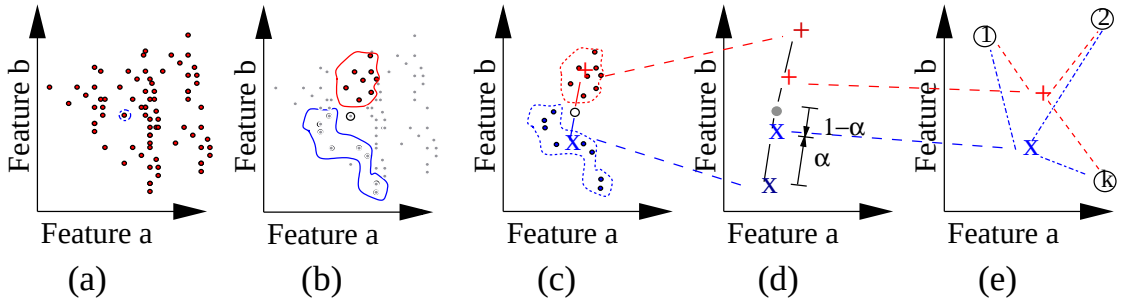



**Figure 5.8: A feature space view of boundary refinement process with butterfly filters. (a) A boundary element $x$ with other elements. (b) $x$ and the two sets of neighbouring elements that are comprised by the butterfly wings, all other elements are not relevant at this moment. (c) The weighted average of each wing. (d) Parameter $\alpha$ balances between the element $x$ and the average of the wings. (e) New positions are compared with the prototypes (1,2,. . . ,k) of the classes, the class that corresponds to the minimum distance is then assigned to $x$.**

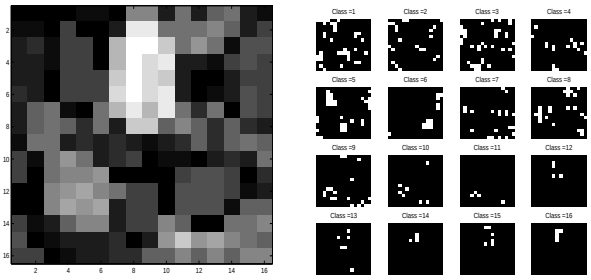
orientations to four in each case: $\theta, \phi = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$.

The effect of butterfly filters bears resemblance to *Anisotropic Diffusion* presented by Perona and Malik [115] where images are smoothed at different scale-spaces, thus removing high frequency noise, while preserving the boundaries of structures of interest. Anisotropic diffusion can enhance edges and be used in multiscale image segmentation. It remains as further work to compare the results of anisotropic diffusion with the butterfly filters.

## 5.4 Supervised Single Resolution vs. Unsupervised Multiresolution

One of the advantages of the multiresolution algorithms is that at the highest level of the $QT$, the reduction of the number of elements allows the use of some algorithms that would be too expensive at a lower level. We will now analyse the use of unsupervised classification.

If classification was to be attempted with $(S^2)^6$ (figure 5.1 (b)) with this *single* feature of the space and no previous knowledge of the different textures, the 16 classes would be distributed along the grey level axis. Figure 5.9 shows the classification into 16




(a) (b)

**Figure 5.9: Classification of figure 5.1 (b): (a) Classes as levels of grey, (b) Individual Classes.**

classes with an agglomerative hierarchical clustering with complete linking [54] (identical results were obtained with the LBG algorithm [97]). The results of the classification are far from impressive; some classes, like 1, 3 or 7, are scattered all around the image.

Of course that the classification does not have to be performed with only one of the measurements. When all the measurements are included into the classifier the results improve significantly. Figure 5.10 (a) shows the final results: propagated downwards through the tree, spatially restored and butterfly filtered of the multiresolution classification. The single resolution classification is presented in (b) for comparison purposes.

The first striking difference is the smoother appearance of the multiresolution classification. The speckled appearance of the single resolution (without LEF) is not present in the multiresolution where the classes appear smooth. A second important difference is revealed if we look at the elements that have been correctly classified (c,d). While the single resolution classification can have a certain number of correctly classified pixels in all the classes, the multiresolution classification is more clear-cut: either it classifies the region correctly or it is totally misclassified. A third important difference appears in the class boundaries; the multiresolution procedure has very sharp boundaries. This property is given by the butterfly filters; this will become clearer when Markov Random Fields are compared with this technique.

While the misclassification is similar; 42.2% with the supervised single resolution against 45.5% of the unsupervised multiresolution, it is important to remember that

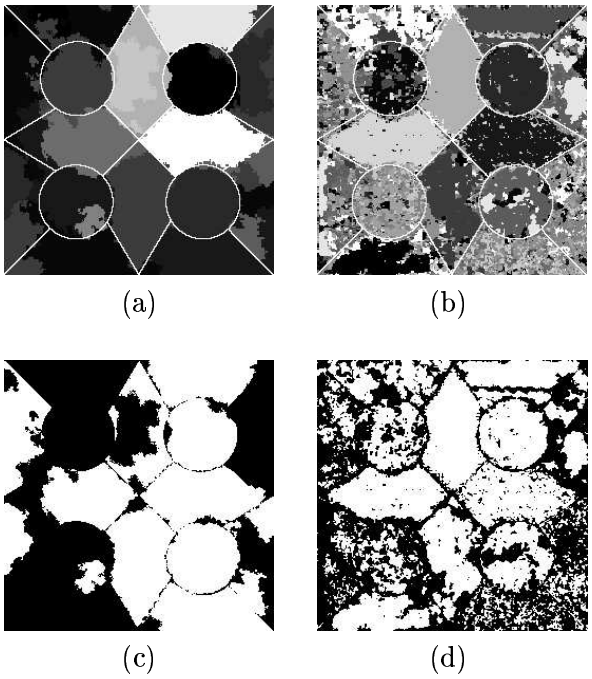



**Figure 5.10: Classification of a 16-class natural texture image. (a) Unsupervised multiresolution, misclassification = 45.5% (Schroeter and Bigun), (b) $\hat{a}_k$ Supervised, misclassification = 42.2%. (c), (d) Correctly classified pixels for (a) and (b) (correct labelled in white).**

the unsupervised single resolution misclassification of this image was 78.6% (table 3.6). Multiresolution processing considerably improves the results. We will now present an improvement to the method.

## 5.5 Positional Contiguity Enhancing Features

The inadequacy of the unsupervised classification previously presented is partly due to the lack of feature context. In this section we introduce a simple modification, which attempts to address this problem and include the spatial position of the elements into the classifier. This modification is the insertion of extra dimensions that correspond to rows and columns for an image and rows, columns and slices for a volume as it was suggested by Jain and Farrokhnia [71]. This procedure is similar to a region growing process without the need of seeds for the regions. We describe the process below.

The grey level histogram for the volumetric data $\mathcal{VD}$ at any level $\mathcal{L}$, is defined [163]

as:

$$h(g) = \frac{\#\{x \in (L_r \times L_c \times L_d) : \mathcal{VD}(x) = g\}}{\#\{L_r \times L_c \times L_d\}}, g \in G$$

where # denotes the number of elements in the set. This histogram can be considered as a marginal function of a higher dimensional space. The univariate distribution represented by the histogram will relate to a function $f_h(g)$:

$$h(g) = f_h(g) = \sum_{1}^{N_r} \sum_{1}^{N_c} \sum_{1}^{N_d} f(\mathcal{R}, \mathcal{C}, \mathcal{D}, (G)^{\mathcal{L}}) \tag{5.7}$$

where $f(\mathcal{R}, \mathcal{C}, \mathcal{D}, (G)^{\mathcal{L}})$ is a joint density of $(G)^{\mathcal{L}}$, the grey level space at the level $\mathcal{L}$ of the Quad Tree, and a series of uniformly distributed spaces $\mathcal{R}, \mathcal{C}, \mathcal{D}$, which we will call *Row* and *Column* and *Slice* co-ordinate. The dimensions of the data to classify, two for images and three for volumetric data, will determine the number of features. The new features – called Positional Contiguity Enhancing (PCE) features – are defined by:

$$\begin{aligned}
&L_r \times L_c \times L_d;\ \mathcal{R} : L_r \times L_c \times L_d \to L_r \quad r \mapsto L_r^i \times L_c \times L_d : L_r^i = \{1 \ldots N_r\} \\
&L_r \times L_c \times L_d;\ \mathcal{C} : L_r \times L_c \times L_d \to L_c \quad c \mapsto L_r \times L_c^i \times L_d : L_c^i = \{1 \ldots N_c\} \\
&L_r \times L_c \times L_d;\ \mathcal{D} : L_r \times L_c \times L_d \to L_d \quad d \mapsto L_r \times L_c \times L_d^i : L_d^i = \{1 \ldots N_d\}
\end{aligned} \tag{5.8}$$

These new features look like a vertically shaded image for $\mathcal{C}$ and a horizontally shaded image for $\mathcal{R}$. To illustrate the additional features figure 5.11 shows the scatter plot of the distribution $f(\mathcal{C}, (G)^{\mathcal{L}})$ with its corresponding marginals. If the data are normalised as suggested in section 3.1, then the values of the PCEs are adjusted to spread the same range of values as the measurement space.

### 5.5.1 Clustering with the PCE features

The measurement space $S$ extracted by the SOP from the original data, and reduced by a Quad Tree up to a level $\mathcal{L}$ will provide the data $(S)^{\mathcal{L}}$ that we will proceed to cluster

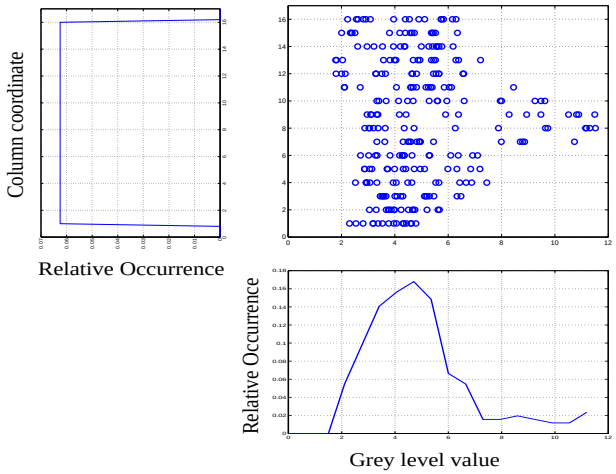



**Figure 5.11: Joint density (scatter plot) and marginal densities of the grey levels of figure 5.1 (b) ($(S^2)^6$) and column co-ordinate feature space $\mathcal{C}$.**

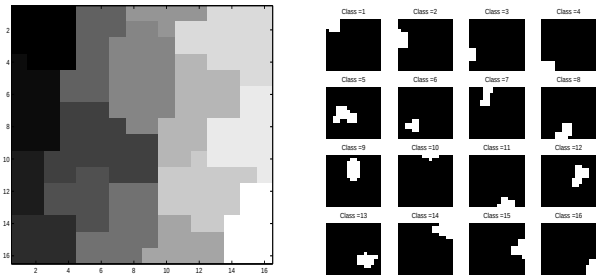
into homogeneous groups. If we include the PCE features to the space, the Euclidean distance $D_2(x_1, x_2) = ||S(x_1) - S(x_2)||_2$ between two elements $(x_1, x_2) \in (S)^{\mathcal{L}}$ will become:

$$\tilde{D}_2(x_1, x_2) = \left((r_{x_1} - r_{x_2})^2 + (c_{x_1} - c_{x_2})^2 + (d_{x_1} - d_{x_2})^2 + (D_2(x_1, x_2))^2\right)^{\frac{1}{2}} \qquad (5.9)$$

where the positional values of the elements are $(r_{x_1}, c_{x_1}, d_{x_1}), (r_{x_2}, c_{x_2}, d_{x_2})$.

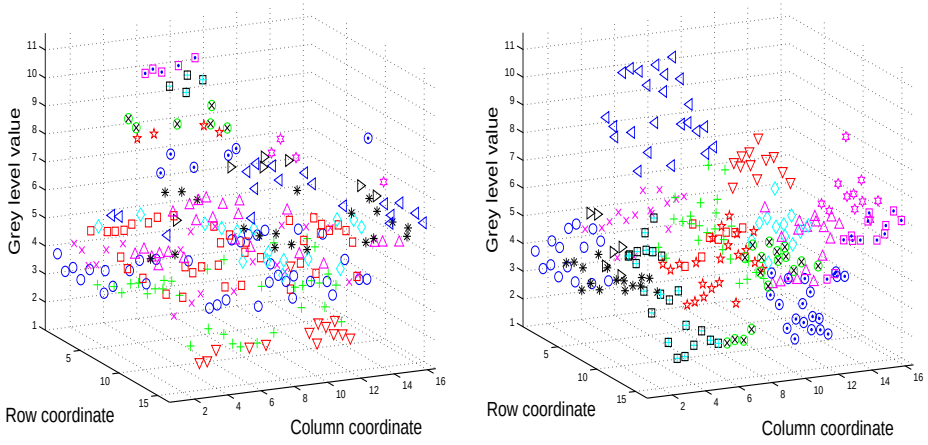
Figure 5.12 shows the classification, unsupervised agglomerative clustering with complete linking, of $(S^2)^6$ plus $\mathcal{R}, \mathcal{C}$. While the clusters may not be as compact in the measurement space, the spatial classification results into 16 contiguous classes without any isolated pixels. Figure 5.13 shows the scatter plot of $\mathcal{R}, \mathcal{C}, (G)^{\mathcal{L}}$ with the corresponding classifications, with and without the new features, and figure 5.14 presents the corresponding marginals.

To visualise the effect of the PCEs the following example is presented. A 2D, 2-measurement test set of $64 \times 64$ elements was generated with random noise with mean grey level of 100 and standard deviation of 10. Classification was performed with 2, 4 and 8 classes with and without PCEs. The results are presented in figure 5.15. As it could be expected, when the PCEs were not introduced into the classifier, the elements




(a) (b)

**Figure 5.12: (a) Classification of $(S^2)^6$ with the new features $\mathcal{R}, \mathcal{C}$ into 16 classes. (b) Individual Classes**



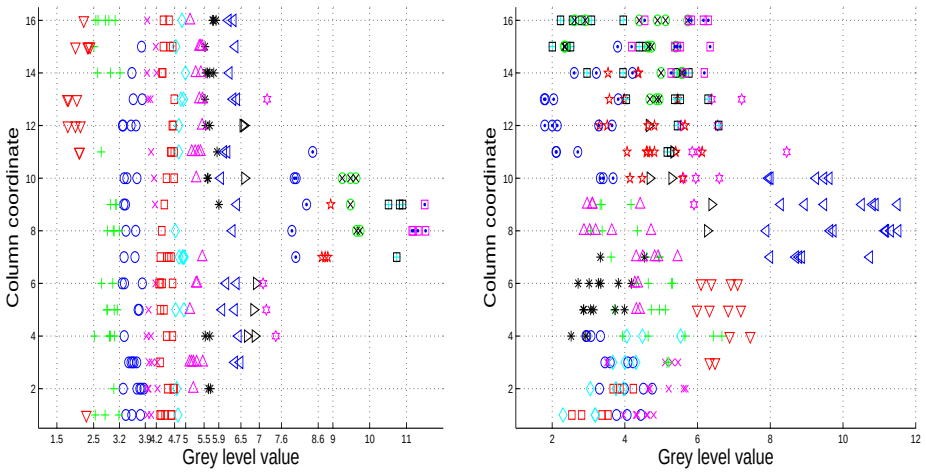

(a) (b)

**Figure 5.13: Clustering of grey levels of figure 5.1 (a) without the new features $\mathcal{R}, \mathcal{C}$ (b) with the new features $\mathcal{R}, \mathcal{C}$.**

were joined into compact clusters in the measurement domain with very low contiguity in the image domain (a-f). In contrast, contiguous results (g-l) were obtained with PCEs while the clusters are not compact at all.

## 5.6 Comparison to Markov Random Fields Models

The operation performed with the butterfly filters bears a resemblance to Markov Random Fields (MRF) models [21, 29, 48] in two ways: the neighbour to neighbour interaction of standard MRF [14] and the interaction of parent and children nodes in the Multiresolution Markov Random Fields (MMRF) models [159]. If we consider any given




(a) (b)

**Figure 5.14: Clustering of grey levels of figure 5.1: (a) Marginal of figure 5.13 (a) over $\mathcal{C}$, (b) Marginal of figure 5.13 (b) over $\mathcal{C}$.**

voxel $x = (r, c, d) \in (L_r \times L_c \times L_d)$, whose grey level is $g$, and that it is assigned to class $k$, $K_x = k$, the *a posteriori* probability model [114] $p(K|g)$ (or $p(K|S(x))$ for the whole measurement space) is given by Bayes' rule:

$$p(K|g) \propto p(g|K)p(K) \tag{5.10}$$

where $p(K)$ is the *a priori* density of the region process and $p(g|K)$ is the conditional density of the observed data. The Markov process implies the existence of a neighbourhood $\mathcal{N}_x$ of $x$ whose elements are the only elements of the data that present an influence to $x$:

$$p(K_x|K_q,\ \forall\ q \neq x) = p(K_x|K_q, q \in \mathcal{N}_x) \tag{5.11}$$

For images, 4-connected or 8-connected neighbourhoods are common, that is, the 4 nearest pixels or 8 including the diagonals. For volumes, 6 or 28 neighbours need to be considered (Figure 5.16) (a,b). The density of $K$ can be described by a Gibbs distribution:

$$p(K) = \frac{1}{Z} exp\left\{ -\sum_{C} V_C(K) \right\} \tag{5.12}$$

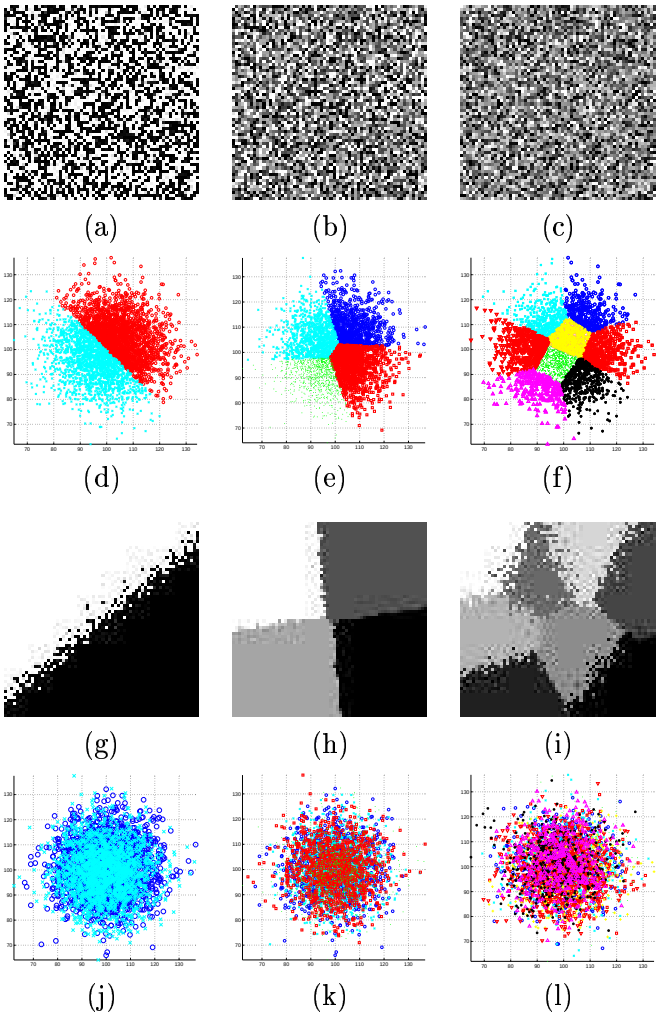



**Figure 5.15: The effect of classifying noise with and without PCEs. Image domain and scatter plot of noise ($\mu = 100,\ \sigma = 10$): (a-f) Classification assuming 2, 4 and 8 classes and no PCEs, (g-l) Classification assuming 2,4 and 8 classes and with PCEs.**

where $Z$ is a normalising constant and $C$ is a series of cliques, a set of points neighbours to each other. $V_C$ are the clique potentials that depend only on the elements that belong to that clique: for two-point cliques, the potentials could be:

$$V_c(K) = \left\{ \begin{array}{ll} -\beta & if\ K_x = K_q \quad and\ x, q \in C \\ \beta & if\ K_x \neq K_q \quad and\ x, q \in C \end{array} \right\} \tag{5.13}$$

With $\beta$ positive, neighbours are more likely to belong to the same class than to different classes. If $\beta = 0$, then the classes are equiprobable and the process reduces to a k-means

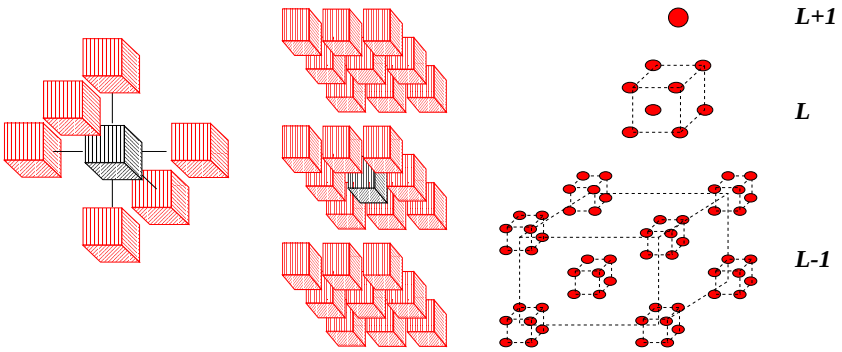



**Figure 5.16: Multiresolution Markov structures: (a) 6 voxel neighbourhood (b) 28 voxel neighbourhood, (c) An Oct-tree with three levels: grandparent $\mathcal{L}+1$, parent $\mathcal{L}$, child $\mathcal{L}-1$.**

clustering [114]. When the analysed field is extended to a multiresolution field, a voxel will be conditioned by an extra term: the *quad-tree parent* [159, 160] that belongs to a parent set. (Three levels of an Oct-tree structure are shown in figure 5.16 (c)). The parent node of an element $x$ at level $\mathcal{L}$ is:

$$P_{x,\mathcal{L}} = (\left\lfloor \frac{r}{2} \right\rfloor, \left\lfloor \frac{c}{2} \right\rfloor, \left\lfloor \frac{d}{2} \right\rfloor, \mathcal{L}+1) \tag{5.14}$$

so the conditional probability is conditioned jointly on the $\mathcal{N}_x$ and $P_{x,\mathcal{L}}$

$$p(K_x | K_q, q \in \mathcal{N}_x \bigcup P_{x,\mathcal{L}}) \tag{5.15}$$

Essentially, the local MRF neighbourhood scans the classes that surround the element being analysed and uses them to change the probability of belonging to a certain class, while the butterfly filters use the grey levels of the surrounding elements to anisotropically smooth the value of a feature. Both steps are used in the latter classifying stage.

The final results of this chapter are presented in figure 5.17. The classification of the 16-texture image of figure 3.1 (f) was supervised with the use of training data to obtain the estimates of the means (prototypes) for each class $\hat{a}_k$ and with PCE features in the level 6 of the $QT$. Figure 5.17 (a) used a MMRF approach towards the boundary

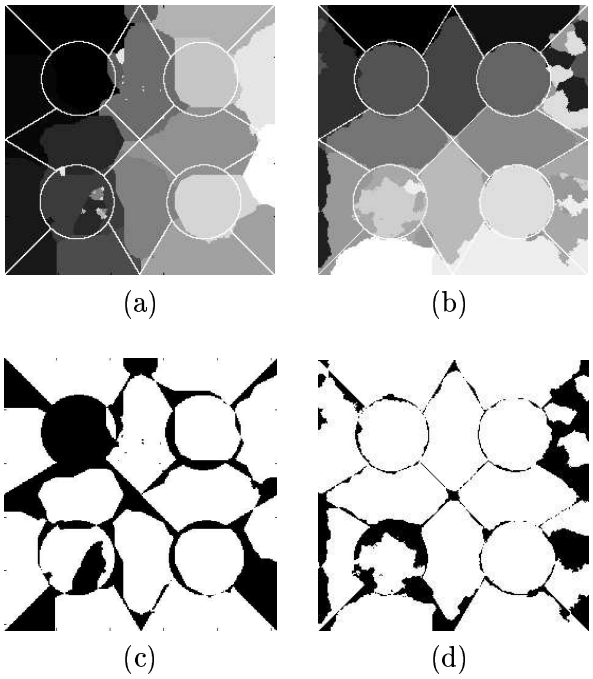



**Figure 5.17: Supervised classification of a 16-class natural texture image with PCE features: (a) MMRF approach, misclassification = 26.5%, (b) M-VTS, misclassification = 16.5%. (c), (d) Correctly classified pixels for (a) and (b) (correct labelled in white).**

refinement and (b) used the butterfly filters (M-VTS), the respective misclassification results were 26.5% and 16.5%. Both cases outperform the single resolution supervised and multiresolution unsupervised. In the M-VTS the borders between classes are much sharper than the MMRF approach; this is due to the anisotropy of the butterflies. Again, the classes in both methods are very smooth, and some of the regions have been completely misclassified.

## 5.7 Summary

This chapter presented a multiresolution algorithm that can improve the results of classification of textured data. In our implementation we selected a Quad Tree as a method for data reduction by the combination of neighbouring elements (considered as siblings) to form a parent element. It is expected that Gaussian pyramids provide similar results. The height of the tree is important for the algorithm; we presented two ways of selecting

**Table 5.1: Comparative misclassification (%) with different algorithms for the image 3.1 (f).**

| Single resolution | | Multiresolution $\mathcal{L}$=5 | |
|---|---|---|---|
| Algorithm | Misc. | Algorithm | Misc. |
| Unsupervised | 78.6 | Unsupervised | 45.5 |
| Supervised, No LEF | 42.2 | Supervised: MMRF with PCE | 26.5 |
| Supervised, LEF $9 \times 9$ | 35.5 | Supervised: M-VTS no PCE | 18.6 |
| Supervised, LEF $27 \times 27$ | 28.7 | Supervised: M-VTS with PCE | 16.5 |

an adequate height: one through the variance of some of the measurements (low pass) and the other with the use of a classifier. The latter one gives a clear idea of the optimal height of the trees. In our experiments, we used 5 levels.

The use of multiresolution allows the use of unsupervised classification techniques since the number of elements and uncertainty in the element values are reduced. Still, if training data sets are available, supervised classification provides better results both for single and multiresolution.

We experimented with a new set of Positional Contiguity Enhancing (PCE) features that improve the classification results. The strength of PCE features for classification relies on the fact that PCE will encourage contiguous regions in the data, either images or volumes, instead of clustering compact regions in the feature domain. An important distinction to be made is that the PCEs are independent of the algorithm used, they are included as part of the classification space. This augmentation of the space has the disadvantage of the increased computational complexity as mentioned by Matas [105], but since this addition will come in a reduced space at the highest level of the tree, it is not that expensive. Moreover, adding 3 dimensions to a space of 28 or 35 is not too costly. It can be expected that the PCE features will be most helpful when classes are distributed in regular compact regions, similar to the ones showed in the previous figures (and as most examples in the literature).

It is also possible to modify the classification algorithms. One option is that presented by Theiler and Gisler [145] who include a weighting factor that balances the ratio of

contiguity and compactness for the clusters. Another solution is that of Matas and Kittler [105] who use global and local statistics computed from the image together with pixel connectivity information to provide a simultaneous feature space-image space clustering. It remains as further work to test these and other algorithms.

Finally, boundary refinement was compared between a MRF approach and butterfly filters (BF). BFs are steerable anisotropic structures that try to capture under each of its wings elements belonging to different classes. The orientation of the filters is determined by the boundary condition of the elements of the space which is propagated downwards from the top level of the tree. When the butterfly is extended to 3D, it can take a conic or pyramidal shape, we chose pyramidal for ease of implementation with satisfactory results. The BFs outperform the Markovian approach. Table 5.1 presents a summary of the results for the image of 3.1 (f).

# Chapter 6

# Results and Discussion

In this chapter a series of experiments with the M-VTS algorithm will be presented. The set of 9 images from Randen [124] will be compared with co-occurrence, due to its popularity, the best of Randen's results, LBP and $p_8$ from [112], and the watershed transformation [104]. These experiments are necessary to compare M-VTS against different classification algorithms and feature extraction techniques.

Then, volumetric experiments are presented with two sets of artificial data and three MRIs of human knees. As an application of the segmentations that can be obtained, the cartilages of two of the MRIs were extracted.

## 6.1 2D Natural Textures

Table 6.1 presents characteristics of the 9 images composed of different natural textures, as well as the classification details for each of them. The measurements were selected from the SOP sub-band filtering (without LEF) through the Bhattacharyya space. Then, for each selected feature, a $QT$ of 5 levels was constructed and the classification was performed at the highest level with and without the PCE features. Butterfly filters were used to refine the boundaries on the descent of the $QT$. To evaluate the performance of the proposed method (M-VTS), a comparison was made against the best results of Randen, the results of Ojala [112] who used Local Binary Patterns (LBP) and

**Table 6.1: Characteristics of the images and their classification details**

| Figure | Size | Classes | Source | Number of Features selected | |
|---|---|---|---|---|---|
| | | | | No PCEs | with PCEs |
| a | $256 \times 256$ | 5 | Brodatz | 21 | 21 |
| b | $256 \times 256$ | 5 | MIT | 35 | 28 |
| c | $256 \times 256$ | 5 | MIT | 28 | 28 |
| d | $256 \times 256$ | 5 | MIT | 14 | 28 |
| e | $256 \times 256$ | 5 | MeasTex | 14 | 21 |
| f | $512 \times 512$ | 16 | Brodatz | 28 | 28 |
| g | $512 \times 512$ | 16 | MIT | 21 | 21 |
| h | $256 \times 640$ | 10 | Brodatz | 21 | 21 |
| i | $256 \times 640$ | 10 | MIT | 28 | 21 |

**Table 6.2: Comparative misclassification (%) results of Malpica [104], Randen [124], Ojala [112] and M-VTS (Bold is the best for each image).**

| Method | Figures | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | a | b | c | d | e | f | g | h | i | Average |
| Co-occurrence | 9.9 | 27.0 | 26.1 | 51.1 | 35.7 | 49.6 | 55.4 | 35.3 | 49.1 | 37.69 |
| Best in Randen | 7.2 | 18.9 | 20.6 | 16.8 | 17.2 | 34.7 | 41.7 | 32.3 | 27.8 | 24.13 |
| $p_8$ (Ojala) | 7.4 | 12.8 | 15.9 | 18.4 | 16.6 | 27.7 | 33.3 | 17.6 | 18.2 | 18.66 |
| LBP (Ojala) | 6.0 | 18.0 | 12.1 | 9.7 | 11.4 | 17.0 | 20.7 | 22.7 | 19.4 | 15.22 |
| Watershed (Malpica) | 7.1 | **10.7** | 12.4 | 11.6 | 14.9 | 20.0 | **18.6** | 12.0 | 15.3 | 13.62 |
| M-VTS (No PCEs) | **3.1** | 14.8 | 8.6 | **7.3** | 5.4 | 18.6 | 32.0 | 14.7 | 20.2 | 13.86 |
| M-VTS (with PCEs) | 3.2 | 14.2 | **7.9** | 9.0 | **4.7** | **16.5** | 29.4 | **8.8** | **12.9** | **11.86** |

multidimensional distributions of signed grey-level differences ($p_8$), and those reported by Malpica [104] who used a multichannel watershed-based algorithm with Wavelet features. The results of Randen's co-occurrence are included in the comparison since they are widely used.

The final classification results are presented in table 6.2 and the following observations can be made.

- It should be noted that co-occurrence can easily be outperformed, it is the worst classification individually and overall.
- The best results presented by Randen were outperformed by all the other methods. Again this was to be expected, since the classification schemes were far more complex than those used by Randen.
- The methods proposed by Ojala outperform those of Randen and have good results, in some cases they are better than Malpica's, but in general they can be

outperformed.

- The multichannel watershed-based algorithm (Malpica [104]) presents very good results, in two cases (b, g) it has the lowest misclassification.
- The M-VTS Multiresolution algorithm proposed in this thesis presents very good results even without the use of PCE features. But it is the use of these features that gives the best overall results to M-VTS. Without the PCEs, Malpica has a better classification. With the PCEs, M-VTS has the lowest misclassification.
- The use of PCEs does not always improve the results. In 7 of the 9 cases (b,c,e,f,g,h,i) the results improved with the use of the PCEs but for cases (a,d) the results did not improve, for (a) it was marginal but for (d) it was considerable. Still, the result is better than all the other methods.

As an indication of the computational complexity of the algorithm presented, we measured the computation time of the programs running with Matlab version 6.5 R13 running on a Linux platform based on a Pentium 4 CPU 2.80 GHz. The time for the 16-class segmentation of figures 3.1(f), was 2.7s for single resolution and 56.3s for M-VTS. No systematic attempt to make the code more efficient was made. The figures corresponding to the classification results are presented below. Figure 3.1 presents the 9 original images. Figure 6.1 shows the boundaries on top of the original images, figure 6.2 shows the results as classified regions, and figure 6.3 shows the pixels that are correctly classified. These latter results are considered by the author to be the most revealing since showing only the labelled classes or only the boundaries on top of the original images can be misleading.

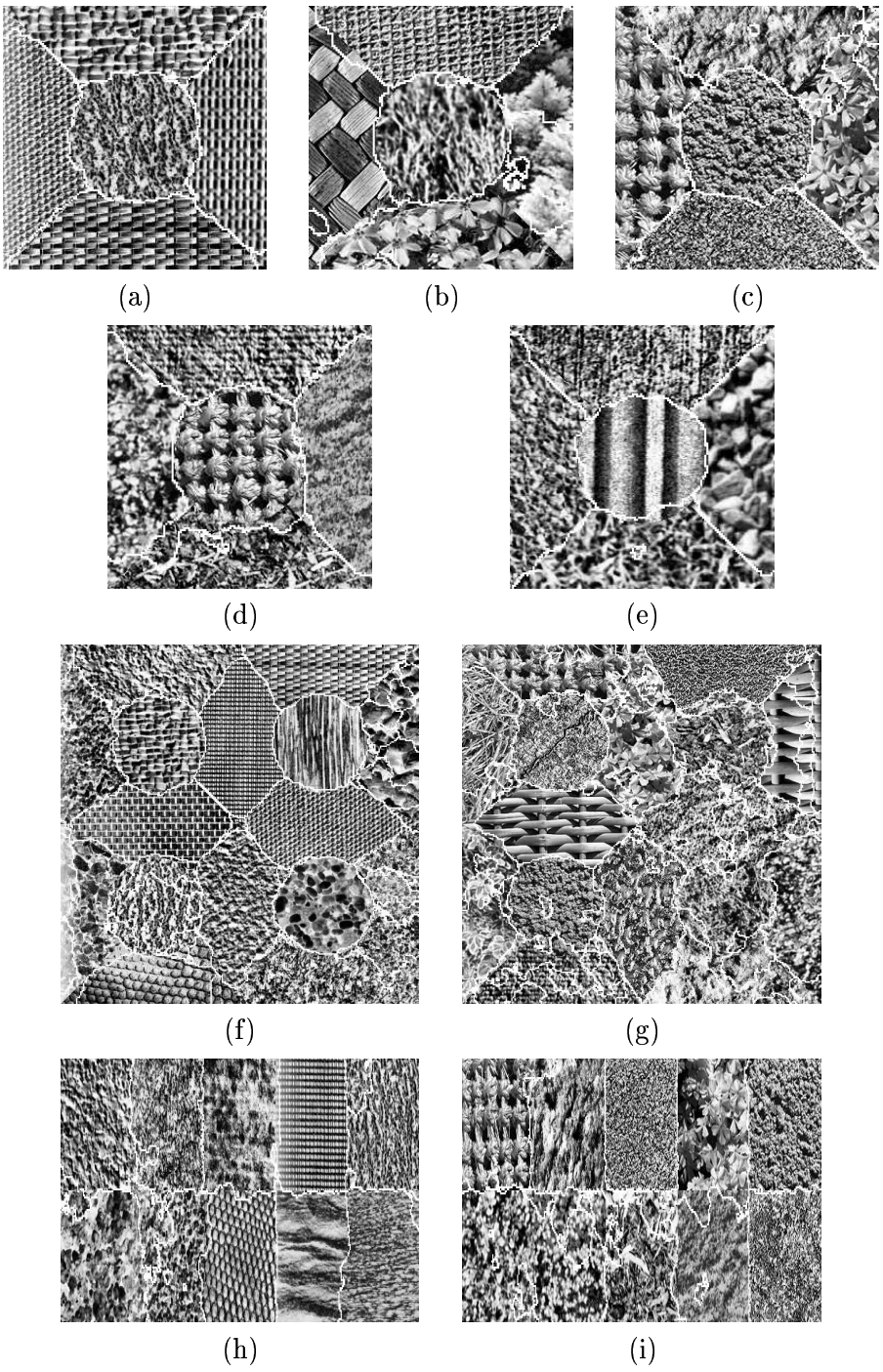



**Figure 6.1: Classification of the images in figure 3.1. Classes boundaries are super-imposed in the images.**

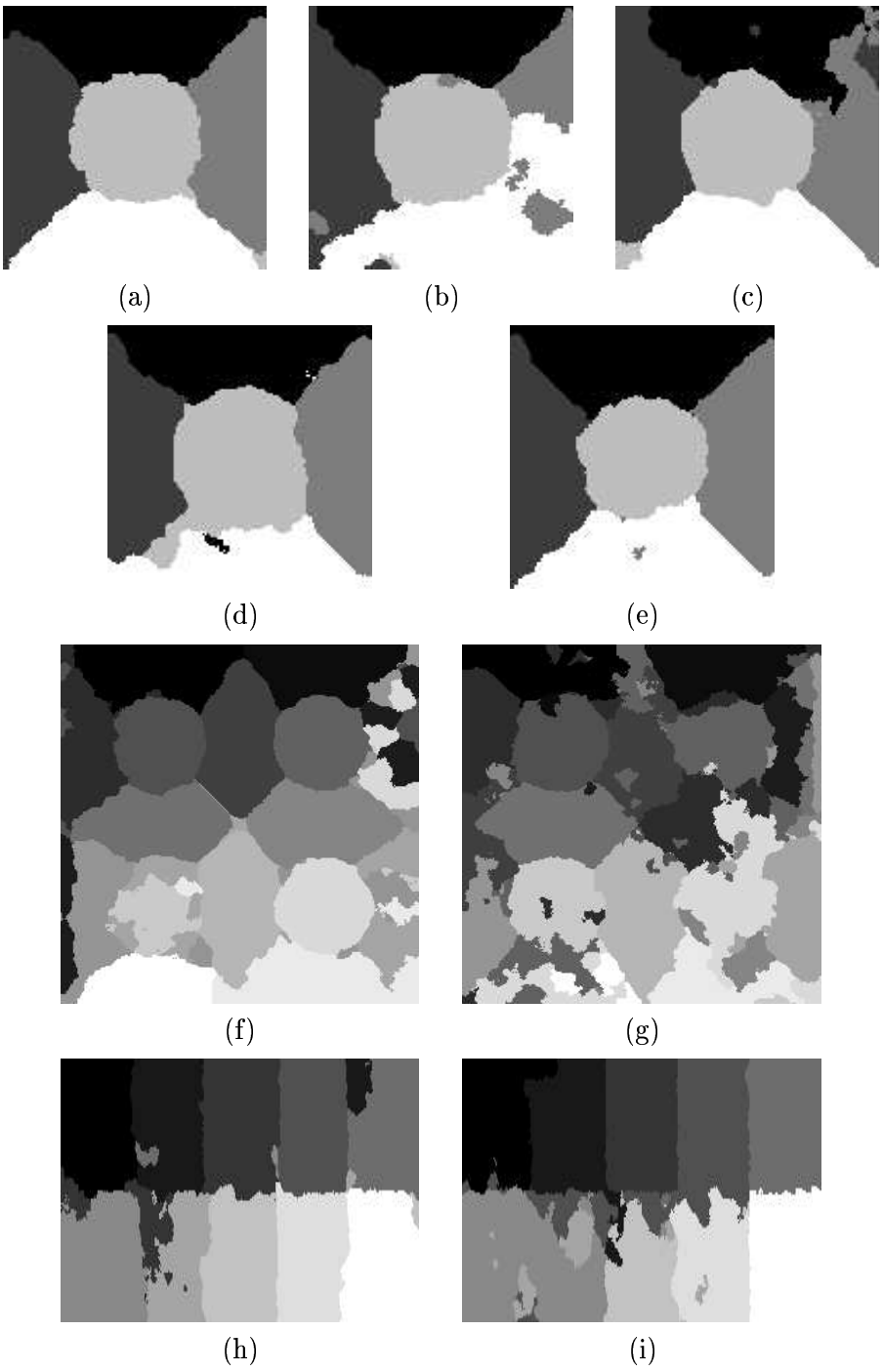



**Figure 6.2: Classification of the images in figure 3.1. Classes are presented as different levels of grey.**

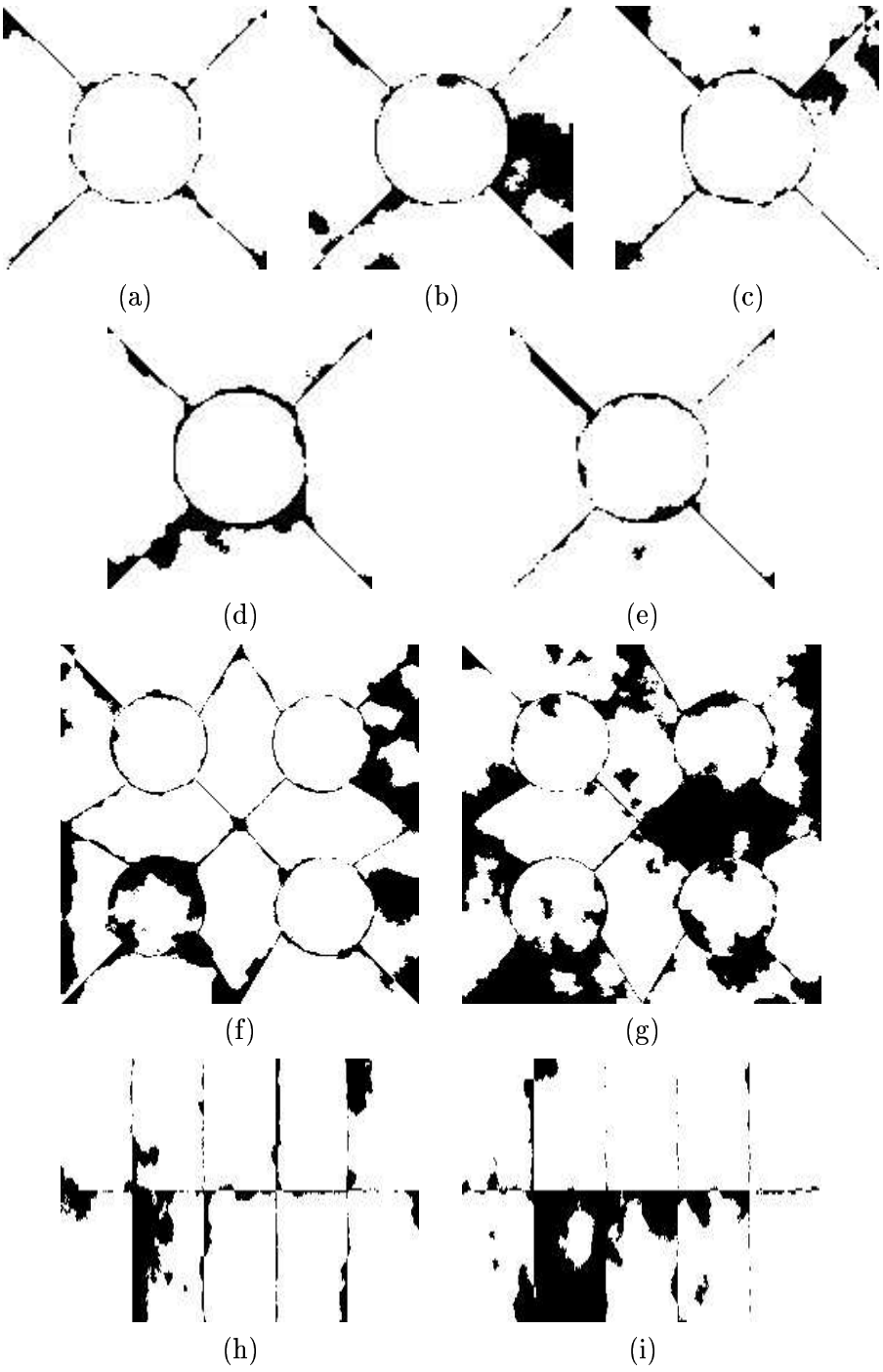
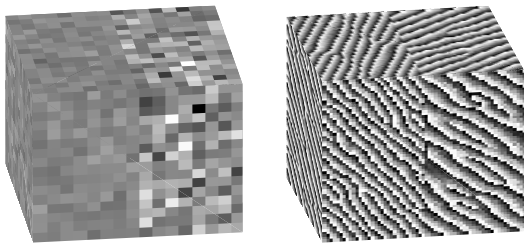



**Figure 6.3: Classification of the images in figure 3.1. Pixels that are correctly classified appear in white.**



(a) (b)

**Figure 6.4: Volumetric test data (a) One measurement of the two bivariate Gaussian distributions with similar means and variances (b) Two oriented patterns with different frequency and orientation.**

## 6.2 3D Artificial Textures

Unlike the 2D texture literature, that has many examples and test images available, up to the best of the author's knowledge, there is not a database available in 3D. We propose the following sets as a benchmark to compare different algorithms and measurement extraction techniques. First, a volumetric set that represents a simple measurement space of 2 dimensions each with Gaussian distributions with the following characteristics:

| Class | Dimensions | $S^1$ | $S^2$ |
|---|---|---|---|
| A | $32 \times 16 \times 32$ | $\mu_1 = 25, \sigma_1 = 2$ | $\mu_2 = 26, \sigma_2 = 4$ |
| B | $32 \times 16 \times 32$ | $\mu_1 = 27, \sigma_1 = 7$ | $\mu_2 = 28, \sigma_2 = 7$ |

The two classes together form a $32 \times 32 \times 32 \times 2$ space. Figure 6.4 (a) shows one measurement of the space and figure 6.5 shows a scatter plot of the data and the corresponding masks. The measurement space was classified unsupervised with the number of classes provided but not the estimates of the means. First it was classified in a single resolution (LBG) and then using M-VTS ($QT$ level $\mathcal{L} = 3$), both with and without PCEs. The classification results for the unsupervised case are presented in figure 6.6 as clouds of points for each class. Figures (a, b) show classes 1 and 2 for LBG without PCEs and (c, d) show classes 1 and 2 for M-VTS (with PCEs). Results are presented in table 6.3.

It is clear that for the single resolution many elements have been incorrectly classified and each class spans the whole volume. For M-VTS there are some incorrectly classified

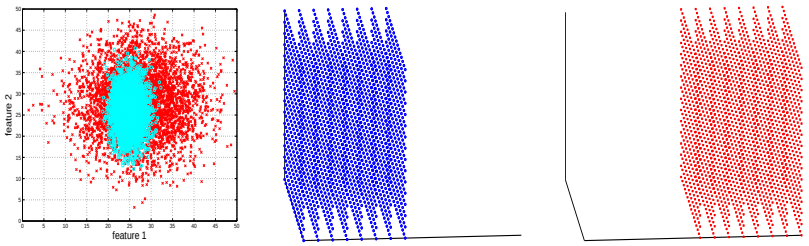


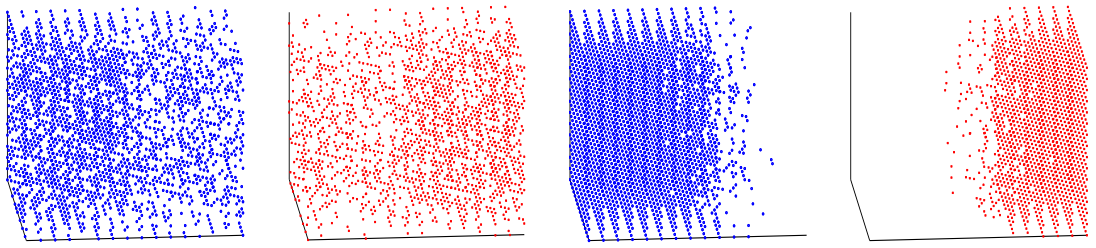

**Figure 6.5: Gaussian volumetric test data: (a) A scatter plot of the data of figure 6.4 (a). (b,c) Masks for each class.**

voxels close to the boundary, but the general shape of the original data is preserved. For this experiment the use of PCEs can improve the results in both classification algorithms, and the use of multilevel classification can improve the classification as well.

It is very important to notice that the original shape and size of the data are of huge importance for the multilevel procedure. At the third level of the $QT$, the sets have been reduced in dimensions to $[8 \times 8 \times 8]$ elements for the Gaussian data and $[16 \times 16 \times 16]$ for the oriented data. If the original shape of the classes were in a pyramidal shape, instead of a cubic or parallelepiped, then, while reducing over the $QT$, the classes will mix among them. By selecting the shape of the regions of the classes, one can control the classification results. This was not the case for the natural textures presented in section 6.1, nor will it be for the human knees MRI data of section 6.3.

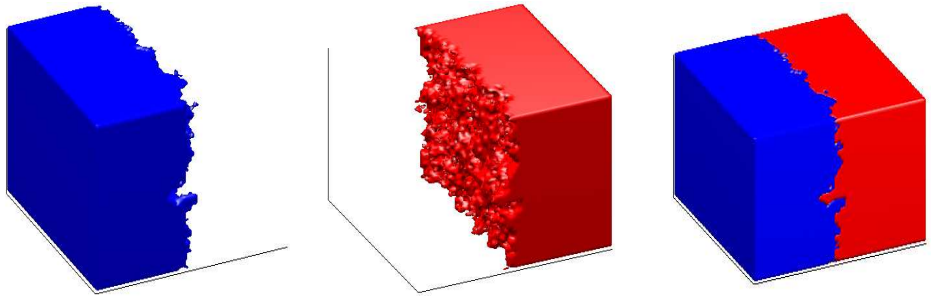
The second set is a $64 \times 64 \times 64$ volume with two oriented patterns of $64 \times 32 \times 64$ elements each, with different frequency and orientation. Figure 6.4 (b) shows this set. The measurement space was extracted and two measurements were manually selected: $S^1$ and $S^3$, and classification was again performed unsupervised with and without PCEs for single and multiresolution. Results are presented in table 6.3.

Figures 6.7 (a, b) present the two classes segmented with M-VTS. Again, some voxels near the boundary are misclassified, but the shape (c) is well preserved. The computational complexity was considerably increased in 3D, for the first set the respective times for LBG and M-VTS were 0.1s and 14.9s and for the second set 0.4s and 54.0s. No systematic attempt to make the code more efficient was made.



(a) (b) (c) (d)

**Figure 6.6: Classification results for Gaussian data: (a) LBG Class 1, (b) LBG Class 2, (c) M-VTS Class 1, (d) M-VTS Class 2.**



(a) (b) (c)

**Figure 6.7: Classification of the oriented data with M-VTS:(a) Class 1 (b) Class 2 and (c) Both classes.**

It should be noted that while multiresolution improves the results in both cases, the result of single resolution and PCEs outperforms M-VTS without PCEs. This may be due to the arbitrary selection of the features from the measurement space; perhaps other features could improve the results.

This example was presented to compare single and multiresolution in a volumetric space, the feature selection was performed manually.

**Table 6.3: Misclassification Results (%) for LBG and M-VTS for the two 3D test sets.**

| Algorithm | Gaussian Data | | Oriented Data | |
|---|---|---|---|---|
| | Without PCEs | With PCEs | Without PCEs | With PCEs |
| LBG | 37.0 | 14.1 | 22.3 | 4.6 |
| M-VTS 3 | 6.6 | 4.7 | 6.2 | 3.0 |

## 6.3 3D Magnetic Resonance Textures

A final set of experiments was conducted with several volumetric MRI sets of human knees acquired under different protocols: one set with *Spin Echo* and two sets with *SPGR*. In the three cases each slice had dimensions of $512 \times 512$ pixels. Some of the main characteristics of the sets are presented in table 6.4. One sample slice from the set is presented in figure 6.8 and the corresponding (2D) measurements extracted with the SOP sub-band filtering are presented in figure 6.9.

A few observations about the images should be made. The increased *Tr* time of the Case 1 results in a bright appearance of the bone and tissue (which are hard to distinguish by the grey level). Cases 2 and 3 have a shorter *Tr* and a fat-suppressed SPGR sequence where the bone and background (hard to distinguish) appear dark and the muscle and cartilage appear bright. These last images are useful when the cartilage of the joint is of importance. In the following subsections a heuristic technique to segment the cartilage based on the bone extracted by the texture segmentation will be presented.

**Table 6.4: Characteristics of the MRI knee sets.**

| Case | Rows | Cols | Slices | Slice thickness (mm) | Pix size (mm) | Tr (ms) | Te (ms) | Acquisition Protocol |
|---|---|---|---|---|---|---|---|---|
| 1 | 512 | 512 | 87 | 1.4 | 0.2539 0.2539 | 1650 | 22 | Spin Echo |
| 2 | 512 | 512 | 64 | 1.5 | 0.2344 0.2344 | 60 | 7 | 3D SPGR |
| 3 | 512 | 512 | 60 | 1.5 | 0.1953 0.1953 | 60 | 3.2 | 3D SPGR |

### 6.3.1 MRI Supervised classification

For the first data set, Case 1, the following classification approach was followed. Four training regions of size $32 \times 32 \times 32$ elements were manually selected for the classes of *background, muscle, bone* and *tissue*. These training regions are small relative to the size of the data set, and they remained as part of the data to classify. Each training

(a) (b) (c)

**Figure 6.8: One sample slice from the knee sets: (a) Case 1, (b) Case 2, (c) Case 3. Data sets b,c provided by Dr. Simon Warfield from Brigham and Women's Hospital.**

sample was filtered with the SOP sub-band filtering scheme, and the results were used to construct the Bhattacharyya space (figure 6.10(a)).

It can be immediately noticed that two bands: $S^{22,54}$, which correspond to the low pass bands, dominate the discrimination while the distance of the pair *bone-tissue* is practically zero compared with the rest of the space. If the marginals were calculated directly, the low pass would dominate and the discrimination of the bone and tissue classes, which are difficult to segment, (table 4.1 in chapter 4), would be easily merged. Figure 6.10 (b) zooms into the Bhattacharyya space of the bone-tissue pair. Here we can see that some features: $S^{12,5,8,38,\ldots,}$ provide discrimination between bone and tissue, and the low pass bands helps discriminate the rest of the classes.

The sequential forward classification was performed with the low pass feature, $S^{54}$, and the first 6 features from the ordered statistics $BS_I$ of the bone-tissue Bhattacharyya space which correspond to: $S^{12,5,8,39,9,51}$.

This selection of features reduced significantly the computational burden since only these features were filtered. The final misclassification obtained was 8.1% with 7 features. Figure 6.10 (c) presents the improvement in the classification as the features are introduced to the classifier.

The previous classification of the data was a full volumetric analysis, with a single resolution supervised classification. The next step was to compare these results with two other different techniques: a per-slice segmentation, and M-VTS classification (with

**Table 6.5: Misclassification results (%) for 2D and 3D single resolution and M-VTS for Case 1 of the MRI sets.**

| Technique | Misclassification (%) |
|---|---|
| 2D Single resolution | 8.6 |
| 3D Single resolution | 8.1 |
| 3D M-VTS | 6.0 |

PCEs, $\mathcal{L}$=3). The results obtained are presented in table 6.5. While the results from the 2D and 3D single resolution are close, the use of multiresolution improves the results by more than 2%. The classification with a multiresolution algorithm improves the results and produces a much smoother region classification. Some of the errors are due in part to magnetic inhomogeneity artefacts across the image which were not handled explicitly. It should be noted that the classification results, although not anatomically perfect, illustrate the utility of the use of texture features in MRI classification. Two slices in sagittal plane with their respective classifications are presented in figure 6.11.

Finally, a volume rendering of the segmented bone is presented in figure 6.12. Four anatomical structures present in the MRI data set are clearly identifiable: *patella, fibula, femur* and *tibia*.

### 6.3.2 MRI Unsupervised classification

Manual selection of training data in MRI data sets can be a slow and difficult process, and it has the disadvantage of training and testing on the same data, even if the training samples are small compared with the size of the data. The following section presents the results for unsupervised segmentation for the SPGR MRI data sets. The sets were classified and the bone was segmented with the objective of using this as an initial condition for extracting the cartilage of the knee.

The first set to be classified in unsupervised scheme was Case 2. No training data were used, but the classification is not fully unsupervised since the number of classes desired, four, was provided by the user and the sub-bands of the SOP were selected

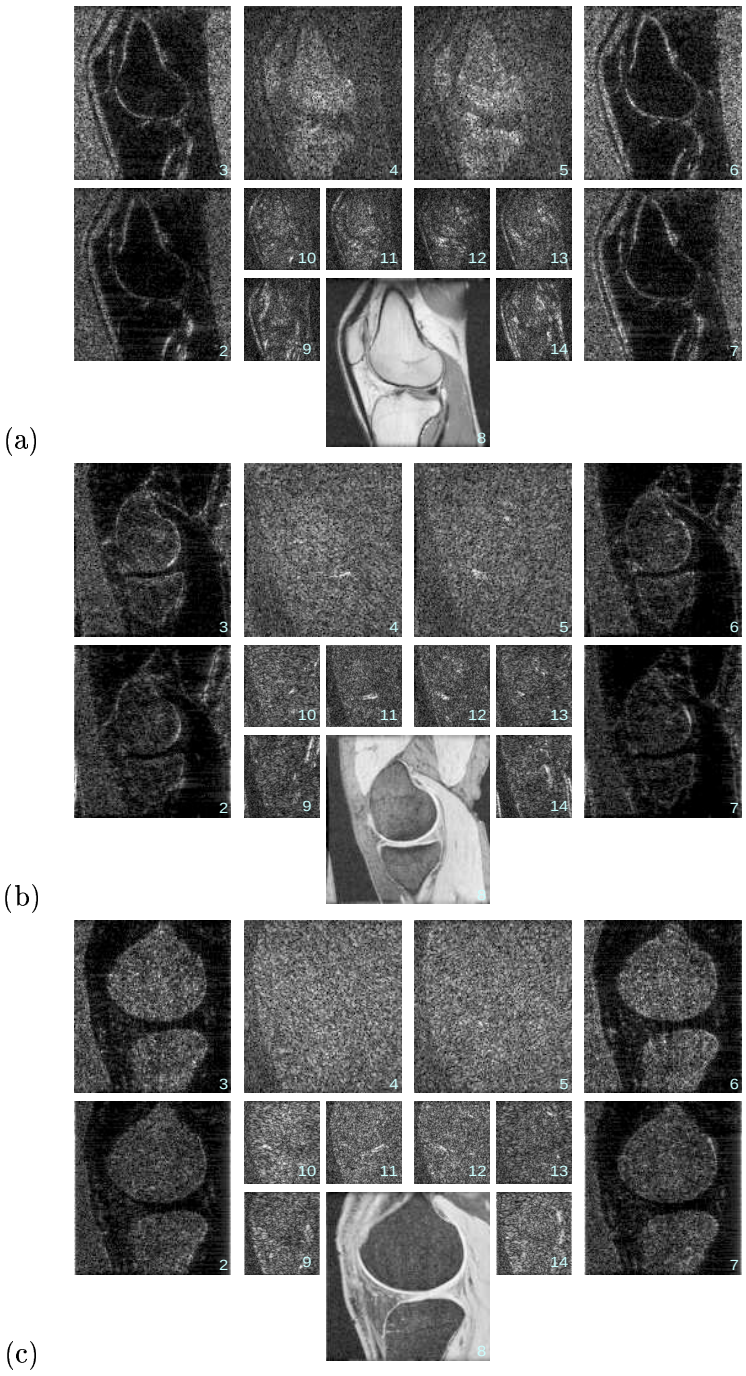
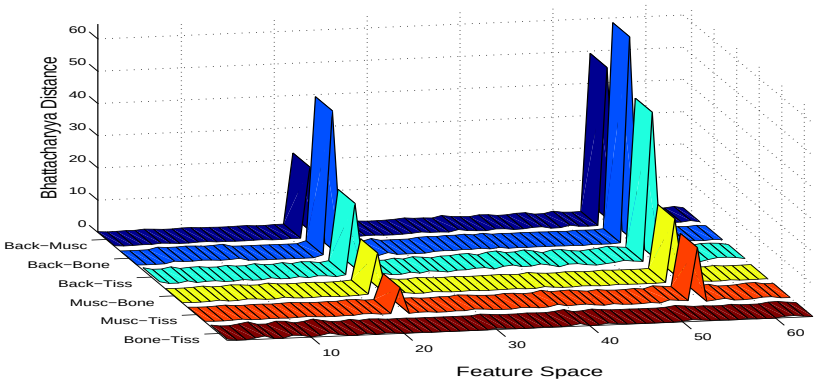
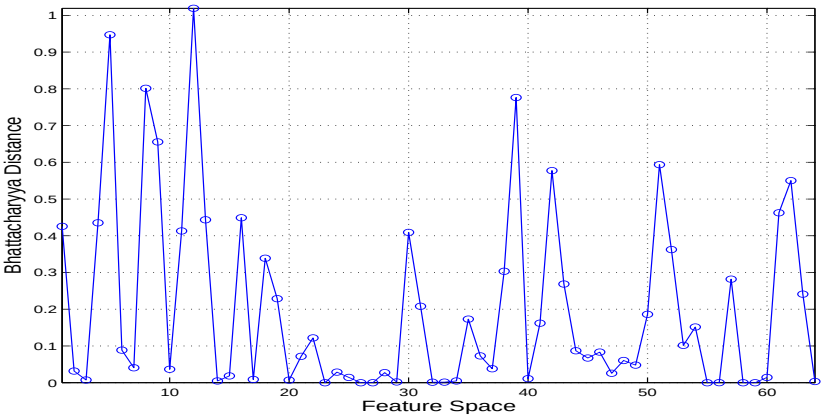


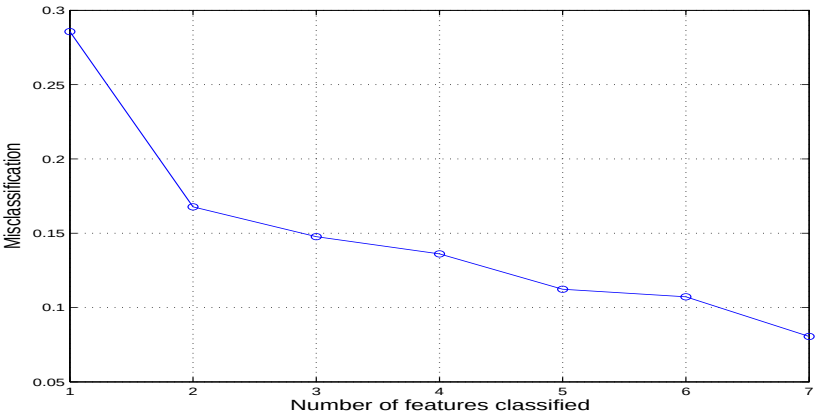

**Figure 6.9: SOP Measurements for the sample slices: (a) Case 1, (b) Case 2, (c) Case 3.**




(a)




(b)




(c)

**Figure 6.10: Human knee MRI (a) Bhattacharyya space $BS$ (3D, order 2), (b) Bhattacharyya space ($BS^i(bone, tissue)$) (c) Sequential misclassification results.**

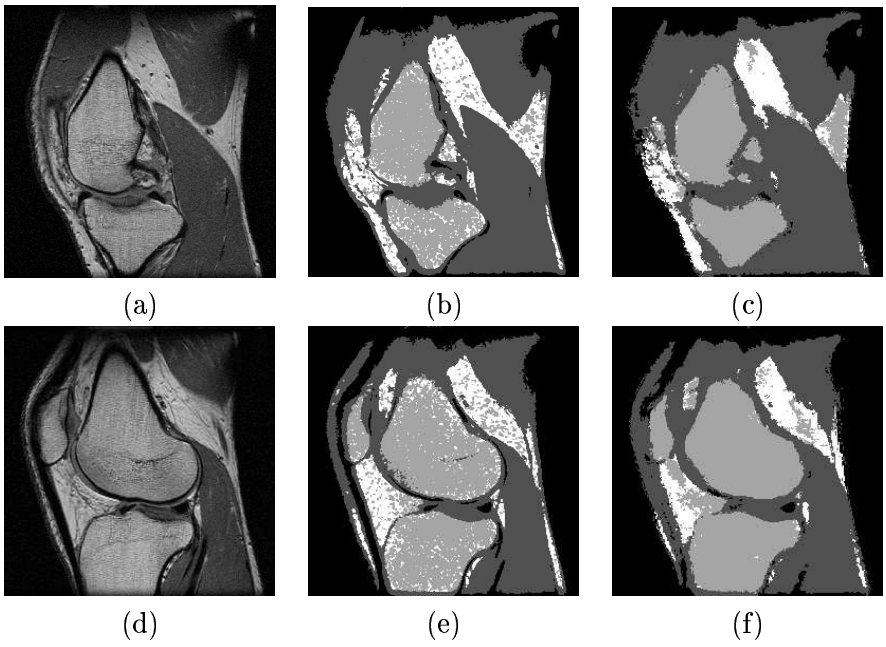



**Figure 6.11: Human knee MRI and its classification: (a,d) Sagittal slices 36 and 56 (b,e) 3D single resolution classification (c,f) M-VTS classification**

manually. The SOP measurement space of Case 2 is shown in figure 6.9 (b). Besides the low pass, the sub-bands that appear to give good discrimination of the classes are $S^{2,3,6,7}$ which correspond to the higher frequencies. For the volumetric classification, three analogous bands were selected, namely $S^{1,5,9}$ plus the low pass $S^{22}$. The corresponding positions are shown in figure 6.13. More features could have been selected but the complexity would have increased.

The performance of the classification schemes was measured on the ability to correctly classify the bone since this class alone will be used to segment the cartilage later on in this chapter. The correct classification was measured by how much bone was classified correctly inside the bone mask ($\tilde{b} \in b$), and how much bone was classified outside the bone mask ($\tilde{b} \in (b)^c$) and their complements ($(\tilde{b})^c \in b$ , $(\tilde{b})^c \in (b)^c$). The knee was classified with LBG and M-VTS at level 3. One slice of the classified results is presented in figure 6.14 and the classification results are presented in table 6.6. As expected, M-VTS presents smoother results and an increase in the correct classification of the bone $\tilde{b} \in b$, yet it fails to reduce the classification of bone outside the region for

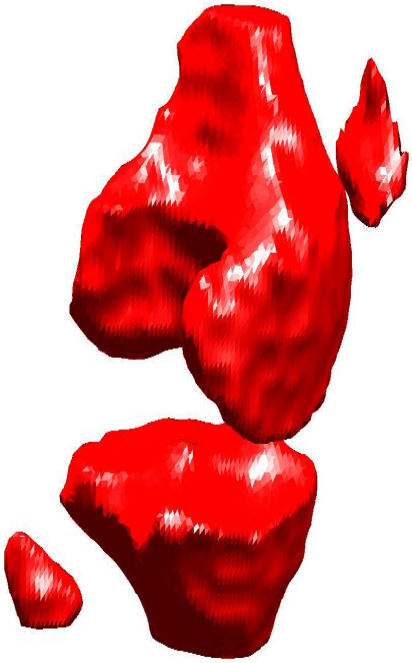


**Figure 6.12: Volume rendering of the segmented bone of Case 1 (misclassification 8.1%).**

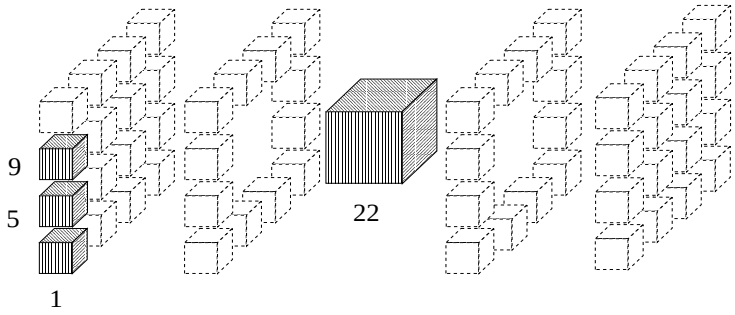



**Figure 6.13: A graphical representation of the features selected for the unsupervised classification**

bone $\tilde{b} \in (b)^c$.

Before proceeding to extract the cartilage from the knees, a heuristic approach that improves the results is proposed for the knee of Case 3. First, one slice of the set (slice 45) was selected and the 2D SOP sub-band measurement space was obtained (Figure 6.9 (c)). From this measurement space, two filtered measurements were selected for their discrimination $S^{3,8}$ which correspond to the low pass of the data and a high pass filter. It can be observed that measurements $S^{2,6,7}$ could be equally good for the discrimination

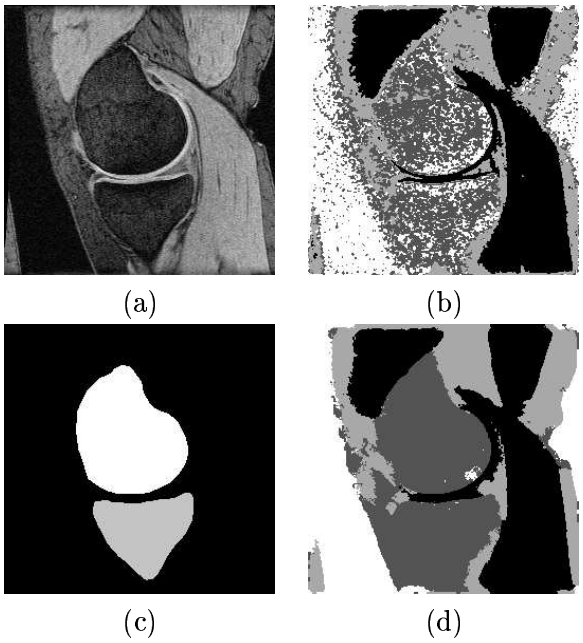



**Figure 6.14: Case 2: Unsupervised classification of human knee SPGR weighted MRI. (a) Sagittal slice 45, (b) LBG single resolution classification, (c) Mask for the bone, (d) M-VTS Classification.**

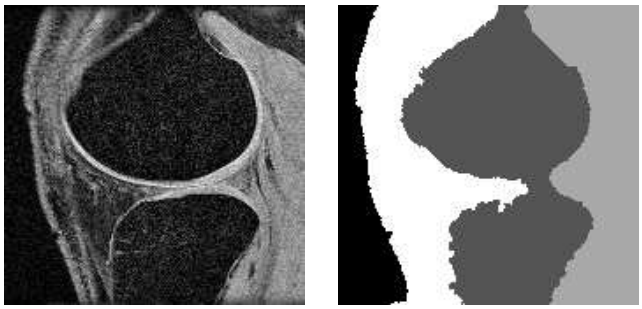


**Figure 6.15: (a) Sagittal slice 45 of Case 2. (b) Corresponding classification which will be used to obtain a set of means.**

as $S^3$, since they filter out the muscle and the tissue.

The feature space for this slice was classified with M-VTS at level 6, using PCEs and an agglomerative clustering algorithm at the highest level. This gave very good visual results (figure 6.15). The next step was to generate the 3D measurement space $S$. As for the previous example, the features $S^{1,5,9,22}$ were manually selected to match those that were selected from the 2D space. $S^{1,5,9}$ correspond to the high frequency spaces while $S^{22}$ corresponds to the low pass.

**Table 6.6: Classification (%) of Bone ($\tilde{b}$) according to the mask for bone ($b$) with k-means, and M-VTS. For Case 3, supervised and unsupervised classification was performed.**

| Knee Set | Algorithm | $\tilde{b} \in b$ | $\tilde{b} \in (b)^c$ | $(\tilde{b})^c \in b$ | $(\tilde{b})^c \in (b)^c$ |
|---|---|---|---|---|---|
| Case 2 | LBG | 67.2 | 21.0 | 32.8 | 79.0 |
| | M-VTS | 89.5 | 21.6 | 10.5 | 78.4 |
| Case 3 | LBG | 42.2 | 22.9 | 57.8 | 77.1 |
| | $\hat{a}_k$ | 64.0 | 11.0 | 36.0 | 89.0 |
| | M-VTS (UnSup) | 75.8 | 3.5 | 24.2 | 96.5 |
| | M-VTS (Sup) | 88.0 | 7.1 | 12.0 | 92.9 |

With the classes that were segmented from the 2D data, a mask was formed, then with one slice of the 3D features, a set of means estimates were obtained, and then these values were used to classify the data in a supervised scheme. The 3D data set was classified in single and multiresolution, both supervised and unsupervised. One slice of the classified results is presented in figure 6.16 and the classification results are presented in table 6.6.

In the two schemes, LBG and M-VTS, the use of the estimate of the means improved the classification, in both cases increasing $\tilde{b} \in b$. In single resolution $(\tilde{b})^c \in (b)^c$ also decreased, but not so in M-VTS, this could be due to an incorrect classification at the top level that is propagated downwards. This disadvantage can be seen in the region of the patella and infrapatellar pad: a section bigger than the actual bone is classified as bone (figure 6.16 (f)). Also upper part of the patella is misclassified: while the $\hat{a}_k$ still detects as bone some of the pixels in this region, the M-VTS method classifies the upper part as background.

Figure 6.17 presents two different views of a cloud of points of the bone class where the misclassification of the patella is visible. The upper part of the patella is classified as background and the lower part extends more than it should do into the infrapatellar pad. In figure 6.17(a), this can be better observed as a hole in the upper part and some scattered points in the lower part.

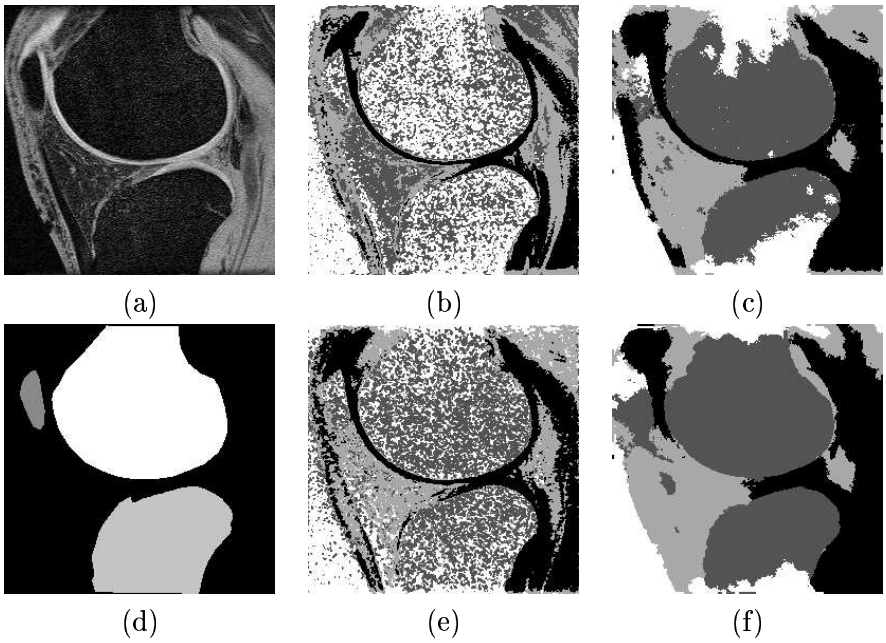
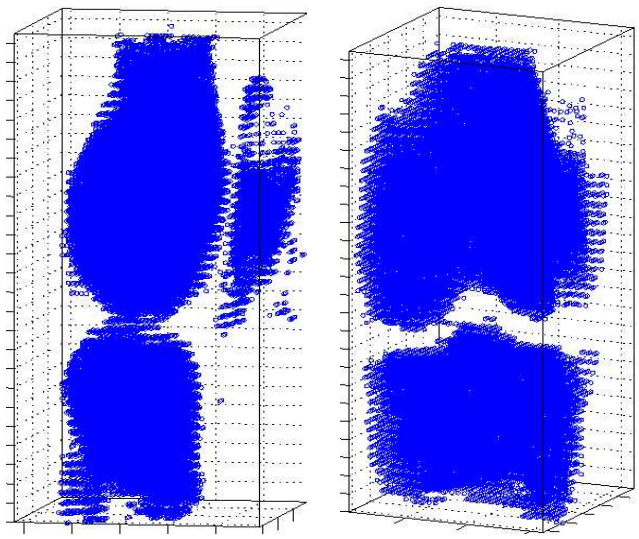



**Figure 6.16: Case 3: Human knee SPGR weighted MRI. (a) Sagittal slice 40, and its mask (d). Single resolution: (b) Unsupervised, (e) Supervised. M-VTS: (c) Unsupervised, (f) Supervised.**

### 6.3.3 Segmentation of the cartilage

Segmentation of articular cartilage of human knees has become a very important field [25, 39, 100, 138, 149, 152], and MRI has played an important role since it is a non-invasive method and generally cheaper than arthroscopy which has been the gold standard for diagnosing and monitoring cartilage damage and repair [148]. The interest in cartilage measurement is related to osteoarthritis which results in the thinning of the cartilage. The possibility of *in vivo* observation of the progression of osteoarthritis, and the evaluation of pharmacological treatments has developed further the interest in cartilage segmentation from MRI. Besides the thickness of the cartilage, there is also interest in analysing the surface topography of the cartilage, its contact areas, curvature characteristics and biomechanics [25]. For these reasons, MRI sequences have been developed to highlight articular cartilage [64, 120]. Fat-suppressed Gradient echo pulse sequences (SPGR) have been particularly good for cartilage segmentation [25, 149]. Once the sequences to highlight the cartilage have been developed, semi-automated methods for



(a) (b)

**Figure 6.17: Two different angles of the segmented bone $\tilde{b}$ (as clouds of points) from Case 3 MRI of the human knee**

the segmentation need to be developed to make the tool clinically valuable [149].

A common technique in the cartilage segmentation is the use of deformable models [138], yet these techniques normally require user intervention as a manual initialisation or seeding of the model [113] or a delineation of cartilage borders [25]. In Pakin's [113] method, besides the initial seeding, the boundaries are manually marked to prevent misclassification and also, prior to clustering, a detection of presence or absence of cartilage in a region is required. In Cohen's [25] method, a set of points is digitised manually along each articular contour curve; this process could be completed in two hours. Other methods require the use of 3D volumetric digital atlases [152] or templates that have been generated by hand segmented cartilages [26, 151].

In this section we propose a simple technique to extract the cartilage without the use of deformable models or seeding techniques. The user has to determine a Region of Interest and a grey level threshold and the bone extracted from the previous chapter is used as a starting point. In order to segment the cartilage out of the MRI sets, two heuristics were used: *cartilage appears bright in the MRIs*, and *cartilage resides in the*

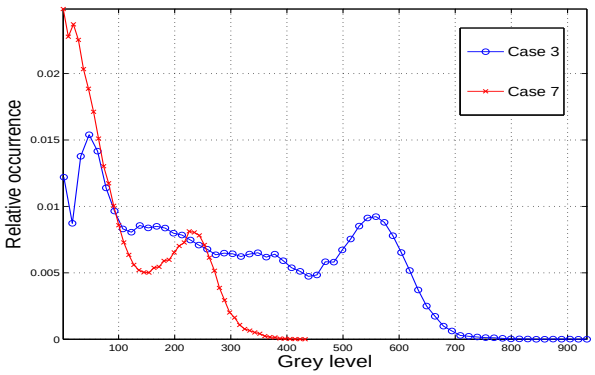
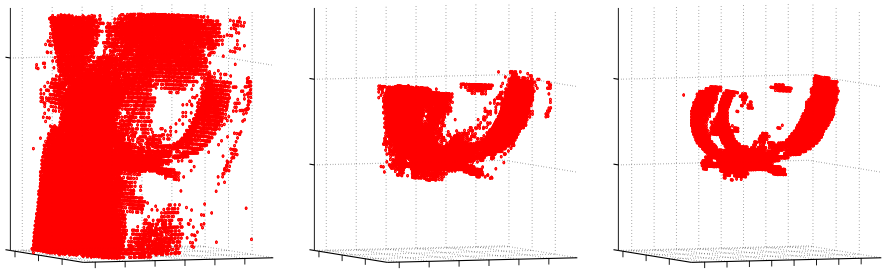



**Figure 6.18: Histograms of Case 2 and Case 3 MRIs.**

*region between bones*, which translated into two rules: threshold the sets above a certain grey level, and discard what is not close to the region of contact between bones. The methodology to extract the cartilage followed these steps:

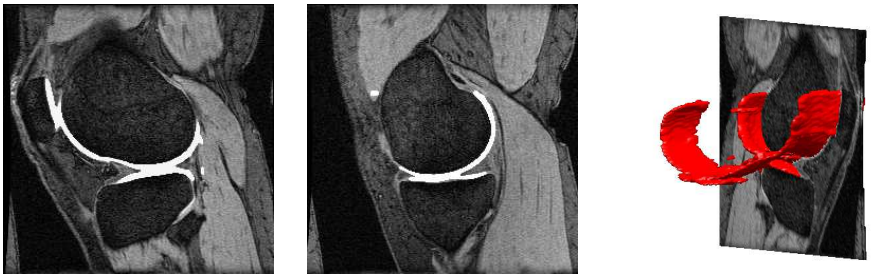
1. Extract the border of the bone segmented by the M-VTS.
2. Dilate this border by a number of pixels to each side (5 voxels in our case).
3. Use this region as a mask and eliminate the elements outside.
4. Threshold the region (grey level $g = 550$ for Case 2, and $g = 280$ for Case 3 figure 6.18).
5. Eliminate isolated elements.

After the previous methodology was used, another heuristic rule was employed since there were some voxels classified as cartilage outside the area of interest, mainly due to the cartilage between fibula and tibia that was not of interest at this moment. A *Region of interest* (ROI) was defined and then the elements that resided outside a cube were discarded. Figure 6.19 presents the segmentation of the cartilage for the knee of Case 2: (a) presents the whole thresholded MRI set, (b) presents the thresholded ROI, and (c) presents the segmented cartilage. Although the shape of the cartilage appears in (a), far too many voxels (that correspond to the muscle) are present. When the ROI is used (b), this improves but still many voxels obscure the shape of the cartilage. When the bone is used as a starting point, the characteristic shape of a saddle is revealed (c). Some elements still appear misclassified but the overall shape is now clear.



(a) (b) (c)

**Figure 6.19: Extraction of the cartilage of Case 2, results as clouds of points. (a) Data thresholded at $g = 550$ (b) The region of interest segmented at $g = 550$, (c) The cartilage extracted with the heuristic method.**



(a) (b) (c)

**Figure 6.20: Cartilage of Case 2. (a) Slice 15 of the set with the cartilage in white, (b) Slice 46 of the set with the cartilage in white. (c) Rendering of the cartilage segmented from Case 2 and one slice of the MRI Set.**

Figure 6.20 presents the results obtained imposed over the original MRI, first in two 2D images, and then a 3D rendering of the cartilage over one slice. These figures allow a visual interpretation of the results. Some false positives can be seen in the area of the muscle (right of the bone in 6.20 (a) and left in 6.20 (b)) but the general shape is visually close to that of the cartilage.

The same procedure was followed with the knee of Case 3 and the results are presented below. The results were visually better; this may be due to the fact that Case 3 was classified in a supervised scheme with the means extracted from one slice classification. Figure 6.21 presents the segmented cartilage for three slices of the set in different view: sagittal slice 18, axial slice 212 and coronal slice 130. Figure 6.21 (a) presents the segmented cartilage. Some false positives appear as small dots in the image. The tibial cartilage also appears a bit ragged but the general shape is correct, notice for instance

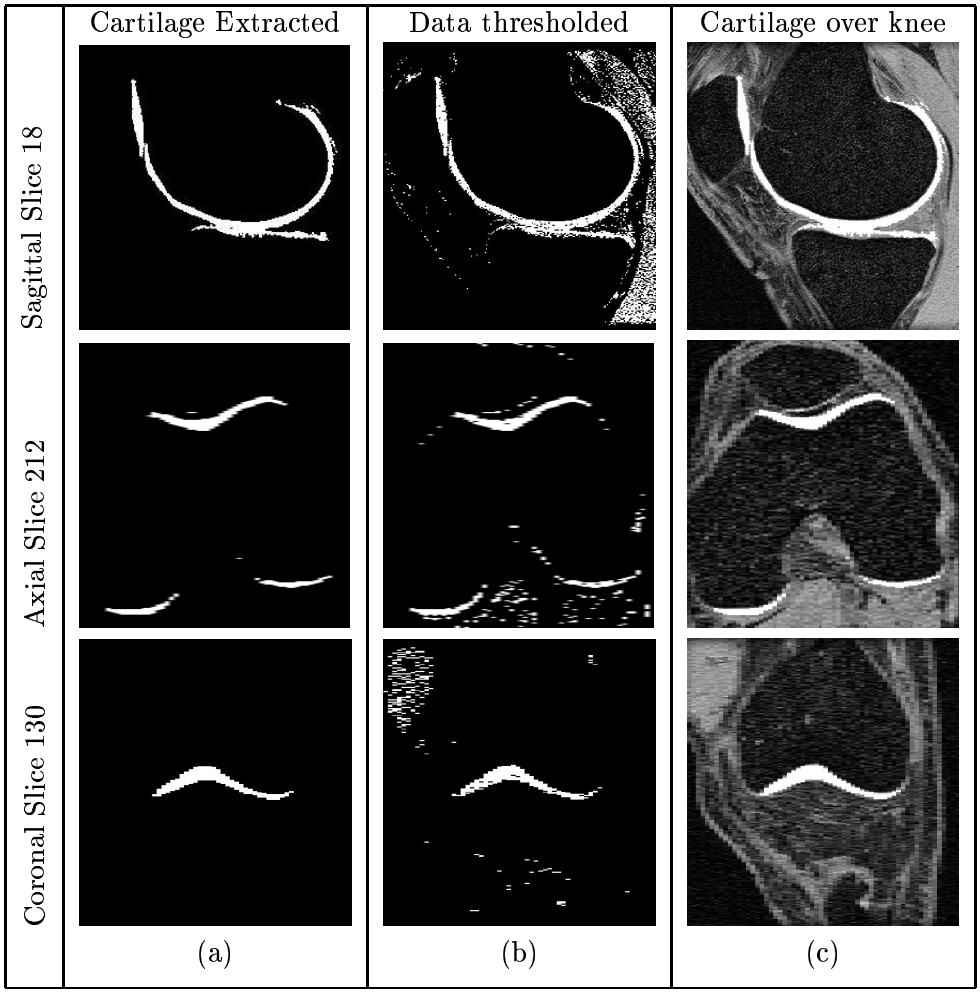



**Figure 6.21: Sagittal, coronal and axial view of the cartilage extracted from knee Case 3. The first column (a) shows the cartilage in the three planes, Second Column (b) shows the data thresholded at the same level used to extract the cartilage $g = 280$, the third column (c) shows the cartilage over the corresponding slice.**

the separation of the patellar cartilage from the femoral cartilage. As a comparison, figure 6.21 (b) presents the thresholded data of the same slices. Figure 6.21 (c) presents the cartilage over the original image.

Finally, 6.22 presents the rendering of the segmented cartilage in two different view angles and with a slice of the original MRI set. In this result, it is clear that the general shape of the cartilage; tibial, femoral and patellar is correctly segmented and the few incorrectly classified voxels could be easily erased from the set.

A simple methodology to extract articulate cartilage has been presented. The

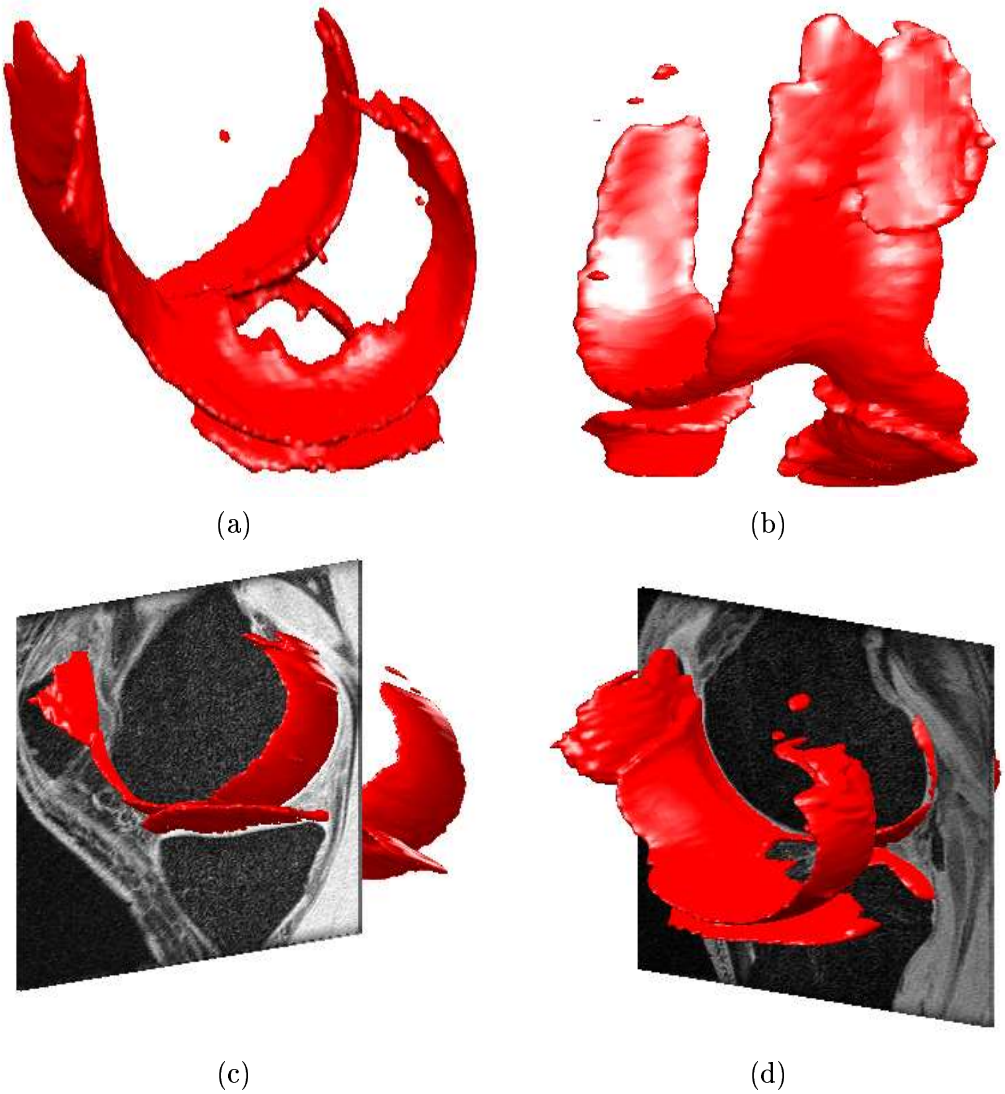


**Figure 6.22: Rendering of the cartilage segmented from Case 3. (a,b) The cartilage from two different view angles, (c,d) The Cartilage and one slice of the MRI Set.**

methodology requires the knowledge of the bone which has been extracted in the previous sections, and the intervention of the user in determining a thresholding level and a ROI. It should be noted that the ROI is basically a cube and not a sophisticated anatomical template. The threshold can be determined from the histogram with relative ease. What is important in this methodology is an adequate segmentation of the bone, since it is used as a starting point to the cartilage extraction. From the results obtained, it can be concluded that the classification based on the texture features and

the M-VTS algorithm is a good way of labelling the bone. This cartilage segmentation technique can be used to measure the thickness of the cartilage, the surface topography, its contact areas or the curvature characteristics.

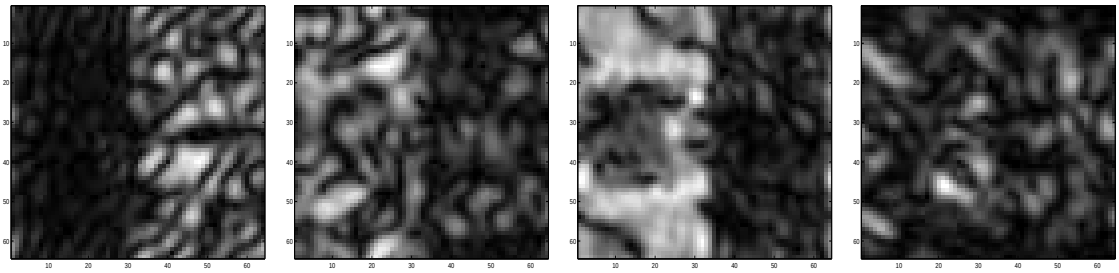
## 6.4 Summary

In section 6.1 we compared the use of SOP measurements and M-VTS against other texture extraction algorithms and classification techniques present in the literature, the proposed methodology outperforms them. The multiresolution nature of the algorithm is very powerful and gives the biggest improvement in the results, but it is the use of PCEs that gives the lowest average misclassification.

In two cases the use of PCEs does not improve the results. One possible explanation for this effect is an incorrect selection of the mean estimates when the new features are added to the space $S$.

In section 6.2, M-VTS algorithm was tested on 3D sets with artificial textures. The results were satisfactory as the objective was to test the method in simple cases before the MRI sets were classified. For the oriented data the features were selected manually through visual examination. In the future a proper method for feature selection in unsupervised cases should be used. The individual histograms of the $S^i$ could help but it may not be a conclusive test, as it will be explained. Figures 6.23 (a,b) show one slice of $S^{1,3}$; the two measurements that were used for the classification. If we now look at the histograms of all the first level of the $QT$ (figure 6.24), immediately one measurement strikes out as stretching more than others, this corresponds to $S^{18}$, one slice is presented in figure 6.23 (c). Perhaps it would be much better for the classification. Yet the histogram does not tell the full story. If we compare those for $S^3$ and $S^{19}$ (figure 6.24 (b)), their distributions look similar but the corresponding slice of $S^{19}$ (figure 6.23 (d)) shows a feature that would not be very good for classification purposes.

In section 6.3 the most important data sets of the thesis were classified: three human knee MRIs. The results that were obtained with the M-VTS allow the distinction of

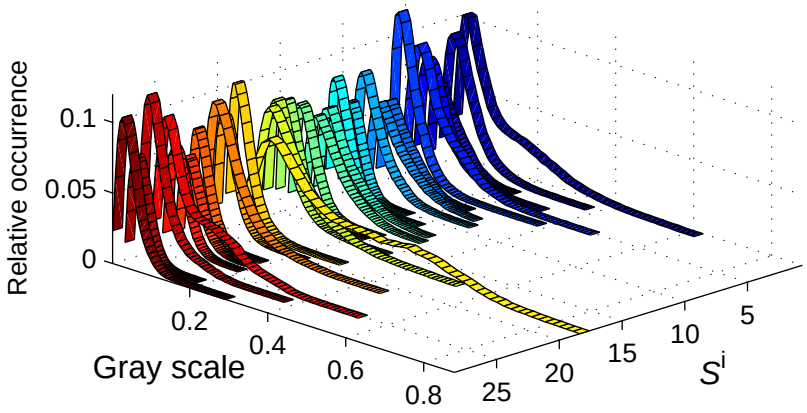


(a) (b) (c) (d)

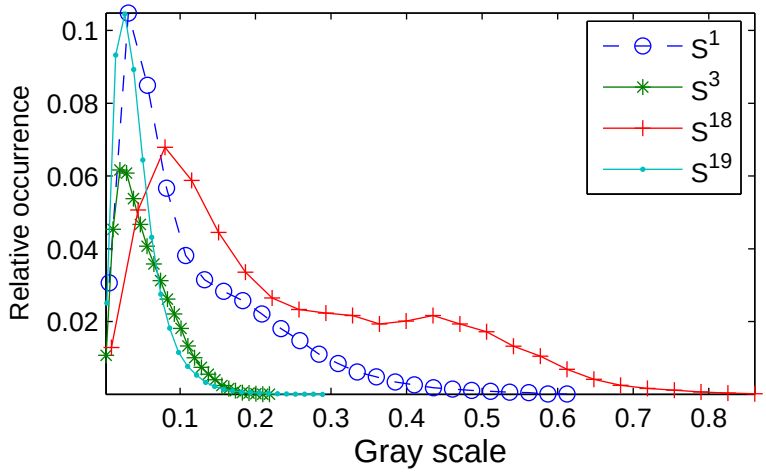
**Figure 6.23: One slice of the $S$ for the oriented texture data: (a) $S^1$, (b) $S^3$, (c) $S^{18}$, (d) $S^{19}$.**

classes that would be hard to distinguish through a grey level comparison; that is bone and tissue in Case 1 and bone and background in Cases 2 and 3. Therefore we can conclude that the M-VTS is capturing some textural properties of the regions and used them to discriminate between the different structures of the human knee. The results presented are not anatomically perfect and would require clinical validation to determine how well classified they are but this would imply a particular problem to solve. For the segmentation of the cartilage from two of the knees, the segmented bone was good enough to provide a starting region.

On a personal communication, Dr Simon Warfield expressed that the cartilages looked *visually satisfactory*. Still, the segmented cartilage would require validation if it were to be used clinically. The use of arthroscopy [33] or other techniques is beyond the scope of this work.




(a)




(b)

**Figure 6.24: (a) Histograms of the measurements of $S$, (b) Histograms of $S^{1,3,18,19}$.**

# Chapter 7

# Conclusions

## 7.1 Summary

A multiresolution algorithm based on Fourier domain filtering was presented for the classification of volumetric textures.

Textural measurements from the Fourier domain were extracted from 2D and 3D data through sub-band filtering with a Second Orientation Pyramid tessellation. Some of the measurements can be selected to form a new feature space; the selection is based on their discrimination powers obtained from a novel Bhattacharyya Space. A multiresolution procedure can improve the classification of these feature spaces: Quad Trees were formed with the features as the lowest level of each tree. Once the classification is performed in the highest level of the tree, the class and boundary conditions of the elements are propagated downwards. A border refining methodology with butterfly filters is performed. This refining technique outperforms an MRF approach. A further improvement is provided by the addition of dimensions to the space at the highest level of the tree.

The algorithm presented was tested with benchmark images in 2D and with 3D data; artificial textures and MRI sets of human knees. The results in 2D outperform several algorithms in the literature and the results in 3D are also satisfactory.

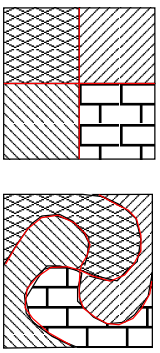
## 7.2 Major contributions of this work

- A review of the existing methodologies for texture analysis in 3D was presented, other 2D techniques were analysed and their extension to 3D considered. The formulation of the problem to solve may favour one technique or other, in our case the intention was to analyse a data set and segment regions that can represent uniform structures, either textures or anatomical features.

- With the previous problem in mind, the proposed extension to 3D of a sub-band filtering with Second Orientation Pyramid provided textural measurements that compared with others present in the literature.

- To solve the problem of feature selection, a novel Bhattacharyya Space was proposed as a technique to measure the discrimination power of pairs of classes. The calculation of the marginals of this space allows us to select the most discriminant sub-bands for a sequential classification, as well as the texture pairs that are easy/difficult to classify throughout the whole measurement space. This Bhattacharyya Space is independent of the filtering and could equally be used to select features from Wavelets, Gabor filters, Laws masks or any other texture measurements.

- The new positional contiguity enhancing features are a simple, yet useful technique to improve considerably the results of the multiresolution algorithm. These features exploit the fact textures extend spatially in 2D and 3D so they will increase the contiguity in the spatial domain at the expense of the compactness in the feature domain. Since the new contiguity features are added only at the highest level of the Quad Tree, the extra computational expense is low. These features are independent of the classification method used. It must be mentioned that the features work better when the regions to classify are compact. Images that have regular shapes, like the top one on the right, would



benefit more than the one on the bottom due to the nature of the features.

## 7.3 Major conclusions from this work

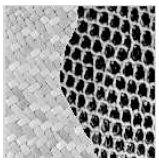
We can conclude that:

- Sub-band filtering in the Fourier domain with a Second Orientation Pyramid is a good way of extracting textural features from data. It is simple to calculate both in 2D or 3D and fast with the use of *FFT*. When compared with other features it was as good as other techniques such as co-occurrence or Gabor filters. Wavelets, Wavelet packets and Wavelet Frames can also provide good texture measures. For our experiments, sub-band filtering was adequate but for other experiments these techniques should also be considered.

- The use of local energy functions can enhance considerably the classification results, the size of the function plays a critical part though. It is important to distinguish the effect of the LEF from the features used, as a bad measurement could be obscured by the use of the LEF or a sophisticated algorithm.

- Multiresolution algorithms are very powerful and they can enhance the results of texture segmentation, but the classification at the highest level is critical since these classes will form the basis of classifications at lower levels. If these classes are incorrect, the final classification can perform worse than a single resolution method.

- Not all images are the same. Some cases in the literature apply algorithms for classification or texture extraction techniques to images that have not been histogram equalised, like the example on the right. It is therefore important to use a common set of test sets to be able to compare methods. For the 2D case Randen's database can be used, for 3D we publish the data sets used in this thesis, the corresponding addresses are:



`http://www.ux.his.no/~tranden/`

`http://www.dcs.warwick.ac.uk/~creyes/m-vts`

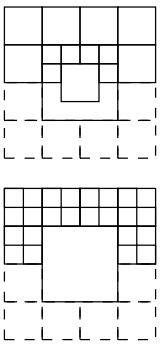
- Pyramidal butterfly filters can improve classification and have better results than MRF because they are anisotropic and steerable.

## 7.4 Suggestions for further research

Regarding the work presented in this thesis several remarks can be made:

- We have segmented anatomical structures from MRIs and measured the performance of M-VTS by comparing against hand-segmented masks. These masks, contrary to those used in the 2D experiments are subjective and would vary depending on the person that performed the segmentation. If M-VTS were to be used in medical applications, extensive clinical validation must be done. If MRI data were used, the effects of inhomogeneities have to be addressed.

- In this thesis we chose the Second Orientation Pyramid to tessellate the Fourier domain. It proved to be a good way of extracting textural measurements for the data that we analysed. Nevertheless other tessellations could be necessary for other textures. One particular extension could be to use the spirit of the Wavelet Packet and further decompose the high pass filters into four subsections as it is shown in the figure on the right.



- So far all the measurements of the Fourier domain were extracted using the magnitude of the inverse Fourier transform. The phase is an unexplored territory that could provide information in other cases. The main problem is the unwrapping in the presence of noise, which is not a trivial problem.

- We have relied in the number of classes or clusters present in the data for the unsupervised cases. If the number of classes to segment is not known, the analysis will change. A possible way of detecting the presence of clusters in the data can be

done with the techniques presented by Fatemi-Ghomi [40]: the distance histogram and the two-point correlation function.

- During the classification process, the representation of the classes does not have to be restricted to either the estimate of the mean or the trained SOM. It could be possible to use a combination of the two. With the use the Bhattacharyya Space, the most distinctive pairs of classes could be selected and then use SOM for those ones, and $\hat{a}_k$ for the other cases.

- PCEs have been used as a binary case, either used or not. They could be used in a continuous manner with a variable $0 \leq \alpha \leq 1$ that determines the contribution of the PCE to the system. In some cases this could present an improvement in the final classification.

- The cartilage segmented in chapter 6 could provide a starting point for more complex techniques. Instead of manually selecting a series of points, the segmented structure could be used.

Finally, in the area of texture analysis an important contribution would be a definition of texture that would be accepted by the vision community, clinicians, geologists, engineers, etc. This author, together with many other researchers would be happy to see a definitive description of texture.

# Appendix A

# Publications by the Author

- Reyes-Aldasoro, C. C. and A. Bhalerao, Volumetric Texture Description and Discriminant Feature Selection for MRI, In *Proceedings Information Processing in Medical Imaging*, Ambleside, Lake District, UK, 21-25 July, 2003, pp. 282-293.
- Bhalerao, A. and C. C. Reyes-Aldasoro, Selecting Discriminant Subbands for Texture Classification, *Proceedings of Eurocast'03*, February 2003, Canarias, Spain.
- Reyes-Aldasoro, C. C. and A. Bhalerao, Volumetric Texture Description and Discriminant Feature Selection for MRI, Volumetric Analysis of MR Images, *IPEM Meetings*, London, UK, 20 November, 2002, pp 8-9.
- Reyes-Aldasoro, C. C. and A. Bhalerao, Sub-band filtering for MR Segmentation, *Medical Image Understanding and Analysis*, Portsmouth, UK, 22-23 July, 2002, pp. 185-188.
- Reyes-Aldasoro, C. C., and A. Bhalerao, Sub-band filtering for MR Segmentation, *COST Meeting*, Bergen, Norway May 2002.

- Reyes-Aldasoro, C. C., A Guide to Co-occurrence Matrix Analysis, Research Report CS-RR-398, Computer Science Research Reports, University Warwick, UK, February 19, 2004.
- Reyes-Aldasoro, C. C., and Bhalerao, A., Classification of Human Knee Data from Magnetic Resonance Images, April 10, 2002, CS-RR-388, Computer Science Research Reports, University of Warwick, UK.

# Appendix B

# Quad Trees and Gaussian Pyramids

The use of multiresolution techniques in Image Processing can be achieved by the use of structures that relate a set of representations of the data at different resolutions. Two of the most common multiresolution techniques are the *Quad/Oct Tree* and the *Gaussian Pyramid*.

The concept of a tree is closely related with the field of data structures [47, 132] where a tree is a structure that contains nodes hierarchically arranged from a *root* node to the *leaf* nodes via some internal nodes (analogous to the branching structure of a real, upside down tree, since in most cases the trees are represented with the root above and the leaves down). Root and internal nodes can have one or more child nodes and can form sub-trees. An example of a tree with 4 levels and 4 children per node is shown in figure B.1.

The term *Quad Tree* ($QT$) [47] refers to a tree where each node is split along all $d$ dimensions, leading to $2^d$ children, since it was originally designed for images, that is, data in 2 dimensions, each node will have 4 children that will be arranged in a rectangle and the nodes will correspond to the NW, NE, SE, SW positions in analogy with cardinal orientation. The term *Oct Tree* refers to a similar structure of data in 3 dimensions.

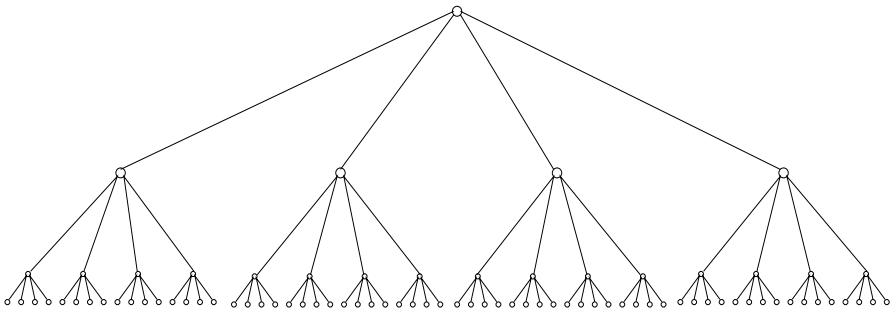


**Figure B.1: The structure of a Quad Tree with 4 levels.**

In our own particular case, the quad/oct trees will follow these properties:

- The dimensions of the data are $N_r = 2^a, N_c = 2^b, N_l = 2^c$,
- Each element of the data will constitute a leaf node,
- Every sub-tree will have equal depth, that is, reaching to the original data itself,
- The parent node will be obtained by the arithmetic mean of the values of the children. This operation implies a low pass filtering effect for every upper level,
- The root node will describe the global mean of the data
- The dimensions of an upper level will be half of the previous one.

A Gaussian Pyramid ($gp$) is similar to the tree, it will also be a structure of different levels, each corresponding to a low pass version of the data. The difference resides in the filtering performed by a convolution with a Gaussian filter kernel rather than a uniform kernel.

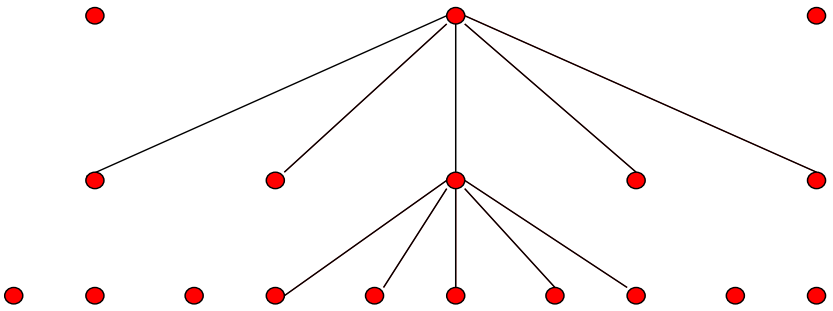


**Figure B.2: Parent-child structure of a Gaussian Pyramid.**

The following description of a $gp$ has been restricted to images for simplicity and follow [17]. Let the original image $\mathcal{I}$ have dimensions for rows and columns $N_r \times N_c$.

Let $L_c = \{1, 2, \ldots, N_c\}$ and $L_r = \{1, 2, \ldots, N_r\}$ be the horizontal and vertical spatial domains, and $G = \{1, 2, \ldots, N_g\}$ the set of grey tones.

Let the original image $\mathcal{I}$ be considered as the bottom level of the *gp*. The next level, $(gp)^2$ is obtained through a low pass filter, or reduction of *gp*. For every level ($\mathcal{L}$), the superior will be obtained then:

$$(gp)^{\mathcal{L}} = REDUCE\left((gp)^{\mathcal{L}-1}\right) \tag{B.1}$$

Each node $(i, j)$ of the higher level is a weighted average of the values of the lower level within a $5 \times 5$ window $w(m, n)$:

$$(gp)^{\mathcal{L}}(i, j) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m, n)(gp)^{\mathcal{L}-1}(2i + m, 2j + n) \tag{B.2}$$

The weights assigned to the nodes within the window are determined by a generating kernel defined by the following constraints:

$$\begin{aligned} w(m, n) &= \hat{w}(m)\hat{w}(n) \\ \sum_{m=-2}^{2} \hat{w}(m) &= 1 \\ \hat{w}(i) &= \hat{w}(-i) \end{aligned} \tag{B.3}$$

which are satisfied when:

$$\begin{aligned} \hat{w}(0) &= a \\ \hat{w}(-1) = \hat{w}(1) &= \frac{1}{4} \\ \hat{w}(-2) = \hat{w}(2) &= \frac{1}{4} - \frac{a}{2} \end{aligned} \tag{B.4}$$

When $a = 0.4$, the kernel has a Gaussian-like shape. The effect of the filtering will reduce by half the number of nodes in the image.

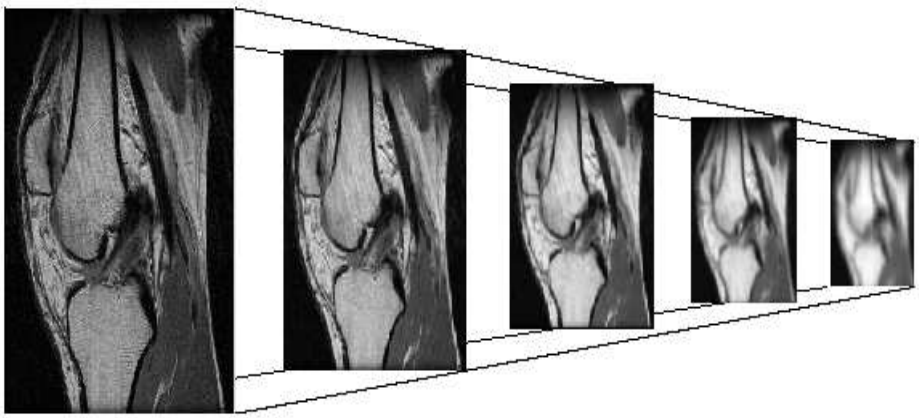


**Figure B.3: Gaussian Pyramid constructed from a MRI of a Human Knee. Images contain 512x512, 256x256, 128x128, 64x64 and 32x32 pixels.**

The reverse of the function $REDUCE$ is $EXPAND$ which interpolates new node values between given values. Let $(gp)^{\mathcal{L},n}$ be the result of expanding $n$ times $(gp)^{\mathcal{L}}$:

$$(gp)^{\mathcal{L},n} = EXPAND\left((gp)^{\mathcal{L},n-1}\right) \tag{B.5}$$

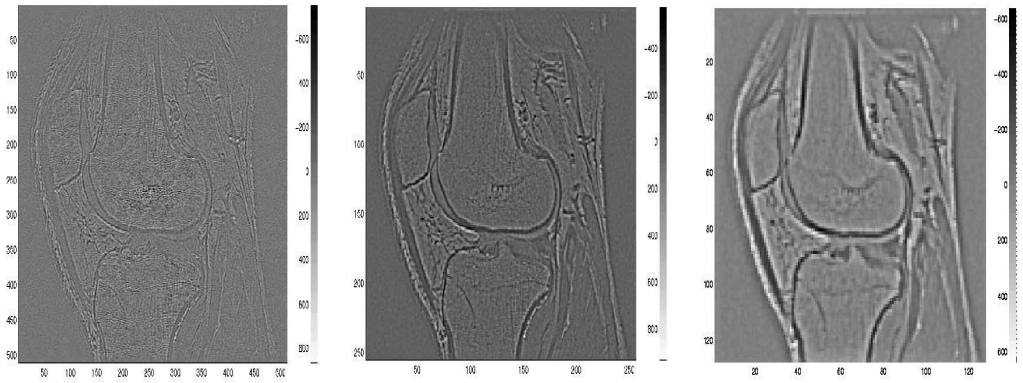
Each node $(i,j)$ of the higher level is a weighted average of the values of the lower level within a $5 \times 5$ window $w(m,n)$:

$$(gp)^{\mathcal{L},n}(i,j) = 4\sum_{m=-2}^{2}\sum_{n=-2}^{2} w(m,n)(gp)^{\mathcal{L},n-1}(\frac{i-m}{2},\frac{j-n}{2}) \tag{B.6}$$

Figure B.3 show a Gaussian pyramid constructed from a MRI of a human knee.

The Laplacian pyramid is derived from the Gaussian pyramid and provides an interesting description of the data with a filter effect that can be used to obtain edges through a zero-crossing detection. The Laplacian pyramid is a sequence of images $LP_0, LP_1, LP_2, \ldots, LP_N$ each being the difference between two levels of the Gaussian Pyramid:

$$LP_L = (gp)^{\mathcal{L}} - EXPAND\left((gp)^{\mathcal{L}+1}\right) = (gp)^{\mathcal{L}} - (gp)^{\mathcal{L}+1,1} \tag{B.7}$$



(a) (b) (c)

**Figure B.4: Three different levels of a Laplacian Pyramid constructed from a MRI slice of a human knee.**

As mentioned before, the Gaussian pyramid presents a low pass filtering effect over the images. Each successive stage of the pyramid is constructed by reducing a previous step. At each pixel, a weighted average of the neighbouring pixels is calculated. By sub-sampling, every successive stage of the pyramid is composed of less pixels. The inverse procedure of expansion generates an image with the same number of pixels as the previous stage of the Gaussian pyramid with a blurred effect. The difference of two images reduced and expanded up to a different stage of the pyramid corresponds to a certain level of the Laplacian pyramid, which has an overall impression of a band pass filter (see figure B.4). The Laplacian pyramid histogram has a Gaussian distribution centred in zero. Therefore, zerocrossings of the image can be obtained and represent edges of some features.

# Appendix C

# MRI and Imaging Technologies

Magnetic Resonance Imaging (MRI) is one of the several techniques used for imaging the Human Body. These techniques focus on obtaining images of different structures of the human body with good accuracy, without the direct surgical intervention. Historically, the first imaging technique is the X-rays, discovered more than a century ago by Röntgen. In X-rays or radiographies as they are commonly known, a film records the attenuation presented by different tissues of the body at the exposure to electromagnetic radiation of very high frequency. The resulting image presents shadows of the structures along the path of the rays.

Since the discovery of X-rays, many different techniques to obtain images from the human body have been developed relying on the use of electromagnetic radiation; microwaves, gamma rays or infrared, as well as sound waves, magnetic fields, sub-atomic particles, or electron beams. Table 1 presents some imaging technologies commonly used today [109].

Magnetic Resonance Imaging is a powerful method of non-invasive imaging of the interior or a living body for medical diagnostic and other purposes [60]. The underlying principle of MRI is that the nuclei of certain atoms have the property of spin and absorb radio waves of a strictly defined frequency when placed in a magnetic field. The atom of hydrogen presents the magnetic resonance effect, and is abundant in the human body

| Technique | Description |
|---|---|
| Film-screen X-ray | Standard X- ray technique. |
| Digital X-ray | Digital version of X-ray technique. |
| Ultrasound | Forms image by reflection of Megahertz frequency sound waves. |
| Magnetic Resonance Imaging | Forms images using radio emissions from nuclear spins. |
| Scintillation | Senses gamma-ray emission of radio active pharmaceuticals. |
| Thermography | Seeks from infrared signature. |
| Electrical impedance imaging | Maps impedance with low-voltage signal. |
| Optical imaging | Measures scattered near-infrared light. |
| Electrical potential measurements | Measures potentials at array of detectors on skin. |
| Positron Emission Tomography | Forms images using emission from annihilation of positrons from radioactive pharmaceuticals. |
| Novel ultrasound | Compound imaging, 3-D and Doppler imaging. |
| Elastography | Uses sound or MRI to infer mechanical properties of tissue. |
| Magnetic Resonance Spectroscopy | Analyses tissue's chemical makeup using radio emissions from nuclear spins. |
| Thermoaccoustic Computed Tomography | Generates short sound pulses using RF pulses and constructs 3-D images from them. |
| Microwave imaging | Uses scattered microwaves. |
| Hall-effect imaging | Picks up vibrations of charged particles exposed to a magnetic field. |
| Magneto-mammography | Senses magnetic contrast agents collected in tumours. |

Source: IEEE Spectrum [109]

**Table C.1: Imaging Technologies commonly used for breast cancer screening and diagnosis.**

in molecules of water. Most MRI is focused on hydrogen, which constitutes around 63% of the human body [43]. However, other atoms not so abundant in the body, such as phosphorus or sodium can be used to form images.

When atoms are placed in a magnetic field $\bar{B}$, the nuclei align with or against the field (figure C.1 (a,b)). Then, a second field, periodic at a certain precession frequency of the atoms, forces them to oscillate around the fixed magnetic field (c). The nuclei have absorbed energy, and oscillate at a frequency proportional to the intensity of the

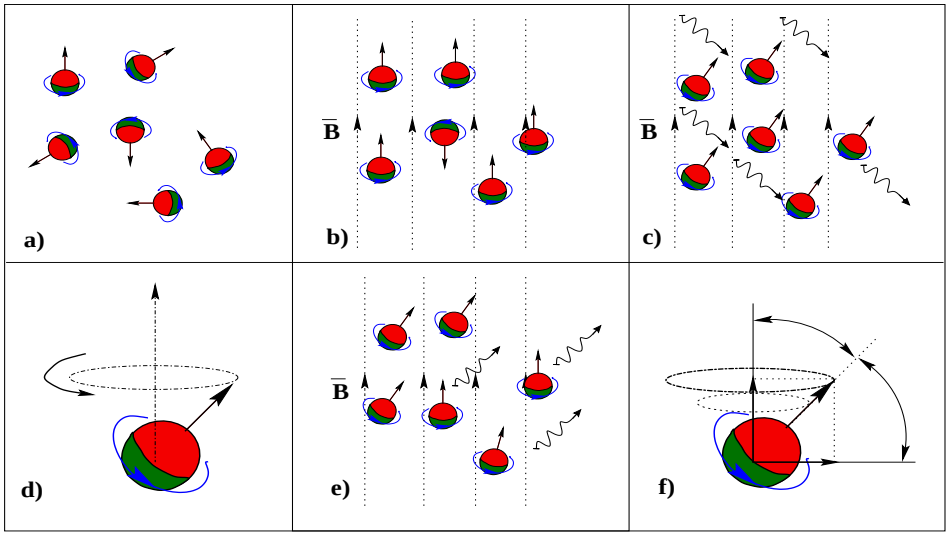



**Figure C.1: Nuclear Magnetic Resonance. (a) Nuclei in natural equilibrium (b) Nuclei aligned with external magnetic field (c) Oscillation of nuclei due to periodic magnetic field (d) Resonance around direction of field (e) Relaxation process and release of energy (f) Parallel and perpendicular components of the magnetic field. Adapted from [129]**

magnetic field (d). This energy is then released (e) over a certain period of time in the process of relaxation towards the original state of the atom. The time of relaxation is specific for the nature of the nuclei. During the return to their state of equilibrium, the nuclei release the energy in form of waves. RF coils are placed around the body that is being imaged, and register both the energy and the time of release and from these data, the image is reconstructed [60]. The relaxation of the atoms to their initial state is governed by two physical nuclear processes related to the components of the orientation, one parallel to the magnetic field that takes a time T1, and one perpendicular to the field in time T2 (f). These two parameters related with time, in addition to the number of protons that have aligned and are releasing energy contained in a tissue, determine the intensity of the magnetic resonance image intensity. The contrast of the image will depend on the relation of these three parameters. If the number of the protons is the only parameter considered, a Proton density (PD) image is obtained.

The resulting image for the MRI process looks like an internal slice or cut of the body. It is similar to the output of a Computed Tomography (CT) image, but there

| **Advantages** |
|---|
| Non-ionising radiation. No known physiological side effects. |
| High soft-tissue contrast. Differences between normal and abnormal tissues. |
| Any arbitrary slice of organ can be imaged. |
| Visualisation of areas deep within bony structures; vertebral canal, skull and cerebrospinal fluid. |
| Natural sources of contrast. |
| Good for angiography, imaging blood flow without catheters or contrast agents. |
| 3D imaging through computer manipulation of successive slices. |
| **Disadvantages** |
| High cost of equipment. |
| Claustrophobia of patients. |
| Long imaging time. |
| Strong magnetic field; unsuitable for patients with metal implants especially pace makers. |
| Images distorted by surgical clamps, wire or surgical stitches. |
| Unable to image calcium; tissue calcification can not be detected. |
| Acoustic noise; high levels of noise during scanning. |

Source: Hennel [60]

**Table C.2: Magnetic Resonance; advantages and disadvantages**

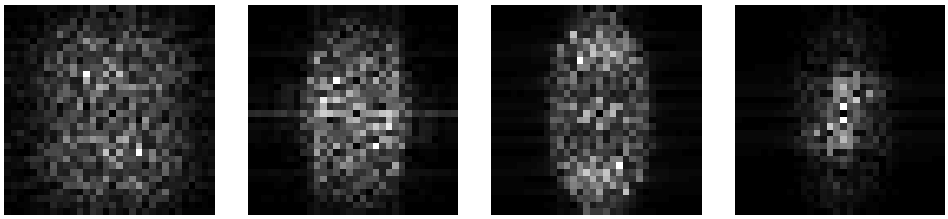
are substantial differences between them. CT uses X-rays, which are absorbed by the bones, therefore, some regions surrounded by bone are better imaged by MRI than CT. Feeney [41] considers that MRI is the best imaging technique for detecting tumours within the brain stem since bone does not obstruct the imaging. Table 2 presents some advantages and disadvantages of MRI. MRI is a vast field; Webb [153] dedicates one chapter for MRI, Hennel [60] is a good introduction and Hornak [66] is a very complete on-line reference. La Recherche [129] has an introductory explanation of magnetism. The Whole Brain Atlas [73] is an on-line atlas with T1, T2 and Proton Density (PD) images of a human brain, Medcyclopedia [117] is a good general reference as well.

Through this thesis, it is assumed that texture is present in magnetic resonance images and in particular in their energy distribution in the Fourier domain. This assumption implies that is different regions could be segmented by their Fourier distribution. Figure C.2 presents the Fourier transform of the four selected regions of figure 2.10 that correspond to background, muscle, bone and tissue. It can be seen that the Fourier

| **Spin Echo** |
| --- |
| Standard pulse sequence used in MR imaging. It uses 90 radiofrequency pulses to excite the magnetisation and 180 pulses to refocus the spins to generate signal echoes. It exists in many forms which consist in the preparation phase - of a 90 radiofrequency pulse which flips the longitudinal magnetisation Mz into the xy-plane, whereby the transverse magnetisation Mxy starts to precess with the Larmor frequency. This preparation phase is then followed in the acquisition phase by a train of refocusing 180 pulses which serve to generate repetitive signal echoes (hence the name spin-echo). The 180 pulses occur at times: TE + iTE, i = 0,1,...,n, and the signal echoes at iTE. |
| **SPGR** |
| short SPoiled GRASS; GE brand name for a spoiled gradient echo pulse sequence. The Gradient-echo pulse sequence, is one of the most frequently used pulse sequences in current day MR imaging often abbreviated GRE sequence. The preparation module of the pulse sequence consists of an excitation pulse which is termed the alpha pulse. It tilts the magnetisation by a flip angle a which is typically between 0 and 90. In the special case where a = 90 the sequence is identical to the so-called partial saturation or saturation recovery pulse sequence. The flip angle can also be slowly increased during data acquisition. Then, the data are not acquired in a steady state, where z-magnetisation recovery and destruction by ad-pulses are balanced, but rather such that the z-magnetisation is 'used up' during imaging by tilting a little more of the remaining z-magnetisation into the xy-plane for each acquired imaging line. The readout- or acquisition module occurs during a free induction decay FID , during which the read gradient is turned on such that localisation of the signal in the readout direction is possible. To accomplish this, the data are sampled during a gradient-echo , which is achieved by properly dephasing the spins before they are rephased by an equal but opposite gradient to generate the echo when the areas under the negative and positive gradients are equal. |

Source: Medcyclopaedia [117]

**Table C.3: Two Magnetic Resonance Protocols**



(a) (b) (c) (d)

**Figure C.2: Fourier transform of the four selected regions from the human knee MRI of figure 2.10: (a) background, (b) muscle, (c) bone and (d) tissue.**

domains are quite different. Also, the different measurements $S^i$ that are obtained by sub-band filtering in the Fourier domain (figure 2.25) show that different structures are highlighted by different filters.

# Appendix D

# The Human Knee Joint

The knee joint (figure D.1) is the most complicated joint in the human body [30], and it is inherently unstable. This condition leads to several injuries and problems like torn meniscus, tendon rupture, and patellar or cruciate ligaments problems, among many others. The knee articulation joins the femur, which has large medial and lateral condyles, with the tibia that has small and shallow condyles. The condyles of the tibia are deepened by crescent shaped rims of fibrocartilage, also called menisci. The knee joint is surrounded by a fibrous capsule that encloses articulations between tibia, femur, fibula and the patella, a thin bone in front of the knee. The joint is strengthened by several ligaments. At the sides, fibular and tibial collateral ligaments prevent side-to-side movements and add strength while the leg is straight. In the front, the strength is given by the tendon of quadriceps femoris and the patella. In the back, oblique popliteal ligament prevent over extension. In the inner part, the cruciate ligaments, anterior and posterior, limit the rotation and the forward and backward motion of the tibia.

The knee joint has two natural movements of *roll* back and forth or the ordinary flexion - extension. This movement, in which the femoral condyles roll over the menisci starts with the muscles of thigh and calf touching at approximately 40°, up to 165°, almost full extent of the leg. From 165° to 180° with the foot firmly upon the ground, the rolling ceases and the femur rotates medially and *glides* backward in a locking mech-

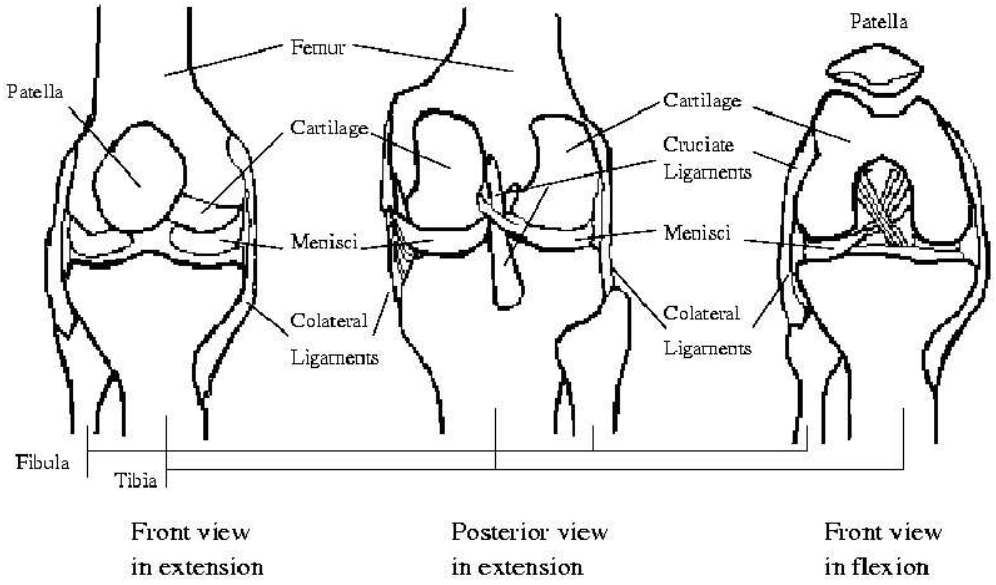



**Figure D.1: A simplified view of a knee joint: anterior and posterior view of leg in extension and anterior view of the leg in flexion.**

anism when the flexion - extension of the leg is accomplished.

# Bibliography

[1] B. Acha, C. Serrano, J. I. Acha, and L. M. Roa. CAD Tool for Burn Diagnosis. In C. Taylor and A. Noble, editors, *Proceedings of Information Processing in Medical Imaging*, pages 282–293, Ambleside, UK, July 2003. (cited on page 24)

[2] L. Alparonte, F. Argenti, and G. Benelli. Fast Calculation of Co-occurrence Matrix Parameters for Image Segmentation. *Electronics Letters*, 26(1):23–24, 4 January 1990. (cited on pages 39, 44)

[3] B. K. Bay. Texture correlation. A method for the measurement of detailed strain distributions within trabecular bone. *Journal of Orthopaedic Research*, 13(2):258–267, 1995. (cited on page 9)

[4] B. K. Bay, T. S. Smith, D. P. Fyhrie, R. B. Martin, D. A. Reimann, and M. Saad. Three-dimensional texture correlation measurement of strain in trabecular bone. In *Orthopaedic Research Society, Transactions of the 44th Annual Meeting*, page 109, New Orleans, Louisiana, 16-19 March 1998. (cited on page 9)

[5] A. Bernasconi, S. B. Antel, D. Collins, N. Bernasconi, A. Olivier, F. Dubeau, G. B. Pike, F. Andermann, and D. L. Arnold. Texture analysis and morphological processing of MRI assist detection of focal cortical dysplasia in extra-temporal partial epilepsy. *Annals of Neurology*, 49(6):770–775, 2001. (cited on page 28)

[6] J. C. Bezdek and S. K. Pal. *Fuzzy Models for Pattern Recognition.* IEEE Press, Piscataway NJ, 1992. (cited on page 15)

[7] A. Bhalerao and N. Rajpoot. Selecting Discriminant Subbands for Texture Classification. In *British Machine Vision Conference (BMVC)*, Norwich, UK, September 2003. (cited on page 87)

[8] J. Bigun. Multidimensional Orientation Estimation with Applications to Texture Analysis and Optical Flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):775–790, 1991. (cited on pages 1, 105)

[9] J. Bigun. Speed Frequency and Orientation Tuned 3D Gabor Filter Banks and Their Design. In *International Conference on Pattern Recognition, ICPR94*, pages 184–187, Jerusalem, Israel, 9-13 October 1994. (cited on page 49)

[10] J. Bigun and J. M. H. du Buf. N-Folded Symmetries by Complex Moments in Gabor Space and Their Application to Unsupervised Texture Segmentation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 16(1):80–87, 1994. (cited on page 49)

[11] C. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, Oxford, UK, 1995. (cited on page 15)

[12] L. Blot and R. Zwiggelaar. Synthesis and Analysis of Solid Texture: Application in Medical Imaging. In *Texture 2002: The 2nd International Workshop on Texture Analysis and Synthesis*, pages 9–14, Copenhagen, 1 June 2002. (cited on pages 2, 9)

[13] B. Boashash. Estimating and Interpreting the Instantaneous Frequency of a Signal; Part I: Fundamentals, Part II: Algorithms. *Proceedings of the IEEE*, 80(4):519–569, April 1992. (cited on page 53)

[14] C. Bouman and B. Liu. Multiple Resolution Segmentation of Textured Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(2):99–113, 1991. (cited on pages 1, 8, 111)

[15] A. C. Bovik, M. Clark, and W. S. Geisler. Multichannel Texture Analysis Using Localized Spatial Filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):55 – 73, 1990. (cited on pages 1, 48, 49, 71)

[16] R. N. Bracewell. *The Fourier Transform and its applications.* McGraw-Hill, New York, 3rd edition, 1986. (cited on pages 3, 4)

[17] P. J. Burt and E. H. Adelson. The Laplacian Pyramid as a compact Image Code. *IEEE Transactions on Communications*, 31(4):532–540, 1983. (cited on pages 12, 97, 152)

[18] A. Carrillat, T. Randen, L. Sønneland, and G. Elvebakk. Seismic stratigraphic mapping of carbonate mounds using 3D texture attributes. In *Extended Abstracts, Annual Meeting, European Association of Geoscientists and Engineers*, Florence, Italy, May 27-30 2002. (cited on page 1)

[19] T. Chang and C. C. J. Kuo. Texture Analysis and Classification with Tree-Structured Wavelet Transform. *IEEE Transactions on Image Processing*, 2(4):429–441, 1993. (cited on pages 29, 62)

[20] M. J. Chantler. Why Illuminant Direction is fundamental to Texture Analysis. *IEE Proceedings in Vision, Image and Signal Processing*, 142(4):199–206, 1995. (cited on page 9)

[21] R. Chellapa and A. Jain. *Markov Random Fields.* Academic Press, Boston, 1993. (cited on page 111)

[22] C. K. Chui. *An Introduction to Wavelets.* Academic Press Inc., Boston, 1992. (cited on page 30)

[23] A. Chung, J. Noble, and P. Summers. Fusing magnitude and phase information for vascular segmentation in phase contrast MR angiography. *Medical Image Analysis Journal*, 6(2):109–128, 2002. (cited on page 53)

[24] D. A. Clausi and M. E. Jernigan. A Fast Method to Determine Co-occurrence Texture Features. *IEEE Transactions on Geoscience and Remote Sensing*, 36(1):298–300, 1998. (cited on pages 39, 42, 44)

[25] Z. A. Cohen, D. McCarthy, S. Kwark, P. Legrand, F. Fogarasi, E. Ciaccio, and G. Ateshian. Knee Cartilage Topography, Thickness, and Contact Areas from MRI: in-vitro Calibration and in-vivo Measurements. *Osteoarthritis and Cartilage*, 7(1):95–109, 1999. (cited on pages 136, 137)

[26] Z. A. Cohen, V. C. Mow, J. H. Henry, W. N. Levine, and G. A. Ateshian. Templates of the cartilage layers of the patellofemoral joint and their use in the assessment of osteoarthritic cartilage damage. *Osteoarthritis and Cartilage*, 11(8):569–579, 2003. (cited on page 137)

[27] G. B. Coleman and H. C. Andrews. Image Segmentation by Clustering. *Proceedings of the IEEE*, 67(5):773–785, 1979. (cited on pages 22, 87)

[28] COST European Cooperation in the field of Scientific and Technical Research. *COST B11 Quantitation of Magnetic Resonance Image Texture*. World Wide Web, http://www.uib.no/costb11/, 2002. (cited on page 2)

[29] G. R. Cross and A. K. Jain. Markov Random Field Texture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(1):25–39, 1983. (cited on pages 1, 111)

[30] J. E. Crouch and R. McClintic. *Human Anatomy and Physiology*. John Wiley & Sons, NY, 1971. (cited on page 162)

[31] O. Cula and K. Dana. 3D Texture Recognition Using Bidirectional Feature Histograms. *International Journal of Computer Vision*, 59(1):33–60, 2004. (cited on page 9)

[32] K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 18(1):1–34, 1999. (cited on page 9)

[33] D. Disler, T. McCauley, C. Kelman, M. Fuchs, L. Ratner, C. Wirth, and P. Hospodar. Fat-suppressed three-dimensional spoiled gradient-echo MR imaging of hyaline cartilage defects in the knee: comparison with standard MR imaging and arthroscopy. *American Journal of Roentgenology*, 167(1):127–132, 1996. (cited on page 143)

[34] M. Dong and R. Kothari. Feature subset selection using a new definition of classifiability. *Pattern Recognition Letters*, 24(9-10):1215–1225, 2003. (cited on page 85)

[35] E. R. Dougherty and M. Brun. A probabilistic theory of clustering. *Pattern Recognition*, 37(5):917–925, 2004. (cited on page 67)

[36] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, New York, 1973. (cited on page 15)

[37] D. Dunn and W. Higgins. Optimal Gabor filters for texture segmentation. *IEEE Transactions on Image Processing*, 4(7):947–964, July 1995. (cited on page 49)

[38] D. Dunn, W. Higgins, and J. Wakeley. Texture segmentation using 2-D Gabor elementary functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):130–149, February 1994. (cited on pages 47, 49)

[39] F. Eckstein, A. Gavazzeni, H. Sittek, M. Haubner, A. Losch, S. Milz, K. Englmeier, E. Schulte, R. Putz, and M. Reiser. Determination of knee joint cartilage thickness using three-dimensional Magnetic Resonance Chondro-Crassometry (3D MR-CCM). *Magn. Reson. Med.*, 36(2):256–265, 1996. (cited on page 136)

[40] N. Fatemi-Ghomi. *Performance measures for wavelet-based segmentation algorithms*. PhD thesis, Centre for Vision, Speech and Signal Processing, University of Surrey, 1997. (cited on pages 29, 95, 149)

[41] J. Feeney. Magnetic Resonance Imaging - A Window into the Human Body. *Mill Hill Essays*, http://www.nimr.mrc.ac.uk/MillHillEssays/1996/mri.htm, 2001. (cited on page 159)

[42] M. Fernández, A. Mavilio, and M. Tejera. Texture Segmentation of a 3D Seismic Section with Wavelet Transform and Gabor Filters. In *International Conference on Pattern Recognition, ICPR00*, volume 3, pages 358–361, Barcelona, September 3-8 2000. (cited on pages 29, 49)

[43] M. Foster. *Magnetic Resonance in Medicine and Biology.* Pergamon Press, New York, 1984. (cited on page 157)

[44] D. Frossyniotis, A. Lykas, and A. Stafylopats. A clustering method based on boosting. *Pattern Recognition Letters*, 25(6):641–654, April 2004. (cited on page 79)

[45] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, 1972. (cited on page 87)

[46] D. Gabor. Theory of Communication. *Journal of the IEE*, 93(26):429–457, 1946. (cited on page 47)

[47] V. Gaede and O. Günther. Multidimensional access methods. *ACM Computing Surveys*, 30(2):170–231, 1998. (cited on pages 97, 151)

[48] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distribution, the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984. (cited on page 111)

[49] C. L. Gilchrist, J. Q. Xia, L. A. Setton, and E. W. Hsu. High-resolution determination of soft tissue deformations using MRI and first-order texture correlation. *IEEE Transactions on Medical Imaging*, 23(5):546–553, 2004. (cited on page 9)

[50] R. C. Gonzalez and R. E. Woods. *Digital Image Processing.* Addison Wesley, Reading Mass, second edition, 1992. (cited on pages 8, 21, 34, 45)

[51] A. Graps. An Introduction to Wavelets. *IEEE Computational Science & Engineering*, 2(2):50–61, 1995. (cited on page 31)

[52] I. Guyon and A. Elisseeff. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3(7-8):1157–1182, 2003. (cited on page 83)

[53] E. Hadjidemetriou, M. D. Grossberg, and S. K. Nayar. Multiresolution histograms and their use for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):831–847, July 2004. (cited on page 12)

[54] D. J. Hand. *Discrimination and classification.* Wiley, Chichester, 1981. (cited on pages 14, 15, 20, 67, 82, 107)

[55] R. M. Haralick. Statistical and Structural Approaches to Texture. *Proceedings of the IEEE*, 67(5):786–804, 1979. (cited on pages viii, 1, 37, 40, 41)

[56] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural Features for Image Classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6):610–621, 1973. (cited on pages viii, 1, 11, 35, 37, 40, 41)

[57] D. Harwood, T. Ojala, M. Pietikäinen, S. Kelman, and S. Davis. Texture classification by center-symmetric auto-correlation, using Kullback discrimination of distributions. Technical Report CAR-TR-678, Computer Vision Laboratory, Center for Automation Research, University of Maryland, College Park, Maryland, 1993. (cited on page 63)

[58] J. K. Hawkins. Textural Properties for Pattern Recognition. In B. Lipkin and A. Rosenfeld, editors, *Picture Processing and Psychopictorics*, pages 347–370, New York, 1970. (cited on page 7)

[59] D. He and L. Wang. Textural Filters Based On The Texture Spectrum. *Pattern Recognition*, 24(12):1187–1195, 1991. (cited on page 54)

[60] J. Hennel, T. Kryst-Widzgowska, and J. Klinowski. *A Primer of Magnetic Resonance Imaging.* Imperial College Press, London, UK, 1997. (cited on pages 156, 158, 159)

[61] S. Herlidou, R. Grebve, F. Grados, N. Leuyer, P. Fardellone, and M.-E. Meyer. Influence of age and osteoporosis on calcaneus trabecular bone structure: A preliminary in vivo MRI study by quantitative texture analysis. *Magnetic Resonance Imaging*, 22(2):237–243, 2004. (cited on page 44)

[62] S. Herlidou, Y. Rolland, J. Y. Bansard, E. L. Rumeur, and J. D. Certaines. Comparison of Automated and Visual Texture Analysis in MRI: Characterization of Normal and Diseased Skeletal Muscle. *Magnetic Resonance Imaging*, 17(9):1393–1397, 1999. (cited on page 44)

[63] S. Herlidou-Même, J. M. Constans, B. Carsin, D. Olivie, P. A. Eliat, L. Nadal-Desbarats, C. Gondry, , E. L. Rumeur, I. Idy-Peretti, and J. D. Certaines. MRI Texture Analysis on texture text objects, normal brain and intracranial tumors. *Magnetic Resonance Imaging*, 21(9):989–993, 2001. (cited on page 44)

[64] C. Heron and P. Calvert. Three-dimensional gradient-echo MR imaging of the knee: comparison with arthroscopy in 100 patients. *Radiology*, 183(3):839–44, June 1992. (cited on page 136)

[65] E. A. Hoffman, J. M. Reinhardt, M. Sonka, B. A. Simon, J. Guo, O. Saba, D. Chon, S. Samrah, H. Shikata, J. Tschirren, K. Palagyi, K. C. Beck, and G. McLennan. Characterization of the Interstitial Lung Diseases via Density-Based and Texture-Based Analysis of Computed Tomography Images of Lung Structure and Function. *Academic Radiology*, 10(10):1104–1118, 2003. (cited on page 2)

[66] J. P. Hornak. *The Basics of MRI.* World Wide Web, http://www.cis.rit.edu/htbooks/mri/bmri.htm, 1996. (cited on page 159)

[67] T.-I. Hsu, A. Calway, and R. Wilson. Analysis of Structured Texture using the Multiresolution Fourier Transform. Technical Report CS-RR-226, Department of Computer Science, University of Warwick, 1992. (cited on page 8)

[68] H. H. S. Ip and S. W. C. Lam. Using an octree-based rag in hyper-irregular pyramid segmentation of texture volume. In *Proceedings of the IAPR Workshop on Machine Vision Applications*, pages 259–262, Kawasaki, Japan, December 13-15 1994. (cited on page 42)

[69] K. Jafari-Khouzani, H. Soltanian-Zadeh, K. Elisevich, and S. Patel. Comparison of 2D and 3D Wavelet features for TLE lateralization. In A. A. Amini and A. Manduca, editors, *Proceedings of SPIE Vol. 5369, Medical Imaging 2004: Physiology, Function, and Structure from Medical Images*, pages 593–601, San Diego, CA, USA, February 14-19 2004. (cited on page 34)

[70] A. K. Jain. Cluster Analysis. In Y. Fu, editor, *Handbook of Pattern Recognition and Image Processing*, pages 33–57, New York, 1986. Academic Press. (cited on page 79)

[71] A. K. Jain and F. Farrokhnia. Unsupervised texture segmentation using Gabor filters. *Pattern Recognition*, 24(12):1167– 1186, 1991. (cited on pages 1, 12, 48, 49, 62, 71, 108)

[72] D. James, B. D. Clymer, and P. Schmalbrock. Texture Detection of Simulated Microcalcification Susceptibility Effects in Magnetic Resonance Imaging of the Breasts. *Journal of Magnetic Resonance Imaging*, 13(6):876–881, 2002. (cited on page 2)

[73] K. A. Johnson and J. A. Becker. *The Whole Brain Atlas.* World Wide Web, http://www.med.harvard.edu/AANLIB, 1999. (cited on page 159)

[74] A. Kadyrov and M. Petrou. The Trace Transform and its Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):811–828, 2001. (cited on page 58)

[75] A. Kadyrov, A. Talepbour, and M. Petrou. Texture Classification with Thousand of Features. In *British Machine Vision Conference (BMVC)*, pages 656–665, Cardiff, UK, 2-5 September 2002. (cited on pages 1, 58, 59)

[76] T. Kailath. The Divergence and Bhattacharyya Distance Measures in Signal Selection. *IEEE Transactions on Communication Technology*, 15(1):52–60, 1967. (cited on page 87)

[77] S. Kamvar, D. Klein, and C. D. Manning. Interpreting and Extending Classical Agglomerative Clustering Algorithms using a Model-based Approach. In C. Sammut and A. G. Hoffmann, editors, *Proceedings of the 19th Conference on Machine Learning*, Sydney, NSW, Australia, July 8-12 2002. (cited on page 80)

[78] T. Kapur. *Model-based three dimensional Medical Image Segmentation.* PhD thesis, AI Lab, Massachusetts Institute of Technology, May 1999. (cited on pages 2, 24)

[79] C. Kervrann and F. Heitz. A Markov Random Field model-based approach to unsupervised texture segmentation using local and global spatial statistics. *IEEE Transactions on Image Processing*, 4(6):856–862, 1995. (cited on page 1)

[80] J. Kittler. Feature Selection and Extraction. In Y. Fu, editor, *Handbook of Pattern Recognition and Image Processing*, pages 59–83, New York, 1986. Academic Press. (cited on pages 82, 83, 84)

[81] H. Knutsson and G. H. Granlund. Texture Analysis Using Two-Dimensional Quadrature Filters. In *IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Image Database Management - CAPAIDM*, pages 206–213, Pasadena, October 1983. (cited on pages 12, 48)

[82] H. Knutsson, C. F. Westin, and G. H. Granlund. Local multiscale frequency and bandwidth estimation. In *Proceedings of the IEEE International Conference on Image Processing*, pages 36–40, Austin, Texas, November 1994. IEEE. (cited on page 53)

[83] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997. (cited on page 85)

[84] T. Kohonen. *Self-Organizing Maps*. Springer, Berlin, Heidelberg, New York, third extended edition, 2001. (cited on pages 76, 105)

[85] V. A. Kovalev, F. Kruggel, H.-J. Gertz, and D. Y. von Cramon. Three-Dimensional Texture Analysis of MRI Brain Datasets. *IEEE Transactions on Medical Imaging*, 20(5):424–433, 2001. (cited on pages 28, 42, 43)

[86] V. A. Kovalev, F. Kruggel, and D. Y. von Cramon. Gender and age effects in structural brain asymmetry as measured by MRI texture analysis. *NeuroImage*, 19(3):895–905, 2003. (cited on pages 28, 42, 43)

[87] V. A. Kovalev and M. Petrou. Multidimensional Co-occurrence Matrices for Object Recognition and Matching. *Graphical Models and Image Processing*, 58(3):187–197, 1996. (cited on pages 29, 42)

[88] V. A. Kovalev, M. Petrou, and Y. S. Bondar. Texture Anisotropy in 3D Images. *IEEE Transactions on Image Processing*, 8(3):346–360, 1999. (cited on pages 2, 28, 29, 42)

[89] V. A. Kovalev, M. Petrou, and J. Suckling. Detection of structural differences between the brains of schizophrenic patients and controls. *Psychiatry Research: NeuroImaging*, 124(3):177–189, 2003. (cited on pages 28, 42)

[90] P. K. Kumar, B. Yegnanarayana, and S. Das. 1-D Gabor for edge detection in texture Images. In *International Conference on Communications, Computers and*

*Devices (ICCCD 2000)*, pages 425–428, IIT Kharagpur, INDIA, December 14-16 2000. (cited on page 50)

[91] A. Laine and J. Fan. Texture Classification by Wavelet Packet Signatures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1186–1191, 1993. (cited on page 29)

[92] Z. Lang, R. Scarberry, Z. Zhang, W. Shao, and X. Sun. A texture-based direct 3D segmentation system for confocal scanning fluorescence microscopic images. In *Twenty-Third Southeastern Symposium on System Theory*, pages 472 – 476, Columbia, SC, 10-12 March 1991. (cited on page 27)

[93] K. Laws. *Textured Image Segmentation*. PhD thesis, University of Southern California, January 1980. (cited on page 27)

[94] R. Lerski, K. Straughan, L. R. Schad, D. Boyce, S. Bluml, and I. Zuna. MR Image Texture Analysis - An Approach to tissue Characterization. *Magnetic Resonance Imaging*, 11(6):873–887, 1993. (cited on page 2)

[95] T. K. Leung and J. Malik. Recognizing Surfaces using Three-Dimensional Textons. In *ICCV (2)*, pages 1010–1017, Corfu, Greece, September 20-25, 1999. (cited on page 9)

[96] C.-T. Li. *Unsupervised Texture Segmentation Using Multiresolution Markov Random Fields*. PhD thesis, Department of Computer Science, University of Warwick, 1998. (cited on page 62)

[97] Y. Linde, A. Buzo, and R. Gray. An Algorithm for Vector Quantizer Design. *IEEE Transactions in Communications*, 28(1):84–95, 1980. (cited on pages 80, 107)

[98] X. Liu and D. Wang. Appearance-Based Recognition Using Perceptual Components. In *International Joint Conference in Neural Networks*, pages 1943–1948, Washington, DC, March 2001. (cited on page 65)

[99] X. Lladó, A. Oliver, M. Petrou, J. Freixenet, and J. Martí. Simultaneous Surface Texture Classification and Illumination Tilt Angle Prediction. In *British Machine Vision Conference (BMVC)*, pages 789–798, Norwich, UK., September 2003. (cited on page 9)

[100] L. M. Lorigo, O. D. Faugeras, W. E. L. Grimson, R. Keriven, and R. Kikinis. Segmentation of Bone in Clinical Knee MRI Using Texture-Based Geodesic Active Contours. In *Medical Image Computing and Computer-Assisted Interventions (MICCAI)*, pages 1195–1204, Cambridge, USA, October 11-13 1998. (cited on pages 2, 25, 136)

[101] D. Mahmoud-Ghoneim, G. Toussaint, J. M. Constans, and J. D. de Certaines. Three dimensional texture analysis in MRI: a preliminary evaluation in gliomas. *Magnetic Resonance Imaging*, 21(9):983–987, 2003. (cited on page 43)

[102] J. Malik and P. Perona. Preattentive texture discrimination with early vision mechanisms. *J. Opt. Soc. Am. A*, 7(5):923–932, 1990. (cited on page 47)

[103] S. G. Mallat. A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989. (cited on page 31)

[104] N. Malpica, J. E. Ortuño, and A. Santos. A multichannel watershed-based algorithm for supervised texture segmentation. *Pattern Recognition Letters*, 24(9):1545–1554, 2003. (cited on pages viii, xi, 64, 65, 96, 118, 119, 120)

[105] J. Matas and J. V. Kittler. Spatial and feature space clustering: applications in image analysis. In V. Hlavac and R. Sara, editors, *Proceedings of the 6th International Conference on Computer Analysis of Images and Patterns*, pages 162–173, Berlin, Germany, 1995. Springer-Verlag. (cited on pages 116, 117)

[106] J. M. Mathias, P. S. Tofts, and N. A. Losseff. Texture Analysis of Spinal Cord Pathology in Multiple Sclerosis. *Magnetic Resonance in Medicine*, 42(5):929–935, 1999. (cited on page 2)

[107] T. O. McKinley and B. K. Bay. Trabecular bone strain changes associated with cartilage defects in the proximal and distal tibia. *Journal of Orthopaedic Research*, 19(5):906–913, 2001. (cited on page 9)

[108] Merriam-Webster. *Merriam-Webster's Collegiate Dictionary*. NY, USA, 11 edition, 2004. (cited on pages 7, 44)

[109] S. K. Moore. Better Breast Cancer Detection. *IEEE Spectrum*, 38(5):50–54, 2001. (cited on pages 156, 157)

[110] F. Neyret. A general and Multiscale Model for Volumetric Textures. In *Graphics Interface*, Québec, Canada, 17-19 May 1995. (cited on page 9)

[111] T. Ojala, M. Pietikäinen, and D. Harwood. A Comparative Study of Texture Measures with Classification based on Feature Distributions. *Pattern Recognition*, 29(1):51–59, 1996. (cited on pages xi, 54, 57, 63, 65)

[112] T. Ojala, K. Valkealahti, E. Oja, and M. Pietikäinen. Texture discrimination with multidimensional distributions of signed gray level differences. *Pattern Recognition*, 34(3):727–739, 2001. (cited on pages viii, 58, 64, 96, 118, 119)

[113] S. K. Pakin, J. G. Tamez-Pena, S. Totterman, and K. J. Parker. Segmentation, surface extraction, and thickness computation of articular cartilage. In M. Sonka and J. M. Fitzpatrick, editors, *Proceedings of SPIE:Medical Imaging: Image Processing*, volume 4684, pages 155–166, San Diego, CA, USA, May 2002. (cited on page 137)

[114] T. N. Pappas. An Adaptive Clustering Algorithm for Image Segmentation. *IEEE Transactions on Signal Processing*, 40(4):901–914, 1992. (cited on pages 112, 114)

[115] P. Perona and J. Malik. Scale-Space and Edge Detection Using Anisotropic Diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990. (cited on page 106)

[116] M. Petrou and A. Kadyrov. Affine Invariant Features from the Trace Transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1):30–44, 2004. (cited on page 58)

[117] H. Pettersson. *Medcyclopaedia: The Encyclopaedia of Medical Imaging.* World Wide Web, http://www.amershamhealth.com/medcyclopaedia/, Standard edition, 2004. (cited on pages 159, 160)

[118] O. Pichler, A. Teuner, and B. J. Hosticka. A comparison of texture feature extraction using adaptive Gabor filtering, pyramidal and tree structured Wavelet transforms. *Pattern Recognition*, 29(5):733–742, 1996. (cited on page 62)

[119] C. Porteneuve, J.-P. Korb, D. Petit, and H. Zanni. Structure-Texture Correlation in Ultra High Performance Concrete: a Nuclear Magnetic Resonance Study. In *Franco-Italian Conference on Magnetic Resonance*, La Londe Les Maures, France, 2nd - 5th May 2000. (cited on page 9)

[120] H. Potter, J. Linklater, A. Allen, J. Hannafin, and S. Haas. Magnetic Resonance Imaging of articular cartilage in the knee. An evaluation with use of fast-spin-echo imaging. *J Bone Joint Surg Am*, 80(9):1276–84, September 1998. (cited on page 136)

[121] N. M. Rajpoot. Texture Classification Using Discriminant Wavelet Packet Subbands. In *Proceedings 45th IEEE Midwest Symposium on Circuits and Systems (MWSCAS 2002)*, Tulsa, OK, USA, August 2002. (cited on page 29)

[122] T. Randen and J. H. Husøy. Multichannel filtering for image texture segmentation. *Optical Engineering*, 33(8):2617–2625, 1994. (cited on page 48)

[123] T. Randen and J. H. Husøy. Texture segmentation with optimal linear prediction error filters. *Piksel'n*, 11(3):25–28, September 1994. (cited on page 71)

[124] T. Randen and J. H. Husøy. Filtering for Texture Classification: A Comparative Study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4):291–310, 1999. (cited on pages v, viii, xi, 45, 49, 58, 64, 65, 66, 67, 69, 70, 71, 81, 118, 119)

[125] T. Randen, E. Monsen, A. Abrahamsen, J. O. Hansen, J. Schlaf, and L. Sønneland. Three-dimensional texture attributes for seismic data analysis. In *Ann. Int. Mtg., Soc. Expl. Geophys., Exp. Abstr.*, Calgary, Canada, August 2000. (cited on pages 1, 28)

[126] T. Randen, L. Sønneland, A. Carrillat, S. Valen, T. Skov, S. I. Pedersen, B. Rafaelsen, and G. Elvebakk. Preconditioning for optimal 3D stratigraphical and structural inversion. In *65th EAGE Conference & Exhibition*, Stavanger, June 2003. (cited on page 50)

[127] A. R. Rao and G. L. Lohse. Towards a texture naming system: identifying relevant dimensions of texture. In *Proceedings of the 4th conference on Visualization*, pages 220–227, San Jose, California, 1993. (cited on page 8)

[128] C. C. Reyes-Aldasoro and A. Bhalerao. Volumetric Texture Description and Discriminant Feature Selection for MRI. In C. Taylor and A. Noble, editors, *Proceedings of Information Processing in Medical Imaging*, pages 282–293, Ambleside, UK, July 2003. (cited on pages 2, 17, 82)

[129] C. Reyraud, J.-P. Renard, and C. Blondel. Le magnétisme. *La Recherche*, Spécial:50–53, Août/Septembre/Octobre 2001. (cited on pages 158, 159)

[130] F. Rousseau, R. Fablet, and C. Barillot. Robust Statistical Registration of 3D Ultrasound Images Using Texture Information. In *IEEE Int. Conf. on Image Processing, ICIP'2003*, Barcelona, Spain, September 2003. (cited on pages 48, 49)

[131] N. Saeed and B. K. Puri. Cerebellum Segmentation Employing Texture Properties and Knowledge based Image Processing : Applied to Normal Adult Controls and Patients. *Magnetic Resonance Imaging*, 20(5):425–429, 2002. (cited on page 2)

[132] H. Samet. The Quadtree and Related Hierarchical Data Structures. *Computing Surveys*, 16(2):187–260, 1984. (cited on pages 12, 97, 151)

[133] A. Sayeed, M. Petrou, N. Spyrou, A. Kadyrov, and T. Spinks. Diagnostic features of Alzheimer's disease extracted from PET sinograms. *Phys. Med. Biol.*, 47(1):137–148, 2002. (cited on pages 2, 61)

[134] L. R. Schad, S. Bluml, and I. Zuna. MR Tissue Characterization of Intracranial Tumors by means of Texture Analysis. *Magnetic Resonance Imaging*, 11(6):889–896, 1993. (cited on page 2)

[135] P. Schroeter and J. Bigun. Hierarchical Image Segmentation by Multi-dimensional Clustering and Orientation-Adaptive Boundary Refinement. *Pattern Recognition*, 28(5):695–709, 1995. (cited on pages 13, 17, 97, 98, 101, 103, 105)

[136] M. Segovia-Martínez, M. Petrou, V. A. Kovalev, and P. Perner. Quantifying Level of Brain Atrophy Using Texture Anisotropy in CT Data. In *Medical Imaging Understanding and Analysis*, pages 173–176, Oxford, UK, July 1999. (cited on pages 2, 28, 29)

[137] R. Sivaramakrishna, K. A. Powell, M. L. Lieber, W. A. Chilcote, and R. Shekhar. Texture analysis of lesions in breast ultrasound images. *Computerized Medical Imaging and Graphics*, 26(5):303–307, September 2002. (cited on page 2)

[138] S. Solloway, C. J. Taylor, C. E. Hutchinson, and J. C. Waterton. Quantification of Articular Cartilage from MR Images Using Active Shape Models. In *ECCV (2)*, pages 400–411, 1996. (cited on pages 136, 137)

[139] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. PWS, Pacific Grove, USA, second edition, 1998. (cited on page 8)

[140] M. Spann and R. Wilson. A quad-tree approach to image segmentation which combines statistical and spatial information. *Pattern Recognition*, 18(3/4):257–269, 1985. (cited on pages 12, 97)

[141] S. Srisuk, K. Ratanarangsank, W. Kurutach, and S. Waraklang. Face Recognition using a New Texture Representation of Face Images. In *Proceedings of Electrical Engineering Conference*, pages 1097–1102, Cha-am, Thailand, November 6-7 2003. (cited on page 58)

[142] K. R. Subramanian, J. P. Brockway, and W. B. Carruthers. Interactive detection and visualization of breast lesions from dynamic contrast enhanced MRI volumes. *Computerized Medical Imaging and Graphics*, 28(8):435–444, December 2004. (cited on page 2)

[143] C. Tai and K. Baba-Kishi. Microtexture studies of PST and PZT Ceramics and PZT Thin Film by Electron Backscatter Diffraction Patterns. *Textures and Microstructures*, 35(2):71–86, 2002. (cited on page 1)

[144] H. Tamura, S. Mori, and T. Yamawaki. Texture Features corresponding to Visual Perception. *IEEE Transactions on Systems, Man and Cybernetics*, 8(6):460–473, 1978. (cited on pages 1, 8)

[145] J. Theiler and G. Gisler. A contiguity-enhanced k-means clustering algorithm for unsupervised multispectral image segmentation. In I. Mirkin and L. Singher, editors, *Proceedings SPIE*, volume 3159, pages 108–118, October 1997. (cited on page 116)

[146] M. Tuceryan and A. K. Jain. Texture Analysis. In C. H. Chen, L. F. Pau, and P. S. P. Wang, editors, *Handbook of Pattern Recognition and Computer Vision*, pages 207–248. World Scientific Publishing, 1998. (cited on pages 1, 8)

[147] M. Unser. Texture Classification and Segmentation Using Wavelet Frames. *IEEE Transactions on Image Processing*, 4(11):1549–1560, 1995. (cited on pages 1, 29, 31, 33)

[148] I. Van Breuseghem. Ultrastructural MR imaging techniques of the knee articular cartilage: problems for routine clinical application. *Eur Radiol*, 14(2):184–192, 2004. (cited on page 136)

[149] K. L. Verstraete, F. Almqvist, P. Verdonk, G. Vanderschueren, W. Huysse, R. Verdonk, and G. Verbrugge. Magnetic Resonance Imaging of cartilage and cartilage repair . *Clinical Radiology*, 59(8):674–689, August 2004. (cited on pages 136, 137)

[150] L. Wang and D. He. Texture Classification Using Texture Spectrum. *Pattern Recognition*, 23(8):905–910, 1990. (cited on page 54)

[151] S. Warfield, C. Winalski, F. Jolesz, and R. Kikinis. Automatic Segmentation of MRI of the Knee. Technical Report 91, Department of Radiology, Brigham and Women's Hospital and Harvard Medical School, July 1998. (cited on page 137)

[152] S. K. Warfield, M. Kaus, F. A. Jolesz, and R. Kikinis. Adaptive, Template Moderated, Spatially Varying Statistical Classification. *Medical Image Analysis*, 4(1):43–55, Mar 2000. (cited on pages 136, 137)

[153] S. Webb, editor. *The Physics of Medical Imaging*. Institute of Physics Publishing, Bristol, 1996. (cited on page 159)

[154] T. P. Weldon, W. E. Higgins, and D. F. Dunn. Efficient Gabor Filter Design for Texture Segmentation. *Pattern Recognition*, 29(12):2005–2015, 1996. (cited on page 49)

[155] C. F. Westin, A. Bhalerao, H. Knutsson, and R. Kikinis. Using Local 3D Structure for Segmentation of Bone from Computer Tomography Images. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, June 1997. IEEE. (cited on page 47)

[156] J. Weszka, C. Dyer, and A. Rosenfeld. A comparative Study of Texture Measures for Terrain Classification. *IEEE Transactions Systems, Man and Cybernetics*, 6(4):269–285, April 1976. (cited on page 1)

[157] R. Wilson, A. D. Calway, E. R. S. Pearson, and A. R. Davies. An Introduction to the Multiresolution Fourier Transform and its Applications. Technical Report CS-RR-204, Department of Computer Science, University of Warwick, UK, 1992. (cited on page 13)

[158] R. Wilson and G. H. Granlund. The uncertainty principle in image processing. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 6(6):758–767, 1984. (cited on page 12)

[159] R. Wilson and C.-T. Li. Multiresolution Random Fields and their Application to Image Analysis. Technical Report CS-RR-361, Department of Computer Science, University of Warwick, UK, 1999. (cited on pages 111, 114)

[160] R. Wilson and C.-T. Li. A Class of Discrete Multiresolution Random Fields and its Application to Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):42–56, 2003. (cited on page 114)

[161] R. Wilson and M. Spann. Finite Prolate Spheroidal Sequences and Their Applications: Image Feature Description and Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(2):193–203, 1988. (cited on page 51)

[162] R. Wilson and M. Spann. *Image Segmentation and Uncertainty.* John Wiley and Sons Inc., New York, 1988. (cited on pages 12, 13, 16, 17, 97, 105)

[163] G. Winkler. *Image Analysis, Random Fields and Dynamic Monte Carlo Methods.* Springer, Berlin, Germany, first edition, 1995. (cited on pages 21, 108)

[164] O. Yu, Y. Mauss, I. J. Namer, and J. Chambron. Existence of contralateral abnormalities revealed by texture analysis in unilateral intractable hippocampal epilepsy. *Magnetic Resonance Imaging*, 19(10):1305–1310, 2001. (cited on page 2)

[165] O. Yu, C. Roch, I. J. Namer, J. Chambron, and Y. Mauss. Detection of late epilepsy by the texture analysis of MR brain images in the lithium-pilocarpine rat model. *Magnetic Resonance Imaging*, 20(10):771–775, 2002. (cited on page 2)

[166] Y. Zhan and D. Shen. Automated Segmentation of 3D US Prostate Images Using Statistical Texture-Based Matching Method. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 688–696, Canada, November 16-18 2003. (cited on pages 2, 49)

[167] S. W. Zucker and R. A. Hummel. A Three-Dimensional Edge Operator. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3(3):324–331, 1981. (cited on page 28)

[168] J. Zupan. *Clustering of Large Data Sets*. Research Studies Press, Chichester, 1982. (cited on pages 80, 81)