

# Upper airways segmentation using principal curvatures



**César Julio Bustacara Medina**

Facultad de Ingeniería

Pontificia Universidad Javeriana

This dissertation is submitted for the degree of  
*Doctor of Philosophy*





# Upper airways segmentation using principal curvatures



Author:

César Julio Bustacara Medina

Dissertation presented to the Engineering Faculty of the Pontificia Universidad Javeriana in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Advisor:

Ing. Leonardo Flórez Valencia, PhD.

PONTIFICIA UNIVERSIDAD JAVERIANA  
FACULTAD DE INGENIERIA  
DOCTORADO EN INGENIERIA  
BOGOTA D.C., COLOMBIA  
August, 2018



# Upper airways segmentation using principal curvatures

Author:

César Julio Bustacara Medina

Dissertation presented to the Engineering Faculty of the Pontificia Universidad Javeriana in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Approved by  
Supervising Committee:

---

Ing. Leonardo Flórez Valencia, PhD., Advisor  
Pontificia Universidad Javeriana, Bogotá D.C.

---

Ing. Pedro Raúl Vizcaya Guarín, PhD., Chair  
Pontificia Universidad Javeriana, Bogotá D.C.

---

Ing. Andrés Navarro Newball, PhD.  
Pontificia Universidad Javeriana, Cali.

---

Ing. Simone Balocco, PhD.  
University of Barcelona, Spain.

Bogotá D.C., Colombia  
August 2018



I would like to dedicate this thesis to:

*God.*

For guiding my way on this wonderful journey of research. For the strength to overcome moments of uncertainty and above all for his infinite kindness and love.

*My loving father Rafael.*

For the examples of honesty, dedication to work and perseverance that characterized him and that he always infused me, for the courage shown to get ahead and for his love.

*My mother Magola.*

For her love and unconditional support, for her advice, her values, for the constant motivation that has allowed me to be a good person, but more than anything, for her presence.

*My beautiful and beloved Maria Emilia.*

For your love and support during all this time, for remaining silent at my side. Because you have taught me the gratitude and generosity that is contained in every detail of life.

*My wonderful son Juan Sebastian.*

Who every day of his life teaches me to cross my limits in the pursuit of happiness. You have been the best teacher of my life.

*My beloved Carlitos and Pipe.*

For received me and have given me their love without conditions.

*My Family.*

For all your support and company both in the pleasant moments and in the difficult ones.



## **Declaration**

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other University. This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where specifically indicated in the text.

César Julio Bustacara Medina

August, 2018





## **Acknowledgements**

I would like to express my sincere gratitude to my advisor Ing. Leonardo Flórez Valencia for the continuous support of my PhD study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my PhD study. In addition, I would like to thank the individuals or groups of people who have provided me the datasets to test my algorithms on. Dr. Luis Felipe Uriza of the Department of Radiology at the Pontificia Universidad Javeriana supplied me with the head-neck CT images used in this thesis. Finally, I must express my very profound gratitude to my friends and colleagues for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thanks for all your encouragement!



## **Abstract**

This dissertation proposes a new approach to segment the upper airways. This proposal allows the extraction of curvilinear structures based on the principal curvatures. The proposal allows extracting these structures from 2D and 3D images. Among the main novelties is the proposal of a new stopping criterion to stop the propagation of the contrast enhancement algorithm (multiscale top-hat morphological operator). In the same way, the proposed stopping criterion is used to stop the anisotropic diffusion algorithms. In addition, a new criterion is proposed to select the principal curvatures that make up the curvilinear structures, which is based on the criteria proposed by Steger, Deng et. al. and Armande et. al. Furthermore, a new algorithm to perform the non-maximum suppression that allows reducing the presence of discontinuities in the border of curvilinear structures is proposed. To extract the edges of the curvilinear structures, a linking algorithm is used that includes a new distance criterion to reduce the appearance of gaps in the final structure. Finally, based on the obtained results, a morphological algorithm is used to close the gaps and a region growing algorithm to obtain the final upper airways segmentation is applied.



# Contents

<b>Contents</b>	<b>xv</b>
<b>List of Figures</b>	<b>xxi</b>
<b>List of Tables</b>	<b>xxv</b>
<b>Nomenclature</b>	<b>xxx</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	2
1.2 Objective . . . . .	7
1.3 Proposed solution . . . . .	8
1.4 Contributions . . . . .	10
1.5 Thesis outline . . . . .	10
<b>2 Preliminar concepts</b>	<b>13</b>
2.1 Upper airways . . . . .	13
2.1.1 Upper airways anatomy and physiology . . . . .	14
2.1.1.1 Nasal cavity . . . . .	15
2.1.1.2 Nasal septum . . . . .	16
2.1.1.3 Nose . . . . .	17
2.1.1.4 External nose . . . . .	18
2.1.1.5 Paranasal sinuses . . . . .	18
2.1.1.6 Turbinates . . . . .	20
2.1.1.7 Pharynx . . . . .	21
2.1.1.8 Larynx . . . . .	23
2.1.2 Airways diseases . . . . .	24
2.1.2.1 Upper airways obstruction . . . . .	25
2.1.2.2 Sleep apnea . . . . .	26

2.2	Image segmentation . . . . .	29
2.3	Differential geometry . . . . .	31
2.3.1	Surfaces . . . . .	32
2.3.2	Curvatures . . . . .	33
2.3.2.1	Types of curvature . . . . .	33
2.3.2.2	Fundamental forms and shape operator . . . . .	34
2.3.2.3	Principal curvatures and vectors . . . . .	38
<b>3</b>	<b>Contrast enhancement using multi-scale morphological operator</b>	<b>43</b>
3.1	State of the art . . . . .	44
3.1.1	Morphological approaches . . . . .	46
3.1.1.1	Morphological operators . . . . .	47
3.1.1.2	Top-Hat transforms . . . . .	48
3.1.2	Multiscale Top-Hat transforms . . . . .	49
3.1.3	Contrast measures . . . . .	51
3.1.3.1	Contrast Improvement Ratio (CIR) . . . . .	51
3.1.3.2	Mean Squared Error (MSE) . . . . .	52
3.1.3.3	Peak Signal-to-Noise Ratio (PSNR) . . . . .	52
3.1.3.4	Edge Content (EC) . . . . .	53
3.2	Method proposed . . . . .	53
3.2.1	Contrast Improvement Ratio Revisited (CIRR) . . . . .	54
3.2.2	Stopping criterion proposed . . . . .	54
3.3	Experimental results . . . . .	55
3.3.1	2D case . . . . .	55
3.3.2	3D case . . . . .	61
<b>4</b>	<b>Smoothing using nonlinear anisotropic diffusion</b>	<b>71</b>
4.1	State of the art . . . . .	72
4.1.1	PDEs in image processing . . . . .	73
4.1.1.1	PDEs classification . . . . .	75
4.1.1.2	Physical problems . . . . .	76
4.1.1.3	Initial values and boundary conditions . . . . .	77
4.1.2	Diffusion process . . . . .	78
4.1.3	Smoothing using diffusion . . . . .	81
4.1.3.1	Linear Isotropic Diffusion Filters . . . . .	82
4.1.3.2	Nonlinear Isotropic Diffusion Filters . . . . .	84
4.1.3.3	Linear Anisotropic Diffusion Filters . . . . .	87

---

4.1.3.4	Nonlinear Anisotropic Diffusion Filters . . . . .	88
4.2	Nonlinear algorithms selected . . . . .	90
4.2.1	Perona-Malik filtering (PM-isotropic) . . . . .	91
4.2.2	Edge enhancing diffusion filtering (EED) . . . . .	91
4.2.3	Conservative edge enhancing diffusion filtering (cEED) . . . . .	93
4.2.4	Coherence enhancing diffusion filtering (CED) . . . . .	94
4.2.5	Conservative coherence enhancing diffusion filtering (cCED) . . . . .	96
4.3	Diffusion stopping criteria . . . . .	97
4.4	Stopping criterion proposed . . . . .	100
4.5	Experimental results . . . . .	100
4.5.1	2D case . . . . .	101
4.5.1.1	Original images . . . . .	101
4.5.1.2	Contrast-enhanced images . . . . .	108
4.5.2	3D case . . . . .	114
4.5.2.1	Original images . . . . .	114
4.5.2.2	Contrast-enhanced images . . . . .	120
<b>5</b>	<b>Features and shapes detection</b> . . . . .	<b>125</b>
5.1	State of the art . . . . .	125
5.1.1	Local features . . . . .	126
5.1.2	Features detectors . . . . .	127
5.1.3	Characteristics of Feature Detectors . . . . .	127
5.1.4	Local features detectors . . . . .	129
5.1.4.1	Edge detection . . . . .	130
5.1.4.2	Corner detection . . . . .	133
5.1.4.3	Blob detection . . . . .	137
5.1.5	Shape-based features . . . . .	144
5.1.5.1	Blob-like detection . . . . .	146
5.1.5.2	Sheet-like detection . . . . .	147
5.1.5.3	Vessel-like detection . . . . .	148
5.2	Proposed method . . . . .	151
5.3	Experimental results . . . . .	154
5.3.1	Ordering criterion of Hessian eigenvalues . . . . .	154
5.3.2	3D synthetic images . . . . .	159
5.3.3	3D medical images . . . . .	160

<b>6</b>	<b>Features and shapes extraction</b>	<b>165</b>
6.1	State of the art . . . . .	166
6.1.1	Ridge definition . . . . .	166
6.1.2	Geometrical properties of the surface of a ridge point . . . . .	167
6.1.3	Extraction schemes . . . . .	167
6.1.3.1	Nevatia and Babu feature extractor . . . . .	167
6.1.3.2	Canny edge detector . . . . .	168
6.1.3.3	Förstner features extractor . . . . .	169
6.1.3.4	Armande, Montesinos and Monga 3D thin net detector . . . . .	170
6.1.3.5	Steger line detector . . . . .	171
6.1.3.6	Aylward and Bullitt ridge detector . . . . .	172
6.1.3.7	PCBR detector . . . . .	173
6.1.4	Non-Maximum suppression (NMS) . . . . .	174
6.1.4.1	Canny's NMS algorithm . . . . .	174
6.1.4.2	Korn's NMS algorithm . . . . .	175
6.1.4.3	Devernay's NMS algorithm . . . . .	177
6.1.4.4	Multiple Directional Non-Maximum Suppression . . . . .	178
6.1.5	Linking and thinning algorithms . . . . .	179
6.1.5.1	Canny's hysteresis algorithm . . . . .	180
6.1.5.2	Monga, Deriche, Malandain, and Cocquerez's hysteresis algorithm . . . . .	181
6.1.5.3	Steger's hysteresis algorithm . . . . .	181
6.1.5.4	PCBR's hysteresis algorithm . . . . .	183
6.2	Proposed methods . . . . .	184
6.2.1	Non-maximum suppression . . . . .	184
6.2.2	Hysteresis thresholding . . . . .	186
6.3	Experimental results . . . . .	190
6.3.1	Non-maximum suppression tests . . . . .	190
6.3.1.1	2D images . . . . .	190
6.3.1.2	3D images . . . . .	194
6.3.2	Hysteresis tests . . . . .	198
6.3.2.1	2D images . . . . .	198
6.3.2.2	3D images . . . . .	200
<b>7</b>	<b>Upper airways segmentation using principal curvatures</b>	<b>205</b>
7.1	State of the art . . . . .	205
7.1.1	Image segmentation . . . . .	206



---

7.1.1.1	Thresholding and region growing . . . . .	207
7.1.2	Upper airways segmentation methods . . . . .	207
7.1.2.1	Vos et al. segmentation . . . . .	208
7.1.2.2	Van Holsbeke et. al. segmentation . . . . .	209
7.1.2.3	Xu et. al. segmentation . . . . .	210
7.1.2.4	Jeong et. al. segmentation . . . . .	211
7.2	Dataset . . . . .	212
7.2.1	Patient data . . . . .	212
7.2.2	CT device specifications . . . . .	213
7.2.3	CT Metadata . . . . .	213
7.2.4	Slices number . . . . .	214
7.2.5	Ground truth (semi-automatic) segmentation . . . . .	215
7.3	Segmentation method proposed . . . . .	216
7.4	Evaluation measures . . . . .	218
7.5	Experimental results . . . . .	219
<b>8</b>	<b>Conclusions</b>	<b>223</b>
	<b>References</b>	<b>225</b>



# List of Figures

1.1	Complications generated by sleep apnea disorders. . . . .	2
1.2	Superimposed recordings data in a patient with OSA (adapted from [1]). . . .	4
1.3	Upper airways segmented using FM. . . . .	6
1.4	Upper airways segmented using CV. . . . .	6
1.5	Main structure of proposed solution. . . . .	8
1.6	Main algorithms included in the proposed solution. . . . .	9
1.7	Thesis outline according to the proposed solution. . . . .	11
2.1	Upper Airways. Left image shows main structures and right image shows main divisions (taken from [2]). . . . .	14
2.2	Nasal cavities. . . . .	16
2.3	External nose and nasal septum. . . . .	16
2.4	Nose anatomy (adapted from <a href="http://physiotherapy.org.nz">http://physiotherapy.org.nz</a> ). . . . .	17
2.5	Paranasal sinuses. . . . .	19
2.6	Tubينات cross section. . . . .	21
2.7	Divisions of the pharynx. . . . .	22
2.8	Tangent circles to a 2D curve. . . . .	32
2.9	Shape space categories using $k_1$ and $k_2$ . (taken from [3]) . . . . .	39
3.1	HSS and BF computation using different window sizes. . . . .	57
3.2	Zoom of lena image in eye region. . . . .	61
3.3	Profile enhancement behavior of lena images. . . . .	61
3.4	HSS and BF computation using different window sizes. . . . .	63
3.5	Zoom of the slice 170 of image1 (left:original image, middle:HSS enhanced image, right:BF enhanced image). . . . .	69
3.6	Profile of the image1 and its enhanced images (blue line:original image, red line: HSS-enhanced image, green line: BF-enhanced image). . . . .	69
4.1	Linear isotropic diffusion . . . . .	84

4.2	Nonlinear isotropic diffusion . . . . .	86
4.3	Linear anisotropic diffusion . . . . .	88
4.4	Nonlinear anisotropic diffusion . . . . .	90
4.5	MSE, PSNR and CIRR measures of smoothed image of lena. . . . .	106
4.6	Profile behavior of smoothing algorithms using lena image. . . . .	108
4.7	MSE, PSNR and CIRR of original and contrast-enhancement image of lena. . . . .	110
4.8	Lena profile of original image, enhanced image, isotropic, CED, cCED, EED, and cEED. . . . .	113
5.1	Local features detectors . . . . .	130
5.2	Examples of sticks, plates, and blobs shapes (taken from [4]). . . . .	145
5.3	Examples of blobs, planes, and line (curvilinear) structures in 2D and 3D space (taken from [5]). . . . .	146
5.4	Vascular models classification. . . . .	150
5.5	Vascular features classification. . . . .	150
6.1	Kernel for non-maximum suppression used by Canny. . . . .	175
6.2	Search paths for the detection of maxima[6]. . . . .	176
6.3	Examples of linear and quadratic interpolation[7]. . . . .	178
6.4	2D linear windows at different directions [8]. . . . .	178
6.5	3D linear window at different directions [8]. . . . .	179
6.6	Two of the eight possible neighborhoods examined during the Steger's linking process. . . . .	182
6.7	Non-maximum suppression neighbors. . . . .	184
6.8	Non-maximum suppression using normal directions. . . . .	185
6.9	Non-maximum suppression using normal directions and tangent directions. . . . .	186
6.10	Possible configuration in hysteresis flow. . . . .	187
6.11	Different hysteresis configurations in 3D. . . . .	188
6.12	Examples of neighbors analyzed in hysteresis algorithm. . . . .	189
6.13	Example of possible holes in the surface. . . . .	190
7.1	Example of Mimics's muscle segmentation [adapted from <a href="http://www.materialise.com">http://www.materialise.com</a> ].	209
7.2	Models of upper airway for three different patients. . . . .	209
7.3	Upper airway segmented and divided into regions. . . . .	210
7.4	Upper airway segmented model. . . . .	211
7.5	Upper airway segmented example. . . . .	212
7.6	Age's patients . . . . .	213
7.7	Number of slices obtained per patient. . . . .	215

---

7.8 Upper airways segmentation method proposed. . . . . 217



# List of Tables

1.1	Estimated annual costs in GPD (U.S. \$billions) (Adapted from [9]) . . . . .	3
2.1	Upper airways structures. . . . .	15
2.2	Pixel label proposed by Haralick et. al. [10]. . . . .	39
2.3	A possible set of principal curvatures in a 4D surface. (taken from [11, 12]). . . . .	40
2.3	A possible set of principal curvatures in a 4D surface. (taken from [11, 12]). . . . .	41
3.1	HSS and BF indexes using CIRR contrast measure. . . . .	56
3.2	Required iterations using HSS and BF indexes. . . . .	58
3.3	Edges of original, HSS, and BF images. . . . .	58
3.3	Edges of original, HSS, and BF images. . . . .	59
3.3	Edges of original, HSS, and BF images. . . . .	60
3.4	HSS and BF indexes using CIRR contrast measure. . . . .	62
3.5	Iterations required using HSS and BF stopping criteria . . . . .	63
3.5	Iterations required using HSS and BF stopping criteria . . . . .	64
3.6	Edges of original, HSS, and BF images. . . . .	64
3.6	Edges of original, HSS, and BF images. . . . .	65
3.6	Edges of original, HSS, and BF images. . . . .	66
3.6	Edges of original, HSS, and BF images. . . . .	67
3.6	Edges of original, HSS, and BF images. . . . .	68
4.1	Types of PDEs and examples . . . . .	76
4.2	Examples of nonlinear isotropic diffusion . . . . .	86
4.2	Examples of nonlinear isotropic diffusion . . . . .	87
4.3	MSE measure - baboon. . . . .	101
4.4	PSNR measure - baboon. . . . .	102
4.5	CIRR measure - baboon. . . . .	102
4.6	MSE, PSNR and CIRR measures. . . . .	103
4.7	Number of iterations using JGTS and BF stopping criteria. . . . .	104

4.8	Edges of images obtained using JGTS and BF stopping criteria. . . . .	105
4.9	Zoom in of eye region of smoothed image of lena. . . . .	106
4.9	Zoom in of eye region of smoothed image of lena. . . . .	107
4.10	Number of iterations using JGTS and BF stopping criteria. . . . .	109
4.11	Smooth images and edges of lena with contrast-enhancement. . . . .	111
4.12	Zoom in of eye region of smoothed image of lena-MSTH. . . . .	112
4.12	Zoom in of eye region of smoothed image of lena-MSTH. . . . .	113
4.13	Edges of Isotropic and cEED smothing algorithms. . . . .	114
4.14	MSE measure of image1. . . . .	115
4.15	PSNR measure of image1. . . . .	115
4.15	PSNR measure of image1. . . . .	116
4.16	CIRR measure of image1. . . . .	116
4.17	MSE, PSNR, and CIRR measures. . . . .	117
4.18	Number of iterations using JGTS and BF stopping criteria. . . . .	118
4.19	Edges of images obtained using JGTS and BF stopping criteria. . . . .	119
4.20	MSE, PSNR, and CIRR behavior for 3D enhanced-images. . . . .	120
4.20	MSE, PSNR, and CIRR behavior for 3D enhanced-images. . . . .	121
4.21	Number of iterations using JGTS and BF stopping criteria. . . . .	122
4.22	Edges of images obtained using JGTS and BF stopping criteria. . . . .	123
4.22	Edges of images obtained using JGTS and BF stopping criteria. . . . .	124
5.1	Test 2D images. First and third columns - original images, second and fourth columns - gradient magnitude. . . . .	154
5.1	Test 2D images. First and third columns - original images, second and fourth columns - gradient magnitude. . . . .	155
5.2	Curvatures using $\lambda_1 \geq \lambda_2$ as ordering criterion. . . . .	155
5.2	Curvatures using $\lambda_1 \geq \lambda_2$ as ordering criterion. . . . .	156
5.3	Curvatures using $ \lambda_1  \geq  \lambda_2 $ as ordering criterion. . . . .	156
5.3	Curvatures using $ \lambda_1  \geq  \lambda_2 $ as ordering criterion. . . . .	157
5.4	Ordering and discrimination criteria. . . . .	158
5.5	3D synthetic images curvatures. . . . .	159
5.6	Gradients and curvatures of 3D medical images. . . . .	160
5.6	Gradients and curvatures of 3D medical images. . . . .	161
5.7	Zoom in of gradients and curvatures of 3D medical images. . . . .	162
5.7	Zoom in of gradients and curvatures of 3D medical images. . . . .	163
6.1	Pixel coordinates. . . . .	176



---

6.2	Non-maximum suppression results for 2D synthetic image. . . . .	191
6.3	Non-maximum suppression results for Lena's image. . . . .	192
6.3	Non-maximum suppression results for Lena's image. . . . .	193
6.4	Number of pixels before and after applying the NMS algorithms to 2D images.	193
6.5	Non-maximum suppression results for 3D synthetic images. . . . .	194
6.5	Non-maximum suppression results for 3D synthetic images. . . . .	195
6.6	Number of pixels before and after applying the NMS algorithms to 3D synthetic images. . . . .	196
6.7	Non-maximum suppression results for 3D medical images. . . . .	196
6.7	Non-maximum suppression results for 3D medical images. . . . .	197
6.8	Number of pixels before and after applying the NMS algorithms to 3D medical images. . . . .	197
6.9	Hysteresis results for a 2D synthetic image. . . . .	198
6.10	Non-maximum suppression results for Lena's image. . . . .	199
6.11	Number of pixels after applying the hysteresis algorithm to 2D images. . . . .	200
6.12	Non-maximum suppression results for 3D synthetic image. . . . .	201
6.13	Number of pixels after applying the hysteresis algorithm to 3D synthetic images.	201
6.13	Number of pixels after applying the hysteresis algorithm to 3D synthetic images.	202
6.14	Hysteresis results for 3D medical image. . . . .	202
6.15	Number of pixels after applying the hysteresis algorithm to 3D medical images.	203
7.1	Patient's characterization . . . . .	212
7.2	CT Scanner specifications . . . . .	213
7.3	CT scanner configuration parameters . . . . .	214
7.4	Ground truth images . . . . .	216
7.5	Upper airways segmentation of image1 using AC, CV and BF methods. . . . .	220
7.6	Nasal cavity segmentation using AC, CV and BF methods. . . . .	220
7.6	Nasal cavity segmentation using AC, CV and BF methods. . . . .	221
7.7	Upper airways segmentation of image5 using AC, CV and BF methods. . . . .	221
7.8	Evaluation measures results. . . . .	222



# Nomenclature

## Roman Symbols

AASM American Academy of Sleep Medicine

CAT Computed Axial Tomography

CIR Contrast Improvement Ratio

CIR Contrast Improvement Ratio

CSA Central Sleep Apnea

CT Computed tomography

EEG Electroencephalogram

EKG Electrocardiogram

EMG Electromyogram

EOG Electrooculogram

GDP Gross Domestic Product

MRI Magnetic Resonance Imaging

MSE Mean Squared Error

MSE Mean Squared Error

OSA Obstructive Sleep Apnea

PDE Partial Differential Equations

PSG Polysomnogram

PSNR Peak Signal-to-Noise Ratio

PSNR Peak Signal-to-Noise Ratio

RG Retroglossal

ROI Region of Interest

RP Retropalatal

SE Structuring element

SNR Signal Noise Ratio

URAO Upper respiratory airways obstruction

# Chapter 1

## Introduction

“You never change things by fighting the existing reality. To change something, build a new model that makes the existing model obsolete.”

– *Buckminster Fuller.*

Medicine area is being revolutionized thanks to the rapid development and proliferation of medical imaging technologies. Medical imaging allows scientists and physicians to acquire potentially life-saving information by looking non-invasively into the human body. The role of medical imaging has expanded beyond the simple visualization and inspection of anatomic structures. It has become a tool for surgical planning and simulation, intra-operative navigation, radiotherapy planning, and for tracking the progress of diseases. The analysis or study of medical imaging has depended mainly on the interpretative capacity of doctors and radiologists, but this task is very time consuming and require a great experience of specialists to reduce the subjectivity in diagnosis[13–15].

In our case, the upper airways segmentation is of significant importance because one of the most prevalent chronic disorders is airway disease. It is a major cause of morbidity and mortality worldwide [16, 17]. In order to understand its underlying mechanisms and to enable assessment of therapeutic efficacy of a variety of possible interventions, noninvasive investigation of the airways in a large number of subjects is of great research interest. Due to its high resolution in temporal and spatial domains, Computed Axial Tomography (CAT or CT) has been widely used in clinical practices for studying the normal and abnormal manifestations of airways diseases[18]. For example, evaluation of the upper respiratory airway obstruction (URAO) is of significant importance, as its physiological effects on ventilation rapidly evolve to secondary body malfunctions. Given the nature and location of the lesions, the invasive handling of the condition with direct bronchoscopy is undesirable; an imaging approach based on CT is preferred. Objective and quantitative evaluation of the obstruction therefore requires the

application of image processing and analysis techniques[19, 20].

## 1.1 Motivation

Sleep disorders are a widely recognized consequence of many neurological pathologies, however, the significance of sleep disorders is underestimated in our society. There are approximately 90 different types of sleep disorders ranging from insomnia to more serious conditions such as restless legs syndrome, narcolepsy and sleep apnea [21, 22]. And the incidence and prevalence of sleep disorders are very high compared to other medical illnesses.

Sleep apnea is a condition characterized by the episodic cessation of breathing during sleep. This generates several complications in the health condition of the patients (see Figure 1.1). Obstructive sleep apnea syndrome (OSA) is the most common sleep apnea, affecting 4% of adults [22, 23], and is secondary to sleep-induced obstruction of the upper airways combined with simultaneous respiratory efforts. OSA results in reduced oxygen saturation which may give rise to hypertension, cardiac arrhythmia, nocturnal angina, and myocardial ischemia. OSA also results in impaired sleep quality leading to excessive daytime sleepiness, deterioration of memory and judgment, altered personality, and reduced concentration[23, 24].

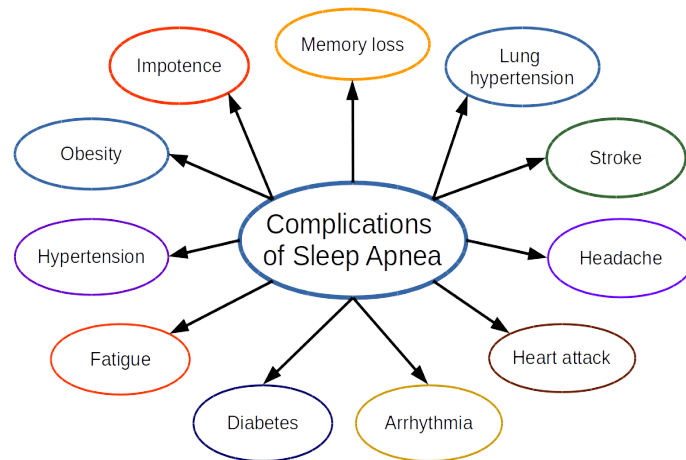


Figure 1.1: Complications generated by sleep apnea disorders.

Among patients with OSA, the sites of obstruction and the narrowing of the upper airways can differ greatly. The retropalatal region, an area posterior to the soft palate (RP), and retroglossal region, an area posterior to the base of tongue (RG), are commonly restricted sites [25]; however, multiple sites of obstruction and narrowing are not uncommon [23].

The consequences of sleep disorders such as medical illness, surgical illness, and accidents may cause serious burden not only in individual health but also in society both medically and

socioeconomically [24, 26]. By reviewing economical costs of sleep disorders, it is possible to see the seriousness of this condition (see Table 1.1). In addition, Hillman et. al. present an analysis of direct and indirect costs of sleep disorders and the fractions of other health impacts attributable to sleep disorders. They include in their study direct health costs of sleep disorders (principally, obstructive sleep apnea, insomnia, and periodic limb movement disorder) and of associated conditions. Also, they are considered indirect financial costs of associated work-related accidents, motor vehicle accidents, and other productivity losses [26].

Table 1.1 presents a consolidation of the costs presented by Hafner et. al. [9] in their comparative analysis of several countries (United States, United Kingdom, Japan, Germany and Canada). Table 1.1 shows the great economic impact associated with sleep disorders using the gross domestic product (GDP<sup>1</sup>) as a reference measure.

Country/Year	2015)	2020	2025	2030	Cumulative costs 2030
<b>U.S.</b>	411.0	433.8	456.1	467.7	<b>7030.7</b>
<b>UK</b>	50.2	53.8	57.6	58.7	<b>877.4</b>
<b>Japan</b>	138.6	145.9	151.7	156.2	<b>2357.2</b>
<b>Germany</b>	60.0	62.3	64.7	69.1	<b>1021.4</b>
<b>Canada</b>	21.4	21.9	22.5	23.4	<b>355.1</b>
<b>Total</b>	<b>681.2</b>	<b>717.7</b>	<b>752.6</b>	<b>775.1</b>	<b>11641.8</b>

Table 1.1: Estimated annual costs in GPD (U.S. \$billions) (Adapted from [9])

Based on the medical and socioeconomic impact that is generated by sleep disorders, it is important to be able to make a diagnosis to detect and treat these disorders. Normally, doctors diagnose sleep apnea based on medical and family histories, a physical exam, and sleep study results. In the physical exam, the doctor checks mouth, nose, and throat for extra or large tissues. Children who have sleep apnea might have enlarged tonsils (Doctors may need only a physical exam and medical history to diagnose sleep apnea in children). Adults who have sleep apnea may have an enlarged uvula or soft palate. The uvula is the tissue that hangs from the middle of the back of the mouth. The soft palate is the roof of the mouth in the back of the throat.

When the medical and family history and the physical exam are not enough, sleep studies are required. Sleep studies are tests that measure how well the patient sleep and how his body responds to sleep problems. These tests can help the doctor to find out whether the patient has a sleep disorder and how severe it is. Sleep studies are the most accurate tests for diagnosing

<sup>1</sup>GDP of a country provides a measure of the monetary value of the goods and services it produces in a specific year.

sleep apnea. There are different kinds of sleep studies. If the doctors think the patient has sleep apnea, they may recommend a polysomnogram (poly-SOM-no-gram; also called a PSG) or a home-based portable monitor.

The American Academy of Sleep Medicine (AASM) provides standards and guidelines for diagnostic polysomnography that include the following:

- Sleep stages are recorded via an electrooculogram (EOG), electroencephalogram (EEG), and electromyogram (EMG).
- Heart rhythm is monitored with a single-lead electrocardiogram (EKG).
- Leg movements are recorded via an anterior tibialis electromyogram.
- Breathing is monitored, including airflow at the nose and mouth (using both a thermal sensor and a nasal pressure transducer), effort (using inductance plethysmography), and oxygen saturation.
- The breathing pattern is analyzed for the presence of apneas and hypopneas.

Figure 1.2 shows a superimposed recordings of the electrooculogram (EOG), electroencephalogram (EEG), electromyogram (EMG), ECG (EKG), sympathetic nerve activity (SNA), respiration (RESP), and blood pressure (BP) during rapid eye movement (REM) sleep in a patient with OSA [1]. The regions marked in the data obtained for EMG and RESP show the intervals in which obstructions occurred and therefore low activity in the sampled signal.

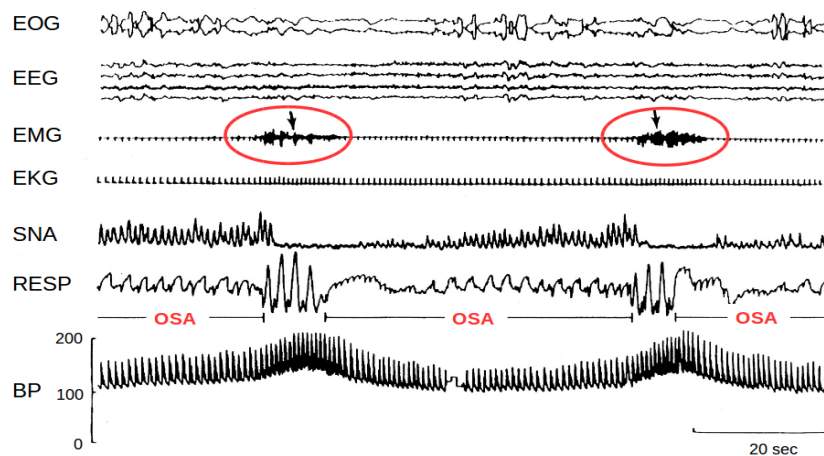


Figure 1.2: Superimposed recordings data in a patient with OSA (adapted from [1]).

The home-based portable monitor is a home-based sleep test with a portable monitor. The portable monitor records some of the same information as a PSG. For example, it may record:



The amount of oxygen in the blood, air movement through the nose while the patient breathe, heart rate, chest movements that show whether patient is making an effort to breathe. The sleep specialists may use the results from a home-based sleep test to help diagnose sleep apnea. They also may use the results to decide whether you need a full PSG study in a sleep centre.

Although polysomnography (PSG) is the method used to confirm the diagnosis of OSA, CT and MRI images have established themselves as important supporting methods in the clinical diagnosis, preoperative evaluation, and post-treatment follow-up of patients who do not respond well to initial therapy. Both CT and MRI images can provide an excellent evaluation of the various anatomical planes of the obstruction site, which enables better clinical assessment as well as better planning for a possible surgical approach [27]. For this reason and considering that patients with sleep-disordered breathing often have physiologic and anatomic abnormalities of the upper airways that are demonstrable while awake, the use of CT images is an excellent tool to identify the anatomic features that predispose patients to these kind of sleep disorders.

In the initial studies in which airway dimensions were measured using CT, the researchers relied on manual tracing of the airway images. These techniques are extremely time consuming and prone to error. Therefore, computer-aided and automated techniques have since been developed to measure airway lumen and wall dimensions. In many airway diseases, the important site of airflow obstruction is the small airways [28, 29]. It has been reported that airway lumens as small as a 0.5mm diameter can be measured using CT [30], but, there are large errors associated with the measurements of small airways when the data are obtained using routine clinical scanning parameters. While many methods have been proposed for the segmentation of the lower airways from CT images [31–45], the segmentation of the nasal cavity and paranasal sinuses has not been addressed. Similarly, no one has proposed a method for nasal passage segmentation from MR images.

Medical image segmentation algorithms often face difficult challenges such as poor image contrast, noise, and missing or diffuse boundaries [46]. For example, tissue boundaries in medical images may be smeared (due to patient movements), missing (due to low Signal Noise Ratio (SNR) of the acquisition apparatus), or nonexistence (when blended with similar surrounding tissues). Under such conditions, without a prior model to constrain the segmentation, most algorithms (including intensity and curve-based techniques) fail—mostly due to the under-determined nature of the segmentation process. Similar problems arise in other imaging applications as well and they also hinder the segmentation of the image. These image segmentation problems demand the incorporation of as much prior information as possible to help the segmentation algorithms to extract the tissue of interest.

Based on the above, several tests were performed to segment the CT images to obtain the upper airways using well known algorithms such as thresholding, region growing, fast marching (FM), Chan and Vese (CV), and Level set. For example, in Figure 1.3 the result obtained using fast marching algorithm is shown. As can be seen in the left image (axial view) and in the right image (coronal view), in thin regions (thin tissue) or narrow passages the segmentation algorithm does not propagate generating an under-segmentation.

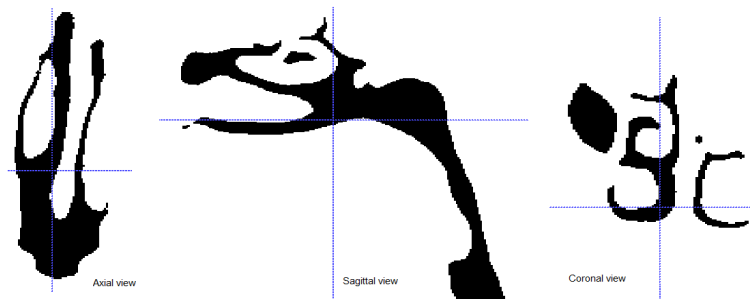


Figure 1.3: Upper airways segmented using FM.

In addition, Figure 1.4 presents an example, which is the result of applying the segmentation algorithm proposed by Chan-Vese. The image on the left (axial view) allows to see in the top part an over-segmentation generated by the algorithm.

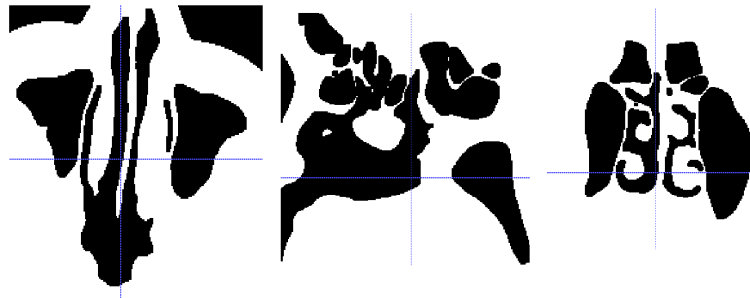


Figure 1.4: Upper airways segmented using CV.

In conclusion, the segmentation obtained is not complete in the region of interest (upper airways), due to the conditions already mentioned as the presence of low contrast and the existence of regions with thin tissue, which are difficult to overcome using traditional techniques. Because of this, the challenge in this research is overcoming the difficulties described, using principal curvatures, in order to segment the upper airways in a more accurately way from CT medical images.

## 1.2 Objective

The aim of this thesis work is to develop an image segmentation algorithm using principal curvatures to segment upper airways from CT images.

To meet this objective the revision of important issues is required. First, contrast enhancement and smoothing of CT images is reviewed.

- Contrast in a CT image is determined by the differential absorption of X-rays by neighboring structures. Problem of automatic upper airways (nose, nasal cavity, paranasal sinuses, nasopharynx and pharynx) segmentation is hard, because low contrast exists between the interior of the airways and its surrounding tissue making very difficult the segmentation process[47, 48].
- In a medical setting, it is often the case that the acquired images suffer from noise, which may ultimately affect the accuracy of analysis and diagnosis, therefore, image smoothing is of critical interest.

Choosing an appropriate method for contrast enhancement or smoothing is not straightforward, due to lack of dependable measures that quantify the quality of the output image.

Second, since the segmentation algorithm is based on principal curvatures, it is necessary to review the selection criteria, the detection and its corresponding extraction.

- Principal curvatures are part of the feature extraction of an image. Therefore, its definition and the computational forms to calculate them must be reviewed.
- According to several authors, principal curvatures allow defining the contour of a surface or hypersurface. Then, the detection and extraction of the curvatures can be considered as a problem of edge detection.

Finally, to compare the results obtained with the proposed algorithm, it is necessary to have a set of reference images. Therefore, the project includes the manual or semi-automatic segmentation of five images in order to have a reference measurement.

- Evaluation of segmentation methods has been problematic because, manual segmentation of airways is a difficult and very time consuming task due to the complexity of the 3D structure of the upper airways. Without a prior model to constrain the segmentation, most algorithms (including intensity and edge based techniques) will fail. Similar problems arise in other imaging applications as well, hindering the segmentation of the image.

### 1.3 Proposed solution

To fulfill the objective of this thesis, a general solution was defined, which is presented in Figure 1.5. As you can see, the solution is composed of four blocks that allows to segment the CT images to obtain the upper airways. Each block is marked according to the user intervention level as automatic (without intervention) or semi-automatic (some parameters entered by the user are required). The first corresponds to the image enhancement and smoothing that permits to reduce the noise and increase the contrast. The second deals with the feature detection with emphasis on the principal curvatures and the shapes classification that can be identified (peaks, valleys, saddles, ridges, etc.). The third block allows to extract the edges that belong to the surface or hypersurface, for this, techniques based on edges and ridges are used. Finally, the fourth block corresponds to the segmentation based on the edges obtained by means of the previous steps.

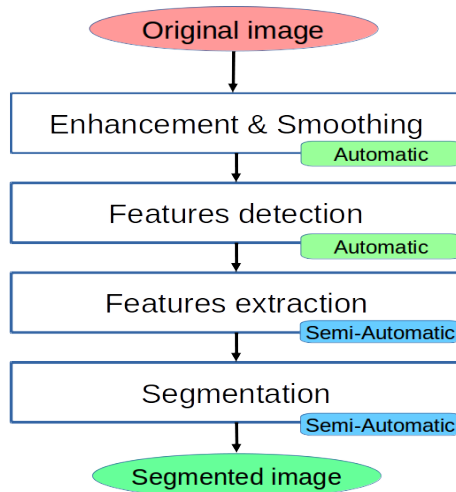


Figure 1.5: Main structure of proposed solution.

The most important algorithms that make up the proposed solution are presented in Figure 1.6. The first corresponds to the algorithm for contrast enhancement based on morphological operators. Specifically, a multi-scale top-hat algorithm is used. The biggest drawback with multi-scale techniques is the definition of the number of iterations. Therefore, a stopping criterion is proposed based on the Contrast Improvement Ratio (CIR) metric. An improvement to the CIR metric to include negative values was made, which is called (CIRR). Subsequently, the stopping criterion called  $CIRR_{index}$  was defined.

The second algorithm corresponds to the image smoothing. Tests were performed with linear and non-linear algorithms, and algorithms based on partial derivative equations (PDEs), specifically diffusion equations, were tested. The non-linear anisotropic diffusion algorithm

proposed by Mirebeau was selected. As it is a diffusion algorithm, it also presents difficulties to determine the number of iterations. Based on this, there are some solutions (stopping criteria) that are included directly in the diffusion equation, however there are some proposals that do not depend on the diffusion equation, for example, Joao, Gambaruto, Tiago, and Sequeira (JGTS) proposed a criterion based on Mean Square Error (MSE). From the review of the state of the art and the tests performed with the stopping criterion used in the contrast enhancement, it was identified that it could also be applied to the anisotropic diffusion.

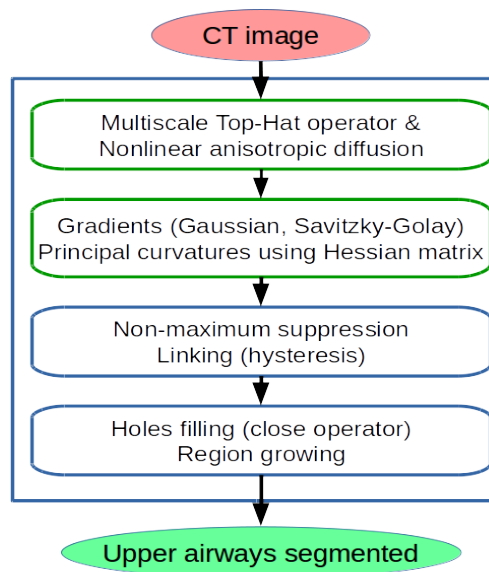


Figure 1.6: Main algorithms included in the proposed solution.

The third algorithm allows to compute the gradient of the 3D images. The 3D version of interpolation algorithm proposed by Savitzky-Golay was implemented. In addition, the derivative Gaussian filter proposed by Deriche was used. Tests were carried out with the two implementations and it was found that Savitzky-Golay generates less attenuation of the theoretical values of the gradient.

The fourth algorithm allows determining the principal curvatures using the Hessian matrix. A criterion is proposed that allows to select the ridge points of the surface. This criterion was compared against that proposed by Steger and that proposed in the PCBR algorithm. The results show that the proposed criterion allows to obtain greater continuity in the surface of the curvilinear structures. The fifth algorithm corresponds to the non-maximum suppression. It is proposed to include more information of each point so as not to generate gaps in the objects surface. The results were compared with the non-maximum suppression algorithm proposed by Canny. The results at first glance are crude, but they allow to maintain the surface without many gaps. This is due to the inclusion of the information of the normal direction and

the direction of the plane tangent to the surface.

The sixth algorithm corresponds to the link of the points on the surface. The procedure is similar to the one proposed in the PCBR algorithm, including a distance criterion between the points found as belonging to the surface. This allows to reduce the presence of gaps and therefore of holes in the surface. The disadvantage is that it does not generate contours of width equal to a voxel.

Finally, to fill in the gaps that may appear on the surface, a morphological closing operator is used. Based on the contours, a region growing algorithm is used to extract the objects of interest.

## 1.4 Contributions

The main contributions of this thesis are:

1. A new stopping criterion is proposed to multi-scale top-hat contrast enhancement.
2. The stopping criterion is used in image smoothing that use isotropic or anisotropic diffusion.
3. A new criterion to detect edges or ridges is proposed.
4. A new non-maximum suppression criterion is proposed.
5. One modification to linking algorithm proposed by Steger to detect edges is proposed.
6. An automatic implementation to compute derivatives using Savitzky-Golay is implemented.
7. A new algorithm to segment curvilinear structures is proposed.

## 1.5 Thesis outline

The main contents of this thesis are presented in six chapters, as shown in Figure 1.7. In each chapter is included the state of the art that support the proposed algorithm or the improvement of the selected algorithm, then, it presents the method used and the tests that allow to see its advantages.

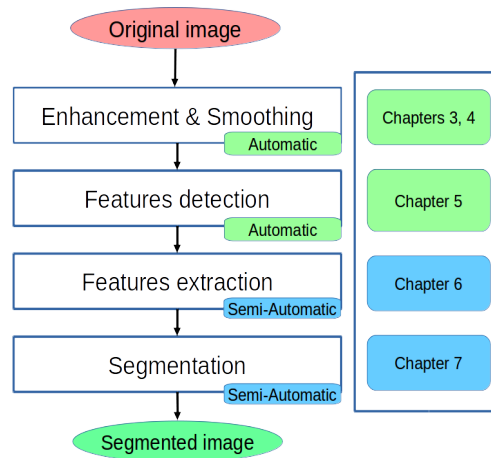


Figure 1.7: Thesis outline according to the proposed solution.

**Chapter 2** provides a background description on upper airways such as anatomy, physiology and common diseases. Then, a brief description about image segmentation techniques are presented. Finally, a basic concepts about differential geometry such as surfaces and curvatures are reviewed.

**Chapter 3** provides a description of state of the art about contrast enhancement, mainly morphological operators. First, the improvement of contrast is described using multi-scale morphological operators in gray-level images (head-neck CT images). Second, an improvement in the Contrast Improvement Ratio (CIR) criterion called Contrast Improvement Ratio Revisited (CIRR) is proposed. Third, the proposed stopping criterion for the multi-scale Top-Hat algorithm is presented. Finally, the experiments and their results are presented for both 2D and 3D images.

**Chapter 4** provides a description of the state of the art about smoothing using partial derivative equations (PDEs) primarily. The linear and nonlinear diffusion techniques are presented, both isotropic and anisotropic. Then the selected diffusion algorithms are presented, as well as the stopping criterion that is not associated with the diffusion equation but is independent of the algorithms. Finally, the tests for each of the algorithms are performed and the best one is selected to be applied to the images resulting from the contrast enhancement. The tests are applied to 2D and 3D images.

**Chapter 5** presents a description of the state of the art about features and shapes detection, this includes detection of corners, edges, blobs and shapes. Subsequently, the proposed method to select the main characteristics using the principal curvatures is presented. Finally, experiments are carried out using 2D and 3D images to show the improvement in the detection of edges or ridges.

**Chapter 6** presents a description of the state of the art on the features extraction, includ-

ing the most relevant algorithms. Then two important elements are presented for features extraction such as the non-maximum suppression and the linking. Subsequently, the proposed method for the non-maximum suppression is presented and compared with traditional techniques. In addition, the proposed linking algorithm is presented. Finally, the tests carried out in 2D and 3D images that demonstrate the advantages of the proposed solution are presented.

**Chapter 7** presents a brief description of the state of the art on segmentation methods with emphasis on upper airways. Then the proposed method to extract the region from the upper airways is presented. Finally, tests with 3D images are realized and the results are compared with previously segmented images.



# Chapter 2

## Preliminar concepts

“[The golden proportion] is a scale of proportions which makes the bad difficult [to produce] and the good easy.”  
– *Albert Einstein.*

In this chapter are presented the concepts related to upper airways physiology and the main diseases present in this area of the body. Then, a brief description of image segmentation techniques is made. Finally, the most relevant concepts about surfaces and curvatures are presented.

### 2.1 Upper airways

Principal organs of the respiratory system are the two lungs, which are in the right and left sides of the chest (thoracic cavity) and are separated from each other by the heart. Air passes into and out of the lungs through a series of cavities, tubes, openings and passages called the upper airways[49]. The flow of air depends on other organs, including the muscular diaphragm and the muscles and bones that make up the wall of the thoracic cavity. Part of each lung consists of tubes called the lower airways; they end in the microscopic saclike alveoli, which make up most of the lungs. The lower airways transport air to and from the alveoli [2, 50].

The respiratory system is comprised of several elements including the central nervous system, the chest wall, the pulmonary circulation, and the respiratory tract. For example, many pulmonary vessels transport blood throughout the lungs. The main functions of the respiratory system include:

1. External respiration: exchange of gases (oxygen and carbon dioxide) between air and blood.

2. Internal respiration: exchange of gases between blood and tissue fluid.
3. Transport of gases to and from the lungs and the tissues.

The respiratory tract can be divided into four distinct segments [51]: the naso-oropharynx (*upper respiratory tract or upper airways*), the conducting airways (*lower respiratory tract or lower airways*), the respiratory bronchioles, and the alveoli. In general, upper airways include all structures located above the trachea, while lower airways are below the trachea and into the lungs. In this case, the segment that is relevant in the segmentation process corresponds to the upper respiratory tract.

### 2.1.1 Upper airways anatomy and physiology

The upper airway starts at the nostrils, extends through the nasal conchae to the nasopharynx, over the uvula to the hypopharynx and larynx, or, at the lips, extends through the oral cavity, over the tongue and below the hard and soft palates, to the hypopharynx and larynx. Figure 2.1 shows the structures that conform the upper airways.

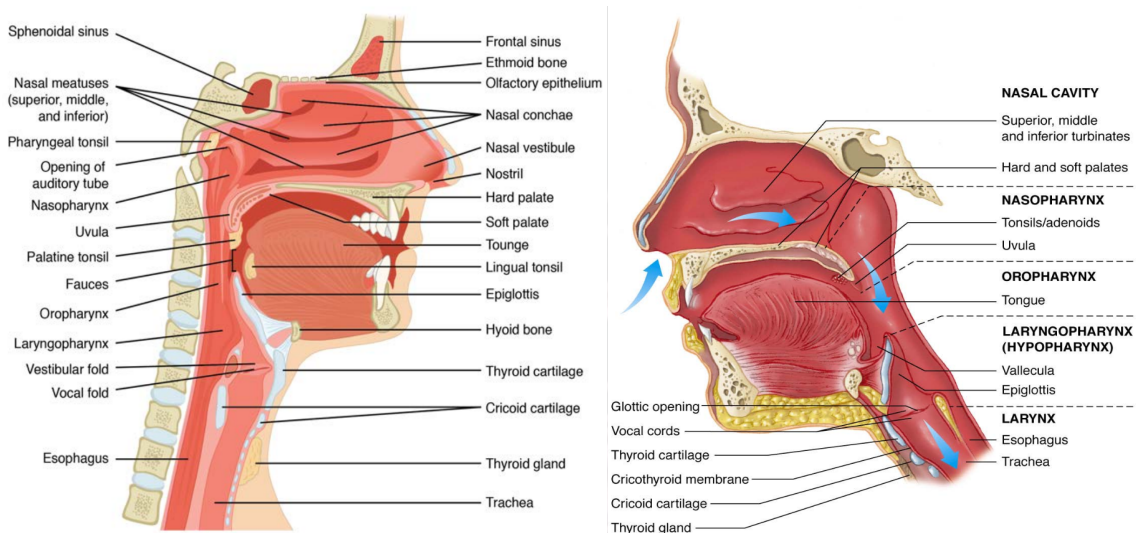


Figure 2.1: Upper Airways. Left image shows main structures and right image shows main divisions (taken from [2]).

The upper airways main functions are warm, filter, and humidify air. According with the literature and the summary presented in table 2.1, upper airways can be separated in five sections as is illustrated in Figure 2.1: Nasal Cavity, Nasopharynx, Oropharynx, Laryngopharynx and Larynx. In this section the main structures without consider special order or importance level are described.

<b>Structure</b>	<b>Description</b>	<b>Function</b>
Nose	Hollow spaces in nose. Supported by bone and cartilage	provides an entrance for air in which air is filtered by coarse hairs inside the nostrils.
Nasal Cavity	Space posterior to the nose that is divided medially by the nasal septum.	
Paranasal Sinuses	Sinuses are air-filled spaces within the maxillary, frontal, ethmoid, and sphenoid bones of the skull.	Open to the nasal cavity and are lined with mucus membrane that is continuous with that lining the nasal cavity. Reduce the weight of the skull and serve as a resonant chamber to affect the quality of the voice.
Oral Cavity	Hollow spaces in mouth	
Pharynx (behind the tongue)	Chamber posterior to oral cavity; lies between nasal cavity and larynx	Connection to surrounding regions. Common passageway for air and food. Aids in producing sounds for speech.
Glottis	Opening into larynx	Passage of air into larynx
Larynx (voice box)	Cartilaginous organ that houses the vocal cords (voice box). Is composed of a framework of muscles and cartilage bound by elastic tissue.	Sound production. Helps keep particles from entering the trachea and also houses the vocal cords.
Trachea (windpipe)	Flexible tube that connects larynx with bronchi.	Passage of air to bronchi

Table 2.1: Upper airways structures.

### 2.1.1.1 Nasal cavity

Nasal cavity is the space between the roof of the mouth and the cranial base, divided vertically in the middle by the nasal septum. Each cavity is wide caudally, and narrow cranially. The roof of the nasal cavity is thin cribriform plate (0.5 mm) and is formed by the nasal spine of the frontal bones, nasal bones, cribriform plate of the ethmoid bone, and anterior body of the sphenoid bone. The floor of the nasal cavity is the hard palate, formed by the palatine processes of the maxillae and the horizontal processes of the palatine bones. The lateral walls of the nasal cavities are formed by the maxilla, palatine, and ethmoid bones. The lateral nasal wall contains the maxillary and ethmoid ostia, plus three or four turbinates. The lateral walls of the nasal cavity also contain three projections called the inferior, middle, and superior nasal turbinates, which separate the nasal cavity into four air chambers, namely the inferior nasal meatus, middle nasal meatus, superior nasal meatus, and the spheno-ethmoidal recess (see Figures 2.2 and 2.3)[18, 52].

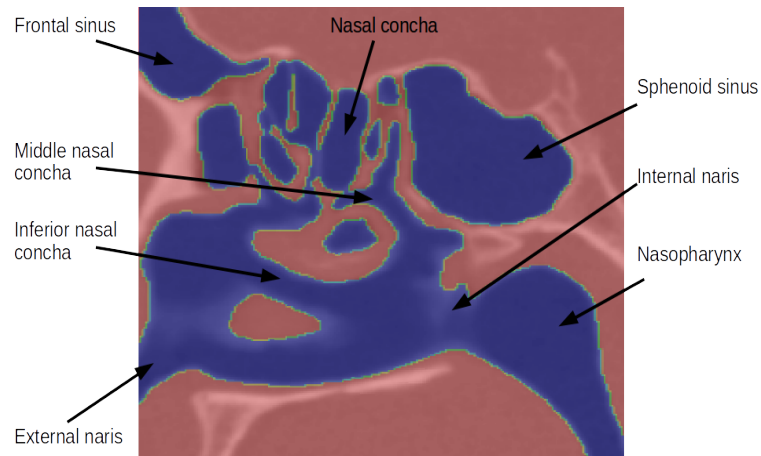


Figure 2.2: Nasal cavities.

### 2.1.1.2 Nasal septum

The nasal septum divides the nasal cavity into two separate compartments (see Figure 2.3), increasing the total mucosal surface area. It consists of an anterior cartilaginous portion, which provides support for the nasal tip, and a posterior bony portion formed by the perpendicular plate of the ethmoid and the vomer[18, 53].

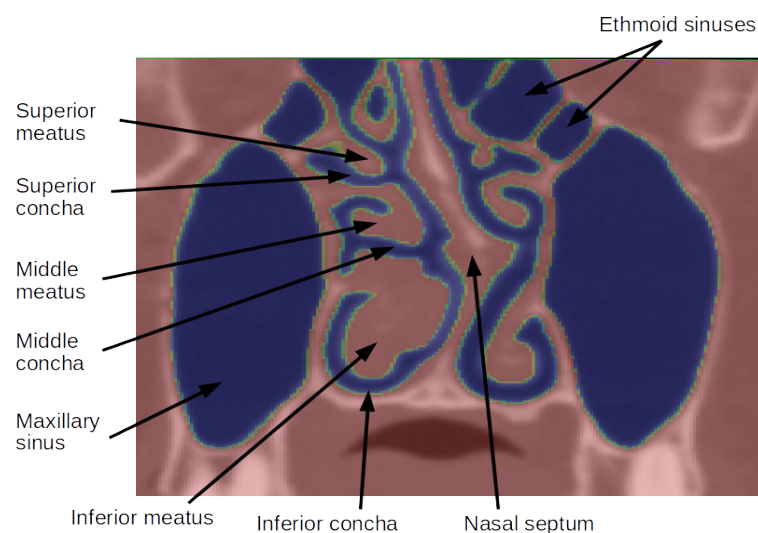


Figure 2.3: External nose and nasal septum.

The major portion of the nasal septum is formed by the perpendicular plate of the ethmoid bone posteriorly and the septal cartilage anteriorly. The vomer completes the posteroinferior portion of the septum. The septum is lined by relatively thin, ciliated respiratory mucosa,

which may regularly undergo squamous metaplasia. The underlying thin lamina propria, although containing seromucinous glands, is tethered to the septal cartilage, restricting reactive polyp formation[51, 54].

Structural deviations of the septum lead to internal nasal asymmetry, which in turn causes compensatory change in turbinate morphology, resulting in changes in nasal airway resistance [55]. The percentage of nasal septal deformities changes with age. A multinational study has shown that septal deformities are present in approximately 90% of adult patients. A straight septum is twice as frequent in females than in males [56].

In addition, small anterior deviations at the level of the nasal valve can lead to significant nasal airway obstruction, whereas large deviations in the posterior portion of the nasal cavity may have no effect on airflow resistance. Also, weakening or collapse of the septal cartilage that results from a septal abscess, surgical intervention, or Wegener granulomatosis leads to loss of nasal tip support compromising the nasal valve and nasal airflow [56].

### 2.1.1.3 Nose

This part of upper airways is above the hard palate and contains the peripheral organ of smell. Some authors indicated that nose includes the external nose and nasal cavities, in our case is considered how another part of upper airways. For this reason, nose is divided into right and left cavities by the nasal septum (see Figures 2.4 and 2.3). The functions of the nose and nasal cavities are: olfaction (the sense of smell), respiration (breathing), filtration of dust, heating and humidification of inspired (inhaled) air, and finally, reception and elimination of secretions from the nasal mucosa, paranasal sinuses, and nasolacrimal ducts[54].

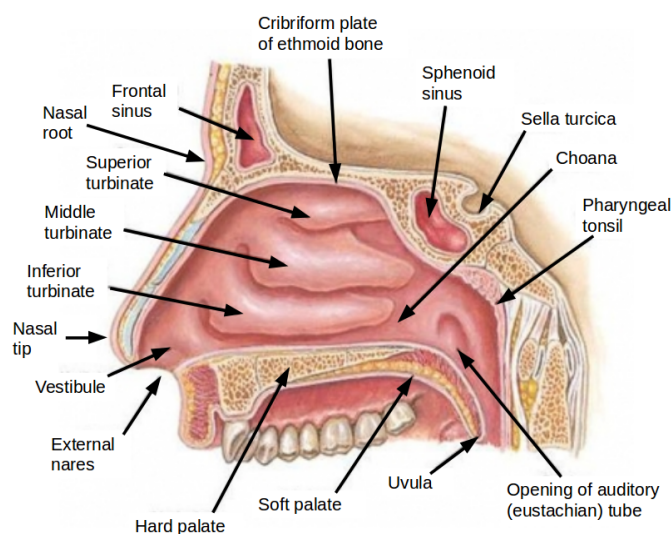


Figure 2.4: Nose anatomy (adapted from <http://physiotherapy.org.nz>).

Some nose anatomical structures are defined as follow as these are not always used with standard meaning.

**Nasal Root:** The most depressed, superior part of the nose along the nasal ridge. **Nasion:** The midline point just superior to the nasal root overlying the naso-frontal suture. **Nasal Bridge:** A saddle-shaped area that includes the nasal root and the lateral aspects of the nose. It lies between the glabella and the inferior boundary of the nasal bone, and extends laterally to the inner canthi. **Nasal Ridge:** The midline prominence of the nose, extending from the nasal root to the tip (also called the dorsum of the nose). **Nasal Base:** An imaginary line between the most lateral points of the external inferior attachments of the alae nasi to the face. **Nasal Tip:** The junction of the inferior margin of the nasal ridge and the columella. Commonly, it is the part of the nose furthest from the plane of the face. In rare circumstances, such as markedly prominent and convex nasal profiles, other parts of the ridge may be further removed from the facial plane. **Ala:** The tissue comprising the lateral boundary of the nose, inferiorly, surrounding the naris. **Columella:** The tissue that links the nasal tip to the nasal base, and separates the nares. It is the inferior margin of the nasal septum[2, 51, 54]. Additionally, information from NIH is used <sup>1</sup>.

#### 2.1.1.4 External nose

The external nose is a pyramidal structure, situated in the midface. It presents a root (or bridge), a dorsum, and a free tip or apex (see Figure 2.4). The two inferior openings are the nostrils (or nares), bounded laterally by the ala and medially by the nasal septum. The superior part of the nose is supported by the nasal, frontal, and maxillary bones; the inferior part includes several cartilages. The continuous free margin of the nasal bones and maxillae in a dried skull is termed the piriform aperture[54].

The external nose varies considerably in size and shape, mainly because of differences in the nasal cartilages. The dorsum of the nose extends from its superior angle, the root (see Figure 2.4), to the apex (tip) of the nose[2, 53].

#### 2.1.1.5 Paranasal sinuses

Paranasal sinuses are air filled hollow sacs seen around the skull bone. These sacs precisely surround the nasal cavity. Figure 2.5 shows four paired sinuses: Maxillary, Frontal, Ethmoidal and Sphenoidal. Each of these sinuses are named according to the facial bones in which it is located. The sinuses are located within the bones of the face and around the nasal cavity and

---

<sup>1</sup>National Human Genoma Research Institute (<https://elementsofmorphology.nih.gov>)

eyes. Normal sinus anatomy is complex, and can be difficult to appreciate on static images alone [52, 53].

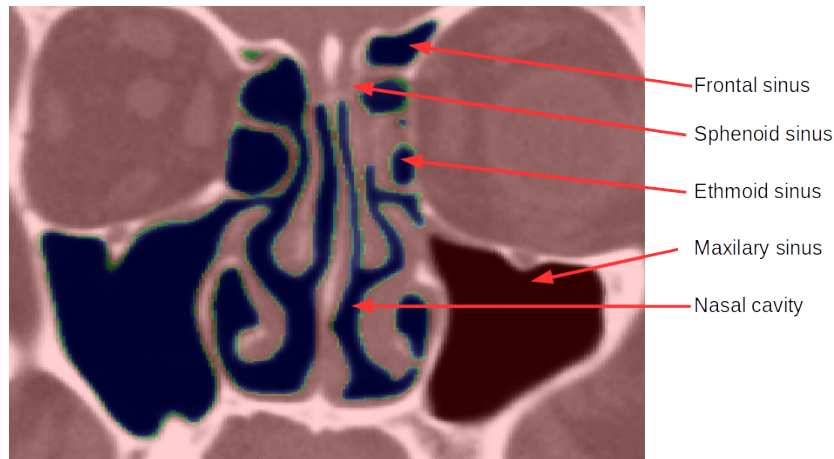


Figure 2.5: Paranasal sinuses.

**Frontal Sinus** These paired sinuses reside between the internal and external cranial tables and drain either via a nasofrontal duct into the frontal recess or more directly into the anterior infundibulum, or less often, into the anterior ethmoid cells, which in turn will open into the infundibulum of the bulla ethmoidalis (see Figures 2.5 and 2.4) [18, 52, 54]. The frontal sinus is the second largest sinuses with 2 – 2.5 cm.

**Ethmoid Sinus** This paired complex of sinuses contains 3-18 cells that are grouped as anterior, middle, or posterior, according to the location of their ostia (see figure 2.5). There is an inverse relationship between the number and size of the cells. Generally, the posterior cells are both larger and fewer than the anterior cells. Each ethmoid labyrinth lies between the orbit and the upper nasal fossa. The left and right groups of ethmoid cells are connected in the midline by the cribriform plate (nasal roof) of the ethmoid bone. The cribriform plate is an important landmark in evaluation of sinonasal tumour stage-violation of the cribriform plate signifies direct extension of the tumour into the anterior cranial fossa. The crista galli is a distinctive pointed bony landmark that extends from the midline of the cribriform plate upward into the floor of the anterior cranial fossa. The perpendicular plate of the ethmoid bone extends downward from the cribriform plate to contribute to the nasal septum. The medial wall of each ethmoid labyrinth is formed by a thin lamella of bone from which arise the middle, superior, and supreme turbinates. The lateral ethmoid wall is formed by the thin lamina papyracea, which separates the ethmoid cells from the orbit. The roof of the ethmoid complex is formed by a medial extension of the orbital plate of the frontal bone, which projects to articulate with



the cribriform plate. This is often referred to as the fovea ethmoidalis[18, 52, 54].

**Sphenoid Sinus** The average adult sphenoid sinus measures 20 mm high, 23 mm long, and 17 mm wide. The relationship of the posterior extension of the sphenoid in relation to the sella turcica is variable. The sphenoid sinus septum is usually in the midline, and anteriorly aligned with the nasal septum (see Figure 2.5). However, it can also deviate far to one side creating two unequal sinus cavities. With the exception of the sinus roof, the other sinus walls are of variable thickness depending on the degree of pneumatization. The sphenoid roof is thin, often measuring only 1 mm. (planum sphenoidale), and is vulnerable to perforation during surgery. The sinus roof relates to the floor of the anterior cranial fossa, anteriorly; the optic chiasm and the sella turcica, posteriorly. The lateral sphenoid wall is related to the orbital apex, the optic canal, the optic nerve, and the cavernous sinus, containing the internal carotid artery. The sinus floor is the roof of the nasopharynx, and the anterior sinus wall is the back of the nasal fossa[18, 52, 54].

**Maxillary Sinus** The maxillary sinus lies within the body of the maxillary bone. Behind the orbital rims, each sinus roof/orbital floor slants obliquely upward so that the highest point of the sinus is in the posteromedial portion, lying directly beneath the orbital apex (see Figure 2.5). The medial antral wall is the inferior lateral wall of the nasal cavity (“party wall”). The curved posterolateral wall separates the sinus from the infratemporal fossa. The anterior sinus wall is the facial surface of the maxilla that is perforated by the infraorbital foramen below the orbital rim. The floor of the sinus is lowest near the second premolar and first molar teeth and usually lies 3-5 mm below the nasal floor. The lower expansion of the antrum is intimately related to dentition. The location of the maxillary sinus ostia, is high on the medial wall. They drain through the ethmoidal infundibulum and then the nasal fossa. This pattern of drainage in the erect position is accomplished by intact ciliary action. The maxillary hiatus is a bony window leading to the interior of the maxillary sinus. The hiatus is normally partially covered by portions of four bones: the perpendicular plate of the palatine bone, posteriorly; the lacrimal bone, anterosuperiorly; the inferior turbinate, inferiorly, and above the turbinate attachment, the uncinat process of the ethmoid bone[18, 52, 54]. The maxillary sinus is the largest sinuses with 3.5 cm high and 2.5 – 3 cm wide.

#### 2.1.1.6 Turbinates

The nasal turbinates serve to increase the surface area within the nasal mucosa to facilitate moistening of inspired air. The turbinates are three, rarely four, scroll-like projections from the lateral nasal wall (see Figure 2.6).



The lower two, referred to as the inferior and middle turbinates, are functionally the most significant. Each turbinate consists of a bony frame with overlying respiratory epithelium. These turbinates interrupt the flow of air into the nasal passage, forcing it through narrow passages that are covered with moist nasal respiratory mucosa. The total surface area available in the nasal mucosa is estimated to be about  $180\text{ cm}^2$ , of which  $10\text{ cm}^2$  is olfactory mucosa and  $170\text{ cm}^2$  is the richly vascularized respiratory mucosa[18, 53, 54].

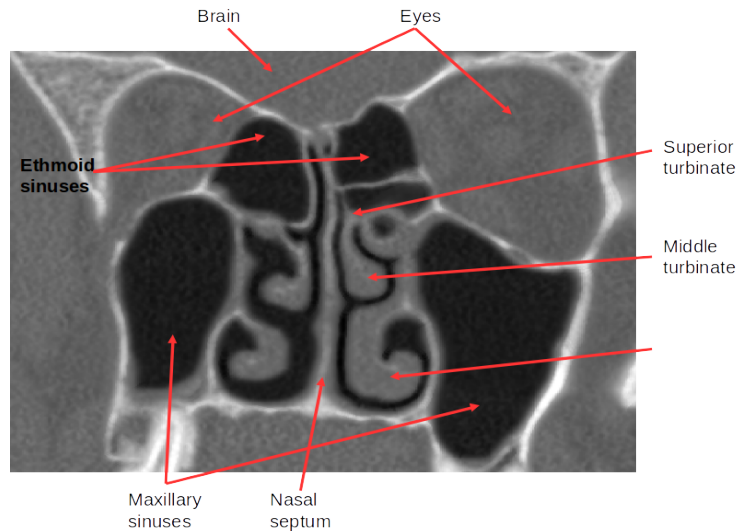


Figure 2.6: Tubinates cross section.

The inferior turbinate has an important role in the defense of the lungs and the physiology of the nose. Trimming of the anterior portion of the inferior turbinate can lead to a decrease in the total nasal resistance to airflow [54] by enlarging the nasal valve, but this should be considered only after potential causes for its enlargement have been investigated. In inferior turbinate reduction surgery, in efforts to preserve turbinate function, one should consider the fact that the area fraction of glands in the lateral mucosa significantly exceeds that of the medial and inferior mucosal layers, whereas that of venous sinusoids varies significantly, with the greatest difference inferiorly.

### 2.1.1.7 Pharynx

The pharynx is a tube formed by skeletal muscle and lined by mucous membrane that is continuous with that of the nasal cavities. The pharynx is divided into three major regions: the nasopharynx, the oropharynx, and the laryngopharynx (see Figure 2.7).

The **nasopharynx** is flanked by the conchae of the nasal cavity, and it serves only as an airway. At the top of the nasopharynx are the pharyngeal tonsils. A pharyngeal tonsil, also called

an adenoid, is an aggregate of lymphoid reticular tissue similar to a lymph node that lies at the superior portion of the nasopharynx. The function of the pharyngeal tonsil is not well understood, but it contains a rich supply of lymphocytes and is covered with ciliated epithelium that traps and destroys invading pathogens that enter during inhalation. The pharyngeal tonsils are large in children, but interestingly, tend to regress with age and may even disappear[2]. The uvula is a small bulbous, teardrop-shaped structure located at the apex of the soft palate. Both the uvula and soft palate move like a pendulum during swallowing, swinging upward to close off the nasopharynx to prevent ingested materials from entering the nasal cavity. In addition, auditory (Eustachian) tubes or pharyngotympanic tubes that connect to each middle ear cavity open into the nasopharynx (this connection is why colds often lead to ear infections)[51].

The nasopharynx functions to transmit humidified air from the nasal cavity down to the oropharynx. The nasopharynx lies above the soft palate and behind the posterior nares or conchae of the nasal cavities. Except for the soft palate, the walls of the nasopharynx are rigid and therefore the nasopharyngeal cavity does not obliterate. The pharyngeal isthmus connects the nasopharynx with the more caudal oropharynx. During swallowing, the pharyngeal isthmus may be sealed off from the oropharynx when the soft palate elevates and the superior pharyngeal constrictor muscle contracts. The nasopharynx also contains lymphoid tissues and mucous glands that serve both immune and non-immune host-defense functions. The nasopharyngeal tonsil is a mucosa-associated lymphoid tissue (MALT) located in the roof and posterior wall of the nasopharynx [18, 51].

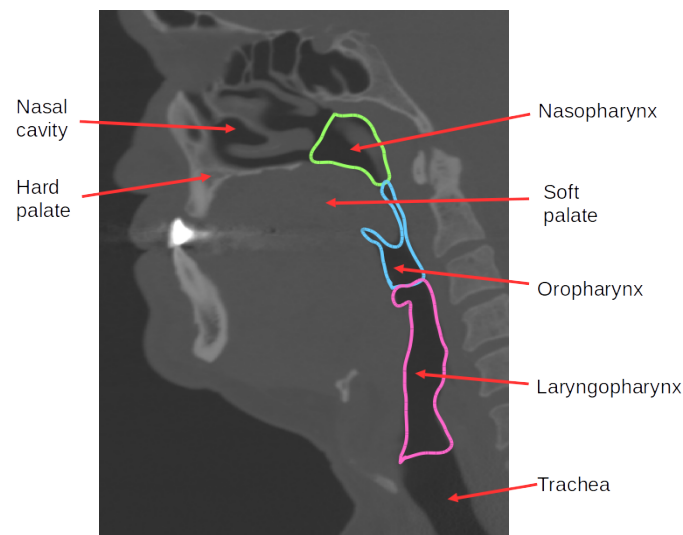


Figure 2.7: Divisions of the pharynx.

The **oropharynx** is a passageway for both air and food. It extends from the soft palate to the upper border of the epiglottis (see Figure 2.7) and it is bordered superiorly by the nasophar-

ynx and anteriorly by the oral cavity. Its anterior border is the pharyngeal part of the tongue and its posterior border is the epithelial mucosa that abuts the body of the second and upper part of the third cervical vertebrae. The oropharynx may be viewed as an “intersection” in the aerodigestive tract because it can transmit inspired air into the trachea and liquid or masticated food into the esophagus. The fauces is the opening at the connection between the oral cavity and the oropharynx. As the nasopharynx becomes the oropharynx, the epithelium changes from pseudostratified ciliated columnar epithelium to stratified squamous epithelium[18]. The oropharynx contains two distinct sets of tonsils, the palatine and lingual tonsils. A palatine tonsil is one of a pair of structures located laterally in the oropharynx in the area of the fauces. The lingual tonsil is located at the base of the tongue. Similar to the pharyngeal tonsil, the palatine and lingual tonsils are composed of lymphoid tissue, and trap and destroy pathogens entering the body through the oral or nasal cavities[51].

The soft palate serves to block swallowed food and liquid from regurgitating up into the nasopharynx, but it also contains mucous glands and taste buds. The soft palate mucous glands are innervated by (i) the lesser palatine nerve (a branch of the maxillary nerve), which contains the secretomotor efferent branches of the postganglionic parasympathetic fibers whose cell bodies are located in the pterygopalatine ganglion, and (ii) postganglionic sympathetic fibers from the carotid plexus. The taste buds are innervated by afferent taste fibers that run in the lesser palatine nerve and through the pterygopalatine ganglion without synapsing to join the greater petrosal nerve, a branch of the facial cranial nerve[2].

The **laryngopharynx** is inferior to the oropharynx and posterior to the larynx (see Figure 2.7). It continues the route for ingested material and air until its inferior end, where the digestive and respiratory systems diverge. The stratified squamous epithelium of the oropharynx is continuous with the laryngopharynx. Anteriorly, the laryngopharynx opens into the larynx, whereas posteriorly, it enters the esophagus[2, 18].

### 2.1.1.8 Larynx

The larynx is a cartilaginous structure inferior to the laryngopharynx that connects the pharynx to the trachea and helps regulate the volume of air that enters and leaves the lungs. The structure of the larynx is formed by several pieces of cartilage[2]:

- Three large cartilage pieces—the thyroid cartilage (anterior), epiglottis (superior), and cricoid cartilage (inferior)—form the major structure of the larynx.
- The thyroid cartilage is the largest piece of cartilage that makes up the larynx. The thyroid cartilage consists of the laryngeal prominence, or “Adam’s apple,” which tends to be more prominent in males.

- The thick cricoid cartilage forms a ring, with a wide posterior region and a thinner anterior region.
- Three smaller, paired cartilages—the arytenoids, corniculates, and cuneiforms—attach to the epiglottis and the vocal cords and muscle that help move the vocal cords to produce speech.

The epiglottis, attached to the thyroid cartilage, is a very flexible piece of elastic cartilage that covers the opening of the trachea. When in the “closed” position, the unattached end of the epiglottis rests on the glottis. The glottis is composed of the vestibular folds, the true vocal cords, and the space between these folds. A vestibular fold, or false vocal cord, is one of a pair of folded sections of mucous membrane. A true vocal cord is one of the white, membranous folds attached by muscle to the thyroid and arytenoid cartilages of the larynx on their outer edges. The inner edges of the true vocal cords are free, allowing oscillation to produce sound. The size of the membranous folds of the true vocal cords differs between individuals, producing voices with different pitch ranges. Folds in males tend to be larger than those in females, which create a deeper voice. The act of swallowing causes the pharynx and larynx to lift upward, allowing the pharynx to expand and the epiglottis of the larynx to swing downward, closing the opening to the trachea. These movements produce a larger area for food to pass through, while preventing food and beverages from entering the trachea[2, 50].

### 2.1.2 Airways diseases

According to Yataco et al., [57], with the increase in respiratory sleep disorders, such as snoring, upper airway resistance syndrome (UARS) and obstructive sleep apnea syndrome (OSAS), the need for better diagnostics and treatment of these disorders became apparent [58]. Treatment of OSAS is important as it is considered a high morbidity, progressive disease [59]. For example, OSAS may affect up to 2% of children, and is associated with lymphoid hyperplasia, craniofacial anomalies, and neurological disorders. Specifically, upper airway obstruction (UAO) is one of the most serious emergencies faced by critical care physicians. Early diagnosis followed by restoration of airflow is essential to prevent cardiac arrest or irreversible brain damage that occurs within minutes of complete airway obstruction [60]. Although a long list of causes may be responsible for acute UAO, management must begin almost immediately after recognition of the problem. If there is an actual or potential obstruction sufficient to cause ventilatory or oxygenation impairment, an intervention to secure the airway is indicated by whatever method appropriate at the time. No single method is suitable in all instances; selection depends on the assessment of the circumstances[59, 60]. The timing

of the intervention, medical, or surgical, is determined based on the condition of the patient. In practice, an elective procedure before acute decompensation is always preferable.

Exist many diseases associated with the upper airways, but in this case only three of them are reviewed: upper airway obstruction, sleep apnea, and tracheal stenosis.

### 2.1.2.1 Upper airways obstruction

UAO may be functional or anatomic and may develop acutely or subacutely. Relapsing polychondritis constitutes a good example of functional UAO caused by lack of a firm cartilaginous structure to support the tracheal wall. Squamous cell carcinoma of the larynx represents an anatomic example of UAO. Although UAO occurs at any level of the upper respiratory tract, laryngeal obstruction has a particular importance because the larynx is the narrowest portion of the upper airway. The narrowest portion of the larynx is at the glottis in adults and the subglottis in infants [58].

Some key recommendations must be considered in critical care of UAO[61]:

- UAO is a life-threatening emergency that requires prompt diagnosis and treatment.
- Severe UAO can be surprisingly asymptomatic at rest if it develops gradually. Sudden clinical deterioration is unpredictable.
- Patients with possible UAO must never be sedated until the airway is secured. Minimal sedation may precipitate acute respiratory failure.
- Achievement of airway patency in total airway obstruction and reestablishment of ventilatory airflow is the first and foremost goal of the treating physicians.
- Critical care physicians must be aware that pharmacologic interventions (epinephrine, steroids, and heliox) provide temporary support but cannot significantly improve mechanical UAO.
- Bronchoscopy constitutes the most accurate diagnostic tool and frequently provides the best way to correct UAO.
- Cricothyroidotomy is the surgical intervention of choice to reestablish airflow when medical interventions have failed.

Three of the most common causes of UAO are: anaphylaxis, croup, and epiglottitis. Additionally, the airway can also be obstructed if the patient inhale a foreign object[59].

1. **Anaphylaxis:** is a severe allergic reaction that can appear within just minutes of coming into contact with an allergen. An allergen is something that triggers an allergy. Anaphylaxis can be fatal. During an anaphylactic reaction, your airways can swell, obstruct-

ing your breathing. Peanuts and bee stings are among the most common anaphylaxis-causing allergens.

2. **Epiglottitis:** is a condition that occurs as a result of the epiglottis becoming swollen. The epiglottis is a flap of cartilage that covers the opening of your windpipe. Swelling can be caused by anything from an infection to simply drinking coffee that is too hot. Epiglottitis can block the flow of air to your lungs so it can be potentially life threatening. This condition is more often found in children aged 1-5 years, who present with a sudden onset of the following symptoms:
  - Sore throat
  - Drooling, difficulty or pain during swallowing, globus sensation of a lump in the throat
  - Muffled dysphonia or loss of voice
  - Dry cough or no cough, dyspnea
  - Fever, fatigue or malaise (may be seen with any URI)
  - Tripod or sniffing posture
3. **Croup:** is a condition that usually causes a harsh, barking cough. The barking cough is caused by an inflamed windpipe and vocal cords. The swollen windpipe causes the vibration of your vocal cords to sound different. Croup is not considered to be a severe condition and it can usually be treated at home.
4. **Foreign Objects:** Inhaling a foreign object, like a bean, nut or bead, can cause a UAO. A foreign object can get stuck the throat or other air passage, causing an obstruction. While foreign objects can be inhaled accidentally at any age, this is most commonly seen in toddlers and small children.

#### 2.1.2.2 Sleep apnea

Sleep apnea is a common breathing disorder that occurs during sleep. One can get the view of the prevalence of the condition from the fact that 3% of the children are affected by OSA in USA, meaning millions of children alone in USA. And that is not counting the adult population, 2-4% of which is said to be affected by OSA as well [62].

Sleep apnea can be obstructive, central and mixed (sometimes referred to as 'complex'). The most common type is **obstructive sleep apnea (OSA)**, which is characterized by repetitive narrowing and/or collapse of the pharyngeal airway (for upper airway anatomy refer to

Figure 2.1)[63]. The other type, **central sleep apnea (CSA)**, shows a lack of drive to breathe, thus resulting in insufficient ventilation. Additionally, both of these pathologies can occur simultaneously, this being referred to as **complex or mixed apneas**. Sleep apneas cause improper gas exchange, frequent catecholamine surges and impaired sleep continuity[63, 64]. The subsequent blood gas disturbance and arousal from sleep stimulate the sympathetic nervous system resulting in surges of blood pressure. All of the mentioned consequences impart stress on the cardiovascular system and thus may lead to serious cardiovascular problems. In addition, when talking about sleep apnea in children, it can lead to problems such as learning disabilities, attention deficit disorder and failure to thrive[62].

However, sleep apnea is a disease which has not been getting an adequate amount of attention in the research community for a long time. But, the strain on the cardiovascular system and other serious problems, such as daytime sleepiness and even neurocognitive dysfunction, that it causes may be severe in advanced cases of the illness, as such it can significantly affect the heart especially and lead to cardiac arrest. Thus, it has been receiving a lot of attention recently.

Sleep apnea is mainly due to the fact that it is quite widespread and may cause rather severe problems, mainly associated with the cardiovascular system[63]. There are many treatment options available, starting from external breathing assisting devices, such as the continuous pharyngeal pressure device, certain medication, up to various surgical intervention methods.

**Obstructive sleep apnea (OSA)** OSA is a common disorder that is characterized by repetitive pharyngeal airway collapse which results in either hypopnea (partially reduced ventilation) or apnea (totally reduced, or absent, ventilation). As a result, hypoxemia (insufficient amount of oxygen in the organism) and hypercapnia (larger than normal concentration of  $CO_2$  in the bloodstream) manifest themselves, despite the patient's continuing efforts to breathe (and stimulate them even further)[60]. Despite the continuing efforts to breathe in, the airway is only opened when the patient awakens. Then hyperventilation occurs, in order to restore the blood gas balance, which was disturbed during the (hypo)apnea event. After this, the patient returns to sleep and the event takes place again. Due to the repetitive nature of these respiratory events, sleep is significantly fragmented, causing insufficient amount of sleep. That results in fatigue in wakefulness, and even in neurocognitive dysfunction which impairs the ability of many day-to-day tasks, such as driving a car. In addition to this, the repeated hyperventilation results in great strain on the cardiovascular system, thus potentially causing serious problems in it, especially if the cardiovascular system is not fully healthy before all sleep apnea related events start[65].

Common symptoms of OSA include snoring, daytime sleepiness, irritability, or difficulty



with focus or concentration. If left untreated, OSA can lead to secondary health issues such as [61, 65]: heart attack, high blood pressure, stroke, heart failure, irregular heartbeats and weight gain [64].

The current treatment options for OSA range from lifestyle changes, to external device-based treatments, such as continuous positive airway pressure (CPAP), to surgery.

- **Lifestyle changes** such as losing weight, exercising regularly, or consuming less alcohol before sleep may help improve sleep. If lifestyle changes alone do not resolve OSA, CPAP is typically prescribed.
- **CPAP** is the most common treatment for OSA. CPAP is an effective treatment that uses a mask to deliver air pressure and keep the airway open. While CPAP is often successful, some people are unable to use or adhere to this treatment. Oral appliances may be prescribed as an alternative to CPAP.
- **Oral appliances** keep the airway open by holding the jaw forward during sleep.
- **Surgical options** may be considered by some people who are unable to use or adhere to CPAP or oral appliances. Traditional sleep apnea surgery is intended to make the airway larger by removing or altering facial or airway anatomy. These anatomy altering surgeries can be painful and involve lengthy recovery times.

**Central sleep apnea (CSA)** Contrary to OSA, which is chiefly identified by efforts to breathe in when there are obstructions in the airway. Central sleep apnea (CSA) is described by the lack of drive to breathe during sleep. The effect it has is similar to OSA, as it results in frequent (depending on the number of events during sleep) night time awakenings, thus also causing daytime sleepiness and strain on the cardiovascular system [61, 64]. In fact, even though CSA and OSA are often separated in their description, frequently they overlap and occur simultaneously in patients [65]. Central apneas might lead to upper airway closure, which in turn may be connected with OSA, in a way that there is reduced drive in all dilator muscles.

**Mixed apneas** Mixed (complex) apneas are a combination of obstructive and central apneas, meaning that it consists of a complete pause of respiration, shortly followed by an obstructive respiratory effort. Mixed apneas can in general be described by the percentage of CSA vs OSA that the patient has. Furthermore, it has been observed that after commencing CPAP (continuous positive airway pressure) treatment, patients develop CSA symptoms when they previously had only OSA, meaning that when the airway collapsibility is reduced, the patient's apnea shifts more to the CSA side [61, 65].



## 2.2 Image segmentation

The process of partitioning an image into meaningful regions is called image segmentation [15, 19, 43, 44, 48, 66–74]. For the regions to be meaningful, they should represent objects or their parts. Difficulties arise when properties within objects vary or boundaries of objects become blurred [75]. The problem is worsened when sensor inaccuracies exist, noise, sampling artifacts, spatial aliasing or partial volume effect is present in the image, generating that the boundaries of structures to be indistinct and disconnected [76]. These variations, which are often unpredictable, make it impossible to develop an automatic method that can segment all images correctly. Because a high accuracy is demanded in the segmentation of medical images, the user has a critical role in examining the results and correcting the possible errors. For this reason, one challenge is to extract boundary elements belonging to the same structure and integrate these elements into a coherent and consistent model of the structure [76].

Segmentation of medical images is an important step in various applications such as visualization, quantitative analysis and image-guided surgery [33]. Numerous segmentation methods have been developed in the past two decades for extraction of organ contours on medical images. Low-level segmentation methods, such as pixel-based clustering, region growing, and filter-based edge detection, require additional pre-processing and post-processing as well as considerable amounts of expert intervention or information of the objects of interest [62].

Image segmentation is perhaps the most studied area in image analysis. A large number of papers on this topic is published annually in image analysis journals and conference proceedings [15, 19, 46, 48, 72, 74, 77–81]. The developed methods often take into consideration various properties of images or objects, and when such properties deviate from the anticipated ones, errors occur. Even for a limited class of images, for instance MRI brain images, various methods have been developed, none of which is guaranteed to work correctly on a new image. This may be because there are sensor variations; variations in the brain's shape, size, and intensity distribution; and variations in intensities of tissues surrounding the brain. Since an error-proof image segmentation method cannot be developed, user assistance is needed to correct the obtained errors. At present, the best one can hope for is to have a segmentation method that can correctly find most areas of an object of interest, and in areas where it makes a mistake, allow the user to correct them.

The computerization of medical image segmentation plays an important role in medical imaging applications. It has found wide application in different areas such as diagnosis, localization of pathology, study of anatomical structure, treatment planning, and computer-integrated surgery. However, the variability and the complexity of the anatomical structures in the human body have resulted in medical image segmentation remaining a hard problem [46, 78].

Segmentation of biomedical images in computer-assisted medical diagnosis is a preliminary operation aimed at detection and localization of regions of interest (ROI) in the images for their more detailed examination. The aim of image segmentation consists in reduction of total image examination time. Manually performed image segmentation is a tiresome operation itself; that is why it is reasonable to include computer-aided image segmentation into advanced image processing procedures as their integral part. In general, two basic types of image segmentation tasks in biomedical image processing arise [62]:

- segmentation of anatomical objects of known location and form, co-occurring with other anatomical objects in the given examination being to be neglected.
- detection and contouring of pathomorphological structures of a priori unknown localization and form, distinguished due to their specific biophysical or biochemical properties manifested by differences in color and/or in texture respecting the environment.

Specifically, airways tree segmentation methods must allow an accurate, automated identification of the human airway tree as depicted in CT examinations. The researchers indicate that the segmentation is a building block for most computerized airway related analyses. Additionally, they consider that there is a relatively high contrast between the airway lumen and the airway wall, allowing straightforward way of extracting an airway tree from CT images. For example, using a three-dimensional (3D) region-growing procedure specifically designed to identify lumen regions. However, in the presence of partial volume effects and/or image noise (artifacts), a purely region-growing based operation frequently leads to leakage into the lung parenchyma (i.e., a sudden explosion) under a given (fixed or constant) threshold. This leakage often occurs in small airways and in cases with severe lung disease (e.g., emphysema), thereby leading to an early termination of the progressive airway tree detection process. Although schemes (e.g., front propagation [82–84]) have been developed to prevent leakage associated with region growing, there is no strategy available that can completely prevent it from occurring.

When applying a region-growing approach, a threshold and a seed location needs to be specified. Given the fact that CT examinations may be acquired under different scanning conditions and/or depict different diseases, it is difficult, if not impossible, to determine an optimal threshold for all cases. To determine an appropriate threshold, Mori *et al.* [85] proposed an intuitive approach of gradually increasing the threshold until a sudden “explosion” appears, indicating the occurrence of a leakage. Similarly, Nakamura *et al.* [86] proposed to adaptively change the threshold based on CT values [Hounsfield (HU)] of the seed area projected from adjacent images. When specifying a seed for region growing, the general idea is to locate the trachea regions that are typically defined as regions with low intensity and circular shapes

[31, 85]. Whereas it is difficult to guarantee that the regions satisfying these criteria are always representative of the trachea, robust and efficient identification of the trachea region is not always an easy task. Hence, an interactive selection of a seed is often implemented for this purpose [32–34, 86].

Despite these limitations, the region-growing approach is very simple and efficient in implementation. Hence, many of the available airway segmentation methods [31–44, 87] employ this procedure as an initial step for large airway identification and thereafter implement additional procedures to identify smaller airways while preventing potential leakage.

In general, there are different taxonomies to classify segmentation methods, for example, depending on the level of user intervention (automatic or semi-automatic)<sup>2</sup> or (supervised and unsupervised), according to the main characteristics (based on morphology, based knowledge or rule based on machine learning based on matching templates, based on the shape analysis) [88], and according to the desired segment regions (regions based methods, edge-based methods and hybrids methods) [69]. In addition, if you want to delve into the different techniques for image segmentation, two important reviews are described by Kirbas et. al. [70] and Lesage et. al. [68].

Finally, according to the review of the state of the art performed, it was identified that there are not many works on the upper airways segmentation. Works found are described in Chapter 7, however, most of them use proprietary software (especially Mimics) to perform the segmentation and they perform it manually or semi-automatically. The obtained result is used for executing the simulation of airflow and thus able to determine where the airflow resistance affects the normal breathing process.

## 2.3 Differential geometry

Differential calculus interprets derivatives by approximating the tangential line at any given point. Differential geometry describes curvature in a similar way by approximating surface curves using tangential circles. Do Carmo [89] presents the fundamental theorem of the local theory of curves states as following:

**Theorem 1.** *Given differentiable functions  $k(s) > 0$  and  $\tau(s)$ , using scalar  $s \in I$ , there exists a regular parametrized curve  $\alpha : I \rightarrow R^3$  such that  $s$  is the arc length,  $k(s)$  is the curvature, and  $\tau(s)$  is the torsion of  $\alpha$ . Moreover, any other curve  $\bar{\alpha}$  satisfying the same conditions,*

---

2

Fully automated algorithms require no manual initialization or interaction and use the same settings for all scans processed. Semi-automated algorithms require user initialization or interaction, which varied from placing a single seed point or selection of certain parameters.

*differs from  $\alpha$  by a rigid motion; that is, there exists an orthogonal linear map  $\rho$  of  $\mathbb{R}^3$ , with positive determinant, and a vector  $c$  such that  $\bar{\alpha} = \rho \circ \alpha + c$ .*

The basic condition is that at every point along a continuous curve there exists a tangential circle that has a radius equivalent to the instantaneous radius of the curve in much the same way the derivative describes the instantaneous slope of a curve[90].

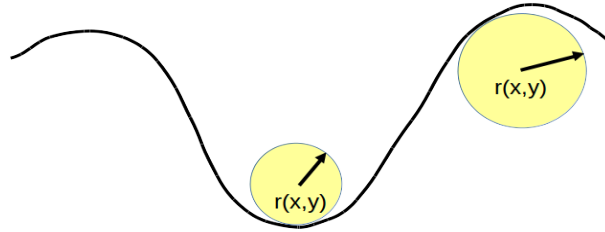


Figure 2.8: Tangent circles to a 2D curve.

Figure 2.8 shows a continuous curve and two points where curvature is computed. Yellow circles are the tangent circles to the curve at the shown points. Curvature is given as  $1/r$  where  $r$  denotes the radius of the fitted circle.

In differential geometry, curves that are “above” the surface are positive, while curves “below” the surface are negative. Then, in Figure 2.8, curvature has positive values at the left circle and negative value at the right circle.

### 2.3.1 Surfaces

A mathematical surface,  $S$ , is a mapping from a two-dimensional parameter domain into a higher-dimensional space, such as  $\mathbb{R}^3$  [91, 92]:

$$S = u \times v \rightarrow \mathbb{R}^d \quad (2.1)$$

The surface must be locally minimum of class  $C^2$ , that is, differentiable up to order 2. This allows computation of the principal curvatures, which help classify the type of surface and provide insight into its intrinsic and extrinsic shape variation. The principal curvatures are directly related to the regional variation of the surface’s normal directions [91].

A surface can be defined as any 2-dimensional set of points within  $\mathbb{R}^n$ . Thus, a general surface is parameterized using:

$$r(x, y) = (r_1(x, y), r_2(x, y), \dots, r_n(x, y)) \quad (2.2)$$

This type of mapping from an open set in  $\mathbb{R}^2$  into  $\mathbb{R}^3$  is known as a Monge patch and is parameterized as:

$$r(x, y) = (x, y, F(x, y)) \quad (2.3)$$

where  $F(x, y)$  is the original image and  $r(x, y)$  is a 2-D surface in  $\mathbb{R}^3$ .

### 2.3.2 Curvatures

Classical differential geometry provides a complete local description of smooth surfaces and guides the selection of surface characteristics. Principal curvatures, mean curvature, gaussian curvature and principal directions are identified as the local second-order surface characteristics that possess several desirable invariance properties and represent extrinsic and intrinsic surface geometry respectively [93–95]. Curvatures and the tensor of curvature are well studied in classical differential geometry (see for example reference [89]). The curvature tensor is closely related to the surface normal and normal curvature evaluation.

#### 2.3.2.1 Types of curvature

There are several different measures of a surface curvature [89] and while each can be useful, no single type of curvature detects all possible surface anomalies. Thus, it is often necessary to calculate several kinds of surface curvature to obtain an accurate impression of the surface's behavior.

In this context, surfaces of volumetric images as existing as the interfaces of homogenous intensity regions. This assumption locates the surface at high gradient points in the image. In addition, this also allows constructing the local frame for the surface directly from the gradient.

Based on the above, at any given point on a parametric surface, i.e. a surface of the form  $r = r(x, y)$ , it is possible to calculate a unit normal vector. This surface normal (unit normal vector) is the line perpendicular to the tangent plane at a given point along a surface. The unit normal vector  $\vec{n}$  is calculated from the first derivatives of the surface function in 2.2 using:

$$\vec{n} = \frac{\frac{\partial r}{\partial x} \times \frac{\partial r}{\partial y}}{\left| \frac{\partial r}{\partial x} \times \frac{\partial r}{\partial y} \right|} \quad (2.4)$$

Now, if a plane containing the normal is rotated about the normal, the intersections of this plane with the surface result in an infinite number of section curves each of which could have a different value for the curvature at the point [96]. However, there are two mutually perpendic-

ular locations of the normal plane where the curvature is a maximum and a minimum. These curvature values are defined to be the principal curvatures,  $k_{max}$  and  $k_{min}$  (frequently defined using the first and second fundamental forms). Then, surface normal is used in computing the fundamental forms and can also be used by itself to show the direction where surface is “facing”. In addition, from the principal curvatures can be calculated the Gaussian curvature and the mean or average curvature.

Gaussian curvature is given by

$$K = k_{max}k_{min} \quad (2.5)$$

and is the most often cited form of curvature. However, from experiments performed, it is not very suitable to define surface contours. Notice that if either one of the principal curvatures is zero, the Gaussian curvature is also zero, even though the other principal curvature may be fluctuating wildly.

Mean curvature is given by

$$H = \frac{k_{max} + k_{min}}{2} \quad (2.6)$$

The problem with displaying only the mean curvature is that minimal surfaces always have  $H = 0$ . Another measure of curvature is the absolute curvature, it is given by

$$A = |k_{max}| + |k_{min}| \quad (2.7)$$

In general, differential geometry states that local surface shape is uniquely determined by the first and second fundamental forms. Gaussian and mean curvature combine these first and second fundamental forms in two different ways to obtain scalar surface features that are invariant to rotations, translations, and changes in parameterization [89, 92, 94, 95].

### 2.3.2.2 Fundamental forms and shape operator

The fundamental forms are used to determine the metric properties of surfaces [92–94]. Modern mathematics favors an equivalent formulation of this knowledge in terms of the metric tensor and the Weingarten mapping (the “shape” operator).

**First fundamental form** The first fundamental form expresses differential surface length. It describes the way of measuring the distances on a surface. An apparatus that enables one to measure the distances is called metric. This is why the first fundamental form is often referred to as the metric form.

To define the fundamental forms of a surface an explicit surface parameterization  $x(u, v)$

is given. Then, the first fundamental form  $I$  of a surface defined by  $x(u, v)$  is given by the following quadratic form:

$$I(u, v, du, dv) = dx \cdot dx = \begin{bmatrix} du & dv \end{bmatrix} \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \begin{bmatrix} du \\ dv \end{bmatrix} = du^T \begin{bmatrix} g \end{bmatrix} du \quad (2.8)$$

where the  $[g]$  matrix elements are defined to be

$$\begin{aligned} g_{11} &= E = x_u \cdot x_u \\ g_{22} &= G = x_v \cdot x_v \\ g_{12} &= g_{21} = F = x_u \cdot x_v \end{aligned} \quad (2.9)$$

where the subscripts denote partial differentiation

$$\begin{aligned} x_u(u, v) &= \frac{\partial x}{\partial u} \\ x_v(u, v) &= \frac{\partial x}{\partial v} \end{aligned} \quad (2.10)$$

$x_u$  and  $x_v$  are referred to as the  $u$  tangent vector and the  $v$  tangent vector functions, respectively, and they may or may not be orthogonal to each other. These two tangent vectors are said to lie in (and form a basis for) the tangent plane  $T(u, v)$  of the surface at the point  $x(u, v)$ .

The  $[g]$  matrix is referred to as the first fundamental form matrix or, more simply, as the metric (or metric tensor) of the surface. Since the vector dot product is commutative, this  $[g]$  matrix is symmetric and only has three independent components.

The first fundamental form  $I(u, v, du, dv)$  measures the small amount of movement  $|dx|^2$  on the surface at a point  $(u, v)$  for a given small movement in the parameter space  $(du, dv)$ . This function is invariant to surface parameterization changes and to translations and rotations of the surface. The first fundamental form and the metric depend only on the surface itself. They do not depend on how the surface is embedded in 3-D space, and they are therefore referred to as intrinsic properties of a surface. In fact, the metric functions  $E$ ,  $F$ , and  $G$  determine all intrinsic properties of a surface [94].

In 3D case, Toriwaki [11] considers a 3D continuous image  $f(x, y, z)$  and a 4D hypersurface  $S : u = f(x, y, z)$ . At an arbitrary point  $P$  on this surface, there exist three principal curvatures. He denotes them by  $k_1$ ,  $k_2$ , and  $k_3$ . They are calculated as follows. If the first and the second derivatives of a 3D gray-tone continuous image  $u = f(x, y, z)$  are denoted by  $f_x$ ,  $f_y$ , and  $f_{xx}$ , and  $f_{xy}$ , etc. Then, two matrices,  $F_1$  and  $F_2$  are considered.  $F_1$  corresponds to the first

fundamental form and it is given by

$$F_1 = \begin{bmatrix} 1 + f_x^2 & f_x f_y & f_x f_z \\ f_y f_x & 1 + f_y^2 & f_y f_z \\ f_z f_x & f_z f_y & 1 + f_z^2 \end{bmatrix} \quad (2.11)$$

where to each component of the diagonal is added one to eliminate the presence of critical points (when the product of diagonal's components is zero).

**Second fundamental form** In contrast to the first fundamental form, the second fundamental form of a surface is dependent on the embedding of the surface in 3-D space and it is therefore considered as an extrinsic property of the surface [94, 95]. The second fundamental form  $II$  is given by

$$II(u, v, du, dv) = -dx \cdot dn = \begin{bmatrix} du & dv \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} du \\ dv \end{bmatrix} = du^T \begin{bmatrix} b \end{bmatrix} du \quad (2.12)$$

where the  $[b]$  matrix elements are defined to be

$$\begin{aligned} b_{11} &= L = x_{uu} \cdot n \\ b_{22} &= N = x_{vv} \cdot n \\ b_{12} &= b_{21} = M = x_{uv} \cdot n \end{aligned} \quad (2.13)$$

where  $n$  is the unit normal vector (see section 2.3.2.1). The double subscript implies second partial derivatives

$$\begin{aligned} x_{uu}(u, v) &= \frac{\partial^2 x}{\partial u^2} \\ x_{vv}(u, v) &= \frac{\partial^2 x}{\partial v^2} \\ x_{uv}(u, v) &= \frac{\partial^2 x}{\partial u \partial v} \end{aligned} \quad (2.14)$$

The  $[b]$  matrix is the second fundamental form matrix and it is also symmetric if the surface is well behaved (i.e., if the mixed partial derivatives are equal). The Gauss-like  $L$ ,  $M$ , and,  $N$  notation is introduced again as above. These definitions allow discussing about the “state” equation for surfaces [89, 94].

The second fundamental form measures the correlation between the change in the normal vector  $dn$  and the change in the surface position  $dx$  at a surface point  $(u, v)$  as a function of a small movement  $(du, dv)$  in the parameter space.

The differential normal vector  $dn$  always lies in the tangent plane  $T(u, v)$ . The ratio of  $II(u, v, du, dv) / I(u, v, du, dv)$  is known as the the normal curvature function  $k_{normal}$ . Normal



curvature at a surface point varies only as a function of the direction of the differential vector  $(du, dv)$  in the parameter space. If  $dn$  and  $dx$  are aligned for a particular direction of  $(du, dv)$ , that direction is called a *principal direction* of the surface at that surface point. The extrema of the normal curvature function at a given point occur in these directions and are referred to as *principal curvatures*.

In 3D case, Toriwaki [11] presents the second fundamental form  $F_2$  and it is given by

$$F_2 = \frac{-1}{D} \begin{bmatrix} f_{xx} & f_{xy} & f_{xz} \\ f_{yx} & f_{yy} & f_{yz} \\ f_{zx} & f_{zy} & f_{zz} \end{bmatrix} \quad (2.15)$$

where

$$D = (1 + f_x^2 + f_y^2 + f_z^2)^{\frac{1}{2}} \quad (2.16)$$

**Third fundamental form** In differential geometry, the third fundamental form is a surface metric denoted by  $III$ . Unlike the second fundamental form, it is independent of the surface normal. The third fundamental form is expressible entirely in terms of the first fundamental form and second fundamental form. If  $H$  be the mean curvature of the surface and  $K$  be the Gaussian curvature of the surface (see section 2.3.2.1), then

$$III - 2 \cdot H \cdot II + K \cdot I = 0 \quad (2.17)$$

**Shape operator** A good way to measure how a regular surface  $S$  bends in  $\mathbb{R}^3$  is to estimate how the surface normal  $n$  changes from point to point. To do this, a linear operator called the *shape operator* (or Weingarten map) is used to calculate the bending of  $S$ . In general, the shape operator can be used to compute how the surface normal changes along the surface. All other curvatures can be derived from the shape operator which in turn is calculated using the first two fundamental forms.

The fundamental forms can also be expressed using symmetric matrices. Here, the matrix form will be used to define the shape operator. Define  $F_1$  and  $F_2$  for 2D case using:

$$F_1 = \begin{bmatrix} E & F \\ F & G \end{bmatrix} \quad (2.18)$$

$$F_2 = \begin{bmatrix} L & M \\ M & N \end{bmatrix} \quad (2.19)$$

Then, the shape operator can be calculated using:

$$W = F_1^{-1}F_2 = \begin{bmatrix} FM - GL & FN - GM \\ FL - EM & FM - EN \end{bmatrix} (F^2 - EG)^{-1} \quad (2.20)$$

The eigenvalues of  $W$  are just the principal curvatures  $k_1$  and  $k_2$  at point  $x(u, v)$ . In particular the determinant of the shape operator at a point is the Gaussian curvature, but it also contains other information, since the mean curvature is half the trace of the shape operator. Which are given respectively by

$$K = \det \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} = k_1 k_2 \quad (2.21)$$

$$H = \frac{1}{2} \text{tr} \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} = \frac{1}{2} (k_1 + k_2) \quad (2.22)$$

### 2.3.2.3 Principal curvatures and vectors

As mentioned above, the principal curvatures provide the maximum and minimum curvature at any given point. The principal vectors are the directions in which these extrema occur. The principal curvatures  $k_1$  and  $k_2$  are a perfectly valid pair of surface curvature descriptors, which are analytically equivalent to the mean and Gaussian curvature pair. For surfaces, the principal curvatures are the two eigenvalues of the  $2 \times 2$  matrix shape operator in  $\mathbb{R}^2$ . And for hypersurfaces, principal curvatures  $k_1$ ,  $k_2$ , and  $k_3$  are the three eigenvalues of the  $3 \times 3$  matrix shape operator in  $\mathbb{R}^3$ . The principal vectors in each case are derived from the corresponding eigenvectors.

The great interest with the principal curvatures is that they allow to classify the points of the surface according to the following categories: valley, crest, flat, pit, peak, saddle and minimal [3, 10, 11, 97–100](see Figure 2.9, where first quadrant (I) is a cup ( $k_1 = k_2$ ), second quadrant (II) is a saddle ( $-k_1 = k_2$ ), third quadrant (III) is a cap ( $-k_1 = -k_2$ ), fourth quadrant (IV) is a saddle ( $k_1 = -k_2$ ), dotted lines are used to show the opposite sign and same sign in regions where  $k_1 = k_2$ , on the axes are the parabolic surfaces, and at the centre is the plane where  $k_1 = k_2 = 0$  [3]).

It is important to mention that the relationship between the principal curvatures depends initially on the dimension of the image (n-dimension). For the 2D case, only two principal curvatures must be considered, being themselves the maximum and the minimum depending on the ordering form. In the 3D case, there are three principal curvatures and in the same way as in the 2D case, the maximum and the minimum must be selected.

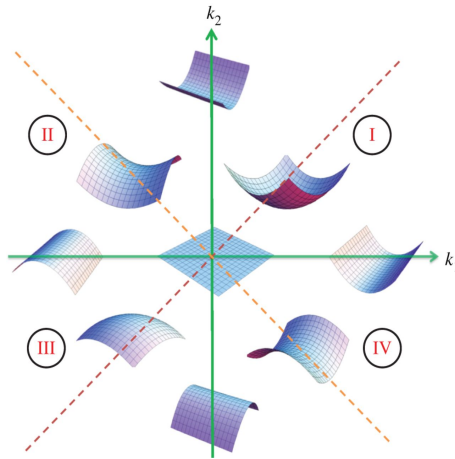


Figure 2.9: Shape space categories using  $k_1$  and  $k_2$ . (taken from [3])

For 2D case, there are several shape classifications [3, 10, 97–100], however, all coincide with the classification presented in Figure 2.9. For example, Table 2.2 presents the classification shapes proposed by Haralick et. al. [10] in the classical paper “The topographic primal sketch”.

Pixel label No Zero-Crossing				Pixel label One Zero-Crossing			
$\ \nabla f\ $	$k_1$	$k_2$	Label	$\ \nabla f\ $	$k_1$	$k_2$	Label
0	0	0	Flat	0	-	-	Peak
+	-	-	Concave hill	0	-	0	Ridge
+	-	0	Concave hill	0	-	+	Saddle
+	-	+	Saddle hill	0	+	-	Saddle
+	0	0	Slope	0	+	0	Ravine
+	+	-	Saddle hill	0	+	+	Pit
+	+	0	Convex hill				
+	+	+	Convex hill				

Table 2.2: Pixel label proposed by Haralick et. al. [10].

Consequently, a point of the valley is mathematically a point of the image’s surface such that  $k_1 > 0$  and  $k_2 = 0$ . In practice, the luminance variations and the noise in real images make the detection of pure valleys almost impossible ( $k_1 > 0$  and  $k_2 = 0$ ).

For 3D case, principal curvatures  $k_1$ ,  $k_2$ , and  $k_3$  of a hypersurface  $S$  have relations among them as shown below [11].

1. Signs and orders in their sizes (20 cases).

2. Orders in sizes of their absolute values and zero and non-zero (26 cases).
3. The sign of the sum  $k_1 + k_2 + k_3$  (3 cases).

According to Toriwaki et. al. [11] and Hirano et. al. [12], in total, there exist 1560 cases of all these combinations. However, Hirano et. al. [12] assume that  $k_1 \geq k_2 \geq k_3$  and indicate that this does not generate loss of generality. While Toriwaki et. al. [11] assume  $|k_1| \geq |k_2| \geq |k_3|$  without loss of generality. In both cases, they consider that 20 cases are enough to characterize the hypersurfaces (see Table 2.3). These curvatures can be used to represent the local shape of the surface  $S$  at a point  $P$  and its vicinity.

Table 2.3: A possible set of principal curvatures in a 4D surface. (taken from [11, 12]).

No.	Simple curvature criteria	Absolute value criteria	Average curvature criteria
1	$k_1 > k_2 > k_3 > 0$	$ k_1  >  k_2  >  k_3  > 0$	$k_1 + k_2 + k_3 > 0$
2	$k_1 > k_2 > k_3 = 0$	$ k_1  >  k_2  >  k_3  = 0$	“
3	$k_1 > k_2 = k_3 > 0$	$ k_1  >  k_2  =  k_3  > 0$	“
4	$k_1 > k_2 = k_3 = 0$	$ k_1  >  k_2  =  k_3  = 0$	“
5	$k_1 = k_2 > k_3 > 0$	$ k_1  =  k_2  >  k_3  > 0$	“
6	$k_1 = k_2 > k_3 = 0$	$ k_1  =  k_2  >  k_3  = 0$	“
7	$k_1 = k_2 = k_3 > 0$	$ k_1  =  k_2  =  k_3  > 0$	“
8	$k_1 = k_2 = k_3 = 0$	$ k_1  =  k_2  =  k_3  = 0$	$k_1 + k_2 + k_3 = 0$
9	$k_1 > k_2 > 0 > k_3$	$ k_1  >  k_2  >  k_3  > 0$	$k_1 + k_2 + k_3 > 0$
10	“	$ k_1  >  k_2  =  k_3  > 0$	“
11	“	$ k_1  >  k_3  >  k_2  > 0$	“
12	“	$ k_1  =  k_3  >  k_2  > 0$	“
13	“	$ k_3  >  k_1  >  k_2  > 0$	“
14	“	“	$k_1 + k_2 + k_3 = 0$
15	“	“	$k_1 + k_2 + k_3 < 0$
16	$k_1 > k_2 = 0 > k_3$	$ k_1  >  k_3  >  k_2  = 0$	$k_1 + k_2 + k_3 > 0$
17	“	$ k_1  =  k_3  >  k_2  = 0$	$k_1 + k_2 + k_3 = 0$
18	“	$ k_3  >  k_1  >  k_2  = 0$	$k_1 + k_2 + k_3 < 0$
19	$k_1 = k_2 > 0 > k_3$	$ k_1  =  k_2  >  k_3  > 0$	$k_1 + k_2 + k_3 > 0$
20	“	$ k_1  =  k_2  =  k_3  > 0$	“
21	“	$ k_3  >  k_1  =  k_2  > 0$	“
22	“	“	$k_1 + k_2 + k_3 = 0$

Table 2.3: A possible set of principal curvatures in a 4D surface. (taken from [11, 12]).

No.	Simple curvature criteria	Absolute value criteria	Average curvature criteria
23	“	“	$k_1 + k_2 + k_3 < 0$
24	$k_1 = k_2 = 0 > k_3$	$ k_3  >  k_1  =  k_2  = 0$	“
25	$k_1 > 0 > k_2 > k_3$	$ k_1  >  k_3  >  k_2  > 0$	$k_1 + k_2 + k_3 < 0$
26	“	“	$k_1 + k_2 + k_3 = 0$
27	“	“	$k_1 + k_2 + k_3 < 0$
28	“	$ k_1  =  k_3  >  k_2  > 0$	“
29	“	$ k_3  >  k_2  >  k_1  > 0$	“
30	“	$ k_3  >  k_1  =  k_2  > 0$	“
31	“	$ k_3  >  k_1  >  k_2  > 0$	“
32	$k_1 > 0 > k_2 = k_3$	$ k_1  >  k_2  >  k_3  > 0$	$k_1 + k_2 + k_3 > 0$
33	“	“	$k_1 + k_2 + k_3 = 0$
34	“	“	$k_1 + k_2 + k_3 < 0$
35	“	$ k_1  =  k_2  =  k_3  > 0$	“
36	“	$ k_2  =  k_3  >  k_1  > 0$	“
37	$k_1 = 0 > k_2 > k_3$	$ k_3  >  k_2  >  k_1  = 0$	“
38	$k_1 = 0 > k_2 = k_3$	$ k_2  =  k_3  >  k_1  = 0$	“
39	$0 > k_1 > k_2 > k_3$	$ k_3  >  k_2  >  k_1  > 0$	“
40	$0 > k_1 > k_2 = k_3$	$ k_2  =  k_3  >  k_1  > 0$	“
41	$0 > k_1 = k_2 > k_3$	$ k_3  >  k_2  =  k_1  > 0$	“
42	$0 > k_1 = k_2 = k_3$	$ k_1  =  k_2  =  k_3  > 0$	“

In general, in the case of an image in a continuous space, the curvature of an hypersurface can be computed using the first partial derivatives and the second derivatives of the function obtained. In other words, the principal curvatures and principal directions correspond respectively to the eigenvalues and the eigenvectors of the Weingarten matrix [10, 89, 100]. Alternatively, Taubin [101] proposed to compute principal curvatures and principal directions using eigen-analysis of an integral matrix. In addition, Monga, Benayoun and Faugeras [102–104] show that is possible to compute locally the gaussian, average, and principal curvatures, and also the principal directions, without fitting a surface patch. They use the gradient direction  $\vec{g}$  to define the tangent plane of the surface. Then, they define a local parametrization by giving two arbitrary perpendicular vectors in the plane. To find the principal directions, they define a unit vector  $\vec{r}$  in the tangent plane of the surface, characterized by its angle  $\theta$  around the normal

$\vec{g}$ . They have shown how to compute the curvature  $k_{\vec{g}}$  of the surface in the direction  $\vec{t}(\theta)$  using the Hessian matrix  $H$  of the 3D image function at this point by

$$k_{\vec{g}} = \frac{-\vec{t}^T H \vec{t}}{\|\vec{g}\|} \quad (2.23)$$

also, they have shown how to compute the principal directions, which are the directions where the surface curvature is extremal, by a search of the extrema of  $k_{\vec{g}}(\theta)$ . Finally, Koller et. al [105], Sato [73, 106], Frangi et. al. [107, 108] calculate the principal curvatures and principal directions using the Hessian matrix, which describes the second-order structures of local intensity variations around each point of a multi-dimensional image.

## Chapter 3

# Contrast enhancement using multi-scale morphological operator

“Vision is computationally intensive; conventional computers are too slow to perform complex visual tasks in real time.”  
– *Azriel Rosenfeld.*

In order to segment the upper airways, it is very important to improve the contrast between the soft tissue areas and airways. Normally, when a segmentation algorithm is applied, there is an over-segmentation due to the low-contrast between the aforementioned regions. In addition to low-contrast, upper airways are composed of several regions with complex structures (p.e. paranasal sinuses, nasal cavities and nasal walls) that can be affected by partial volume, artefacts and noise. Therefore, to reduce the impact of low-contrast presence on CT images it is necessary to include enhancement techniques. However, the application of these techniques depends directly on the type of images (highly application dependent), the quality of the original images and the structures that are desired to enhance [109, 110]. In other words, an enhancement technique that works well for CT images may not work as well for MR images, and in the same way, a technique that works well to enhance the venous system may not work for the respiratory system. Traditional approaches to solving the above tasks have used mainly linear systems tools, nowadays a new understanding has been matured that linear approaches are not well suitable or even fail to solve problems involving geometrical aspects of the image. Therefore, there is a need for nonlinear geometric approaches. A powerful nonlinear methodology that can successfully solve the above problems is mathematical morphology [111]. For the above reasons and challenges, this chapter presents the use of an enhancement technique for head-neck CT images based on multi-scale morphological transformations to improve the contrast in upper airways region. Additionally, a stopping criterion is proposed

for the multi-scale enhancement algorithm based on the contrast improvement ratio (CIR).

### 3.1 State of the art

Medical imaging is the result of applying a process that allows creating visual representations of the interior of a body in order to support the clinical analysis and the medical intervention. Medical imaging seeks to reveal internal structures where different organs are in various depths, but normally the image does not have enough quality to be analyzed by the physician. For instance, in an image taken from the chest by CT, X-ray, ultrasound or MR organs such as skin, heart, lung, bone, ligaments, vessels, cartilage, and lymphatic fluid appear simultaneously in an image while overlapping or hiding by other structures [109, 112]. In general, the existence of several objects overlapping in an image and the close proximity of adjacent voxel values in medical images make the diagnostic process a difficult task [112]. To reduce the impact of partial volume, artefacts, noise, and low-contrast in the segmentation process are necessary to include an image enhancement technique.

Image enhancement is an important approach in digital image processing, which is a problem-oriented procedure [113]. In addition, image enhancement is a process that principally focuses on processing an image in such a way that the processed image is more suitable than the original one for an specific application [109]. The goals of image enhancement include the improvement of the visibility, perceptibility and detectability. First, visibility is defined as how clearly objects can be seen, or how far you can see clearly, usually because of the lighting conditions. This is determined by the degree to which something is seen by an observer. Second, perceptibility is the capability to see or notice some regions inside the image. Finally, detectability is the ability to notice features that are partly hidden or not clear inside the image or region. To achieve these goals is necessary to include tasks such as cleaning the image from various types of noise, enhancing the contrast among adjacent regions or features permitting to improve the visual appearance of the image or to provide “better” representation for subsequent automated image processing (analysis, detection, segmentation, and recognition) [114]. For instance, image enhancement is usually followed by (or is done simultaneously with) detection of features such as edges, peaks, and other geometric features which is of paramount importance in low-level vision [111].

A vast range of image enhancement techniques can be find in the literature [109, 113–118]. Each of them has an aim in the image enhancement, for example, contrast enhancement, intensity transformation, edge enhancement or digital mosaic production. In order to increase the visibility and detectability in CT images of the upper airways, contrast enhancement is used to obtain better differentiation between the airways and surrounding soft tissue.



Most contrast enhancement algorithms operate by amplifying local variations in colour or intensity within an image, because the human visual system has limited ability to detect small variations in pixel intensity or colour within homogeneous regions in an image, these variations will be very difficult to perceive. One side effect of this contrast enhancement process is that any noise which is present in an image is typically amplified too. Fortunately, most contrast enhancement methods have parameters which can be manually adjusted to specify the amount of enhancement to be performed, so the appropriate contrast level for image display can be obtained. In addition, the contrast enhancement is not done until the restoration processes are completed (p.e. noise or geometric distortions reduction).

In particular, the contrast enhancement of image refers to the amount of gray differentiation that exists between various features in digital images. It is the range of the brightness present in the image. The images having a higher contrast level usually display a larger degree of gray scale difference as compared to lower contrast level. The contrast enhancement is a process that allows image features to show up more visibly by making best use of the intensity presented on the display devices. For example, in certain images, there may be exhaustive utilization of the entire dynamic range of grayscale of the image but the contrast over a small region may be poor. These types of images suffer from poor local contrast [67, 109, 115, 119]. In conclusion, an image with poor contrast has variations which are gradual and difficult to detect visually. While an image with high contrast will typically have large intensity or colour variations separating different objects in an image, so it is easy to visually locate object boundaries and distinctive features within objects.

The methods proposed to enhance image are broadly classified into two categories, including spatial domain and frequency domain methods. The term spatial domain refers to the image plane itself, and approaches in this category are based on direct manipulation of pixels in an image. Frequency (transform) domain processing methods operate on transforms of the image (such as the Fourier, wavelet, and cosine transforms) [109, 113–118]. While traditional approaches for solving the above tasks have used mainly tools of linear systems, nowadays a new understanding has matured that linear approaches are not well suitable or even fail to solve problems involving geometrical aspects of the image. Thus there is a need for nonlinear geometric approaches. In conclusion, contrast enhancement algorithms can be classified in terms of two properties: an algorithm utilizes either spatial or frequency domains, and it incorporates either linear or nonlinear operations.

The most widely used methods include various contrast manipulations and histogram equalization [67]. Classic contrast manipulation is usually based on globally defined stretching function (or called transfer function). Histogram clipping might be needed before pixel-by-pixel stretching. Traditionally histogram equalization is also global technique in the sense

that the enhancement is based on the equalization of the histogram of the entire image. However, it is well recognized that using only global information is often not enough to achieve good contrast enhancement (for example, global approaches often cause an effect of intensity saturation) [120].

Jobson et. al. developed a technique called retinex model [121], in which the contribution of each pixel within the local window is weighted by computing the local average based on Gaussian function. A later version, called multiscale retinex model [122], gives better results but it is computationally more intensive. Shinkar et. al. developed a technique for contrast enhancement based on wavelet decomposition and reconstruction and it has been used for medical image enhancement, especially for mammography images [120].

A more recently developed technique is based on mathematical morphology that was invented in the early 1960s by Georges Matheron and Jean Serra who worked on the automatic analysis of images occurring in mineralogy and petrograph [123, 124], to analyze binary images from geological. In biomedical data as well as to formalize and extend earlier or parallel work [74, 125] on binary pattern recognition based on cellular automata and Boolean/threshold logic. Then, in the late 1970's it was extended to gray-level images [124]. After this, in the mid 1980's it was brought to the mainstream of image/signal processing and related to other nonlinear filtering approaches [111]. Finally, in the late 1980's and 1990's it was generalized to arbitrary lattices [126]. The above evolution of ideas has formed what we call nowadays the field of morphological image processing, which is a broad and coherent collection of theoretical concepts, nonlinear filters, design methodologies, and applications systems [111]. For example, in [127] a morphological filter is proposed for sharpening medical images. In this method, after locating edges by gradient-based operators, a class of morphological filter is applied to sharpen the existing edges. In fact, morphology operators, through increasing and decreasing colors in different parts of an image, have an important role in processing and detecting various existing objects in the image [112]. Another example is presented in [109], where locating edges in an image using morphology gradient has comparable performance with that of classic edge-detectors such as Canny and Sobel.

### 3.1.1 Morphological approaches

Mathematic morphology (Serra, 1982) is a powerful non-linear tool for extracting image components, which is useful in the representation and description of region shape, such as boundaries, skeletons, and the convex hulls. It is based on set theory and is a powerful tool for extracting structural characteristics in an image and is useful for characterizing shape information [124]. An image is regarded as a set (binary image) or a function (greyscale image), acted upon by a collection of nonlinear operators using structuring elements. This set of op-

erators is based on set theory and defined on an abstract structure known as an infinite lattice. Additionally, the structuring element, which indicates the shape characteristics in an image, is generally a small and simple binary image, such as a disc, square or line segment [119].

These operators were first systematically examined by Matheron and Serra in 1964 [128] and they are an extension of Minkowski's set theory [124, 126]. Morphological operators include erosion, dilation, opening, closing, rank filters (including median filters), top hat transforms, and other derived transforms. These operations can be defined on binary or greyscale images in any number of dimensions. The extension of morphologic transformations from binary image processing to gray scale processing using max and min operations is done by Sternberg and Haralick et. al. [117]. Besides, they can also be defined with Euclidean (isotropic) or non-Euclidean (geodesic) metrics [129]. The main application areas for the tools of mathematical morphology have been medical imaging, material sciences, and machine vision, where morphological transformations are particularly useful for image analysis and image enhancement [130][129].

Morphological operations rely on set theory, thus a set of operators as union, intersection, inclusion and complement can be applied, they are used in the preliminary and final processing on images such as thickening, thinning, filtering or enhancing [131]. Besides these operations, top-hat and bottom-hat transformations are also well-known for morphological filtering in image processing and enhancing. Top-hat transformation is an operation that extracts small elements and details from the given images. The top-hat transform is computed by subtracting the image's opening from the original image, while the bottom-hat transform is calculated as the difference between the image's closing and the original image [131].

Morphological methods generally include exploring of complete image by a structuring element and at each coordinates the structuring element performs some operation with the neighboring elements [130]. These methods use mathematical principles and relationships between categories to extract the components of an image, which are useful in describing the shape of regions [112].

### 3.1.1.1 Morphological operators

If  $A(x, y)$  and  $B(u, v)$  describe the gray-level image and the structural element respectively, erosion and dilation operators are defined as:

$$A \ominus B = \min_{u,v} \{A(x+u, y+v) - B(u, v)\} \quad (3.1)$$

$$A \oplus B = \max_{u,v} \{A(x+u, y+v) - B(u, v)\} \quad (3.2)$$

The erosion operator reduces the size of objects. This operator increases the size of holes in an image and removes very small details of that image. Removing bright areas under the mask makes the final image looks darker than the original image. The dilation operator acts in reverse; in other words, it increases and decreases the size of objects and holes in the image respectively. The opening operator is equivalent to the application of the erosion and dilation operations on the same image respectively (Eq. 3.1) while the closing operator acts in reverse (Eq. 3.2):

$$A \circ B = (A \ominus B) \oplus B \quad (3.3)$$

$$A \cdot B = (A \oplus B) \ominus B \quad (3.4)$$

The opening operator removes weak connections between objects and small details while the closing operator removes small holes and fills cracks[109, 112].

### 3.1.1.2 Top-Hat transforms

Top-hat transform is a well known and commonly used morphological operation for locally extracting structures from given images (Meyer, 1979). Structural extraction is unrelated to grey level correction and is independent of background brightness. Top-hat morphological processing uses gray scale opening to extract regional maxima or objects which differ in brightness from the surrounding background in images with uneven background intensity. There are two types of top-hat operations: white top-hat (WTH) and black top-hat (BTH). WTH and BTH extract structures brighter and darker than the surrounding areas, respectively. The top-hat is an operation that highlights the local peaks of the image, meanwhile, bottom-hat operation permits to highlight the valleys of image. In both cases the sizes of extracted structures are smaller than that of the structuring element [118, 119, 132].

As was mentioned before, the WTH is used to extract bright or white features of image related to the used structuring element. It is given as the difference of the original image and the opened image [118]. It is defined as:

$$WTH(f) = f - (f \circ g) \quad (3.5)$$

where  $f$  is the image and  $g$  is the structuring element.

The high intensity regions, i.e., the features that cannot accommodate the structuring element are removed by performing a structural opening. The resulting image contains all residual features (i.e. peaks and ridges) removed. Then, subtracting the residual features to the

original image has the effect of accentuating high-intensity (high) structures [132].

Similarly, the BTH is used to extract the darker or black features of image related to the used structuring element. The BTH is the difference between the closed image and the original image [118]. It is defined as:

$$BTH(f) = (f \cdot g) - f \quad (3.6)$$

Using a structural closing the dual residual features (i.e. valleys and roughs) are obtained. Then, original images is subtracted from the resulting image to accentuate low-intensity (dark) structures[132]. To obtain the final morphological contrast enhancement using top-hat transforms to enlarge the contrast between the white and black regions of image, the WTH and BTH are used. One way of image enhancement based on top-hat transform (ETH) is adding the white image regions on and subtracting the black image regions from the original image as follows [117, 128]:

$$ETH(f) = (f + WTH(f)) - BTH(f) \quad (3.7)$$

$ETH(f)$  is the final enhanced image.

### 3.1.2 Multiscale Top-Hat transforms

The algorithm works through the extraction of two types of features from the image: multi-scale image regions at each scale and multiscale image regions between neighboring scales [118, 133].

The first type of multi scale image features is the extracted multi-scale image regions at each scale. At each scale, the extracted white image regions should be brighter than other image regions, which means the gray values of the extracted white image regions should be large. So, the real white image regions of all scales should be the large gray values of all scales. Then, the extracted white multi-scale image regions at all scales could be the maximum gray values of all scales as follows[118, 133]:

$$f_w^C = \max_{0 \leq i \leq n} \{WTH_i\} \quad (3.8)$$

where,  $f_w^C$  is the extracted white multi-scale image regions at all scales.

Similarly, the extracted black image regions at each scale should be brighter than other image regions, which means the gray values of the extracted black image regions should be large. So, the real black image regions of all scales should be the large gray values of all scales. Then, the extracted black multi scale image regions at all scales could be the maximum gray

values of all scales as follows [118, 133]:

$$f_b^C = \max_{0 \leq i \leq n} \{BTH_i\} \quad (3.9)$$

where,  $f_b^C$  is the extracted black multi scale image regions at all scales.

The second type of multi scale image features is the detail image regions between neighboring scales. These detail image regions represent the white and black detail image features at different scales. The white detail image regions between neighboring scales  $i$  and  $i + 1$  can be expressed as follows [118, 133]:

$$WTH_{i(i+1)} = WTH_{i+1} - WTH_i \quad (3.10)$$

Because the gray values of white image regions should be large in the result of  $WTH$ , the final multi scale white detail image regions should be the maximum gray values of all the extracted white detail image regions as follows [118, 133]:

$$f_w^D = \max_{0 \leq i \leq n} \{WTH_{i(i+1)}\} \quad (3.11)$$

where,  $f_w^D$  represents the final multi scale white detail image regions.

Similarly, the black detail image regions between neighboring scales  $i$  and  $i + 1$  can be expressed as follows [118, 133]:

$$BTH_{i(i+1)} = BTH_{i+1} - BTH_i \quad (3.12)$$

Because the gray values of black image regions should be large in the result of  $BTH$ , the final multi scale black detail image regions should be the maximum gray values of all the extracted black detail image regions as follows [118, 133]:

$$f_b^D = \max_{0 \leq i \leq n} \{BTH_{i(i+1)}\} \quad (3.13)$$

where,  $f_b^D$  represents the final multi scale black detail image regions.

An efficient image enhancement algorithm should not only enlarge the contrast between the bright and black image regions, but also enhance the image details. Therefore, to efficiently enhance images, the extracted white and black detail image regions are added on the extracted white and black image regions as follows [118, 133]:

$$f_w = f_w^C + f_w^D \quad (3.14)$$

$$f_b = f_b^C + f_b^D \quad (3.15)$$

Using the final extracted white and black image regions ( $f_w$  and  $f_b$ ) through multi scale top-hat transform, image  $f$  could be enhanced following [118, 133]:

$$f_{Enh} = f + f_w - f_b = f + (f_w^C + f_w^D) - (f_b^C + f_b^D) \quad (3.16)$$

$$f_{Enh} = f + \left( \max_{0 \leq i \leq n} \{WTH_i\} + \max_{0 \leq i \leq n} \{WTH_{i(i+1)}\} \right) - \left( \max_{0 \leq i \leq n} \{BTH_i\} + \max_{0 \leq i \leq n} \{BTH_{i(i+1)}\} \right) \quad (3.17)$$

where  $f_{Enh}$  denotes the resulting multi-scale enhanced image obtained.

### 3.1.3 Contrast measures

Theoretically, any enhanced image should allow better perception of desirable features in comparison to the original image. The performance of image enhancement algorithms is often difficult to quantify. In addition, it is not easy to find one correct and effective measure to quantify the quality of the enhanced image.

#### 3.1.3.1 Contrast Improvement Ratio (CIR)

Contrast which is defined as the difference in visual properties of pixels makes an object distinguishable from other objects and the background. In gray-scale images, contrast is determined by the difference in the brightness of the object and its surroundings. CIR (Wang et al., 2003 [134]) measures the effect of contrast enhancement on image quality [119]. In this method, the mean value of luminance is computed in two different concentric rectangular windows centered on each pixel. More specifically, CIR defines the local contrast as the following ratio:

$$c(x, y, z) = \frac{|p - a|}{|p + a|} \quad (3.18)$$

where  $p$  and  $a$  are the average values of gray levels in the center window and the surrounding window of the voxel location  $(x, y, z)$  respectively [134]. The inner window is a 3x3x3 and the outer one is a 7x7x7 box. Here  $c(x, y, z)$  is the contrast measurement and is in the range of  $[0, 1]$ . Finally, CIR is defined as the following ratio using the enhanced and original image

local contrast measurements.

$$CIR = \frac{\sum_{(x,y,z) \in R} (c(x,y,z) - \hat{c}(x,y,z))^2}{\sum_{(x,y,z) \in R} c^2(x,y,z)} \quad (3.19)$$

where  $R$  is the region of interest and  $c$  and  $\hat{c}$  are the local contrast measurements in original and enhanced images respectively [112].

### 3.1.3.2 Mean Squared Error (MSE)

The MSE is a measure of the quality of an estimator. It is always non-negative, and values closer to zero are better. According to Wang and Bovik [Wang and Bovik 2009], MSE remains the standard criterion for the assessment of signal quality and fidelity; it is the method of choice for comparing competing signal processing methods and systems, and, perhaps most importantly, it is the nearly ubiquitous preference of design engineers seeking to optimize signal processing algorithms. The goal of a signal fidelity measure is to compare two signals by providing a quantitative score that describes the degree of similarity/fidelity or, conversely, the level of error/distortion between them. To define MSE, it is assumed that given  $I$  and  $I_e$  be the original and enhanced image, respectively, where the size of image is  $m \times n$ . Then, MSE is computed as

$$MSE = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I_e(i,j) - I(x,y))^2 \quad (3.20)$$

### 3.1.3.3 Peak Signal-to-Noise Ratio (PSNR)

One problem with MSE is that it depends strongly on the image intensity scaling. To avoid this problem, PSNR scales the MSE according to the image range. Let  $I$  be an image with  $L$  gray levels and  $m \times n$  the size of image. Then, PSNR is defined as:

$$PSNR = 10 \log_{10} \left( \frac{(L-1)^2}{MSE} \right) \quad (3.21)$$

$$PSNR = 10 \log_{10} \left( \frac{\sum \sum (L-1)^2}{\sum \sum (I_e(i,j) - I(x,y))^2} \right) \quad (3.22)$$

A bigger value of PSNR indicates that the corresponding algorithm enhances images better and produces less noise.



### 3.1.3.4 Edge Content (EC)

Let  $I$  be an image, EC is computed as

$$EC = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} |\nabla I| \quad (3.23)$$

Contrast changes exist over the entire image and EC takes into account all the contrast changes even for pixels which are very close (adjacent) to each other.

## 3.2 Method proposed

To enhance the contrast of head-neck CT images, the use of multiscale top-hat morphological operators is proposed. The implementation was made using the equation 3.17. Enhancement using the multiscale top-hat morphological operator allows increased contrast in regions, especially at the edges, where the original image has low contrast. But one difficulty is to determine the appropriate iterations number to stop the algorithm execution, as well as to identify that the obtained image has better contrast characteristics than the original image. At each iteration, the radius of the structural element is increased. For 2D images, a disc is used, whereas for 3D images a sphere (ball) is used. Normally, the way to stop the algorithm is to define a number of iterations which is given by the user.

For the first tests performed, the size of the structural element was defined between 1 and 20. Afterwards, the images were examined in search of the image that presented a better contrast definition. However, this task is quite manual and depends directly on the user's experience. In addition, this selection is quite subjective. For this reason, the literature was revised in search of a stopping criterion to stop the contrast enhancement algorithm.

In the revised literature, a stopping criterion proposed by Hassanpour et. al. [112] was found. They proposed the use of CIR to determine the final image in the iterative process. In each step, the contrast improvement is measured by CIR through comparing the previous and the current enhanced images. This procedure continues while the returned values from CIR have evolved in an increasing manner. It is CIR that compares the images resulted from two consecutive stages. When CIR returns a smaller value, the procedure ends and the resulting image is the previous one. This can be defined as:

$$CIR_i - CIR_{i-1} > 0 \quad (3.24)$$

where  $i$  indicates the iteration. If the value of the present comparison is larger than the previous one, the process continues (difference is greater than zero). On the other hand, if

CIR returns a smaller value, this means that the contrast of the new enhanced image is less than that of the previous image (difference is less than zero). Therefore, the image of the previous step is selected as the best enhanced image.

Nevertheless, the results were not satisfactory for the initial tests, for this reason, the equations of the contrast measurement and the stopping criterion were revised, which led us to propose a variation in the CIR equation and to propose a new stopping criterion to stop the contrast enhancement algorithm.

### 3.2.1 Contrast Improvement Ratio Revisited (CIRR)

After several experiments was detected that not for all cases traditional CIR works fine. The problem occurs when  $p$  and  $a$  values have different sign. In these cases the contrast measurement is not in the range  $[0,1]$ . The above generates an error in the final value of CIR. To correct this calculation error it is necessary to validate the signs of  $p$  and  $a$  as follows:

$$c(x, y, z) = \begin{cases} \frac{|p-a|}{|p+a|} & \text{if } \text{sign}(p) = \text{sign}(a) \\ \frac{|(a+(a-p))-a|}{|(a+(a-p))+a|} & \text{if } p < 0 \text{ and } a > 0 \\ \frac{|p-(p+(p-a))|}{|p+(p+(p-a))|} & \text{if } p > 0 \text{ and } a < 0 \end{cases} \quad (3.25)$$

Finally, CIRR is defined in the same way that CIR as follow:

$$CIRR = \frac{\sum_{(x,y,z) \in R} (c(x, y, z) - \hat{c}(x, y, z))^2}{\sum_{(x,y,z) \in R} c^2(x, y, z)} \quad (3.26)$$

The initial tests showed better results with this modification of the original criterion.

### 3.2.2 Stopping criterion proposed

Based on the criteria proposed by Hassanpour et. al. [112] and João et. al. [135], the following stopping criterion is proposed.

In each step, the contrast enhancement is measured by CIRR through comparing the absolute difference between previous and current enhanced images. This value is divided by the absolute value of CIRR of current iteration to obtain the *CIRR index*. The index is computed as:

$$CIRR \text{ index}_i = \frac{|CIRR_i - CIRR_{i-1}|}{|CIRR_i|} \quad (3.27)$$

and the enhancement process is iterated while  $CIRR\ index_i$  is greater than a specific constant  $\varepsilon$  defined by the user. In general, the stopping criterion is defined as:

$$\frac{|CIRR_i - CIRR_{i-1}|}{|CIRR_i|} > \varepsilon \quad (3.28)$$

The choice of  $\varepsilon$  is influenced by the need for a small value to identify a convergence of solution, and large enough to make the iterative procedure less computationally demanding. According to João et. al. [135], who defined a stopping criterion for anisotropic diffusion,  $\varepsilon$  value can be  $10^{-2}$ .

### 3.3 Experimental results

The experimentation process was divided into two parts. In the first part, the level of contrast using the MSTH technique and the behavior of the stopping criteria, the proposed one (BF) and the one defined by Hassanpour et al. (HSS) using 2D images is obtained. In the second part, based on the results achieved previously, the MSTH technique and the stopping criteria are applied to 3D images.

In addition to the use of the stopping criteria, a qualitative measure of contrast enhancement, the Canny edge detection filter [136] is applied to the original and resulting images. This allows to observe if with the improvement of contrast a greater amount of edges can be detected. For this, Canny filter is used with predefined parameters.

#### 3.3.1 2D case

Evaluation was conducted on five traditional 2D images that are used in image processing (baboon, barbara, boat, cameraman and lena). In this evaluation a disk-shaped structuring element (SE) was used. The minimum size (radius) of SE was 1 and the maximum was defined to 20. This was made in this way to evaluate the behavior of the HSS and BF stopping criteria.

Table 3.1 shows the result of applying the multi-scale top-hat morphological enhancement algorithm for four images (baboon, barbara, cameraman and lena). In the table 3.1 are presented the results of evaluating CIRR for each iteration, called value and the HSS and BF indexes for each one. CIRR value was computed using an inner window of radius 1 and outer window of radius 2 (inner window is a 3x3 and the outer one is a 5x5 square).

According to the definition of the HSS and BF indices, the values in which the enhancement algorithm should be stopped are highlighted in table 3.1. In red, the stop iterations are presented for the HSS criterion and in blue for the BF criterion. For the baboon image case,

HSS requires two iterations, whereas BF requires five iterations. For barbara image, HSS needs one iteration and BF eight iterations. For cameraman image, two and five iterations are required respectively. Finally, for lena image, they take three and seven iterations.

Table 3.1: HSS and BF indexes using CIRR contrast measure.

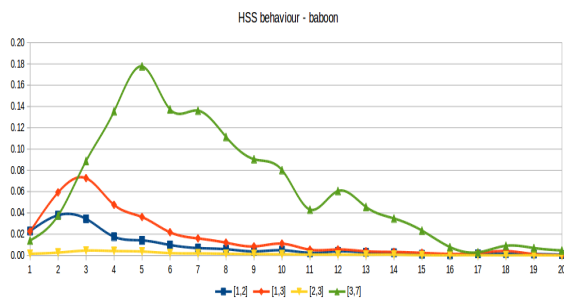
Iter	baboon			barbara			cameraman			lena		
	CIRR			CIRR			CIRR			CIRR		
	value	index		value	index		value	index		value	index	
		HSS	BF		HSS	BF		HSS	BF		HSS	BF
1	0.0229	0.0229	1.0000	0.0136	<b>0.0136</b>	1.0000	0.0176	0.0176	1.0000	0.0031	0.0031	1.0000
2	0.0607	<b>0.0378</b>	0.6230	0.0238	0.0102	0.4276	0.0550	<b>0.0374</b>	0.6802	0.0107	0.0076	0.7088
3	0.0951	0.0344	0.3618	0.0327	0.0089	0.2730	0.0809	0.0259	0.3205	0.0219	<b>0.0112</b>	0.5126
4	0.1126	0.0175	0.1556	0.0374	0.0047	0.1253	0.1038	0.0228	0.2201	0.0307	0.0088	0.2863
5	0.1267	0.0141	<b>0.1115</b>	0.0453	0.0079	0.1747	0.1225	0.0187	<b>0.1530</b>	0.0411	0.0103	0.2511
6	0.1365	0.0098	0.0719	0.0514	0.0061	0.1187	0.1314	0.0089	0.0678	0.0485	0.0074	0.1531
7	0.1435	0.0069	0.0483	0.0582	0.0069	0.1181	0.1378	0.0064	0.0463	0.0550	0.0065	<b>0.1178</b>
8	0.1493	0.0058	0.0392	0.0649	0.0067	<b>0.1033</b>	0.1437	0.0059	0.0414	0.0605	0.0055	0.0910
9	0.1531	0.0038	0.0248	0.0701	0.0051	0.0729	0.1484	0.0047	0.0314	0.0661	0.0057	0.0859
10	0.1580	0.0049	0.0309	0.0741	0.0041	0.0551	0.1513	0.0029	0.0195	0.0716	0.0054	0.0759

Two variables that arise when calculating the CIRR measure are the size of the inner and outer windows. Therefore, the HSS and BF indexes were calculated by varying these variables. The curves representing the behavior of the HSS and BF indexes for the baboon and lena images are shown in Figure 3.1 and the number of iterations for all images is presented in Table 3.2. It can be observed that the behavior does not vary significantly using the BF index while the HSS index shows a greater variation. It can be concluded that the BF index is less sensitive to the size of the windows to calculate the CIRR value. In addition, it is recommended to select small windows size to reduce the time required to compute the CIRR values.

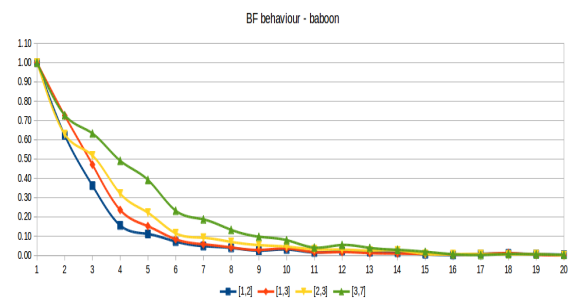
According to Table 3.2, the images with the highest variation in the iterations number using HSS index are barbara and lena. While for the iterations number using BF index the greater variation is presented for the boat. This behavior that is identified through Figure 3.1 and Table 3.2 may be explained by the following reasons:

- The stopping criterion proposed by Hassanpour et. al. [112] only takes into account the increasing part in the function and this means that the number of iterations is smaller than the BF criterion proposed.

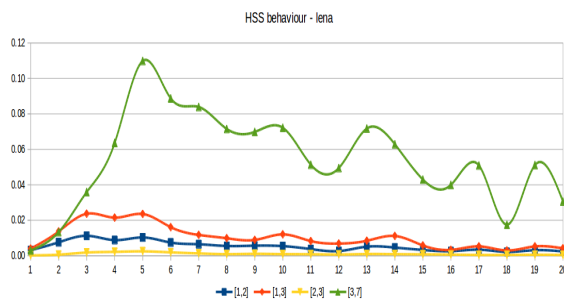
- According to the function behavior, the steady state of the contrast enhancement has not achieved a stabilization point generating in the case of HSS a sub-image enhancement.
- When the size of the windows used for the calculation of CIRR is small, for example, 1 or 2, the HSS criterion is more sensitive, as can be seen in the case of the barbara image.
- In the case of the BF index proposed, it has a more uniform behavior independent of the windows size used. The BF criterion includes more information in the enhancement stabilization process.



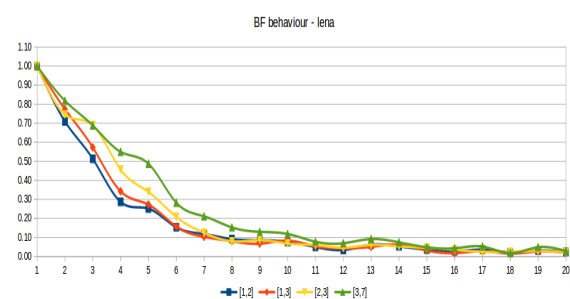
a) HSS behavior for baboon image



c) BF behavior for baboon image



b) HSS behavior for lena image



d) BF behavior for lena image

Figure 3.1: HSS and BF computation using different window sizes.

Image	Technique	[inner, outer] radius			
		[1,2]	[1,3]	[2,3]	[3,7]
baboon	HSS	2	2	2	4
	BF	5	5	6	7
barbara	HSS	1	2	4	4
	BF	8	8	8	11
boat	HSS	2	2	2	4
	BF	8	6	8	8
cameraman	HSS	2	2	3	2
	BF	5	5	6	6
lena	HSS	3	2	4	4
	BF	7	7	7	9

Table 3.2: Required iterations using HSS and BF indexes.

To obtain more information about the impact of the results achieved when applying the MSTH filter and the stopping criteria, a set of tests was performed using the 2D images. The objective is to compute the edges of the images using the Canny algorithm with predefined parameters. In addition, a fixed size of the inner and outer window (1 and 3 respectively) to calculate CIRR was used. The results obtained are shown in Table 3.3, in which the first column corresponds to the original image and the edges obtained, the second column is the resulting image when the HSS stopping criterion is used, and finally, the third column corresponds to the results of applying the BF criterion.

Table 3.3: Edges of original, HSS, and BF images.

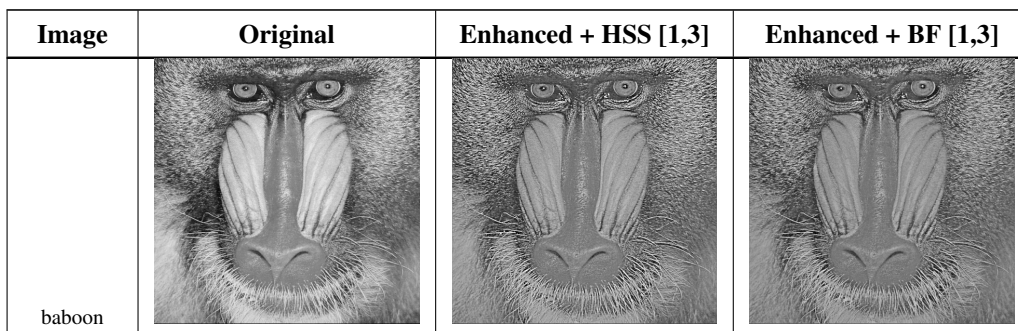


Table 3.3: Edges of original, HSS, and BF images.














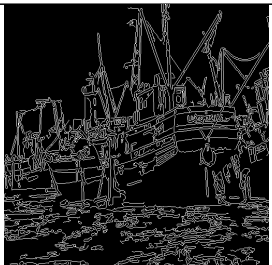
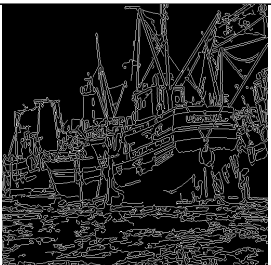

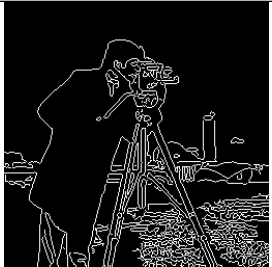






Image	Original	Enhanced + HSS [1,3]	Enhanced + BF [1,3]
edges			
barbara			
edges			
boat			
edges			



Table 3.3: Edges of original, HSS, and BF images.

Image	Original	Enhanced + HSS [1,3]	Enhanced + BF [1,3]
cameraman			
edges			
lena			
edges			

As can be seen in Table 3.3, the edges map of the obtained images using the BF stopping criterion presents more details than the images obtained using the HSS criterion. In addition, the original images have fewer edges than the enhanced images.

Additionally, the profile of one of the rows of the lena image was defined, both for the original image and those obtained using the HSS and BF stopping criterion. Figure 3.2 shows the zoom in the eye area for the three images respectively. In Figure 3.3, the profile for row 100 between columns 200 and 400 is shown.



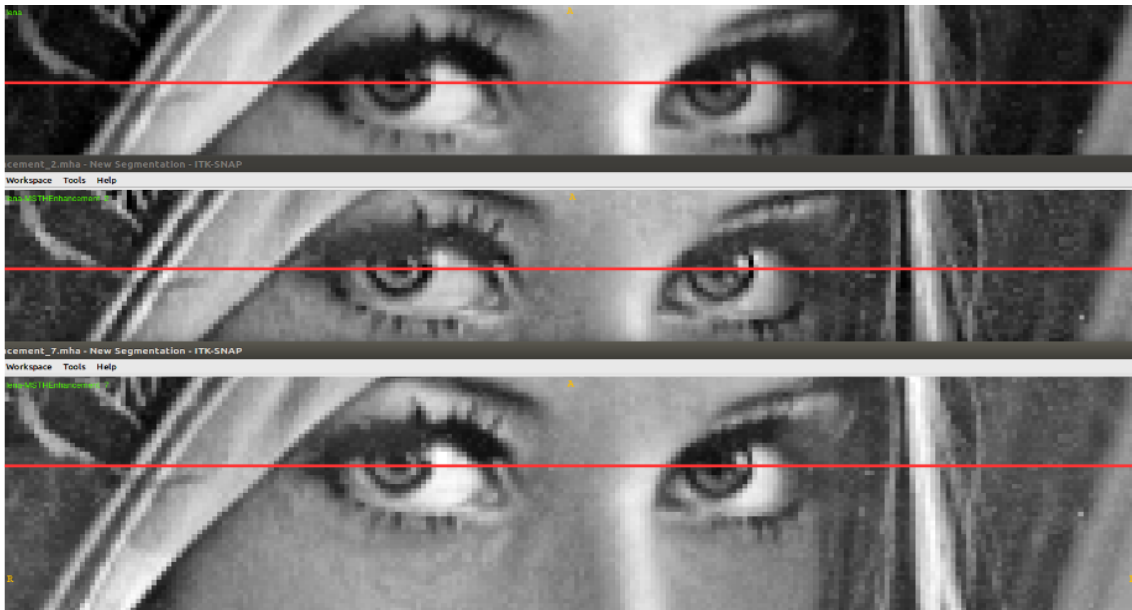


Figure 3.2: Zoom of lena image in eye region.

Figure 3.3 presents the profile behavior of images in Figure 3.2, the greatest enhancement of the lena image using the stopping criterion BF is obtained (green line).

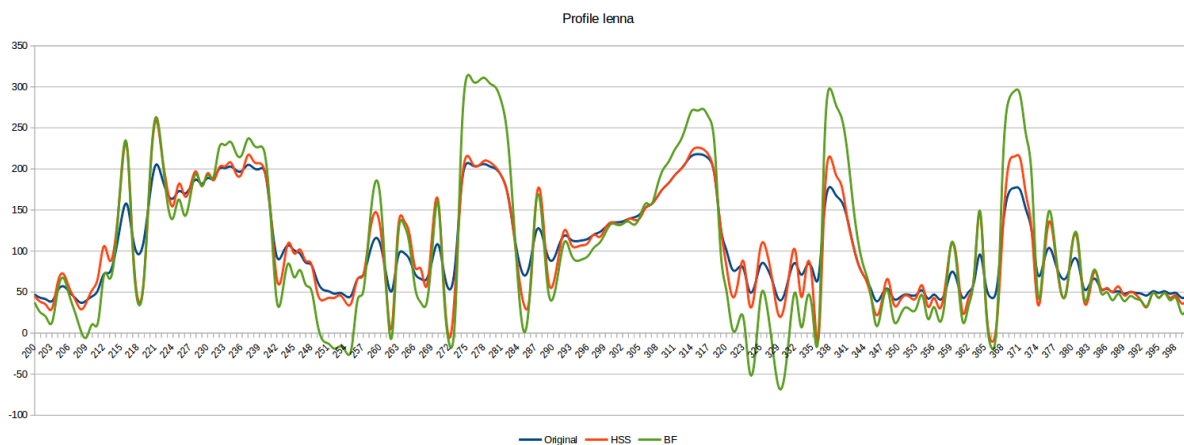


Figure 3.3: Profile enhancement behavior of lena images.

### 3.3.2 3D case

For the 3D tests, ten medical images of head-neck were used. These images have different sizes and resolution. Based on the results obtained in the 2D images, it was decided to use 3x3x3 and 5x5x5 windows (inner window radius is 1 and the outer one is 2) to compute

the HSS and BF stopping criteria. A ball was used as the structural element (SE) of the morphological operator. The radius of the ball is increased at each iteration to obtain multi-scale contrast enhancement.

The number of iterations initially defined to see the behavior of the stopping criteria is 10. Table 3.4 shows the results obtained for images 1, 3, 5 and 7. For each image the value of the contrast measure (CIRR) and the values of the HSS and BF indexes. In addition, Table 3.4 highlights the values in which the stopping criteria are reached, in red the values for HSS and in blue the values for BF.

Table 3.4: HSS and BF indexes using CIRR contrast measure.

Iter	image1			image3			image5			image7		
	CIRR			CIRR			CIRR			CIRR		
	value	index		value	index		value	index		value	index	
		HSS	BF		HSS	BF		HSS	BF		HSS	BF
1	0.0392	<b>0.0392</b>	1.0000	0.0289	0.0289	1.0000	0.0319	0.0319	1.0000	0.0377	<b>0.0377</b>	1.0000
2	0.0753	0.0361	0.4792	0.0601	<b>0.0312</b>	0.5191	0.0641	<b>0.0322</b>	0.5027	0.0734	0.0357	0.4861
3	0.1015	0.0262	0.2581	0.0833	0.0232	0.2783	0.0857	0.0216	0.2525	0.0988	0.0253	0.2567
4	0.1137	0.0122	<b>0.1073</b>	0.0951	0.0118	<b>0.1241</b>	0.0966	0.0109	<b>0.1125</b>	0.1122	0.0134	<b>0.1198</b>
5	0.1226	0.0089	0.0724	0.1053	0.0102	0.0969	0.1065	0.0099	0.0932	0.1233	0.0111	0.0903
6	0.1271	0.0046	0.0361	0.1109	0.0056	0.0502	0.1129	0.0064	0.0563	0.1295	0.0062	0.0478
7	0.1301	0.0029	0.0225	0.1145	0.0036	0.0315	0.1174	0.0046	0.0390	0.1339	0.0044	0.0325
8	0.1322	0.0021	0.0158	0.1171	0.0026	0.0225	0.1206	0.0032	0.0262	0.1364	0.0025	0.0187
9	0.1338	0.0016	0.0122	0.1192	0.0020	0.0170	0.1242	0.0035	0.0285	0.1378	0.0014	0.0099
10	0.1348	0.0010	0.0071	0.1205	0.0014	0.0113	0.1269	0.0028	0.0219	0.1388	0.0010	0.0075

In Table 3.4 is observed that the behavior of the stopping criterion HSS for images 1 and 7 is totally decreasing and the contrast enhancement algorithm is stopped after the first iteration, whereas for images 3 and 5 it stops at the second iteration. For the stopping criterion BF the behavior is more constant and the execution of the contrast enhancement algorithm is stopped at iteration 4 for all the images.

To complement the analysis performed for 2D images, the behavior of the stopping criterion is calculated using different sizes for the inner and outer window in the computation of the CIRR contrast measurement. Figure 3.4 shows the results obtained for image 1 and 3 using both stopping criteria. In the case of HSS, it can be observed that as the outer radius increases compared to the inner radius, the stopping criterion is more reliable. For example,

using radio (1, 3) for an inner and outer window (red line in Figure 3.4) gives a maximum in the second iteration. In the case of BF, the behavior is the same for radio windows (1, 2) as for radio windows (1, 3). The required number of iterations is equal to 4.

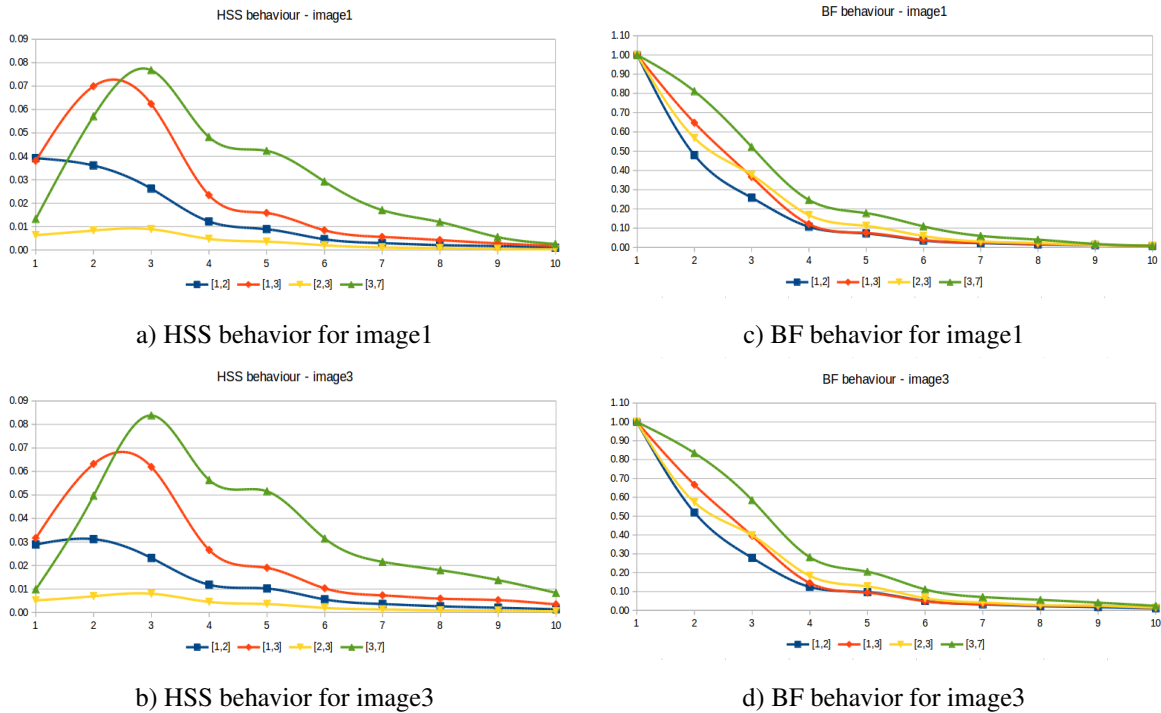


Figure 3.4: HSS and BF computation using different window sizes.

The result of applying the different radii for the inner and outer window in the calculation of the CIRR contrast measure that is used by the two stopping criteria of the MSTH contrast enhancement algorithm is presented in Table 3.5. As can be seen, the results obtained by stopping criterion BF is the same for the 10 test images, that is, it is constant regardless of the size of the image and its spacing. On the other hand, the stopping criterion HSS varies when the window radius is (1, 2) for images 1, 2, 7 and 8. For other windows radii, the number of iterations is the same for all images except for the image10.

Table 3.5: Iterations required using HSS and BF stopping criteria

Image	Stopping criteria	[inner, outer] radius			
		[1,2]	[1,3]	[2,3]	[3,7]
image1	HSS	1	2	3	3
	BF	4	4	5	6

Table 3.5: Iterations required using HSS and BF stopping criteria

Image	Stopping criteria	[inner, outer] radius			
		[1,2]	[1,3]	[2,3]	[3,7]
image2	HSS	1	2	3	3
	BF	4	4	5	6
image3	HSS	2	2	3	3
	BF	4	4	5	6
image4	HSS	2	2	3	3
	BF	4	4	5	6
image5	HSS	2	2	3	3
	BF	4	4	5	6
image6	HSS	2	2	3	3
	BF	4	4	5	6
image7	HSS	1	2	3	3
	BF	4	4	5	6
image8	HSS	1	2	3	3
	BF	4	4	5	6
image9	HSS	2	2	3	3
	BF	4	4	5	6
image10	HSS	2	3	3	3
	BF	4	4	5	6

In addition, as mentioned above, it appears that the number of iterations using the stopping criterion HSS is half of the iterations required using the stopping criterion BF.

Table 3.6 presents a comparative analysis of the 10 original images, those obtained using the stopping criterion HSS and those obtained using the criterion BF. In addition, it presents the edges obtained using the Canny algorithm with predefined parameters.

Table 3.6: Edges of original, HSS, and BF images.

Image	Original	Enhanced+HSS [1,2]	Enhanced+BF [1,2]
-------	----------	--------------------	-------------------

Table 3.6: Edges of original, HSS, and BF images.



















Image	Original	Enhanced+HSS [1,2]	Enhanced+BF [1,2]
image1			
			
image2			
			
image3			
			

Table 3.6: Edges of original, HSS, and BF images.

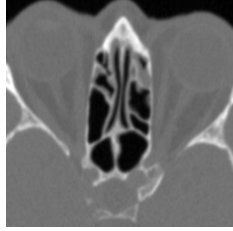

















Image	Original	Enhanced+HSS [1,2]	Enhanced+BF [1,2]
image4			
			
image5			
			
image6			
			

Table 3.6: Edges of original, HSS, and BF images.




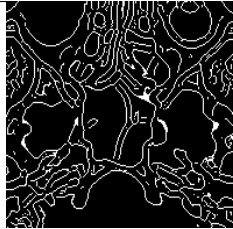

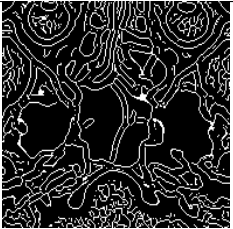
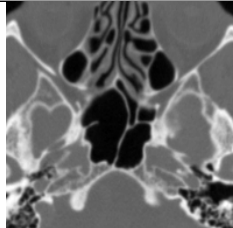
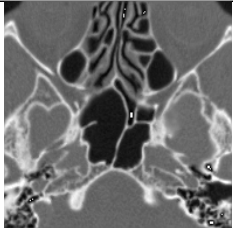
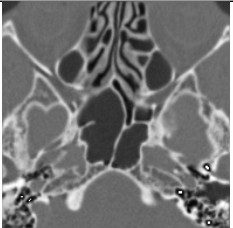
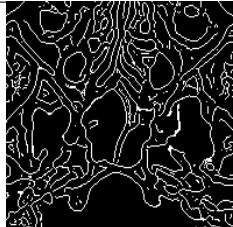












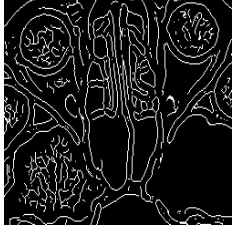
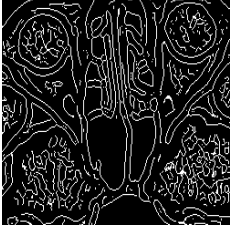
Image	Original	Enhanced+HSS [1,2]	Enhanced+BF [1,2]
image7			
			
image8			
			
image9			
			

Table 3.6: Edges of original, HSS, and BF images.

Image	Original	Enhanced+HSS [1,2]	Enhanced+BF [1,2]
image10			
			

In the case of the stopping criterion BF, which allows propagating more iterations in the contrast enhancement algorithm, a number of edges obtained are greater than in the other cases.

To see the behavior of the two stopping criteria in more detail, for image1 a slice was taken and the image was zoomed in, as can be seen in Figure 3.5. The slice 170 of image1 was taken because it presents several areas where airways are separated by thin tissue.

In the Figure 3.6, it can be appreciated slightly that the obtained image using the number of iterations generated by the criterion BF allows obtaining a better contrast.

The profile of the slice 170 of image1 and its enhanced images was analyzed for column 100 as shown in Figure 3.6. Column 100 was selected because it has areas of thin tissue between the airways and low contrast.

It can be seen that the profile of the enhanced image using the stopping criterion BF (green line) is better than that obtained using the criterion HSS (red line). There is a significant improvement in the areas that separate the cavities from the airways.



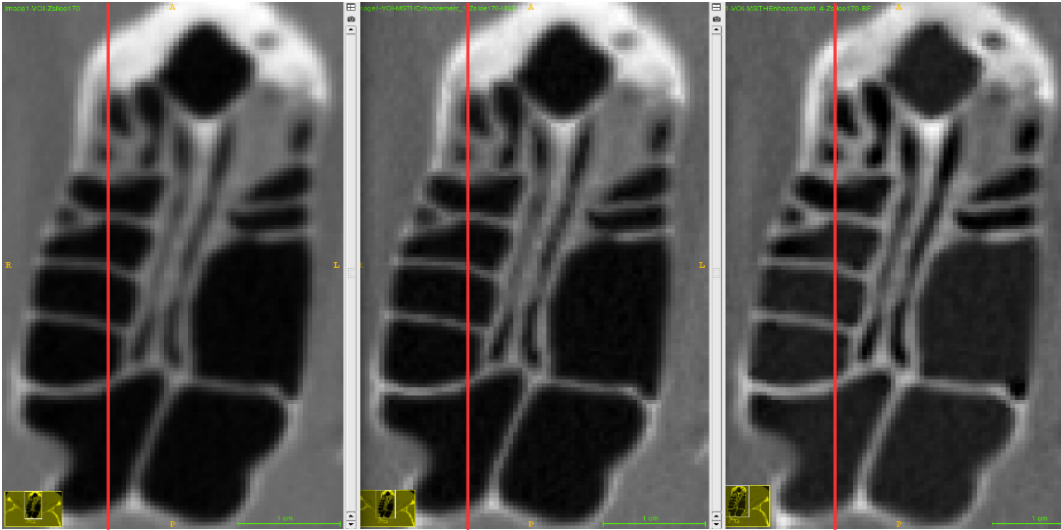


Figure 3.5: Zoom of the slice 170 of image1 (left:original image, middle:HSS enhanced image, right:BF enhanced image).

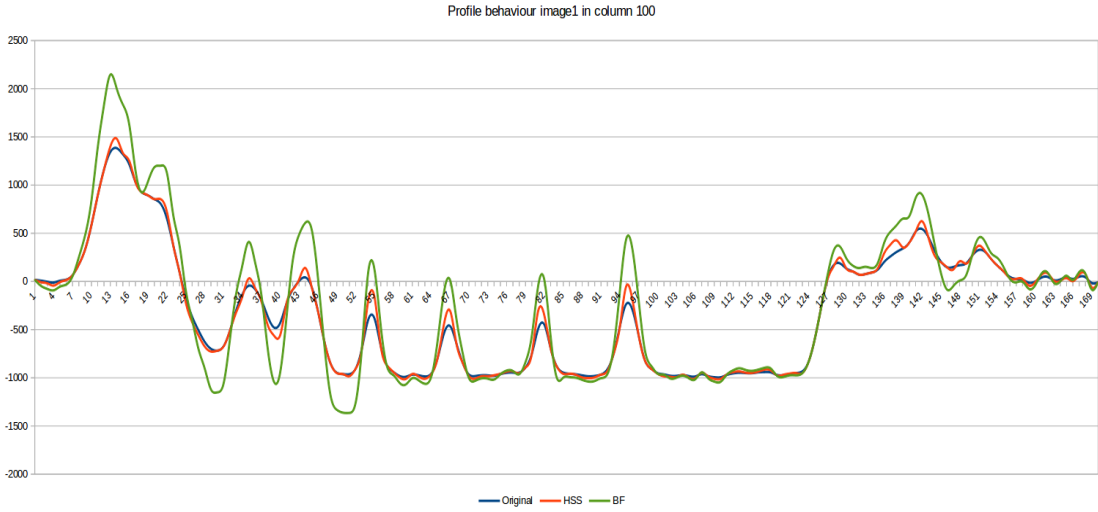


Figure 3.6: Profile of the image1 and its enhanced images (blue line:original image, red line: HSS-enhanced image, green line: BF-enhanced image).

Additionally, it can be seen in figure 3.6 that the improvement in the contrast when using the stopping criterion HSS does not allows obtaining a significant difference in the areas mentioned.



## Chapter 4

# Smoothing using nonlinear anisotropic diffusion

“Success is not final, failure is not fatal: it is the courage to continue that counts.”

– *Winston Churchill.*

The conventional image denoising techniques, such as averaging filter, median filter or 2D Gaussian filter are efficient in reducing the amount of noise, but also have the disadvantage of blurring the image edges [67, 137]. For this reason, numerous edge preserving techniques based on Partial Differential Equations (PDEs) have been developed [137]. In this case, the image affected by noise is smoothed by the diffusion techniques which modify it via a PDE. Diffusion filters and their discrete implementation using finite differences were introduced in the image processing area by Perona and Malik [138]. The general diffusion scheme includes both the Gaussian filter and the nonlinear isotropic and anisotropic diffusion, formalized by the same compact mathematical base.

Specifically, nonlinear anisotropic diffusion filtering is a procedure based on nonlinear evolution PDEs which seeks to improve images qualitatively by removing noise while preserving details and even enhancing edges. However, well-known implementations are sensitive to parameters which are necessarily tuned to sharpen a narrow range of edge slopes; otherwise, edges are either blurred or staircased [139].

For the above reasons, the purpose of this chapter is to select a nonlinear anisotropic diffusion filter which sharpen edges over a wide range of slope scales and which reduce noise while conserving feature boundaries. To this end, a comparative analysis of five diffusion methods is performed. Four of them are anisotropic nonlinear diffusion methods and the fifth is the Perona Malik method.

Another purpose of this chapter is to define a stopping criterion for the diffusion process.

To meet this goal, two stopping criteria were compared. The first is the stopping criterion proposed by Joao et. al. [135], which is based on the Mean Squared Error (MSE). The second is a proposed method based on the CIRR contrast measure. According to the performed tests, better results were found using the proposed algorithm.

## 4.1 State of the art

Medical images typically suffer from one or more of the following imperfections, low resolution (in the spatial and spectral domains), high level of noise, low contrast, geometric deformations and/or presence of imaging artifacts. These imperfections can be inherent to the imaging modality (e.g., X-rays offer low contrast for soft tissues, ultrasound produces very noisy images, and metallic implants will cause imaging artifacts in CT) or the result of a deliberate trade-off during acquisition. For example, finer spatial sampling may be obtained through a longer acquisition time. However that would also increase the probability of patient movement and thus blurring. To remove noise while preserving details and even enhancing edges techniques based on Partial Differential Equations (PDEs) have been used. The idea of using the PDE diffusion equations in image denoising and restoration arose from the use of the Gaussian filter in multiscale image analysis. Convolving an image with a two or three dimensional Gaussian filter is equivalent to the solution of the diffusion equation in two or three dimensions [137].

The applications of PDEs to image processing date back to the 1960s [140, 141]. However, the research work about PDEs in image processing and computer vision can be traced back to the earliest Nagao et. al. (1979) and Rudin et. al. (1992), study of image smoothing and image enhancement. But this technique did not draw much attention until the introduction of the concept of scale space by Koenderink [142] and Witkin [143] in the 1980s. Then, Perona and Malik's work on anisotropic diffusion [138] further drew great interest from researchers towards PDE-based methods. Nowadays, PDEs have been successfully applied to many problems in image processing and computer vision [144–148], e.g., denoising [138], enhancement [149], inpainting [150], segmentation [151], stereo and optical flow computation.

Nonlinear anisotropic diffusion is a variant of the heat equation, generalized in two regards: nonlinearity and anisotropy. Nonlinearity in diffusion means that diffusion tensors are automatically generated from the processed image. Anisotropy means that the smoothing induced by the PDE can be favored in some directions and prevented in others. This is specified by local eigenvectors and eigenvalues of the diffusion tensor field [152]. Diffusion coefficients are thus location and direction dependent, generalizing the approach of Perona and Malik [138] which is only location dependent.

Nonlinear anisotropic diffusion is a powerful image processing technique, which allows to simultaneously remove the noise and enhance sharp features in two or three dimensional images. Anisotropic diffusion is understood here in the sense of Weickert [153], meaning that diffusion tensors are anisotropic and reflect the local orientation of image features.

Weickert [154] proposed two nonlinear anisotropic diffusion algorithms. The first one is called Edge Enhancing Diffusion (EED), which allows smoothing while preserving the edges. The second one is called Coherence Enhancing Diffusion (CED), which allows smoothing based on the structures (flow-like) present in the images. Based on the filters defined by Weickert, several methods have been proposed. For example, Bazan et. al. [155] proposed a new approach based on nonlinear anisotropic diffusion and bilateral filtering for electron tomography of mitochondria. Dong et. al. [156] introduced a source term in the CED filter to restore the initial image and contrast lost by pure diffusion filters. Surya [] proposed an adaptive coherence enhancement diffusion filter (CED) combining anisotropic diffusion and diffusion functions derived from the structural tensor. Mirebeau et. al. [157] proposed two variants to the Weickert's algorithms. The first is associated with the EED algorithm, which is called Conservative variant of EED (cEED). The second is called Conservative variant of CED (cCED). The main distinction lies in the definition of the diffusion parameters of the diffusion tensor.

The methods proposed by Perona-Malik (nonlinear isotropic), Weickert (EED and CED) and Mirebeau et. al. (cEED and cCED) to fulfil the first objective of this chapter were selected. This consists of comparing and selecting the method that allows smoothing head-neck CT images containing the upper airways without attenuating the border information.

This chapter is an attempt to summarize PDE's and their solutions applied to image diffusion. The chapter first presents in section 4.1.1 the fundamental of PDEs in image processing. Next, the diffusion process is covered in section 4.1.2. Then, smoothing using anisotropic diffusion based on PDEs is presented in section 4.1.3. The chapter also covers in section 4.2 a brief description of the nonlinear diffusion methods selected. Then, in section 4.3 and section 4.4, stopping criteria of diffusion process and the proposed criterion are presented. Finally, experimental results are presented in section 4.5.

### 4.1.1 PDEs in image processing

Many mathematical approaches have been investigated for applications in image processing and computer vision (e.g., fractals and self-similarity, wavelets, pattern theory, stochastic point process, random graph theory) [146]. In particular, methods based on PDEs have been extremely popular in the past few years [158–160]. In image processing, methods based on PDEs consider to images as approximations of continuous objects, namely functions. The

basic idea is to deform a given function (image) with a PDE, and obtain the desired result as the PDE solution. Sometimes, as in the case of color images, a system of coupled PDEs is used. The challenge behind this technique is in the design and analysis of these PDEs [146]. Ideas on the use of PDEs in image processing go back at least to Gabor and, a bit more recently, to Jain [146, 158]. However, the field really took off thanks to the independent works of Koenderink [142] and Witkin [143]. These researchers rigorously introduced the notion of scale space, that is, the representation of images simultaneously at multiple scales [146].

In general, a PDE is an equation stating a relationship between function of two or more independent variables and the partial derivatives of this function with respect to these independent variables. The dependent variable  $f$  is used as a generic dependent variable throughout. In most problems in engineering and science, the independent variables are either space  $(x, y, z)$  or space and time  $(x, y, z, t)$ . The dependent variable depends on the physical problem being modeled. The order of a PDE is the order of the highest derivative in the relation. Hence for a positive integer  $m$ , the general form of an  $m$ th-order PDE in a domain  $\Omega \subset \mathbb{R}^n$  is given by

$$F(x, u, \nabla u(x), \nabla^2 u(x), \dots, \nabla^m u(x)) = 0, \text{ for } x \in \Omega \quad (4.1)$$

Here  $F$  is a function which is continuous in all its arguments, and  $u$  is a  $C^m$ -function in  $\Omega$ . A  $C^m$ -solution  $u$  satisfying the above equation in the pointwise sense in  $\Omega$  is often called a classical solution.

PDEs can be classified as linear or nonlinear. **Linear PDE** is one in which all of the partial derivatives appear in linear form and none of the coefficients depends on the dependent variable, i.e., there is no product of the dependent variables or its derivatives. The coefficients may be functions of the independent variables, in which case the PDE is a linear with variable coefficients. On the other hand, **nonlinear PDE** contains coefficients that depend on the dependent variable, or the derivatives appear in a nonlinear form. In addition, if a PDE is linear in its highest order derivatives, it is called a **quasi-linear PDE**.

Additionally, PDE is called **homogeneous** if the equation does not contain a term independent of the unknown function and its derivatives. In other words, when the dependent variable (and its derivatives) appear in terms with degree exactly one. A PDE is **non-homogeneous** if it contains a term that does not depend on the dependent variable, that is, when it contains terms which only depend on the independent variable.

According to Suri et. al. [161], PDE's have recently dominated the fields of computer vision, image processing and applied mathematics due to the following abilities:

1. to transform a segmentation modeling problem into a PDE framework.
2. to embed and integrate regularizers into these models.

3. to solve PDE's using finite difference methods (FDM).
4. to link between PDE's and the level set framework for implementing finite difference methods.
5. to extend the PDE framework from 2-D to 3-D or even higher dimensions.
6. to control the degree of PDE in the image processing domain.
7. to provide solutions in a fast, stable and closed form.
8. to handle interactively image segmentation in the PDE framework.

In particular, several methods have been presented in noise removal and smoothing using PDEs. This has been used for quite some time but recently, robust techniques for image smoothing have been developed (see Perona and Malik [138], Gerig et al. [162], Alvarez et al. [163], Kimia et al. [164], Sapiro et al. [146], Caselles et al. [145], Weickert [153], Black et al. [165], Arridge et al. [166], Bajla et al. [167], Olver et al. [168], Scherzer et al. [169], Romeny et al. [170] and Nielsen et al. [171]).

#### 4.1.1.1 PDEs classification

An important ingredient of a systematic theory of PDEs is a classification scheme which identifies classes of equations with common properties. The “class” of an equation determines the nature of boundary and initial conditions which may be imposed, the nature of singularities which solutions may have and the nature of methods which can be used to approximate a solution. For example, a few problems are governed by a single first-order PDE, numerous problems are governed by a system of first-order PDEs, some problems are governed by a single second-order PDE, numerous problems are governed by a system of second-order PDEs, and a few problems are governed by fourth-order PDEs. The classification of PDEs is most easily explained for a single second-order PDE. The general quasilinear (i.e., linear in the highest-order derivative) second-order nonhomogeneous PDE in two independent variables is:

$$A f_{xx} + B f_{xy} + C f_{yy} + D f_x + E f_y + F f = G \quad (4.2)$$

where the coefficients  $A$  to  $C$  may depend on  $x$ ,  $y$ ,  $f_x$ , and  $f_y$ , the coefficients  $D$  to  $F$  may depend on  $x$ ,  $y$ , and  $f$ , and the nonhomogeneous term  $G$  may depend on  $x$  and  $y$ . The classification of Equation 4.2 depends on the sign of the discriminant,  $B^2 - 4AC$ , as follows:

Class	$B^2 - 4AC$	Example	
Elliptic	$B^2 - 4AC < 0$	$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = 0$	Laplace equation
Parabolic	$B^2 - 4AC = 0$	$\frac{\partial f}{\partial t} = \alpha \frac{\partial^2 f}{\partial x^2}$	Diffusion equation
Hyperbolic	$B^2 - 4AC > 0$	$\frac{\partial^2 f}{\partial t^2} = c^2 \frac{\partial^2 f}{\partial x^2}$	Wave equation

Table 4.1: Types of PDEs and examples

This classification depends only on the coefficients of the highest order derivatives.

#### 4.1.1.2 Physical problems

Physical problems fall into one of the following three general classifications: equilibrium problems, propagation problems and eigenproblems. Each of these three types of physical problems has its own special features, its own particular type of governing partial differential equation, and its own special numerical solution method. A clear understanding of these concepts is essential if meaningful numerical solutions are to be obtained.

First, equilibrium problems are steady-state problems in closed domains  $D(x, y)$  in which the solution  $f(x, y)$  is governed by an elliptic PDE subject to boundary conditions specified each point on the boundary  $\Omega$  of the domain. Equilibrium problems are jury problems in which the entire solution is passed on by a jury requiring satisfaction of all internal requirements (i.e., the PDE) and all the boundary conditions simultaneously. A classical example of an equilibrium problem governed by an elliptic PDE is steady heat diffusion (i.e., conduction) in a solid. The governing PDE is Laplace equation.

Second, propagation problems are initial-value problems in open domains (open with respect to one of the independent variables) in which the solution  $f(x, t)$  in the domain of interest  $D(x, t)$  is marched forward from the initial state, guided and modified by boundary conditions. Propagation problems are governed by parabolic or hyperbolic PDEs. The majority of propagation problems are unsteady problems. The diffusion equation is an example of an unsteady propagation problem. A classical example of a propagation problem governed by a parabolic PDE is unsteady heat diffusion in a solid. The governing PDE is the diffusion equation.

Finally, eigenproblems are special problems in which the solution exists only for special values (i.e., eigenvalues) of a parameter of the problem. The eigenvalues are to be determined in addition to the corresponding configuration of the system.

In conclusion, parabolic and hyperbolic PDEs govern propagation problems and elliptic PDEs govern equilibrium problems.



### 4.1.1.3 Initial values and boundary conditions

A differential equation governs a family of solutions. A particular member of the family of solutions is specified by the auxiliary conditions imposed on the differential equation. The typical problem in PDEs consists of finding the solution of a PDE (or a system of PDEs) subject to certain boundary and/or initial conditions. The nature of boundary and initial conditions which lead to well-posed problems depends in a very essential way on the specific PDE under consideration.

An initial condition is a requirement for which the dependent variable is specified at some initial state, and a boundary condition is a requirement that the dependent variable or its derivatives must satisfy on the boundary of the domain of the PDE.

Four types of boundary conditions are:

1. **The Dirichlet boundary condition.** If the dependent variable along the boundary is prescribed.
  - $f$  is specified on the boundary.
2. **The Neumann boundary condition.** If the value of the derivative normal (gradient) to the boundary is specified.
  - $\frac{\partial f}{\partial x}$  is specified on the boundary.
3. **The Robin boundary condition.** If the imposed boundary condition is a linear combination of the Dirichlet and Neumann types.
  - $af + b\frac{\partial f}{\partial x}$  is specified on the boundary.
4. **The Mixed boundary condition.** Frequently the boundary condition along a certain portion of the boundary is of Dirichlet type and, on another portion of the boundary is of a Neumann type.

One of the above types of boundary conditions must be specified at each point on the boundary of the closed solution domain. Different types of boundary conditions can be specified on different portions of the boundary. For unsteady or steady propagation problems, the auxiliary conditions consist of an initial condition (or conditions) along the time (or timelike) boundary and boundary conditions on the physical boundaries of the solution domain. No auxiliary conditions can be applied on the open boundary in the time (or timelike) direction. In conclusion, proper specifications of the type and number of auxiliary conditions is a necessary condition to obtain a well-posed problem.

Finally, the relationship between image processing and PDEs was proposed by Koenderink [142], who indicated that “any image can be embedded in a one-parameter family of derived images (with resolution as the parameter) in essentially only one unique way if the constraint that no spurious detail should be generated when the resolution is diminished, is applied”. Koenderink proposed that the structure of this family is governed by the well known diffusion equation, which is a parabolic, linear, partial differential equation of the second order. Since then, the PDE models have been increasingly used in many image processing tasks such as restoration, multiscale representation, inpainting, smoothing and edge detection [138, 172]. In particular, the diffusion equation has been successfully used for image smoothing, restoration and regularization [138, 146, 153, 163, 173, 174]. The diffusion equation is an important partial differential equation that describes the distribution of heat in a given region over time.

#### 4.1.2 Diffusion process

Most people have an intuitive impression of diffusion as a physical process that equilibrates concentration differences without creating or destroying mass. This physical observation can be cast in a mathematical formulation [175] using a diffusion equation. This equation is a partial differential equation based on repetition which describes density fluctuations in a material undergoing diffusion. The equation can be written as:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = \nabla \cdot (D(u(\mathbf{x}, t), \mathbf{x}) \nabla u(\mathbf{x}, t)) \quad (4.3)$$

where  $u(\mathbf{x}, t)$  is the density of the diffusing material at location  $\mathbf{x} = (x, y, z)$  and time  $t$ .  $D(u(\mathbf{x}, t), \mathbf{x})$  denotes the collective diffusion coefficient for density  $u$  at location  $\mathbf{x}$ . If the diffusion coefficient does not depend on the density, i.e.,  $D$  is constant, then Equation 4.3 is reduced to the following linear equation:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = D \nabla^2 u(\mathbf{x}, t) \quad (4.4)$$

Equation 4.4 is also called the heat equation and also describes the distribution of a heat in a given region over time. Equation 4.4 can be derived from the **continuity equation**, which states that a change in density in any part of the system is due to inflow and outflow of material into and out of that part of the system. Effectively, no material is created or destroyed:

$$\frac{\partial u}{\partial t} + \nabla \cdot j = 0 \quad (4.5)$$

where  $j$  is the flux of the diffusing material. Equation 4.4 can be obtained from the last

equation when combined with the Fick's first law, which assumes that the flux of the diffusing material in any part of the system is proportional to the local density gradient:

$$j = -D \cdot \nabla u(\mathbf{x}, t) \quad (4.6)$$

This equation states that a concentration gradient  $\nabla u$  causes a flux  $j$  which aims to compensate for this gradient. The relation between  $\nabla u$  and  $j$  is described by the **diffusion tensor**  $D$ . It is a positive definite symmetric matrix that provides the directionality of diffusion (smoothing). In other words, the interesting part is the tensor itself. An eigenvector decomposition of  $D$  will produce an orthogonal system of diffusion directions together with the variance of diffusion (the eigenvalues) in each of the directions.

When  $j$  and  $\nabla u$  are parallel the diffusion is called **isotropic**, otherwise, is defined **anisotropic** ( $j$  and  $\nabla u$  are not parallel).

Rewritten equation 4.5, the continuity equation can be expressed as:

$$\frac{\partial u}{\partial t} = -\nabla \cdot j \quad (4.7)$$

If this equation is plugged into Fick's law, the **diffusion equation** is obtained as:

$$\frac{\partial u}{\partial t} = \text{div}(D \cdot \nabla u(\mathbf{x}, t)) \quad (4.8)$$

Equation 4.8 represents the partial derivative equation (PDE) that governs the smoothing process, where  $u(\mathbf{x}, t)$  is the image at time  $t$ ,  $\text{div}$  represents the divergence operator,  $\nabla u(\mathbf{x}, t)$  is the gradient of the image, and  $D$  corresponds to the diffusion tensor. According to the literature, if the diffusion tensor is constant over the whole image in every direction, the diffusion process is called **isotropic**. In the other hand, the diffusion process is called **anisotropic** if the diffusion tensor is different for each location [152, 170, 176, 177].

The whole effect of the diffusion depends upon the choice of the diffusion tensor  $D$ :

1.  $D$  is scalar and  $D \neq D(u)$ . This is the simplest form. The diffusion equation is linear [ $D \neq D(u)$ ] and isotropic ( $D$  is scalar). In this case, it can be shown that the diffusion corresponds to low-pass nonadaptive filtering with a Gaussian kernel whose width increases with the square root of time ( $\sigma \propto \sqrt{2t}$ ).
2.  $D$  is scalar and  $D = D(u)$ . When the diffusion equation is nonlinear, that is, the diffusivity depends on the local image structure [ $D = D(u)$ ], more interesting results are achieved. Nonlinear isotropic diffusion was introduced to image processing by Perona and Malik (1990) [138]. Its most important application is edge-preserving smoothing.

Nonlinear diffusion can be compared to adaptive (low-pass) filtering in the sense that the smoothing kernel depends on the local image structure.

3.  $D$  is tensor and  $D = D(u)$ . The diffusion tensor introduces anisotropy to the diffusion by treating different orientations differently: diffusion is allowed in some directions and inhibited in others. It makes sense to align the principal axes of the diffusion with the orientations of the image. In other words,  $D$  is constructed such that its eigenvectors follow the local orientation of the image. This leaves many options for  $D$  because we can still adjust the eigenvalues of  $D$ . In images where orientation carries the bulk of the information (like fingerprints and seismic images), interesting results are obtained when diffusion (i.e., smoothing) is allowed only along the structure. This can be achieved by properly setting the eigenvalues of  $D$ .

Based on the concepts described above and according to Weickert and Brox (2002) [160], the diffusion process (filtering) of some image  $f = (f(x, y, z))^T$  may be based on one of the following evolutions:

1. Homogeneous diffusion ([160] in the scalar case):

$$\partial_t u = \Delta u \quad (4.9)$$

2. Linear isotropic diffusion ([178] in the scalar case):

$$\partial_t u = \operatorname{div} \left( g \left( |\nabla f|^2 \right) \cdot \nabla u \right) \quad (4.10)$$

3. Nonlinear isotropic diffusion [162]:

$$\partial_t u = \operatorname{div} \left( g \left( |\nabla u|^2 \right) \nabla u \right) \quad (4.11)$$

4. Linear anisotropic diffusion ([160] in the scalar case):

$$\partial_t u = \operatorname{div} \left( D \left( \nabla f \cdot \nabla f^T \right) \nabla u \right) \quad (4.12)$$

5. Nonlinear anisotropic diffusion [175]:

$$\partial_t u = \operatorname{div} \left( D \left( \nabla u \cdot \nabla u^T \right) \nabla u \right) \quad (4.13)$$

All diffusion methods use  $f$  as the initial condition:

$$u(\mathbf{x}, 0) = f(\mathbf{x}) \quad (4.14)$$

Some authors use  $u$  as the initial condition, but this is because they simply consider that the image  $u(\mathbf{x})$  is the noisy input image.

The function  $g$  denotes a scalar-valued diffusivity, and  $D$  is a tensor (positive definite diffusion matrix). The diffusivity  $g(s^2)$  is a decreasing function in its argument. Moreover, they assume that the flux function  $g(s^2)s$  is nondecreasing in  $s$ . One may use [179]

$$g(s^2) = \alpha + \frac{1}{\sqrt{\beta^2 + s^2}} \quad (4.15)$$

with some small positive numbers  $\alpha$  and  $\beta$ .

### 4.1.3 Smoothing using diffusion

The field of image restoration (sometimes referred to as image deblurring or image deconvolution) is concerned with the reconstruction or estimation of the uncorrupted image from a blurred and noisy one. Essentially, it tries to perform an operation on the image that is the inverse of the imperfections in the image formation system. In the use of image restoration methods, the characteristics of the degrading system and the noise are assumed to be known a priori. In practical situations, however, one may not be able to obtain this information directly from the image formation process. The goal of blur identification is to estimate the attributes of the imperfect imaging system from the observed degraded image itself prior to the restoration process. The combination of image restoration and blur identification is often referred to as blind image deconvolution [11]. Noise filtering techniques that maintain image contrast while decreasing image noise have the potential to optimize the quality of computed tomography (CT) images. To smooth CT images using diffusion techniques there are two alternatives, one using isotropic diffusion and the other using anisotropic diffusion.

Isotropic diffusion can be divided into linear and nonlinear. Linear, when  $D$  is a constant and independent of  $\mathbf{x}$ , it leads to a linear diffusion equation, with a homogeneous diffusivity. In this case, all locations in the image, including the edges are smoothed equally. Nonlinear, when  $D$  is a constant and dependent of image locations  $\mathbf{x}$ , it leads to a nonlinear diffusion equation, with a homogeneous diffusivity according to each location. In this case, image smoothing depends of image locations [153, 160].

In the same way, anisotropic diffusion can be divided into linear and nonlinear. Linear,

when the diffusion tensor  $D$  is a function of  $\mathbf{x}$ , i.e., depends on the space. In this case, all locations in the image, including the edges are smoothed equally but with different value in each direction. Nonlinear, when the diffusion tensor  $D$  is a function of  $\mathbf{x}$ , i.e., it depends on the space. But in this case, all locations in the image, including the edges are smoothed nonequally in each location.

#### 4.1.3.1 Linear Isotropic Diffusion Filters

The linear isotropic diffusion (heat) equation is the oldest and best investigated PDE method in image processing. To determine the linear isotropic diffusion equation first begins with the general diffusion equation, which is defined as [180]:

$$\partial_t u(\mathbf{x}, t) = \operatorname{div}(D(|\nabla u(\mathbf{x}, t)|) \cdot \nabla u(\mathbf{x}, t)) \quad \text{on } \Omega(0, \infty) \quad (4.16)$$

with  $u$  as initial condition,

$$u(\mathbf{x}, 0) = u(\mathbf{x}) \quad \text{on } \Omega \quad (4.17)$$

and using Neumann boundary condition. Where  $u(\mathbf{x})$  denote a grayscale (noisy) input image,  $D$  is the diffusion tensor (function of  $\mathbf{x}$ ), and  $u(\mathbf{x}, t)$  be initialized with  $u(\mathbf{x}, 0) = u(\mathbf{x})$ . In addition,  $t > 0$  and  $\mathbf{x} \in \Omega$ , where  $\Omega \subset \mathbb{R}^d$  is open. The unknown is  $u : \Omega \times [0, \infty) \rightarrow \mathbb{R}$ ,  $u = u(\mathbf{x}, t)$ , and the gradient  $\nabla$  is taken with respect to the spatial variables  $\mathbf{x} = (x_1, \dots, x_d)$ .

According to Weickert [152], the diffusion filter utilizes the unit matrix as diffusion tensor:

$$D(|\nabla u_\sigma|) = I \quad (4.18)$$

replacing the diffusion tensor  $D$  (Equation 4.18) in diffusion equation (Equation 4.16), the linear isotropic diffusion process can be defined by the equation

$$\partial_t u(\mathbf{x}, t) = \operatorname{div}(I \cdot \nabla u(\mathbf{x}, t)) = \nabla \cdot (\nabla u(\mathbf{x}, t)) = \nabla^2 u(\mathbf{x}, t) = \Delta u(\mathbf{x}, t) \quad (4.19)$$

This diffusion process has the properties of [151]:

- (i) linearity, the diffusion filter strength at all locations is the same
  - (ii) isotropy, the diffusion filter strength in different directions at any location is the same.
- Equation 4.19 is solved using a Gaussian function as:

$$u(\mathbf{x}, t) = \begin{cases} G_{\sqrt{2t}} * u(\mathbf{x}) & (\mathbf{x} \in \mathbb{R}^d, t > 0) \\ 0 & (\mathbf{x} \in \mathbb{R}^d, t < 0) \end{cases} \quad (4.20)$$

is called the fundamental solution of the heat equation. Where  $G_{\sqrt{2t}}$  is the Gaussian kernel. It is defined as [152, 180, 181]:

$$G_{\sqrt{2t}}(\mathbf{x}) = \frac{1}{(4\pi t)^{d/2}} e^{\left(-\frac{|\mathbf{x}|^2}{4t}\right)} \quad \mathbf{x} \in \mathbb{R}^d, t > 0 \quad (4.21)$$

The standard deviation  $\sigma = \sqrt{2t}$  of a Gaussian determines its shape and represents the strength of its related Gaussian filter (the larger the standard deviation, the stronger the Gaussian filter). This is the most well-known form of diffusion filtering called homogeneous isotropic linear diffusion and more commonly known as Gaussian smoothing (GS).

In addition, the linear diffusion process can be defined by the equation

$$\begin{aligned} \partial_t u(\mathbf{x}, t) &= \operatorname{div}(c \cdot \nabla u(\mathbf{x}, t)) \\ u(\mathbf{x}, 0) &= u(\mathbf{x}) \end{aligned} \quad (4.22)$$

where  $c$  is a diffusivity parameter that is constant across the image, making it a linear isotropic equation. This is solved in the same way using a Gaussian function:

$$\begin{aligned} u(\mathbf{x}, t) &= G_{\sqrt{2ct}} * u(\mathbf{x}) \\ u(\mathbf{x}, 0) &= u(\mathbf{x}) \end{aligned} \quad (4.23)$$

where  $G_{\sqrt{2ct}}$  is the Gaussian kernel. It is defined as [152, 180, 181]:

$$G_{\sqrt{2ct}}(\mathbf{x}) = \frac{1}{4\pi ct} e^{\left(-\frac{|\mathbf{x}|^2}{4ct}\right)} \quad \mathbf{x} \in \mathbb{R}^N, c > 0, t > 0 \quad (4.24)$$

In summary, the lineal isotropic diffusion filter can be seen as an evolution process with an artificial time variable  $t$  denoting the diffusion time where the input image is smoothed at a constant rate in all directions. In addition, the resulting image at diffusion time can be obtained by convolving the original image using a Gaussian kernel with standard deviation  $\sigma = \sqrt{2t}$  or  $\sigma = \sqrt{2ct}$  [142, 143, 152]. Since this process deals with digital images, solving the linear diffusion equation requires discretization in both spatial and time coordinates. Normally, central differences are the typical choices for the spatial derivatives.

Thus, linear diffusion can be regarded as a low-pass filter. The correspondence between the diffusion time variable  $t$  and the standard deviation  $\sigma$  clearly depicts the effect of  $t$  on the evolving images. The higher the value of  $t$ , the higher the value of  $\sigma$ , and the more smooth the image becomes. As the diffusion process evolve to coarser scales, the evolving images become more and more simplified since the diffusion process removes the image structures at finer scales.

Figure 4.1 uses the radii (of the circles) to represent the value of the standard deviation of the Gaussians. The circles themselves represent the “isotropic” property of the diffusion process, i.e., the Gaussian filter at each location has the same strength (circular shape) in all directions. The property of linearity is reflected by the uniform size of all circles [151].

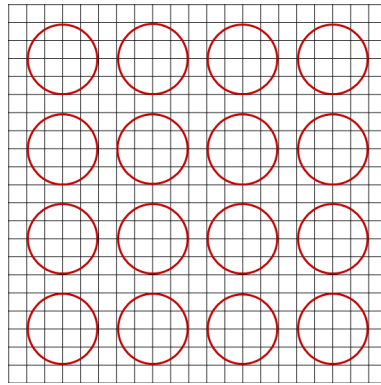


Figure 4.1: Linear isotropic diffusion

This technique goes back to Marr and Hildreth [182], Witkin [143], and Koenderink [142]. It is widely used in image processing community, since it is equivalent to convolving the original image with Gaussians of increasing size. Nevertheless, it smoothes noise within a region in the same way as it blurs semantically important structures like edges, lines, or other details [152]. Additionally, according to Barbu [180], the main disadvantage of this classical procedure is that the denoising operator has no localization property, that is, the solution to the heat equation (Equation 4.16) propagates with infinite speed. In other words, if  $u_0 = 0$  outside a certain domain  $\Omega_t$ , it does not follow that the same thing happens for  $u(t)$  outside  $\Omega_t \subset \mathbb{R}^N$ . This means that the flux instantly affects the image  $u_0$  and this can destroy the edges (see [183]).

#### 4.1.3.2 Nonlinear Isotropic Diffusion Filters

The main theory behind nonlinear diffusion models is to use nonlinear PDEs to create a scale space representation that consists of gradually simplified images where some image features such as edges are maintained or even enhanced. The earliest nonlinear diffusion model proposed in image processing is the so-called anisotropic diffusion by Perona and Malik [138, 184].

The goal of nonlinear diffusion filtering is to reduce smoothing in the presence of edges [138, 185]. In order to avoid the blurring of these edges, one should construct a diffusion filter which reduces the diffusivity at those locations which are good candidates for being an edge [152].



Nonlinear diffusion filtering creates a family of simplified images  $\{u(\mathbf{x}, t)\} | t \geq 0$  of some scalar initial image  $u(\mathbf{x})$  by solving the PDE [185]:

$$\frac{\partial u}{\partial t} = \text{div}(g(|\nabla u|) \nabla u) \quad \text{on } \Omega(0, \infty) \quad (4.25)$$

with  $u$  as initial condition,

$$u(\mathbf{x}, 0) = u(\mathbf{x}) \quad \text{on } \Omega \quad (4.26)$$

and reflecting (homogeneous Neumann) boundary conditions:

$$\partial_\nu u = 0 \quad \text{on } \partial\Omega(0, \infty) \quad (4.27)$$

where  $\nu$  denotes the outer normal on the image boundary.

Equation 4.25 denotes the evolution of image  $u(\mathbf{x})$  per unit time. Here,  $g(\cdot)$  is the conductance function that controls the diffusivity. This can be achieved by a monotonically decreasing diffusivity function  $g$  which correlates the amount of smoothing with the image gradient magnitude  $|\nabla u|$  (strong diffusivity in areas that have small gradients and weak in areas that have large gradients). The diffusivity  $g(|\nabla u|)$  is a nonnegative function that controls the amount of diffusion. Usually, it is decreasing in  $|\nabla u|$ .

This ensures that strong edges are less blurred by the diffusion filter than noise and low-contrast details. The diffusivity parameter should be made dependent on some characterization of image structure.

The diffusion time  $t$  determines the amount of simplification: For  $t = 0$  the original image  $u$  is recovered, and larger values for  $t$  result in more pronounced smoothing. As a consequence, the diffusivities become variable in both space and time. As the evolution proceeds, the areas that have large discontinuities are preserved, whereas those that have small discontinuities such as plain areas or noise are smoothed out. The Neumann boundary condition indicates that there is no orthogonal diffusion to the image borders  $\partial\Omega$ . In theory, a globally constant image is the solution of the equation [186].

The original Perona-Malik [138] equation proposed for nonlinear diffusion was:

$$\frac{\partial u}{\partial t} = \nabla \cdot (g(|\nabla u|) \nabla u) \quad (4.28)$$

with homogeneous Neumann boundary conditions and the initial condition  $u(\mathbf{x}, 0) = u(\mathbf{x})$ ,  $u$  denoting the input image.

The diffusivity now becomes a function of gradients, so at the edge point the diffusion is completely inhibited and in smooth regions diffusion is allowed. However, computing gra-

dients for a noisy image is an ill-posed problem [174, 186]. This problem occurs mainly because, in most cases, the variational function that is correlated with Equation 4.25 is not convex [175]. A solution was pointed out by Cottet and Germain [187], that suggest the use of Gaussian smoothing before computing gradients. This modification lays the foundation for a well-behaved nonlinear isotropic diffusion process.

In addition, forward and backward diffusions simultaneously exist, which indicates that there are potentially multiple solutions to the equation. Fortunately, various regularization strategies in the research are sufficiently good to turn it well-posed for practical use [175].

Figure 4.2 shows a nonlinear isotropic diffusion using the Gaussian filters whose strengths are represented by the size of the circles. The direction indifference of the circles corresponds to the isotropy property and the different sizes of circles represent the nonlinearity property [151].

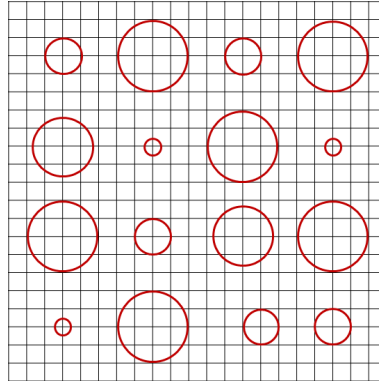


Figure 4.2: Nonlinear isotropic diffusion

Some examples of nonlinear isotropic diffusion are presented in Table 4.2.

Table 4.2: Examples of nonlinear isotropic diffusion

Technique	Equations
Perona-Malik [138]	$\partial_t u = \text{div} (g(\nabla u(\mathbf{x}, t)) \cdot \nabla u(\mathbf{x}, t))$ $g_1(\nabla u) = e^{-\left(\frac{ \nabla u }{K}\right)^2}$ $g_2(\nabla u) = \frac{1}{1 + \left(\frac{ \nabla u }{K}\right)^2}$
Catté et.al. [174]	$\partial_t u = \text{div} (g(\nabla u_\sigma(\mathbf{x}, t)) \cdot \nabla u(\mathbf{x}, t))$
Torkamani-Azar et.al. [188]	$\partial_t u = \text{div} (g\{\nabla[h(\mathbf{x})]\}(\nabla u_\sigma(\mathbf{x}, t)) \cdot \nabla u(\mathbf{x}, t))$ $c(\mathbf{x}, t) = g\{\nabla[h(\mathbf{x}) * u(\mathbf{x})]\} \cdot \nabla u(\mathbf{x}, t)$ $h(\mathbf{x}) = \left(\frac{\beta}{2}\right) \cdot \exp[-\beta \cdot ( x  +  y )]$

Table 4.2: Examples of nonlinear isotropic diffusion

Technique	Equations
Weickert [175]	$g(\mathbf{x}, t) = \begin{cases} 1, &  \nabla u_{\sigma}(\mathbf{x}, t)  = 0 \\ 1 - e^{-\frac{c_m}{(\ \nabla u(\mathbf{x}, t)\ /k)^{2m}}}, &  \nabla u_{\sigma}(\mathbf{x}, t)  > 0 \end{cases}$
Black et.al. [165, 189]	$g(\mathbf{x}, t) = \begin{cases} \frac{1}{2} \left[ 1 - ( \nabla u(\mathbf{x}, t)  / \sigma_e)^2 \right]^2, &  \nabla u(\mathbf{x}, t)  \leq \sigma_e \\ 0 & \textit{otherwise} \end{cases}$
Monteil and Beghdadi [190]	$g(\mathbf{x}, t) = \frac{1}{2} [1 - \tanh(\gamma \cdot (\ \nabla u(\mathbf{x}, t)\  - k))]$
Charbonnier et. al. [191]	$g(\ \nabla u\ ) = \frac{1}{\sqrt{1 + \frac{\ \nabla u\ ^2}{k^2}}}$

#### 4.1.3.3 Linear Anisotropic Diffusion Filters

Anisotropy in diffusion means that the smoothing induced by the PDE can be favored in some directions and prevented in others. The anisotropic diffusion approach is basically a modification of the linear diffusion (or heat equation) to eliminate specially his inability to preserve edges [192]. The term anisotropic is reserved for the case where the diffusivity is a tensor-valued function, varying with both the edge location and its orientation.

In addition, one way of introducing regularization to the Perona-Malik model is through anisotropic diffusion. Förstner and Gülch [193] and Bigün and Granlund [194] concurrently introduced the matrix field of the structure tensor for image processing, and it is the basis for today's anisotropic diffusion models.

Specifically, in linear anisotropic diffusion process the constant eigenvalues are responsible for the linear part of the name, while the steering of the eigenvectors is what provided the word anisotropic in the nomenclature. The diffusion matrix-based equation is defined as

$$\frac{\partial u}{\partial t} = \text{div} \left( D \left( |\nabla u|^2 \right) \cdot \nabla u \right) \quad (4.29)$$

where  $D$  is the diffusion matrix. The eigenvectors of the diffusion matrix provide the required steering while the eigenvalues as a function of gradients, add the non-linearity character. To keep a well-defined problem, in the solution is used a Gaussian function that provides a mathematical tractability to the whole process. To obtain a linear anisotropic diffusion the eigenvalues of the diffusion matrix are kept fixed.

The linear anisotropic diffusion equation has a convolution solution with a non-uniform

Gaussian of the form (2D-case):

$$G_{\lambda_u \lambda_v}(u, v) = \frac{1}{\sqrt{2\pi\lambda_u}} e^{-\frac{u^2}{2\lambda_u}} \frac{1}{\sqrt{2\pi\lambda_v}} e^{-\frac{v^2}{2\lambda_v}} \quad (4.30)$$

where  $(u, v)$  are the rotated coordinates obtained using eigenvectors of the diffusion matrix. The eigenvalues  $\lambda_u, \lambda_v$  represent the standard deviations of the Gaussian in  $u$  and  $v$  direction, respectively. Normally, for noisy images, one of the eigenvalues is set to be much smaller than the other one, resulting in a non-uniform Gaussian function with more generalized elliptical support [195].

Figure 4.3 shows a linear anisotropic diffusion process illustrated by the shapes and sizes of the Gaussian filters. The ‘‘anisotropy’’ property is reflected by the oval shape of the filter kernels that represents different filter strengths in different directions. The linearity is reflected by the uniform size and shape of the ovals.

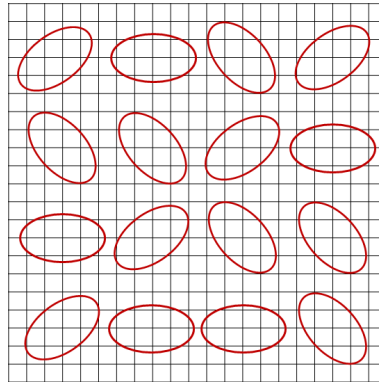


Figure 4.3: Linear anisotropic diffusion

#### 4.1.3.4 Nonlinear Anisotropic Diffusion Filters

The idea of nonlinear anisotropic diffusion was pioneered by Nitzbeg et. al. [196] and Cottet et al. [187]. Later on, Weickert [154] put forward a formal method for enhancing the elongated structure, referred to as coherence-enhanced diffusion (CED).

Nonlinear anisotropic diffusion filtering is a procedure based on nonlinear evolution PDEs which seeks to improve images qualitatively by removing noise while preserving details and even enhancing edges. In the anisotropic case not only the amount of diffusion is adapted locally to the data but also the direction of smoothing. It allows for example to smooth along image edges while inhibiting smoothing across edges. This can be achieved by replacing the scalar-valued diffusivity function by a matrix-valued diffusion tensor [185].

The eigenvectors of the diffusion tensor define the principal directions of smoothing and

the corresponding eigenvalues define the amount of smoothing. Weickert based the diffusion tensor on the structure tensor [175], which describes structures in the image using first order derivative information.

In general, any nonlinear anisotropic diffusion can be described by the equation

$$\frac{\partial u}{\partial t} = \text{div}(D(\nabla u) \nabla u) \quad (4.31)$$

where  $u$  is the smoothed image that is initialized with the input image  $f$  (that is  $u(\mathbf{x}, 0) = f(\mathbf{x})$ ), and  $D$  represents a matrix-valued diffusion tensor that describes the smoothing directions and the corresponding diffusivities [184]. In this case, the diffusion tensor  $D$  is a function of  $\mathbf{x}$ , i.e., depends on the space. Additionally,  $D$  is a positive definite symmetric matrix [153, 160]. The idea is to adaptively choose the diffusion coefficient  $D$  such that intra-regions become smooth while edges of inter-regions are preserved [177]. As  $D$  must be a nonnegative function of gradient magnitude so that small variations in intensity such as noise or shading can be well smoothed, and edges with large intensity transition are retained. It is generally given by an exponential function or an inverse quadratic function, and determined by the gradient magnitude with respect to a predetermined edge strength threshold [177].

Thus the given image  $u$  is usually convolved with a Gaussian kernel  $G_\sigma$  with a relatively small standard deviation  $\sigma$  as a presmoothing step. Cottet and Germain [187] and Weickert [152] devise a diffusivity matrix of the form:

$$D_\sigma = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} v_1^T \\ v_2^T \\ v_3^T \end{bmatrix} \quad (4.32)$$

where the vectors  $v_i$  are the eigenvectors of the structure tensor and the parameters  $\lambda_i$  are functions of the eigenvalues of the structure tensor. The images's structure tensor is defined as [175]:

$$J_\rho(\nabla u_\sigma) = G_\rho * (\nabla u_\sigma \cdot \nabla u_\sigma^T) \quad (4.33)$$

where  $G_\rho$  is the Gaussian kernel with standard deviation  $\rho$  (integration scale), over which the orientation information is averaged, and  $\nabla u_\sigma$  is the gradient of the image  $u$  at scale  $\sigma$ . Principle axis transformation gives the eigenvectors and eigenvalues of  $J_\rho(\nabla u_\sigma)$  [197].

Figure 4.4 shows a nonlinear anisotropic diffusion process illustrated by ovals of different sizes and different orientations. The ratio of two dimensions of the ovals can be arbitrarily different. The orientations of the ovals can also be random. This is the diffusion filter with

the ultimate freedom in terms of the changes of filter strength location-wise or direction-wise [151].

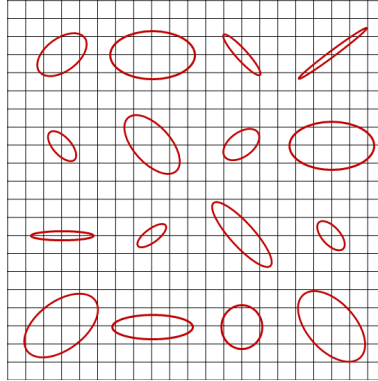


Figure 4.4: Nonlinear anisotropic diffusion

Two specializations of nonlinear anisotropic diffusion were introduced by Weickert, edge-enhancing diffusion (EED) and coherence-enhancing diffusion (CED) [153]. Both were initially defined in two dimensions. EED was designed to smooth noise while enhancing edges and CED was designed to enhance line-like textures. CED is essentially one dimensional diffusion [154], since there is either diffusion in one direction or almost no diffusion at all. In addition, Mirebeau et. al. [157] proposed a conservative variant of both the EED and CED method. These variants are called cEED and cCED respectively.

## 4.2 Nonlinear algorithms selected

In the same way that several researchers, three-dimensional properties of EED and CED are reviewed [155, 195, 197]. In addition, determine how EED and CED can be used in restoration of images which contain upper airways information is crucial. The main reason is that in this kind of images appear plate-like (line-like), vessel-like (tubular-like), and flow-like structures. Additionally, how to combine the advantages of EED and CED. First, EED in three dimensions becomes plate enhancing diffusion, it filters noise from homogeneous areas and enhances plate-like structures. Second, CED preserves small structures and enhances tubular structures.

Medical images consist of many structures of different shape, size and contrast. If plain EED would be applied to medical images, it would filter noise and preserve plate-like structures, such as the boundaries of larger organs, but it would also blur vessels and smaller structures. On the other hand, applying plain CED would preserve smaller structures and filter vessels, but would not filter the noise and plate-like structures properly [197].

### 4.2.1 Perona-Malik filtering (PM-isotropic)

The most popular nonlinear anisotropic diffusion technique is the influential denoising scheme developed by Perona and Malik in 1987[138]. Their approach reduces diffusivity at those locations having a larger likelihood to represent image edges. The Perona-Malik filter is characterized by the following nonlinear diffusion equation:

$$\begin{cases} \frac{\partial u}{\partial t} = \operatorname{div} (g(|\nabla u(\mathbf{x}, t)|) \cdot \nabla u(\mathbf{x}, t)), & \mathbf{x} \in \Omega \\ u(\mathbf{x}, 0) = u(\mathbf{x}), & \mathbf{x} \in \Omega \\ u(\mathbf{x}, t) = 0, & \mathbf{x} \in \partial\Omega \end{cases} \quad (4.34)$$

with the noisy image  $u(\mathbf{x})$  as the initial condition. Obviously, the diffusivity coefficient that controls how much image smoothing is performed in  $\mathbf{x} = (x_1, \dots, x_d)$  is  $C(x, t) = g(|\nabla u(\mathbf{x}, t)|)$ . The value of  $C(x, t)$  should be lower when  $\mathbf{x}$  is part of an edge, and higher when it is not. The function  $g$  that controls the blurring intensity should be monotonous decreasing for this reason. Perona and Malik considered two diffusivity function variants, which are:

$$g_1(\nabla u) = e^{-\left(\frac{|\nabla u|}{K}\right)^2} \quad (4.35)$$

$$g_2(\nabla u) = \frac{1}{1 + \left(\frac{|\nabla u|}{K}\right)^2} \quad (4.36)$$

where  $K > 0$  represents the diffusivity conductance, being the parameter that controls the diffusion process.

The Perona-Malik diffusion algorithm produces good smoothing results while preserving image edges for a long time. Thus, the boundaries remain stable for high values of time variable  $t$ .

### 4.2.2 Edge enhancing diffusion filtering (EED)

The nonlinear diffusion model proposed by Perona and Malik [138] employs a scalar-valued diffusivity function to guide the smoothing process. The diffusivities are reduced at the image locations where the magnitude of image gradient  $|\nabla u|$  is large, and as a result, the edges are preserved or even enhanced. In [153], Weickert suggested an alternative approach that additionally takes direction of the image gradients into account. The suggested model is an anisotropic nonlinear diffusion model with better edge enhancing capabilities [184].

In general, any anisotropic nonlinear diffusion can be described by the equation

$$\begin{cases} \frac{\partial u}{\partial t} = \operatorname{div}(D(\nabla u(\mathbf{x}, t)) \cdot \nabla u(\mathbf{x}, t)), & \mathbf{x} \in \Omega \\ u(\mathbf{x}, 0) = u(\mathbf{x}), & \mathbf{x} \in \Omega \\ u(\mathbf{x}, t) = 0, & \mathbf{x} \in \partial\Omega \end{cases} \quad (4.37)$$

where  $u$  is the input image and  $D$  represents a matrix-valued diffusion tensor that describes the smoothing directions and the corresponding diffusivities [? ] .

Similarly, for Perona-Malik type nonlinear diffusion,  $D(\nabla u) = g(|\nabla u|)I$ . Such a choice reduces the amount of smoothing at image edges, but in an equal amount in all directions. In actual anisotropic setting, the diffusion tensor  $D$  is defined as a function of the structure tensor, to 2D case, is given by

$$J(\nabla u) = \nabla u \cdot \nabla u^T = \begin{bmatrix} u_x^2 & u_x u_y \\ u_x u_y & u_y^2 \end{bmatrix} \quad (4.38)$$

The structure tensor  $J(\nabla u)$  can be interpreted as an image feature describing the local orientation information. It has an orthonormal basis of eigenvectors  $v_1$  and  $v_2$  with  $v_1 \parallel \nabla u$  and  $v_2 \perp \nabla u$ , and the corresponding eigenvalues  $\lambda_1 = |\nabla u|^2$  and  $\lambda_2 = 0$ . It is important to note that noise significantly affects the tensor estimation [184]. Thus the given image  $u$  is usually convolved with a Gaussian kernel  $G_\sigma$  with a relatively small standard deviation  $\sigma$  as a presmoothing step and the structure tensor is computed accordingly by using

$$\nabla u_\sigma = \nabla(G_\sigma * u) \quad (4.39)$$

The main idea behind edge enhancing diffusion is to use the structure tensor as an image/edge descriptor to construct a diffusion tensor that reduces the amount of smoothing across the edges while smoothing is still carried out along the edges. In order to perform this, Weickert [153] proposed to utilize same orthonormal basis of eigenvectors  $v_1 \parallel \nabla u_\sigma$  and  $v_2 \perp \nabla u_\sigma$  estimated from the structure tensor  $J(\nabla u_\sigma)$  satisfying

$$\frac{\lambda_1(|\nabla u_\sigma|)}{\lambda_2(|\nabla u_\sigma|)} \rightarrow 0, \text{ for } |\nabla u_\sigma| \rightarrow \infty \quad (4.40)$$

eigenvalues are chosen using

$$\lambda_1(|\nabla u_\sigma|) = \begin{cases} 1 & \text{if } |\nabla u_\sigma| = 0 \\ 1 - e^{-\frac{3.31488}{(|\nabla u_\sigma|/\lambda)^8}} & \text{otherwise} \end{cases} \quad (4.41)$$



$$\lambda_2(|\nabla u_\sigma|) = 1 \quad (4.42)$$

where  $\lambda$  denotes the contrast parameter.

Such a choice preserves and enhances image edges by reducing the diffusivity  $\lambda_1$  perpendicular to edges for sufficiently large values of  $|\nabla u_\sigma|$ . Specifically, the diffusion tensor is given by

$$D = \begin{bmatrix} (u_\sigma)_x & -(u_\sigma)_y \\ (u_\sigma)_y & (u_\sigma)_x \end{bmatrix} \cdot \begin{bmatrix} \lambda_1(|\nabla u_\sigma|) & 0 \\ 0 & \lambda_2(|\nabla u_\sigma|) \end{bmatrix} \cdot \begin{bmatrix} (u_\sigma)_x & -(u_\sigma)_y \\ (u_\sigma)_y & (u_\sigma)_x \end{bmatrix}^{-1} \quad (4.43)$$

EED in three dimensions becomes plate enhancing diffusion, it filters noise from homogeneous areas and enhances plate-like structures[197]. Diffusion decreases if the gradient magnitude increases compared to the contrast parameter  $\lambda$ , indicating a plate-like structure. If the gradient magnitude is much smaller than  $\lambda$ , isotropic diffusion is performed. EED filter enhances the plane-like structures while reducing the noise. The eigenvalues of the 3-D EED diffusion tensor are defined as [198]

$$\lambda_1 = \begin{cases} 1 & \text{if } |\nabla u_\sigma| = 0 \\ 1 - e^{-\frac{3.31488}{(|\nabla u_\sigma|/\lambda)^8}} & \text{if } |\nabla u_\sigma| > 0 \end{cases} \quad (4.44)$$

$$\lambda_2 = 1 \quad (4.45)$$

$$\lambda_3 = 1 \quad (4.46)$$

with a threshold parameter  $C = 3.31488$  as defined in [153, 175].

### 4.2.3 Conservative edge enhancing diffusion filtering (cEED)

An undesirable side effect using EED filter is that the image is blurred close to the angles of its contour set. Mirebeau et. al. [157] considered that such salient features should be preserved, hence they introduced a Conservative variant of EED (cEED) for which  $\mu_1$  can be small, when appropriate, so as to prevent diffusion around the angles of its countour set.

$$\mu_i = 1 - (1 - \alpha) e^{-\left(\frac{\lambda}{\lambda_i}\right)^m} \quad (4.47)$$

If all eigenvalues are set equal  $\mu_1 = \dots = \mu_d$ , then the diffusion tensors are isotropic, in other words scalar multiples of the identity. The following isotropic variant of EED is close

in spirit to the Perona-Malik model: diffusion is prevented in the neighborhood of the image contours, regardless of direction. This construction is implemented purely for comparison with the anisotropic ones, and does not take advantage of the innovative numerical scheme developed by Mirebeau et. al. [157]

$$\mu_i = 1 - (1 - \alpha) e^{-\left(\frac{\lambda}{\lambda_d}\right)^m} \quad (4.48)$$

cEED filter enhances the plane-like structures while reducing the noise.

#### 4.2.4 Coherence enhancing diffusion filtering (CED)

It is essentially a one-dimensional smoothing strategy in a multidimensional image, it is of outmost importance to have a precise realization of the desired smoothing direction. When the goal consists, e.g., of closing gaps in an interrupted line-like structure, it is clear that slight deviations from the correct smoothing direction will destroy any desired filter effect and result in a deterioration of the line by introducing blurring artifacts [199].

This direction sensitivity constitutes an additional problem for the design of appropriate algorithms for diffusion filtering that has not been addressed in the computer vision literature so far [199].

Three-dimensional nonlinear diffusion filters have been investigated first by Gerig et al. [162] in the isotropic case, and by Rambaux and Garçon (referenced by Weickert [175]) in the edge-enhancing anisotropic case. A generalization of coherence-enhancing anisotropic diffusion to higher dimensions was first proposed by Weickert [154]. A recent three-dimensional PDE-based filter by Krissian et al. [200] and a 3-D reaction-diffusion process by Payot et. al. [201] may be related to these anisotropic diffusion techniques [154].

Weickert [153, 199] provided a unified tensor diffusion formulation which evolves the initial image under a parabolic nonlinear PDE given by,

$$\begin{cases} \frac{\partial u}{\partial t} = \operatorname{div} (D (J_\rho (\nabla u \sigma)) \cdot \nabla u), & \mathbf{x} \in \Omega \\ u(\mathbf{x}, 0) = u(\mathbf{x}), & \mathbf{x} \in \Omega \\ u(\mathbf{x}, t) = 0, & \mathbf{x} \in \partial\Omega \end{cases} \quad (4.49)$$

where  $u(\mathbf{x}, t)$  is the evolving image,  $t$  denotes the diffusion time, and

$$D = \begin{pmatrix} a & b \\ b & c \end{pmatrix} \quad (4.50)$$

is the diffusion tensor, a positive definite symmetric matrix that may be adapted to the local image structure [199].

This local image structure is measured by the so-called structure tensor (scatter matrix, second-moment matrix, Förstner interest operator) [9, 16, 23, 26, 34] which is given by

$$J_\rho(\nabla u_\sigma) = G_\rho * (\nabla u_\sigma \cdot \nabla u_\sigma^T) \quad (4.51)$$

The function  $G_\rho$  denotes a Gaussian with standard deviation  $\rho$ , and  $u_\sigma = G_\rho * u$  is a regularized version of  $u$  that is obtained by convolution with a Gaussian  $G_\sigma$ . The eigenvectors of  $J_\rho$  give the preferred local orientations, and the corresponding eigenvalues denote the local contrast along these directions. The resultant sequence of images  $\{u(\cdot, t)\}_{t=0}^T$ , for a finite time  $T$  represents a nonlinear scale space. Here the diffusion tensor  $D$  is dependent on the image information via the structure tensor  $J_\rho(\nabla u_\sigma)$ . The structure tensor is highly robust under isotropic additive Gaussian noise [24], and it can be implemented efficiently [21] [199].

The matrix that represents of structure tensor  $J_\rho$  is defined as

$$J_\rho(\nabla u_\sigma) = \begin{bmatrix} J_{11} & J_{12} \\ J_{12} & J_{22} \end{bmatrix} \quad (4.52)$$

In a two-dimensional image the structure tensor encode local information with first order directional derivatives and using equations 4.51 and 4.52, a new representation is obtained which is given by

$$J_\rho(\nabla u_\sigma) = \begin{bmatrix} G_\rho * (u_\sigma)_x^2 & G_\rho * (u_\sigma)_x (u_\sigma)_y \\ G_\rho * (u_\sigma)_x (u_\sigma)_y & G_\rho * (u_\sigma)_y^2 \end{bmatrix} \quad (4.53)$$

The eigenvalues of structure tensor  $J_\rho$  matrix can be calculated as,

$$\mu_{1,2} = \frac{1}{2} \left( J_{11} + J_{22} \pm \sqrt{(J_{11} - J_{22})^2 + 4J_{12}^2} \right) \quad (4.54)$$

or, in an equivalent way,

$$\mu_{1,2} = \frac{1}{2} \left( \text{trace}(J_\rho) \pm \sqrt{\text{trace}^2(J_\rho) + 4 \det(J_\rho)} \right) \quad (4.55)$$

and the first eigenvector  $(\cos \alpha, \sin \alpha)^T$  satisfies

$$\begin{pmatrix} \cos \alpha \\ \sin \alpha \end{pmatrix} \parallel \begin{pmatrix} 2J_{12} \\ J_{22} - J_{11} + \sqrt{(J_{11} - J_{22})^2 + 4J_{12}^2} \end{pmatrix} \quad (4.56)$$

Weickert [154, 199] proposed the following particular choices for steering smoothing for coherence enhancement diffusion (CED), where diffusion tensor  $D$  of CED uses the same eigenvectors as the structure tensor, and its eigenvalues are assembled via

$$\lambda_1 := \gamma \quad (4.57)$$

$$\lambda_2 := \begin{cases} \gamma & \text{if } \mu_1 = \mu_2 \\ \gamma + (1 - \gamma) e^{\frac{-\beta}{(\mu_1 - \mu_2)^2}} & \text{else,} \end{cases} \quad (4.58)$$

with  $\beta > 0$  is known as the coherence factor (if the coherence is inferior to  $\beta$  the flux is increasing with the coherence while if the coherence is larger than  $\beta$  the flux decreases as the coherence grows),  $\gamma \in (0, 1)$  is a small parameter added to keep the tensor diffusion matrix  $D$  positive definite.

Note that  $(\mu_1 - \mu_2)^2$  measures the coherence within a window of scale  $\rho$ . This particular choice obtained good diffusion results when the structures are oriented in one particular direction, however can smooth out corners and other singularities as multiple directional information is lost.

Finally, the condition number of  $D$  is thus bounded by  $1/\gamma$ , and the entries of  $D$  are

$$\begin{aligned} a &= \lambda_1 \cos^2 \alpha + \lambda_2 \sin^2 \alpha \\ b &= (\lambda_1 - \lambda_2) \sin \alpha \cos \alpha \\ c &= \lambda_1 \sin^2 \alpha + \lambda_2 \cos^2 \alpha \end{aligned} \quad (4.59)$$

CED filter enhances curvilinear structures and small blob-like structures. In addition, this approach can complete interrupted lines and gaps as well as preserving edges. However, a problem of CED filtering is that it is susceptible to noise. For more details on coherence-enhancing anisotropic diffusion in  $m$ -dimensions see [154].

#### 4.2.5 Conservative coherence enhancing diffusion filtering (cCED)

A more reliable coherence detector is  $\lambda_m - \lambda_i \gg \lambda + \lambda_i$ , which leads to a Conservative variant of CED (cCED). Precisely [157]: (note that  $\mu_d = \alpha$ )

$$\mu_i = \alpha + (1 - \alpha) e^{-\left(\frac{\lambda + \lambda_i}{\lambda_d - \lambda_i}\right)^m} \quad (4.60)$$

### 4.3 Diffusion stopping criteria

Filtering process involves the solution of the anisotropic diffusion equations as a time-marching problem, a possible approach is to halt the filtering when a certain set of metrics falls below a predefined threshold [135]. In addition, the definition of the number of iterations (diffusion time  $t$ ) based on the metrics selected to stop the diffusion process is crucial to obtain a good image reconstruction[202]. For example, if  $t$  is too small, the reconstructed signal is very noisy; if  $t$  is too large it is smooth and discontinuities are lost. In conclusion, automatically stopping the diffusion process is a challenging task. Normally, the stopping criterion depends on the image characteristics and on the parameters of the diffusion equation.

Several authors have addressed this issue in the past in an attempt to devise an optimal stopping criterion [202, 203]. A brief review of previous works on the stopping criteria is presented below.

- Sporring and Weickert (1999) [204]

This is focused on the maximal entropy change by scale to estimate the size of image structures. They argued that the minimal change by scale indicates especially stable scales with respect to evolution time, and conjectured that these scales could be good candidates for stopping times in nonlinear diffusion processes. In addition, this is based on the signal to noise ratio (SNR) and the relative variance at time  $t$  and the initial image [203]. The authors pointed out that the monotonically decreasing 'relative variance',  $0 \leq \text{var}(u)/\text{var}(u_0) \leq 1$ , could be used to measure the distance of  $u$  from the initial state  $u_0$  and, by prescribing an appropriate value for the relative variance, it can constitute a good criterion for stopping the nonlinear diffusion [202].

- Capuzzo and Ferretti (2001) [205]

They determine the optimal time by finding the minimum of a performance index which balances the computing and stopping costs. This is then applied to the regularized Perona-Malik equation. Their method requires a constant that is found by experimentation using a typical image with similar details and discontinuities as the image to be processed. This is a rather vague requirement and they demonstrate that one only needs some approximation to the constant [203].

- Mrazek and Navara (2003) [206]

They choose the stopping criteria so that the correlation of the signal  $u(T)$  and noise  $u(0) - u(T)$  in the filtered image is minimized. This method is applicable to any images where the noise to be removed is uncorrelated with the signal, under the assumptions

that the filter used is suitable for the given type of data, and that neither the additive noise nor the filtering procedure alter the average gray value; no other knowledge (e.g. the noise variance, training data etc.) is needed [206]. This method is applied to several nonlinear filters both isotropic and anisotropic [203]. In addition, this requires no prior estimation of the noise statistics [207].

Proposed method is called decorrelation criterion. This selects the time  $T$  as the time that minimizes the correlation

$$T \equiv \arg \min_t \frac{\text{cov}(u(0) - u(t), u(t))}{\sqrt{\text{var}(u(0) - u(t)) \cdot \text{var}(u(t))}} \quad (4.61)$$

- Gilboa, Sochen and Zeevi (2004)[116]

Stopping criterion is based on obtaining of minimal SNR, i.e. one stops the process when filtering more signal than noise. This is done by estimating the covariance of the image and the noise. This method require an estimate of the standard deviation of the noise  $\sigma_{n0}$  of the input noisy image  $u(0)$ , which is considered to be a priori known [207]. They also compare the advantages and disadvantages of the approaches that use the covariance [203]. The condition for selecting the value of parameter  $T$  is

$$T = \arg \min_t \frac{\partial_t \text{cov}(\bar{N}, u(0) - u(t))}{\partial_t \text{var}(u(0) - u(t))} \quad (4.62)$$

The variance of noise  $\bar{N}$  of the original image is considered a priori known.

- Bazán and Blomgren (2007) [202]

This stopping criterion is inspired by observation of the behavior of the correlation between the noise-free image and the filtered image,  $\text{corr}(f, u)$ , and the correlation between the noisy image and the filtered image,  $\text{corr}(u_0, u)$ . Although the former measure is only available in experimental settings it helps validate the usefulness of the latter.

The nonlinear diffusion process starts from the observed (noisy) image,  $u_0(x)$ , and creates a set of filtered images,  $u(x, t)$ , by gradually removing noise and details from scale to scale until, as  $t \rightarrow \infty$ , the image converges to a constant value. During this process the correlation between the noise-free image and the filtered image increases as the filtered image moves closer to the noise-free image. This behavior continues until it reaches a peak from where the measure decreases as the filtered image moves slowly towards a constant value. During the same process the correlation between the noisy image and the filtered image decreases gradually from a value of 1.0 (perfect correlation), to a constant value, as the filtered image becomes smoother [155].

By comparing both measures, they observed that as  $\text{corr}(f, u)$  reaches its maximum (the best possible reconstructed image), the curvature of  $\text{corr}(u_0, u)$  changes sign. They suggested that a good stopping point of the diffusion process is where the second derivative of  $\text{corr}(u_0, u)$  reaches a maximum [155].

- Tsotsios and Petrou (2013) [207]

The method examines directly the quality of the edges in every iteration. It evaluates, in every iteration, the quality of a percentage of the true edges of the image, taking into consideration the contrast and the noise brightness fluctuations around them, and leads to a judicious choice of the stopping time  $T$  that corresponds to the maximum overall quality of the edges [207]. This method requires an estimate of the standard deviation of the noise  $\sigma_{n0}$  of the input noisy image  $u(0)$ , which is considered to be a priori known [207]. The proposed method has five steps that finally compute the stop time  $T$  as

$$T = \arg \max_t \frac{1}{N} \sum_{i=1}^N Q_i(t) \quad (4.63)$$

where  $N$  is the number of edgels and  $Q(t)$  reflects the quality of the edges within the image, in every iteration.

- Joao, Gambaruto, Tiago and Sequeira (2016) [135]

The relative residual error of Mean Square Error (MSE) measure is the metric chosen for this purpose, specifically

$$\frac{|MSE_{t+1} - MSE_t|}{|MSE_{t+1}|} < \varepsilon_1 \quad (4.64)$$

where  $\varepsilon_1 = 10^{-2}$ . The choice of  $\varepsilon_1$  is influenced by the need for a small value to identify a convergence of solution, and large enough to make the iterative procedure less computationally demanding.

In addition, They propose to use Structural Similarity Index Metric (SSIM) in combination with above criterion, using a threshold value of  $SSIM < \varepsilon_2$  and  $\varepsilon_2 = 0.7$ . The choice of  $\varepsilon_2$  is influenced by the importance of allowing the image to evolve and deviate from the original, and yet not to allow too large a distortion that will make the image unrecognizable compared to the original.

In conclusion, the optimal number of iterations is obtained when  $|MSE_{t+1} - MSE_t| / |MSE_{t+1}| < 10^{-2}$  and  $SSIM_\beta(t+1) < 0.7$ .

This, depending on the size of each image and respective data set, can be rather computationally expensive; therefore, a parallel implementation was used, which proved to be effective.

## 4.4 Stopping criterion proposed

Sporring and Weickert pointed out that the monotonically decreasing 'relative variance',  $0 \leq \text{var}(u)/\text{var}(u_0) \leq 1$ , could be used to measure the distance of  $u$  from the initial state  $u_0$  and, by prescribing an appropriate value for the relative variance, it can constitute a good criterion for stopping the nonlinear diffusion.

The CIRR measure is an increasing function that reaches its steady state when  $t \rightarrow \infty$ . The residual error of the CIRR measure is a decreasing function of values between one and zero. Then, applying the same idea presented by Sporrying and Weickert, it can be indicated that the residual error of the CIRR measure can be used as a stopping criterion to halt diffusion processes.

The residual error of CIRR measure is computed as:

$$CIRR\ index_t = \frac{|CIRR_t - CIRR_{t-1}|}{|CIRR_t|} \quad (4.65)$$

where  $t$  is the diffusion time. Diffusion process is iterated while  $CIRR\ index_t$  is greater than a specific constant  $\varepsilon$  defined by the user. In general, the stopping criteria is defined as:

$$\frac{|CIRR_t - CIRR_{t-1}|}{|CIRR_t|} > \varepsilon \quad (4.66)$$

The choice of  $\varepsilon$  is influenced by the need for a small value to identify a convergence of solution, and large enough to make the iterative procedure less computationally demanding. According to João et. al. [135], who defined a stopping criterion for anisotropic diffusion,  $\varepsilon$  value can be  $10^{-2}$ .

## 4.5 Experimental results

Nonlinear anisotropic diffusion algorithms require several parameters such as diffusion time, lambda, noise scale and feature scale. Therefore, to select the most appropriate diffusion algorithm to preserve the edge information is a complex task. For this reason, the experiments were divided into two parts. The first part is related to the automatic definition of the diffusion time for both two-dimensional and three-dimensional images using the stopping criterion



presented by Joao et. al. [135] and the proposed criterion. The second part is associated with the selection of the algorithm that generates better results with respect to the image quality measures as Mean Square Error (MSE), Peak Signal-Noise Ratio (PSNR), and Contrast Improvement Ratio Revisited (CIRR). As a qualitative measure of smoothing, the Canny edge detection filter is used [136]. The filter is applied to the resulting images by using each of the stopping criteria.

### 4.5.1 2D case

In the first part, the stopping criterion proposed by Joao et. al. [135] is based on the MSE quality measure, this criterion is called JGTS. The proposed criterion is based on the CIRR measure and it is called BF. These two stopping criteria are evaluated.

#### 4.5.1.1 Original images

Five images were selected that are used traditionally in image processing. The images are the baboon, barbara, boat, cameraman and lena. Each of them has different characteristics that allow evaluating the quality of the smoothing obtained for each of them according to each stopping criterion.

Initially smoothing is calculated using each of the selected diffusion algorithms (Isotropic, CED, cCED, EED, cEED). The number of iterations (diffusion time) applied is initially set to 10. In each iteration the MSE, PSNR and CIRR quality measures are calculated. The results for the baboon image are presented in Tables 4.3, 4.4 and 4.5 respectively.

Table 4.3: MSE measure - baboon.

Iteration	Isotropic	CED	cCED	EED	cEED
1	5.3973	65.5733	51.4294	89.5345	19.3840
2	11.6884	138.7730	114.5920	175.5490	43.7887
3	18.4879	199.1690	168.3340	240.0070	68.1375
4	25.8533	248.2130	215.3060	289.1190	91.5854
5	33.6677	289.3180	254.8620	327.9470	113.9010
6	41.8621	322.1530	287.1910	359.6730	135.1300
7	50.3696	349.7260	316.1980	385.8300	155.3750
8	59.1242	373.7520	341.0820	407.9260	174.5190
9	68.0437	394.3030	362.3720	426.9830	192.5880
10	77.0767	412.0270	381.8430	443.7070	209.6550

Table 4.4: PSNR measure - baboon.

Iteration	Isotropic	CED	cCED	EED	cEED
1	40.8090	29.9635	31.0187	28.6109	35.2564
2	37.4532	26.7078	27.5393	25.6868	31.7172
3	35.4619	25.1386	25.8691	24.3286	29.7969
4	34.0056	24.1826	24.8002	23.5200	28.5125
5	32.8587	23.5171	24.0678	22.9728	27.5655
6	31.9126	23.0502	23.5491	22.5717	26.8233
7	31.1091	22.6935	23.1312	22.2668	26.2170
8	30.4132	22.4050	22.8022	22.0250	25.7124
9	29.8029	22.1725	22.5393	21.8267	25.2845
10	29.2616	21.9815	22.3120	21.6598	24.9157

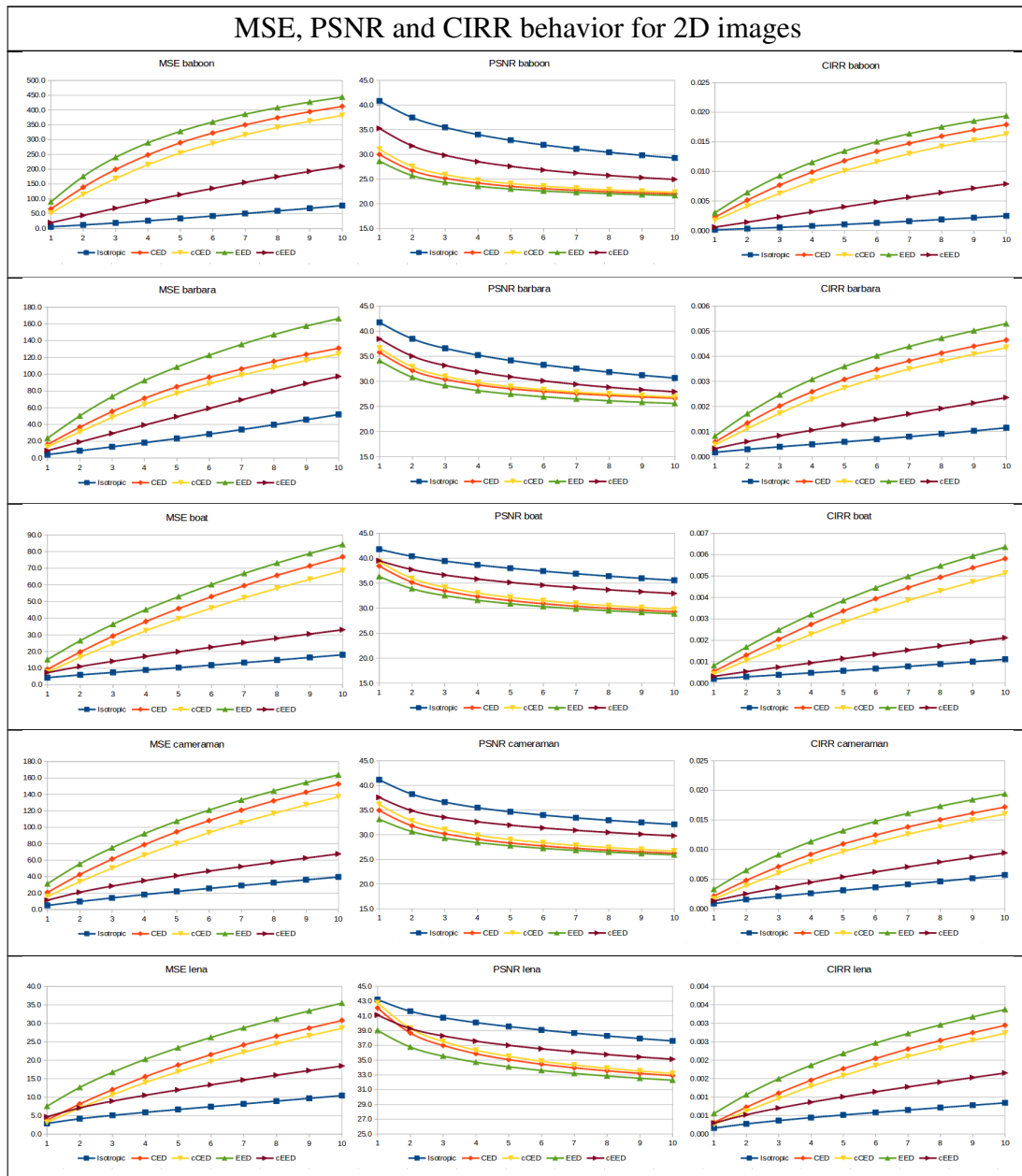
Table 4.5: CIRR measure - baboon.

Iteration	Isotropic	CED	cCED	EED	cEED
1	0.0001	0.0023	0.0017	0.0030	0.0006
2	0.0003	0.0051	0.0041	0.0064	0.0014
3	0.0005	0.0077	0.0063	0.0092	0.0023
4	0.0008	0.0099	0.0083	0.0115	0.0032
5	0.0010	0.0118	0.0101	0.0134	0.0040
6	0.0013	0.0134	0.0116	0.0150	0.0048
7	0.0016	0.0147	0.0130	0.0164	0.0056
8	0.0019	0.0159	0.0142	0.0175	0.0064
9	0.0022	0.0170	0.0153	0.0185	0.0072
10	0.0025	0.0179	0.0163	0.0194	0.0079

Based on the data obtained for each image, the behavior of each quality measure is plotted as shown in Table 4.6. On the left side of each figure is the behavior of the MSE quality

measure, in the middle the behavior of the PSNR measure is presented and in the right part the behavior of the CIRR measure.

Table 4.6: MSE, PSNR and CIRR measures.



It is observed that the behavior of the MSE and CIRR quality measures are increasing while the PSNR quality measure is decreasing. The MSE measure is increasing because each

time the original image is smoothed, the difference between it and the smoothed image is greater. In addition, it can be seen that each of the curves tends to a stable state when the number of iterations grows. This allows that MSE measure can be used as a stopping criterion in the diffusion process. In the case of the PSNR measure, its behavior is decreasing because the signal-to-noise ratio is decreasing. This is because image smoothing reduces signal information. For its part, the CIRR measure is increasing and visually has a similar behavior to the MSE measure, therefore, CIRR can be used as a stopping criterion as it does with the MSE measure.

According to the defined stopping criteria (JGTS and BF), Table 4.7 presents the number of iterations required in the smoothing process for each of the diffusion algorithms. As can be observed, the number of iterations required according to each criterion does not differ significantly from the different images used. For example, in the algorithm CED and EDD the number of iterations required is the same for the two stopping criteria, except for the cameraman and baboon image respectively.

Table 4.7: Number of iterations using JGTS and BF stopping criteria.


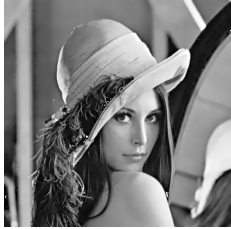
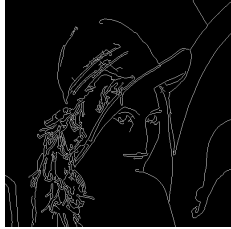
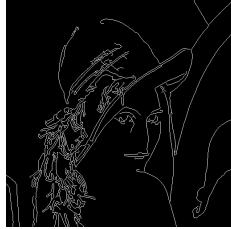
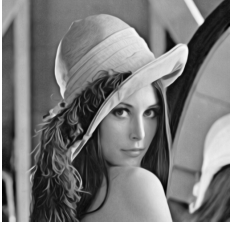
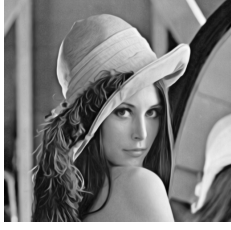
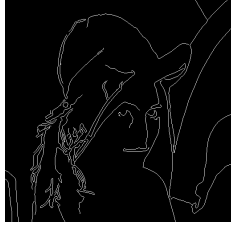
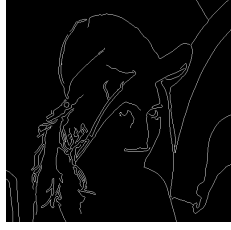

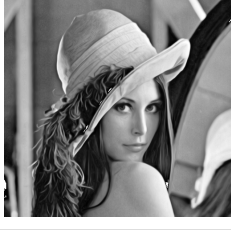







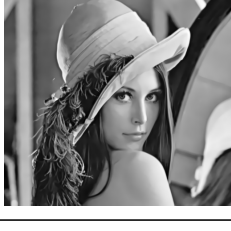


Image	Index	Isotropic	CED	cCED	EED	cEED
baboon	JGTS	10	6	6	5	8
	BF	10	6	7	6	9
barbara	JGTS	10	6	7	6	9
	BF	10	6	6	6	9
boat	JGTS	8	7	7	7	7
	BF	10	7	8	7	9
cameraman	JGTS	8	7	7	6	7
	BF	9	6	7	6	8
lena	JGTS	6	7	7	6	6
	BF	7	7	7	6	7

As the number of iterations defined by each stopping criteria is almost the same, the smoothed images have few differences. In the same way, the detected edges would not go significantly. To see the results obtained with a greater level of detail, the image of lena was selected. The results for each diffusion algorithm appear in the rows of Table 4.8. The first column corresponds to the images obtained using the JGTS stopping criterion. The second

column presents the images obtained by using the BF stopping criterion. Columns three and four present the edges of the images of columns one and two respectively.

With the results obtained using the two stopping criteria, it can be concluded initially that the two criteria satisfy the requirements to stop the diffusion algorithms.

Table 4.8: Edges of images obtained using JGTS and BF stopping criteria.

Diffusion	JGTS	BF	Edges - JGTS	Edges - BF
Isotropic				
CED				
cCED				
EED				
cEED				

The second part corresponds to the selection of the algorithm that smooths the images better. In order to discriminate the selected algorithms and to identify which of them improves the image without losing too much level of detail, the image of lena was selected. Based on the information of the image quality measurements presented in Figure 4.5, it can be identified that the behavior of MSE and CIRR indicate that the algorithm that best maintains the information is isotropic, followed by cEED, cCED, CED and EED respectively. In addition, according to the values obtained from the PSNR measure, the order of the algorithms remains the same.

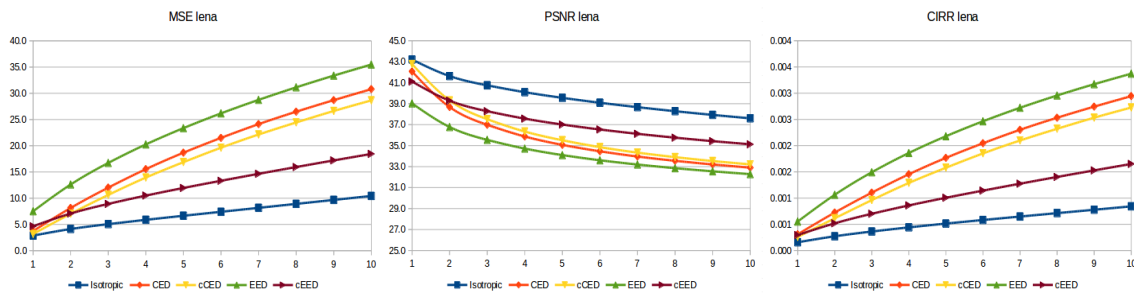


Figure 4.5: MSE, PSNR and CIRR measures of smoothed image of lena.

After exploring the lena image, the eye region was identified to zoom in and see in a greater level of detail the effect of the smoothing algorithms. Table 4.9 presents the original image and the images obtained when applying the three algorithms that show a better behavior with respect to the selected quality measures. As can be seen, the original image differentiates a semi-circular region in the centre of the eye. This region is maintained when applying isotropic softening and cEED, however, when cCED softening is applied, that region becomes blurred. The latter behavior is maintained when applying the CED and EED algorithms (see Table 4.8).

Table 4.9: Zoom in of eye region of smoothed image of lena.

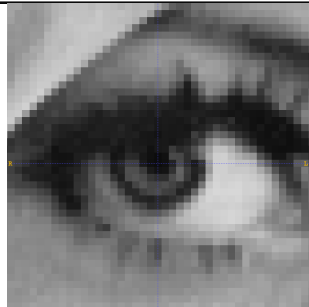
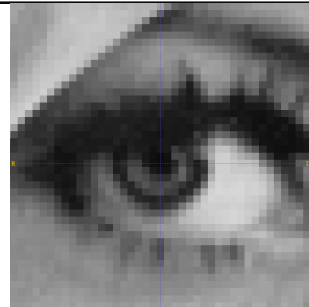
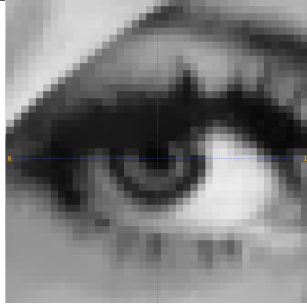
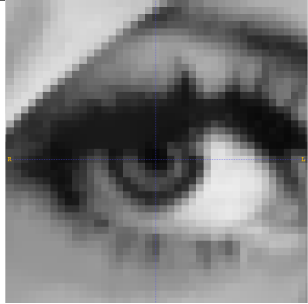
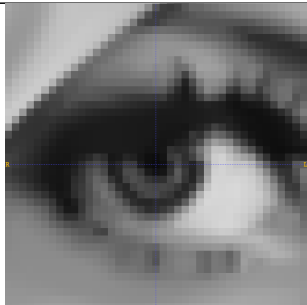
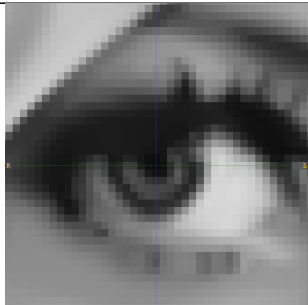
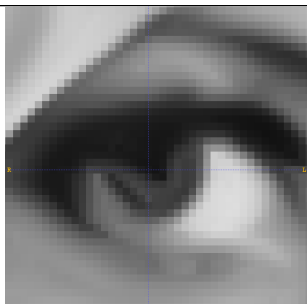
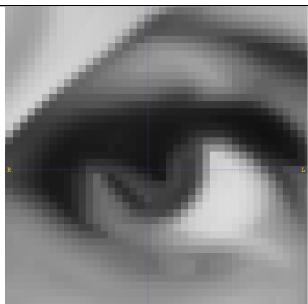
Diffusion	JGTS	BF
Original		

Table 4.9: Zoom in of eye region of smoothed image of lena.

Diffusion	JGTS	BF
Isotropic		
cEED		
cCED		

In addition, it is observed that the isotropic algorithm presents a lower smoothing in some regions compared to the cEED and cCED algorithms, for example, in the upper left region of the images, it is seen that the isotropic algorithm presents a more stepped variation than the results of the cEED and cCED algorithms.

To visually identify the impact of the smoothing algorithms, row 266 of the lena image was selected. Figure 4.6 shows the behavior of the original image and the images obtained from applying each smoothing algorithms. As can be seen, the isotropic diffusion algorithm generates a profile very close to the original image and therefore the image quality measures are better. The cEED algorithm maintains the intensity in the areas where edges are present and in regions with low-intensity variation it makes good smoothing, for example, in the

interval [386, 398]. The other smoothing algorithms generate a loss of information at the edges and attenuate their intensity, causing some of them to be eliminated, for example in the intervals [260, 272] and [320, 335].

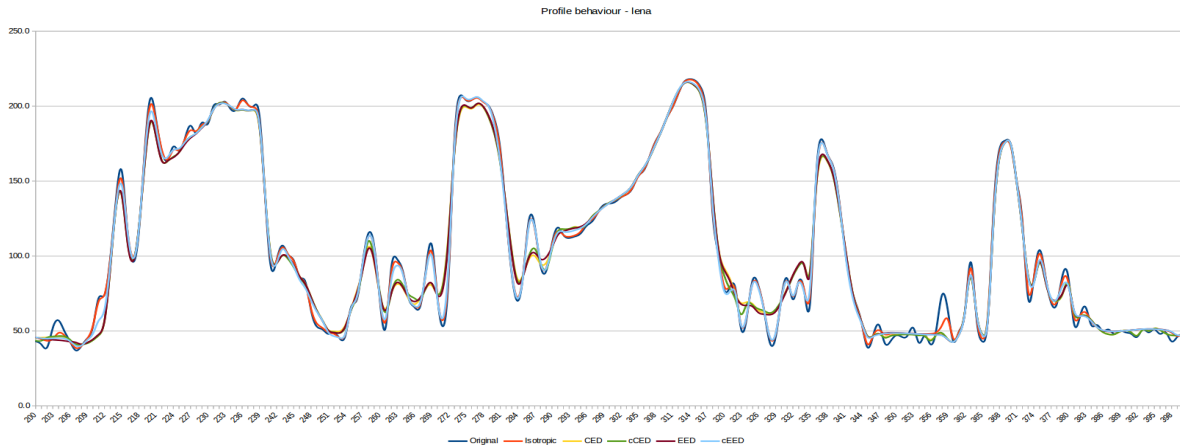


Figure 4.6: Profile behavior of smoothing algorithms using lena image.

Based on the elements mentioned above, the initial alternative to smooth the images by preserving the edge information corresponds to the nonlinear anisotropic diffusion algorithm cEED.

#### 4.5.1.2 Contrast-enhanced images

In this part of the tests, the five original images with contrast enhancement are used. The algorithm used to improve the image contrast was the multiscale top-hat morphological operator (MSTH).

The sequence of the tests is the same as in the previous section. First, the smoothing of the images was done using each of the five algorithms (Isotropic, CED, cCED, EED, and cEED), for which it was defined the number of iterations equal to 10. Then, for each of the iterations, the image quality using MSE, PSNR and CIRR is obtained. Finally, based on the MSE and CIRR measures, the JGTS and BF stopping criteria were calculated respectively.

The results for the images with contrast-enhancement are presented in Table 4.10. In parenthesis, the number of iterations obtained for the original images is placed when the value obtained for the images with contrast enhancement is different. The JGTS stopping criterion presents a difference in the number of iterations required by the smoothing algorithms on nine occasions, whereas the BF criterion does it on twenty-one occasions. From the nine variations of the JGTS criterion on four occasions, the number of iterations required by the smoothing



algorithm was increased, while from the twenty-one variations of the BF criterion, only on one occasion did the number of iterations increase.

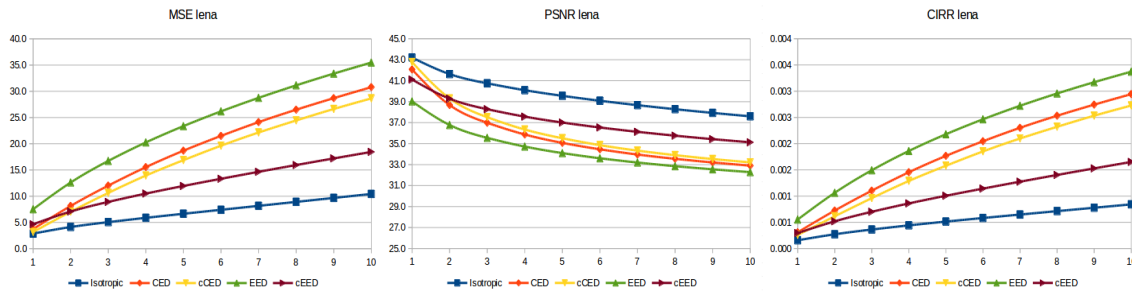
Image	Index	Isotropic	CED	cCED	EED	cEED
baboon	JGTS	10	6	6	5	10 (8)
	BF	10	4 (6)	5 (7)	4 (6)	9
barbara	JGTS	10	7 (6)	7	6	10 (9)
	BF	9 (10)	5 (6)	5 (6)	4 (6)	7 (9)
boat	JGTS	7 (8)	7	7	6 (7)	7
	BF	10	5 (7)	6 (8)	5 (7)	8 (9)
cameraman	JGTS	10 (8)	6 (7)	7	6	7
	BF	7 (9)	5 (6)	5 (7)	4 (6)	6 (8)
lena	JGTS	5 (6)	7	7	6	5 (6)
	BF	9 (7)	5 (7)	5 (7)	5 (6)	7

Table 4.10: Number of iterations using JGTS and BF stopping criteria.

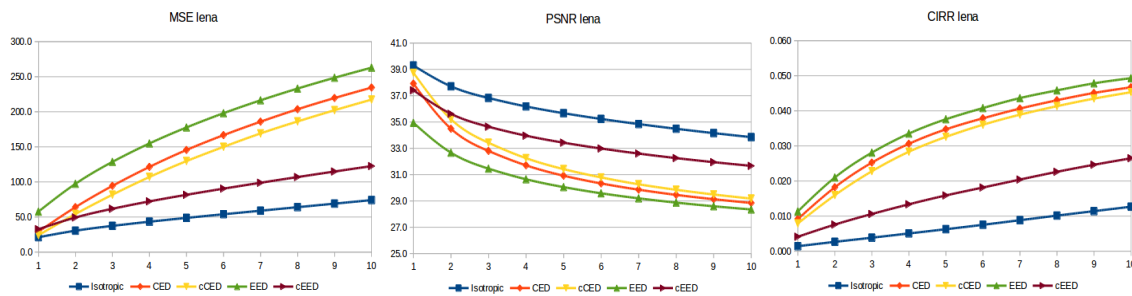
Figure 4.7 shows the behavior of the MSE, PSNR and CIRR image quality measures for lena image (original and contrast-enhancement). Figure 4.7 a) shows the curves obtained for the quality measures when applying each of the smoothing algorithms to the original image. Figure 4.7 b) presents the behavior of the quality measures when applying the smoothing algorithms to lena image with contrast enhancement.

As can be seen in Figures 4.7 a) and 4.7 b), the behavior of the curves has the same tendency, but the values are different.

In the case of the MSE measure, the scale of values increases when the image has contrast-enhanced, however, the slope of the curves is smaller. This circumstance causes the JGTS index to reduce the number of iterations required to stop the diffusion in the smoothing algorithms. For the PSNR measure, the values are reduced when the image presents an contrast-enhancement, although the slopes of the curves remain invariant. The CIRR measure presents a small change in the behavior of the curves of the quality measures obtained when using the image with contrast-enhancement. This change is reflected in the slope reduction, generating a steady state in the diffusion process and reducing the number of iterations required to stop the smoothing algorithms. The most significant changes are given for the CED, cCED and EED algorithms as shown in Table 4.10.



a) MSE, PSNR and CIRR of original image



b) MSE, PSNR and CIRR of MSTH image







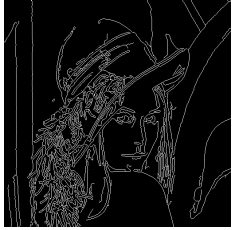



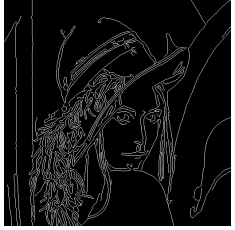



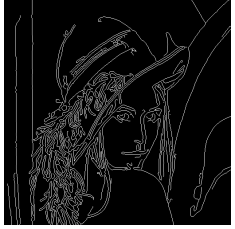
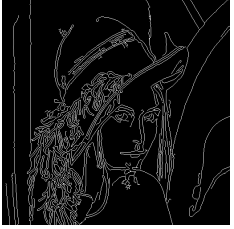


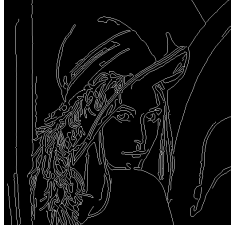
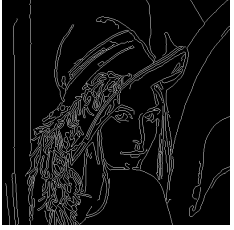


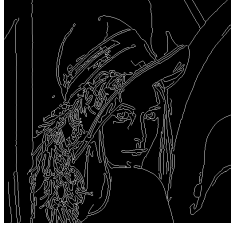
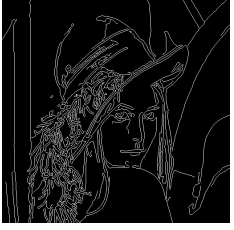
Figure 4.7: MSE, PSNR and CIRR of original and contrast-enhancement image of lena.

Initially, it can be concluded that the CIRR quality measure allows obtaining better results on the image quality, therefore, the BF stopping criterion proposed allows to have a better precision to stop the execution of the smoothing algorithms.

However, more information is required to indicate that the BF stopping criterion presents better results than the JGTS criterion. Therefore, the edges of the images obtained by each of the smoothing algorithms are calculated using the two stopping criteria. To calculate the edges, the algorithm proposed by Canny was used using predefined and equal parameters for all images. The results are presented in Table 4.11. Each of the rows corresponds to their order to the image of lena with contrast-enhancement, the images obtained by using the isotropic, CED, cCED, EED and cEED algorithms, respectively. The second column corresponds to the resulting images when using the JGTS stopping criterion. The third column to the images obtained when using the BF stopping criterion proposed and the two last columns present the edges of the images of column two and three respectively.

As shown in Table 4.11, the images obtained when using the JGTS stopping criterion generate a smaller number of edges with respect to the images obtained when using the BF stopping criterion.

Table 4.11: Smooth images and edges of lena with contrast-enhancement.

Image	JGTS	BF	Edges - JGTS	Edges - BF
lena-MSTH				
Isotropic				
CED				
cCED				
EED				
cEED				

To determine in a more objective way which smoothing algorithm generates better results a zoom of the eye region of lena is shown in Table 4.12. In the rows the images are shown in the following order, image with contrast-enhancement, and smoothed images obtained after apply isotropic, cEED and cCED smoothing algorithms respectively.

Table 4.12: Zoom in of eye region of smoothed image of lena-MSTH.


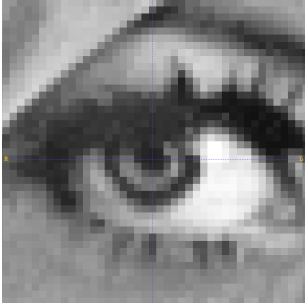


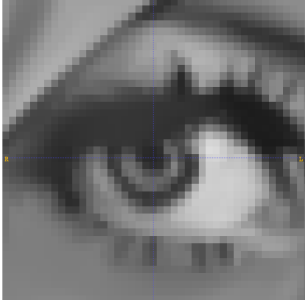
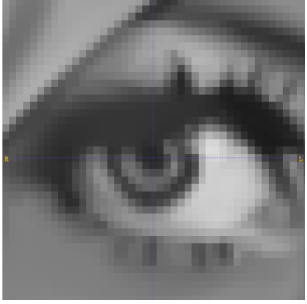
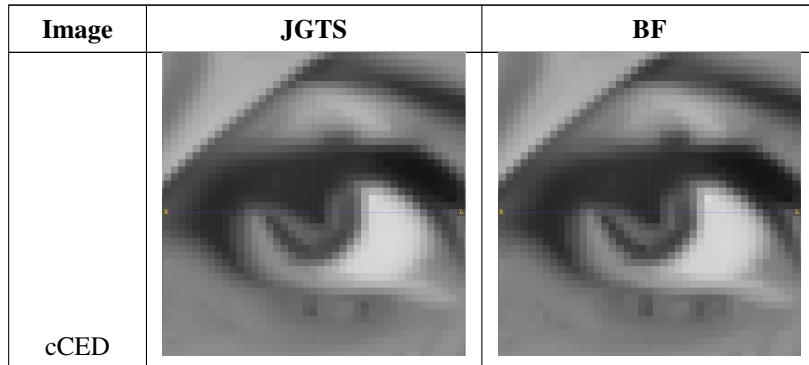
Image	JGTS	BF
Enhanced		
Isotropic		
cEED		

Table 4.12: Zoom in of eye region of smoothed image of lena-MSTH.



The behavior of the smoothing algorithms presents the same behavior as with the original image of lena. Specifically, the isotropic algorithm presents less smoothing in some regions with respect to the results obtained when using the cEED and cCED algorithms. Furthermore, the cCED algorithm generates a more blurred region in the central area of the eyes with respect to the results produced by the isotropic and cEED algorithms. In addition, the cEED algorithm generates more homogeneous smoothing regions outside the edges of the regions of interest than the cCED algorithm.

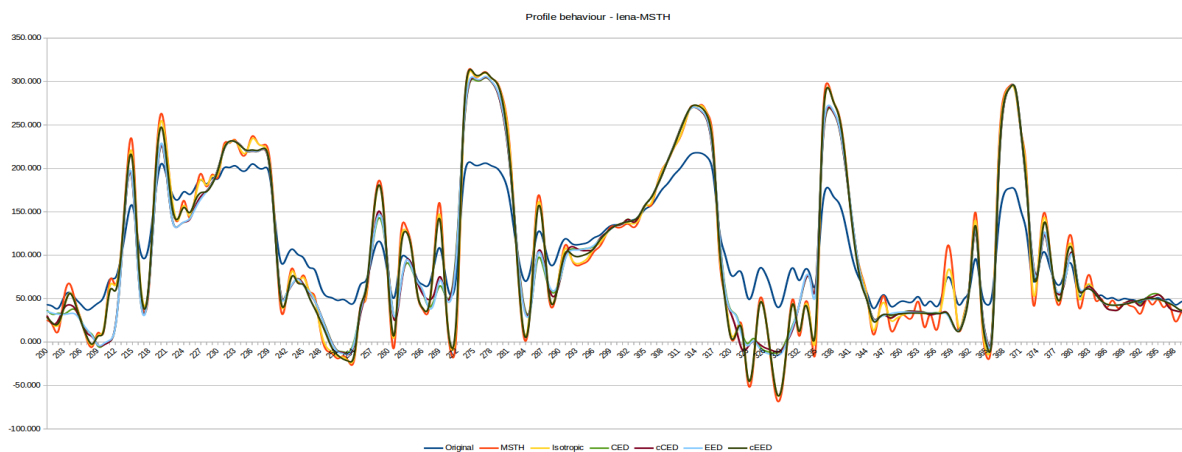


Figure 4.8: Lena profile of original image, enhanced image, isotropic, CED, cCED, EED, and cEED.

Figure 4.8 shows the profile behavior of row 266 of the original image, contrast-enhanced image, and the images obtained from applying each smoothing algorithms. In the same way

as for the original image, the isotropic diffusion algorithm generates a profile very close to the lena-MSTH image and therefore the image quality measures are better. The cEED algorithm maintains the intensity in the areas where edges are present and in regions with low-intensity variation it makes good smoothing. The other smoothing algorithms generate a loss of information at the edges and attenuate their intensity, causing some of them to be eliminated.

In conclusion, the isotropic and cEED smoothing algorithms preserve information in boundary regions between different regions. In addition, using the BF stopping criterion allows obtaining more edges than when using the JGTS criterion. This can be seen in Table 4.13, which presents the edges of the images obtained by using the isotropic and cEED smoothing algorithms applying each of the stopping criteria.

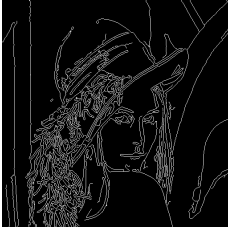
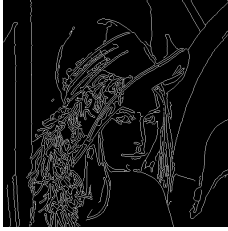


Smoothing	JGTS	BF
Isotropic		
cEED		

Table 4.13: Edges of Isotropic and cEED smoothing algorithms.

## 4.5.2 3D case

For the tests with 3D images were selected ten CT images of head-neck. It is proceeded in a similar way to the 2D case, ie, the original images are used first to evaluate the quality measures and to apply the smoothing algorithms in order to identify which stopping criterion performs better. Contrast-enhanced images are then used to identify if there is any change in the behavior of the smoothing algorithms and in the stopping criteria.

### 4.5.2.1 Original images

Using the ten original CT images, the isotropic, CED, cCED, EED and cEED smoothing algorithms are applied. Each algorithm is applied during ten iterations. For each iteration, the

MSE, PSNR and CIRR quality measures are calculated. The results obtained for the "image1" are presented in Tables 4.14, 4.15 and 4.16.

Table 4.14 presents the behavior of the MSE measure for each of the smoothing algorithms. As it is observed, initially the MSE values are lower for the isotropic, CED and cCED algorithms, however, the behavior of the EED and cEED algorithms is changing as the algorithms evolve. As a result, after the fifth iteration, the algorithms that have the lowest value for the MSE measure are the EED, cEED and isotropic, respectively.

Table 4.14: MSE measure of image1.

Iteration	Isotropic	CED	cCED	EED	cEED
1	112.001	77.337	68.603	155.403	152.838
2	174.797	207.413	184.993	279.909	273.805
3	214.714	349.871	312.740	383.744	374.220
4	246.135	497.161	445.154	480.766	467.881
5	273.531	646.661	580.202	574.370	558.264
6	298.858	797.932	717.965	666.805	647.565
7	322.910	951.567	858.653	758.583	736.321
8	346.146	1105.830	999.920	849.815	824.650
9	368.806	1259.880	1141.160	941.174	913.183
10	391.134	1413.370	1282.890	1032.330	1001.610

Table 4.15 presents the behavior of the PSNR measure for the smoothing algorithms analyzed. The results obtained after the ten iterations tend to be similar for the CED, cCED, EED and cEED algorithms, whereas the isotropic algorithm generates the highest value for the PSNR measure.

Table 4.15: PSNR measure of image1.

Iteration	Isotropic	CED	cCED	EED	cEED
1	43.064	47.737	48.105	42.150	42.193
2	41.064	43.076	43.400	39.578	39.628
3	40.244	40.586	40.890	38.231	38.285
4	39.748	38.929	39.225	37.271	37.327

Table 4.15: PSNR measure of image1.

Iteration	Isotropic	CED	cCED	EED	cEED
5	39.390	37.705	37.993	36.517	36.574
6	39.101	36.737	37.011	35.890	35.948
7	38.855	35.932	36.195	35.355	35.412
8	38.637	35.255	35.511	34.889	34.946
9	38.439	34.676	34.928	34.475	34.531
10	38.255	34.172	34.418	34.103	34.159

Table 4.16 shows the evolution of the CIRR measure for the smoothing algorithms. This table differs in the behavior obtained for the 2D case because the EED and cEED algorithms have a better behavior than the isotropic, CED and cCED algorithms. The results are consistent with the definition of the quality measure and with the characteristics of the diffusion algorithms.

Table 4.16: CIRR measure of image1.

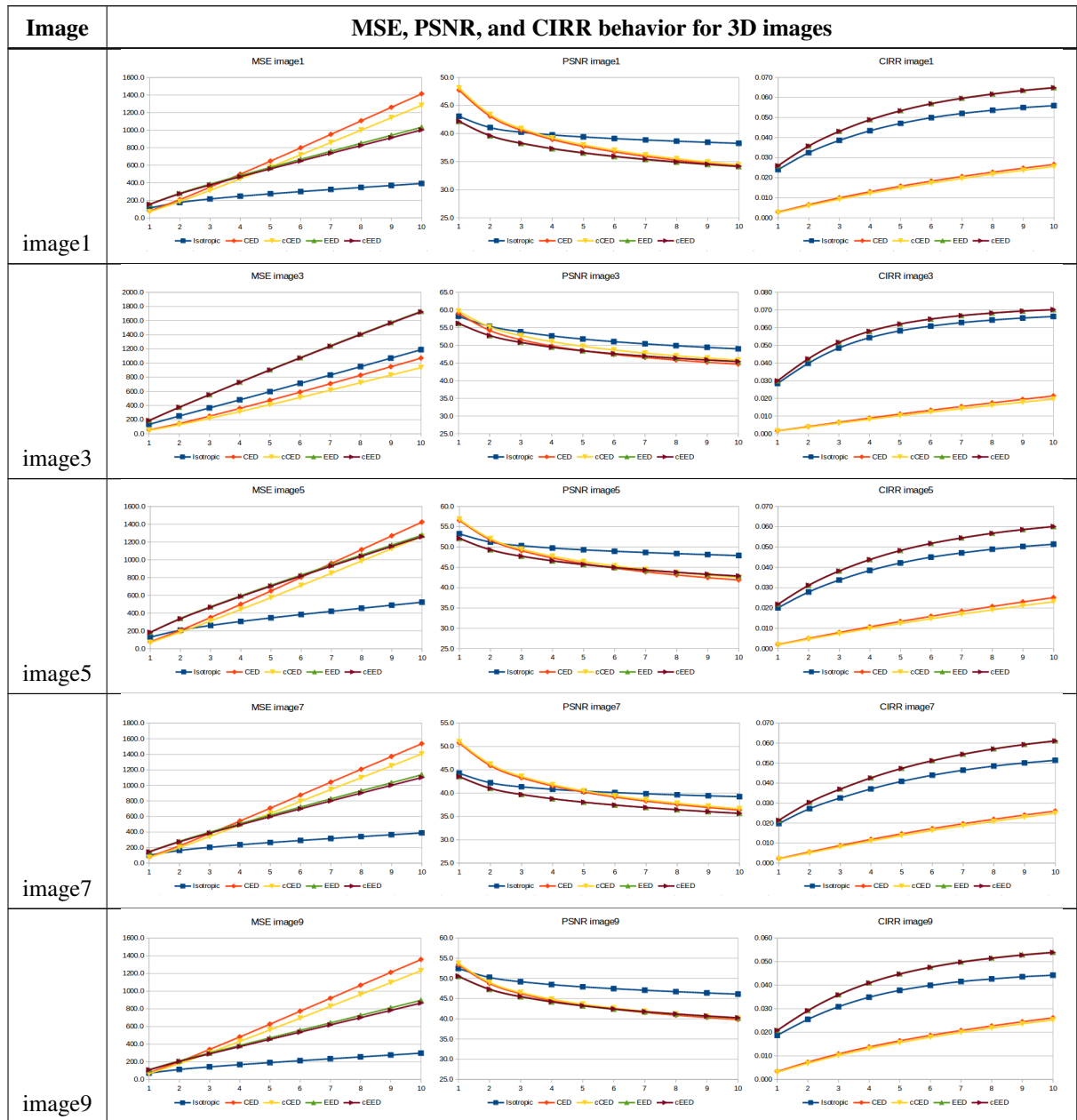
Iteration	Isotropic	CED	cCED	EED	cEED
1	0.02397	0.00283	0.00255	0.0260	0.0259
2	0.03242	0.00652	0.00598	0.0358	0.0357
3	0.03857	0.00991	0.00920	0.0432	0.0431
4	0.04339	0.01297	0.01213	0.0490	0.0488
5	0.04706	0.01575	0.01482	0.0534	0.0533
6	0.04989	0.01827	0.01730	0.0570	0.0569
7	0.05202	0.02059	0.01959	0.0597	0.0595
8	0.05361	0.02274	0.02172	0.0618	0.0617
9	0.05494	0.02473	0.02369	0.0636	0.0635
10	0.05592	0.02659	0.02554	0.0649	0.0648

Table 4.17 shows the behavior of the MSE, PSNR and CIRR quality measures for images 1, 3, 5, 7 and 9. The algorithms to be selected are those that generate the lowest value for the MSE measure and a higher value for PSNR and CIRR measures. However, it should be



considered that as the image is smoothed the value of the PSNR measure decreases, for that reason, the EED and cEED algorithms present a lower value than the other algorithms.

Table 4.17: MSE, PSNR, and CIRR measures.



The number of iterations defined by each of the stopping criteria for the ten test images is presented in Table 4.18. As can be seen, isotropic diffusion is similar using the two stopping

criteria except for image3 and image4. In the case of the EED and cEED diffusion, the number of iterations defined by BF criterion is half the number of iterations required by the JGTS criterion. JGTS stopping criterion generates a greater number of iterations required to stop the diffusion in all the test images with respect to BF criterion. In addition, the number of iterations defined by the BF criterion is the same for the isotropic, EED and cEED diffusion algorithms.

Image	Index	Isotropic	CED	cCED	EED	cEED
image1	JGTS	5	10	10	8	8
	BF	4	7	7	4	4
image2	JGTS	5	10	10	8	8
	BF	4	7	7	4	4
image3	JGTS	10	10	10	9	9
	BF	4	9	9	4	4
image4	JGTS	10	10	10	10	10
	BF	4	9	9	4	4
image5	JGTS	5	10	10	8	8
	BF	4	8	8	4	4
image6	JGTS	5	10	10	8	8
	BF	4	8	8	4	4
image7	JGTS	5	10	10	9	9
	BF	4	8	8	4	4
image8	JGTS	5	10	10	9	9
	BF	4	8	8	4	4
image9	JGTS	6	10	10	9	9
	BF	4	6	7	4	4
image10	JGTS	6	10	10	9	9
	BF	4	6	7	4	4






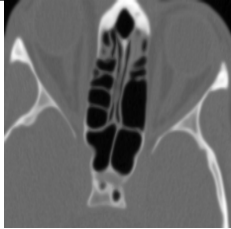














Table 4.18: Number of iterations using JGTS and BF stopping criteria.

To see the impact of the number of iterations in the diffusion algorithms, the edges of the image1 for the images generated by each of them were calculated. The algorithm proposed by Canny was used for this purpose. As the largest variation in the number of iterations was presented in the EED and cEED diffusion algorithms, it is expected that there is a significant variation in the edges.

The results of the edges are presented in Table 4.19. In the rows is found each of the

diffusion algorithms. In the second column the images obtained by using the JGTS stopping criterion for each diffusion algorithm. In the third column the images using the BF criterion. In the fourth and fifth columns, the edges correspond to the second and third columns respectively.

Table 4.19: Edges of images obtained using JGTS and BF stopping criteria.

Diffusion	JGTS	BF	Edges - JGTS	Edges - BF
Isotropic				
CED				
cCED				
EED				
cEED				

As can be seen in Table 4.19, there is no variation in the edge detection in the images obtained using the EED and cEED diffusion by applying the two stopping criteria. In addition, isotropic diffusion presents results similar to EED and cEED diffusion. On the other hand, the diffusion CED and cCED allow detecting a greater number of edges with respect to the other three algorithms, independent of the stopping criterion used. This is due to the fact that the CED and cCED algorithms apply less smoothing in the internal regions of the image structures.

In conclusion, the edges detected in the images obtained using each one of the diffusion algorithms are equal independent of the stopping criterion used. Therefore, it is considered that the BF stopping criterion is more efficient than the JGTS criterion because it allows stopping the diffusion in a smaller number of iterations.

#### 4.5.2.2 Contrast-enhanced images

Using the images with contrast enhancement diffusion is calculated using the algorithms defined for ten iterations. For each iteration, the quality measures of the images are evaluated, which are presented in Table 4.20. For visualization reasons, only information for image1, image3, image5, image7 and image9 is presented. The behavior of the MSE measure does not present significant changes, except for the image3. The curves of the PSNR measure show greater changes, for example, for all the images, the PSNR curves obtained from the isotropic diffusion have a higher value than when using the other algorithms. In addition, the cEED and EED algorithms present a better PSNR value with respect to the CED and cCED algorithms. The curves of the CIRR measurement maintain the same form as for the case of the original images.

Table 4.20: MSE, PSNR, and CIRR behavior for 3D enhanced-images.

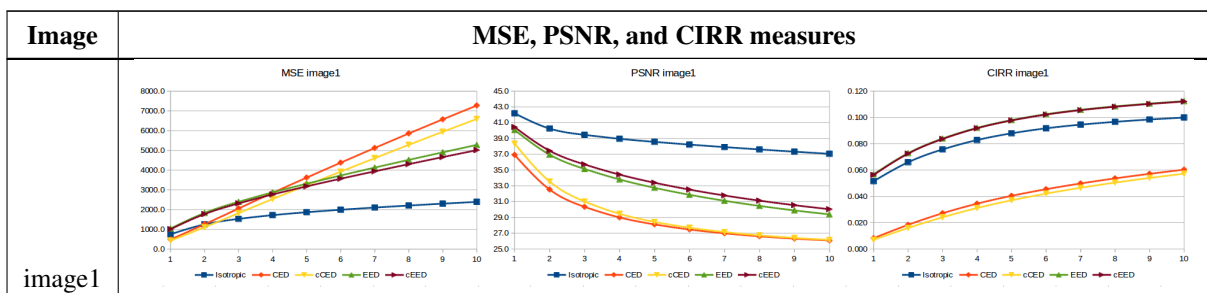
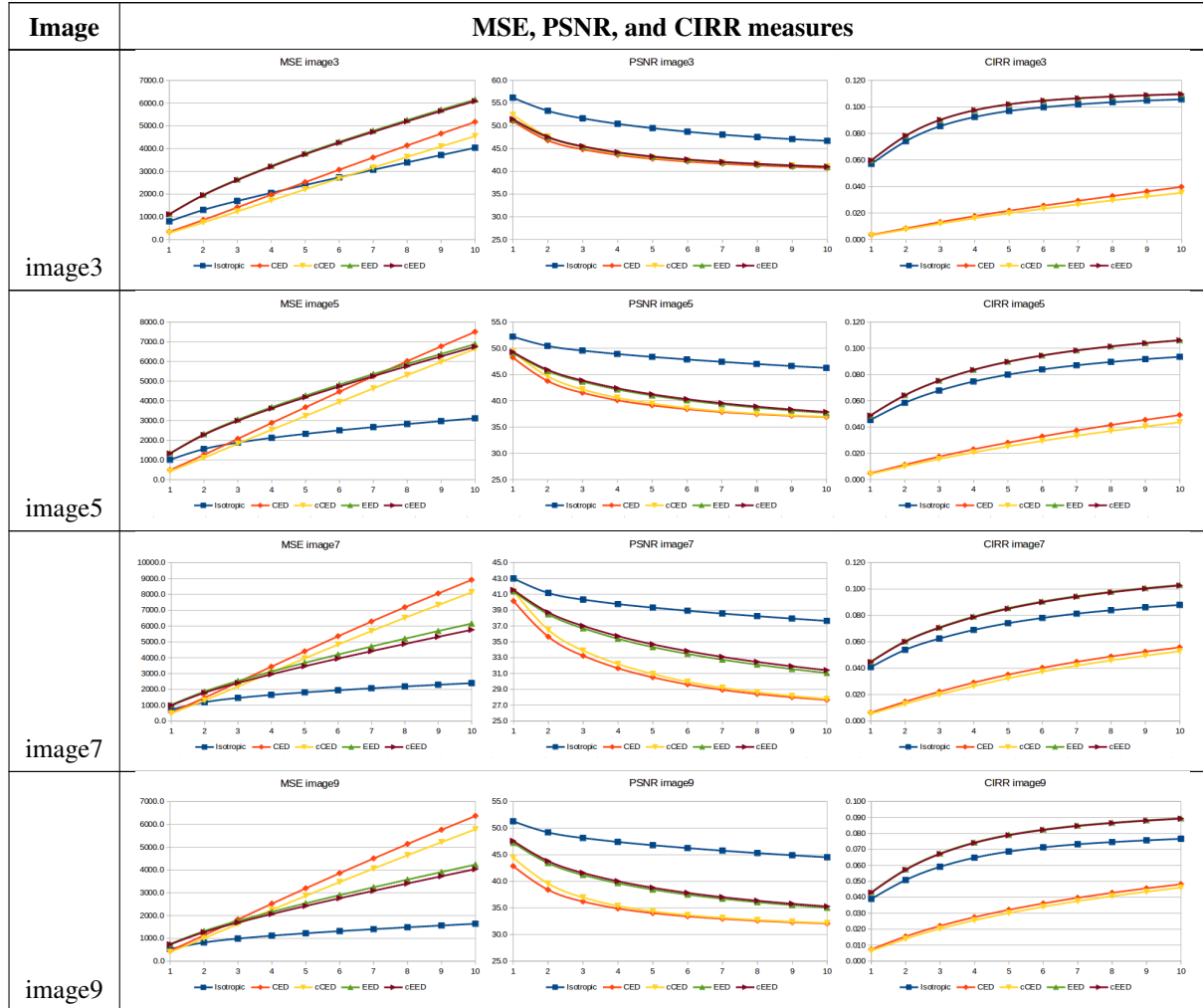


Table 4.20: MSE, PSNR, and CIRR behavior for 3D enhanced-images.



The number of iterations defined by each of the stopping criteria for the different smoothing algorithms is presented in Table 4.21. In addition, the number of iterations required for the original images is presented in parenthesis. This in order to compare the variation in each of the cases. As can be seen, the variation in the number of iterations required for the isotropic, EED and cEED algorithms using the BF stopping criterion was reduced by one for most cases. On the other hand using the criterion JGTS reduced the number of iterations but not in the same proportion. For the CED and cCED smoothing algorithms, the changes were more varied for the two stopping criteria.

The isotropic algorithm presents a number of iterations of four and three for the JGTS and

BF criteria respectively, which implies that there is no greater variation for this case. The only exceptions correspond to image3 and image4 where the number of iterations defined by the JGTS criterion in seven and ten respectively, while for the BF criterion is three for the two images.

The CED and cCED algorithms require a greater number of iterations in comparison to the others. In the case of the JGTS criterion for most cases, it was nine whereas for the BF criterion it was six.

In the EED and cEED algorithms, the number of iterations defined by the JGTS criterion is double or more with respect to BF criterion. For the JGTS criterion the average is seven and for the BF criterion, is three.

Image	Index	Isotropic	CED	cCED	EED	cEED
image1	JGTS	4 (5)	9 (10)	9 (10)	6 (8)	6 (8)
	BF	3 (4)	6 (7)	6 (7)	3 (4)	3 (4)
image2	JGTS	4 (5)	9 (10)	9 (10)	6 (8)	6 (8)
	BF	3 (4)	6 (7)	6 (7)	3 (4)	3 (4)
image3	JGTS	7 (10)	9 (10)	10	7 (9)	7 (9)
	BF	3 (4)	8 (9)	8 (9)	3 (4)	3 (4)
image4	JGTS	10	10	10	9 (10)	9 (10)
	BF	3 (4)	9	9	3 (4)	3 (4)
image5	JGTS	4 (5)	9 (10)	9 (10)	6 (8)	6 (8)
	BF	3 (4)	8	7 (8)	3 (4)	3 (4)
image6	JGTS	4 (5)	9 (10)	9 (10)	7 (8)	7 (8)
	BF	3 (4)	8	7 (8)	3 (4)	3 (4)
image7	JGTS	4 (5)	9 (10)	9 (10)	7 (9)	7 (9)
	BF	3 (4)	7 (8)	7 (8)	4	4
image8	JGTS	4 (5)	9 (10)	9 (10)	7 (9)	7 (9)
	BF	3 (4)	6 (8)	7 (8)	4	4
image9	JGTS	4 (6)	9 (10)	9 (10)	7 (9)	7 (9)
	BF	3 (4)	6	6 (7)	3 (4)	3 (4)
image10	JGTS	4 (6)	9 (10)	9 (10)	7 (9)	7 (9)
	BF	3 (4)	6	6 (7)	3 (4)	3 (4)

Table 4.21: Number of iterations using JGTS and BF stopping criteria.

Table 4.22 presents the result of calculating the edges for each one of the images produced by the smoothing algorithms using each of the stopping criteria.

The changes in the edges of the images obtained from the isotropic algorithm are smaller, however, it is observed that more edges are generated in the image obtained when BF criterion is used.

The edges of the images obtained from the EED and cEED algorithms using the BF criterion present better results than those obtained using the JGTS criterion. It is important to remember that the number of iterations defined by BF criterion is half of the JGTS criterion.

Table 4.22: Edges of images obtained using JGTS and BF stopping criteria.

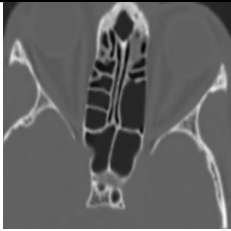
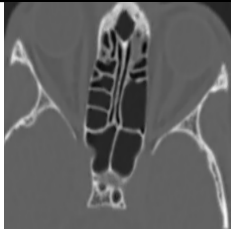


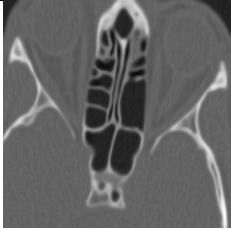
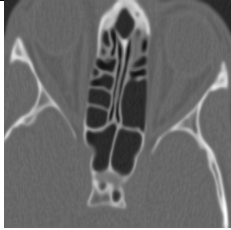


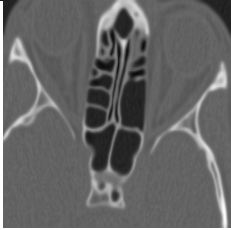
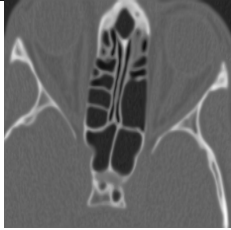






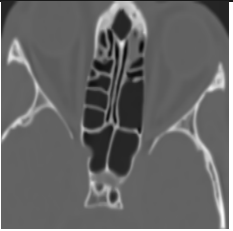



Diffusion	JGTS	BF	Edges - JGTS	Edges - BF
Isotropic				
CED				
cCED				
EED				

Table 4.22: Edges of images obtained using JGTS and BF stopping criteria.

Diffusion	JGTS	BF	Edges - JGTS	Edges - BF
cEED				

The difference between the edges of the images produced by the CED and cCED algorithms are not significant. But as in the previous case, the BF criterion requires fewer iterations than the JGTS criterion.



# Chapter 5

## Features and shapes detection

“Geometry is not true, it is advantageous.”

– *Henri Poincare.*

Generally, feature detection is a necessary phase for further image processing such as object recognition, image retrieval, etc. In the literature, many kinds of feature such as segments, outlines, corners, interest points have been studied. However, these features are highly sensitive to noise and do not describe the local structures of object which are widely useful for generic object recognition. As an aim of this thesis is the segmentation of the upper airways from head-neck 3D images, it is necessary to detect features or shapes related to thin tissue, elongated structures, and objects with low contrast in the presence of noise and nearby structures with similar intensity. Ridges satisfy these demands.

### 5.1 State of the art

Features and shapes detection is a fundamental and important problem in computer vision and image processing. It is a low-level processing step which serves as the essential part for computer vision based applications [76]. Feature and shapes detection is referred as the identification of interested image primitives and structures (e.g. points, lines/curves, regions, and curvilinear structures), for the purpose of highlighting salient visual cues in digital images [76]. It use pixel intensities as the input and image structures indicating different characteristic properties as the output.

Feature detection aims to locate significant feature regions on images depending on their intrinsic characteristics and applications. These regions can be defined in global or local neighborhood and distinguished by shapes, textures, sizes, intensities, statistical properties, and so on [208]. Although the application scopes of visual features are widely different, it is the ulti-

mate goal to extract features or shapes with high stability effectively and efficiently. Generally, the difficulties in features and shapes detection are caused by the changes in scale, viewpoint, illumination, image quality, etc [76]. Local feature detection methods analyze local intensity patterns to find regions that satisfy desired uniqueness or stability criteria. These methods detect image features such as edges, lines, corners, circles, ellipses, and so on. In general, local features detectors show their advantages of robustness to object occlusion or image deformation. Global features describe an image as a whole and can be interpreted as a particular property of the image involving all pixels. In addition, global features include contour representations, shape descriptors, and texture features. Global descriptors generally used in image retrieval, object detection and classification.

Shapes detection methods analyze local structures, which are called curvilinear structures. They represents a line or a curve with some width, and it differs from conventional line or curve features, which are usually extracted based on edges. Curvilinear structures are more structured features and contain more information than edges. They can be found in most natural images, and their detection is particularly useful. For instance, the human body contains various types of line, sheet, and blob structures. For example, blood vessels, bone cortices, and nodules are characterized by line, sheet, and blob structures, respectively [73, 209]. Three-dimensional local structures have been shown to be useful for 3D modeling of anatomical structures to improve their extraction and quantification [73, 105, 107, 210–214].

This chapter presents a brief review of features and shapes detection methods. Next, a new method to detect local structures based on principal curvatures is presented. Then, proposed method is compared with PCBR algorithm [215] using 2D images. Finally, tests using 3D synthetic images and 3D medical images with proposed method are performed.

### 5.1.1 Local features

As mentioned by Tomasi et.al. [216], several definitions of a “good feature” have been proposed in the literature, based on an a priori notion of what constitutes an “interesting” window. According to Schmid et. al. [217] a “interest point” mean any point in the image for which the signal changes two-dimensionally. Tuytelaars et. al. [218], mentioned that the term local feature (also known as interest points, key points, and salient features) can be either points, regions or even edge segments. They can be defined as a specific pattern which unique from its immediately close pixels, which is generally associated with one or more of image properties [76, 218]. These properties such as edges, contours, corners and regions are much related to human visual perception [76, 219, 220]. Tuytelaars et. al. [218] emphasized that most local feature detectors are in fact extractors, and that the features themselves are usually covariant, in general, distortions of the image. However the term “detector” can be used for

all procedures which extract some relevant features, irrespective of their invariance properties [221].

For example, Moravec propose to use windows with high standard deviations in the spatial intensity profile [222], Marr, Poggio, and Ullman prefer zero crossings of the Laplacian of the image intensity [223], and Kitchen, Rosenfeld, Dreschler, and Nagel define corner features based on first and second derivatives of the image intensity function [224], [225].

By Hassaballah et. al. [226], image features can be divided into global features and local features. Global features (e.g., color and texture) aim to describe an image as a whole and can be interpreted as a particular property of the image involving all pixels. While, local features aim to detect keypoints or interest regions in an image and describe them. Schmid et. al. [217] divided methods into three categories: contour based, intensity based and parametric model based methods. Rodehorst et. al. [227] classify methods into three categories: contour based, intensity based and model based.

### 5.1.2 Features detectors

Interest operators detect salient image features, which are distinctive in their neighborhood and they are reproduced in corresponding images in a similar way [227]. A wide variety of interest point and corner detectors exist in the literature [76, 218, 226, 228, 229]. Based on these advances, over the last decades image feature detectors and descriptors have become popular tools, for example, in the image processing, computer vision and image recovery, additionally, they are being applied widely in a large number of applications[226].

Tuytelaars and Mikolajczyk [218] describe a set of characteristics that the detectors must satisfy. These are listed below.

### 5.1.3 Characteristics of Feature Detectors

As mentioned by Tuytelaars et. al. [218], good features detectors should have the following properties:

- *Repeatability*: Given two images of the same object or scene, taken under different viewing conditions, a high percentage of the features detected on the scene part visible in both images should be found in both images.
- *Distinctiveness/informativeness*: The intensity patterns underlying the detected features should show a lot of variation, such that features can be distinguished and matched.
- *Locality*: The features should be local, so as to reduce the probability of occlusion and to allow simple model approximations of the geometric and photometric deformations

between two images taken under different viewing conditions (e.g., based on a local planarity assumption).

- *Quantity*: The number of detected features should be sufficiently large, such that a reasonable number of features are detected even on small objects. However, the optimal number of features depends on the application. Ideally, the number of detected features should be adaptable over a large range by a simple and intuitive threshold. The density of features should reflect the information content of the image to provide a compact image representation.
- *Accuracy*: The detected features should be accurately localized, both in image location, as with respect to scale and possibly shape.
- *Efficiency*: Preferably, the detection of features in a new image should allow for time-critical applications.

Hassaballah et al. [226] included:

- *Generality*: the feature detection algorithm should be able to detect features that can be used in different applications.

Additionally, Dickscheid et. al. [221] included others important properties of a good detector:

- *Robustness*: The features should be robust against typical distortions such as image noise, different lighting conditions, and camera movement.
- *Sparseness*: The amount of data given by the features should be significantly smaller compared to the image itself, in order to increase efficiency of subsequent processing.
- *Speed*: A feature detector should be fast.
- *Completeness*: Given that the above requirements are met, the information contained in an image should be preserved by the features as much as possible. In other words, the amount of information coded by a set of features should be maximized, given a desired degree of robustness, sparseness, and speed.

Tuytelaars et. al. [218] indicate that *repeatability*, arguably the most important property of all, it can be achieved in two different ways: either by invariance or by robustness.

- *Invariance*: When large deformations are to be expected, the preferred approach is to model these mathematically if possible, and then develop methods for feature detection that are unaffected by these mathematical transformations.

- *Robustness*: In case of relatively small deformations, it often suffices to make feature detection methods less sensitive to such deformations, i.e., the accuracy of the detection may decrease, but not drastically so. Typical deformations that are tackled using robustness are image noise, discretization effects, compression artifacts, blur, etc. Also geometric and photometric deviations from the mathematical model used to obtain invariance are often overcome by including more robustness.

Finally, according to Dickscheid et. al. [221], *complementarity* of features plays a key role when using multiple detectors in an application. It is strongly related to *completeness*, as the information coded by sets of complementary features is higher than that coded by redundant feature sets.

### 5.1.4 Local features detectors

In the literature different detectors and descriptors have been presented. Achieved results vary according to the used images and parameters, therefore assessments of the performances are required. Previous works describing and comparing feature detectors have been reported by Ziou et. al. [230], Schmid et. al., 2000 [217], Zuliani et. al., 2004 [231], Mikołajczyk et. al., 2005 [232], Rodehorst et. al., 2006 [227], Moreels et. al., 2007 [233], Tuytelaars et. al., 2007 [218], Gauglitz et. al. [228], Miksik et. al., 2012 [234], Li et. al., 2015 [76], and Hassaballah et. al., 2016 [226].

To better describe the recent progress in feature detection methods, related concepts are firstly clarified.

1. *Edge* refers to pixel at which the image intensities change abruptly. Image pixels are discontinuous at different sides of edges.
2. *Contour/boundary* has ambiguous definitions. Since the main focus is centered on low-level characteristics, they are defined as intersecting lines/curves of different segmented regions.
3. *Corner* refers to the point at which two different edge directions occur in the local neighborhood. It is the intersection of two connected contour lines.
4. *Region* refers to a closed set of connected points. Nearby and similar pixels are grouped together to compose the interest region.

These significant descriptors can facilitate the pattern classifiers with their compactness and discrimination. Frequently used features are corners, edges and blobs (see Figure 5.1).

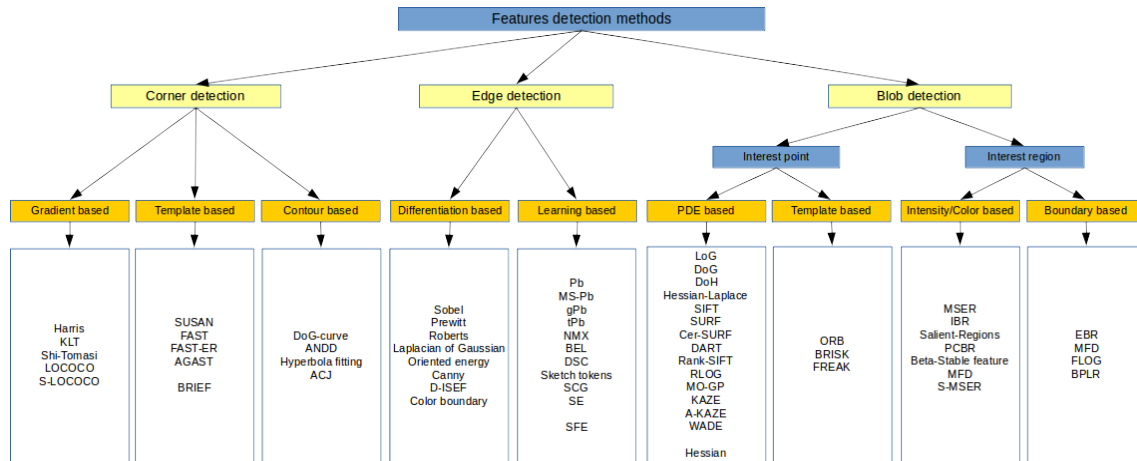


Figure 5.1: Local features detectors

In general, good features describe information contained in an image compactly and effectively, and facilitate higher-level processes such as matching, segmentation or recognition. Based on the classification presented in Figure 5.1, below the most relevant feature detection methods are briefly described.

#### 5.1.4.1 Edge detection

Edge refers to sharp changes in image brightness. According to Kitchen et. al. [235], edge is the boundary between two adjacent regions in an image, each region homogeneous within itself, but differing from the other with respect to some given local property. Thus, an edge should ideally be line-like. [235]. Li et. al. [76], proposed that contour/boundary can be viewed as the generalized definition of edge which indicates the intersection of different regions.

Edge detection has been a challenging task in low level image processing. One of the reasons is because, the development of the optimal or ideal edge detection scheme is difficult for the absence of evident and clear ground truth data, on which to evaluate performances unambiguously [236, 237]. Other, due to the presence of noise and quantization of the original image, during edge detection it is possible to locate intensity changes where edges do not exist. For similar reasons, it is also possible to completely miss existing edges. The degree of success of an edge-detector depends on its ability to accurately locate true edges. For similar reasons, it is also possible to completely miss existing edges [238]. Additional, edge localization is another problem encountered in edge detection. The addition of noise to an image can cause the position of the detected edge to be shifted from its true location. Further, difficulty in any edge detection system arises from the fact that the sharp intensity transitions which indicate an

edge are sharp because of their high-frequency components. As a result, any linear filtering or smoothing performed on these edges to suppress noise will also blur the significant transitions.

Differential operation is used to capture the strength and position of discontinuities in image brightness. Recently efforts have been devoted to multi-resolution edge analysis, sub-pixel edge detection and hysteresis thresholding [76]. In addition, other approaches are available for edge detection, some are based on error minimization, maximizing an object function, neural network, fuzzy logic, wavelet approach, bayesian approach, morphology, and genetic algorithms [239].

As shown in Figure 5.1, edge detection methods can be classified in differentiation-based and learning-based. Differentiation-based filters are convolved to identify edge points. Learning based methods are modeled as a machine learning based framework to discriminate edge points from smooth regions [76].

First, *differentiation-based methods* depend on the derivatives order. First-order differentiation based gradient operators appear in pairs for the two-dimensional case (e.g., Prewitt, Sobel). By those operators, gradients at different orientations are computed. Local maximas of gradient magnitudes are recorded as edges. Second-order differentiation filters such as Laplacian of Gaussian (LoG) find zero-crossings as the edge positions. Gaussian smoothing is necessary since differential operation is sensitive to noise. Directional differentiations such as Oriented Energy (OE) [240] adopt a batch of filters at different orientations to obtain brightness changes. An early survey on differentiation based edge detection is given in [238].

Canny edge detector [136, 241] is based on the computational theory of edge detection. The edge detection is modeled as an optimization problem with three criteria such as good detection, good localization and single-pixel response. The steps of Canny edge detection are filtering, hysteresis thresholding, edge tracking and non-maximal suppression. Edge responses are firstly obtained by filtering with gradient operators. The edges are then traced and determined by hysteresis thresholds. Only the pixels with maximal magnitude in the gradient direction can be recorded as edge points. Canny edge detector still outperforms several new detectors and is still widely applied today. An extended Canny edge detection theory (D-ISEF) has been proposed recently by McIlhagga in [242], with the objective to amend the computational theory of Canny in order to generate theoretically finite edge curves. McIlhagga finds that the optimal step edge detector, according to the Canny criteria, is the derivative of an Infinite Symmetric Exponential Filter (ISEF), proposed by Shen and Castan [243].

Second, *learning-based detection methods* require boundaries that are manually marked and provided in the training and validation set. Multiple low-level image cues are extracted and combined into the model for edge response prediction. Figure 5.1 lists the typical statistical learning based edge detection methods. Some of them are described below.



Probability of boundary (Pb) edge detection [244] use features extracted from an image patch (local information includes brightness gradient, texture gradient and color gradient) to estimate the posterior probability of a boundary passing through the center point. A boundary model based on such local information is likely to be integral to any perceptual organization algorithm that operates on natural images, whether based on grouping pixels into regions or grouping edge fragments into contours.

Multi-scale probability of boundary (MS-Pb) [245] integrates the multi-scale edge responses of Pb. It combines strengths from both large-scale detection (robust but poor localization) and small-scale detection (detail-preserving but sensitive to clutter). The localization cues represent the distance from pixels to the closest peak Pb responses at respective scales. And the relative contrast cues indicate the normalized edge responses in local regions.

Global probability of boundary (gPb) [246, 247] is a high-performance contour detector using a combination of local and global cues. It linearly combines Pb edge responses in three scales and the global information into contour detection. Multi-scale image cues are combined into an affinity matrix which defines the similarity between pixels. Spectral clustering is used to globally compute the eigenvectors of the affinity matrix which correspond to the contour information.

Texture based probability of boundary (tPb) [248] called Sledge is related to research on tracking edges for boundary detection. It uses the average of texture variations in randomly placed pairs of windows to estimate salient edge responses.

NMX [249] detects boundaries by iterating two separate steps—namely, classifying each image edge independently based on fixed edge descriptors, and grouping detected boundaries for the next iteration. In this case, edge features are fed into the AnyBoost classifier whose optimization criterion is based on the approximation of F-measure for boundary detection.

Sketch tokens [250] is an approach to both learning and detecting local contour-based representations for mid-level features. Lim et. al., called sketch tokens to features (standard shapes such as straight lines and junctions to richer structures such as curves and sets of parallel lines) that are learned using supervised mid-level information in the form of hand drawn contours in images. Patches of human generated contours are clustered to form sketch token classes and a random forest classifier is used for efficient detection in novel images.

Boosted edge learning (BEL) [251] is a supervised learning algorithm for edge and boundary detection. In the learning stage, the algorithm selects and combines a large number of features (such as histograms of DoG responses and Haar wavelets) across different scales in order to learn a discriminative model. It is based on the probability Boosting by decision trees. Generic features such as histograms of DoG responses and Haar wavelets are generated at different locations and scales. Those generic features are sequentially selected by Boosting



decision trees to construct the discriminative model for edge determination.

Structured forest based edge detection (SFE) [252, 253] is based on the learning of randomized decision trees. The inputs are color and gradient channels as well as pairwise differences. Each structure forest labels edge pixels inside the patch. The final edge responses are the aggregation of random forests.

Discriminative sparse coding (DSC) [254] is a multiscale method to minimize least-squares reconstruction errors and discriminative cost functions under  $\ell_0$  or  $\ell_1$  regularization constraints. It is applied to edge detection, category-based edge selection and image classification tasks. In general, DSC is an algorithm used to learn the sparse representation of images.

Sparse code gradient (SCG) [255] is an algorithm that uses sparse coding to automatically produce generic features. In the algorithm, an extra linear classifier is trained to combine multi-scale reconstruction errors of sparse coding and obtain the edge responses. Sparse codes serve as the input local cues and support vector machine (SVM) classification is used to learn the model to discriminate edges from non-edges.

#### 5.1.4.2 Corner detection

Corner is an important local feature in images. It is defined as the intersecting point of two connected straight edge lines [76, 256, 257]. In addition, it is a point that has high curvature and lie in the junction of different brightness regions of images [258].

In image understanding, it mathematically refers to the point at which two dominant and different gradient orientations exist. Wealthy information can be obtained at the neighborhood of corners. Compared with edges, corners are unique in local regions, which are favored for wide baseline matching. Corner detection along with multi-scale analysis is a straightforward and important way to identify interest points. Reversely, corner can also be viewed as interest point at a fixed scale.

According to Noble [259], all corner detectors have used a measure of '*cornerness*'  $C$ , defined as the product of gradient magnitude and the rate of change of gradient direction with gradient magnitude (a measure of '*cornerness*'). That is, declare a corner if the cornerness is above threshold and the pixel is an edge point.

Generally, corner detection methods can be further divided into three classes. First, classical gradient based corner detection is based on gradient calculation. Second, template based corner detection is based on the comparison of pixels. In recent years, templates are combined with machine learning techniques (i.e., decision trees) for fast corner detection. Finally, contour based detection is based on the results of contour and boundary detection. It relies on the prediction of edge responses to identify corners. Next, the most relevant methods for each category will be described.

*Gradient based corner detectors* are the most popular methods in the literature. Moravec's detector [222] is specifically interested in finding distinct regions in the image that could be used to register consecutive image frames. It has been used as a corner detection algorithm in which a corner is a point with low self-similarity. The detector tests each pixel in a given image to see if a corner is present. It considers a local image patch centered on the pixel and then determines the similarity between the patch and the nearby overlapping patches. The similarity is measured by taking the sum of square differences (SSD) between the centered patch and the other image patches.

Harris corner detector [260] is a mathematical operator that finds features in an image. It is popular because it is rotation, scale and illumination variation independent. It is an improvement of Moravec's corner detector [222] based on the convolution operator with simple linear gradient masks. The masks are used to get two directional linear gradients using convolution operator. The directional response obtained is used to construct the covariance matrix ( $M$ ). The final cornerness function is defined by:

$$\text{cornerness} = \det(M) - k \cdot \text{trace}(M)^2 \quad (5.1)$$

or using the eigenvalues  $\lambda_1, \lambda_2$  of  $M$  (2D case)

$$\text{cornerness} = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2 \quad (5.2)$$

where parameter  $k$  is chosen manually, often set up as  $k = 0.04$ .

It can be seen from the above that Harris corner detector is based on the auto-correlation of gradients on shifting windows.

There are other gradient based corner detection methods proposed in early papers, such as KLT [216] and Shi-Tomasi corner detector [261]. According to Li et. al. [76], the main difference lies in the cornerness measurement function. Since the calculation of gradients is sensitive to noise, gradient based corner detection suffers from the disadvantages of noise-sensitivity. Besides, the matrix for measurement function needs to be computed inside the window, which makes the computational complexity quite high. That is another drawback of traditional gradient based corner detection. To tackle with the high complexity, recently efforts have been devoted on the approximation of gradient based cornerness measurements.

Low-complexity corner detector (Lococo) [262] is proposed based on the classical corner detectors Harris and KLT. These algorithms are redesigned to reduce their complexity. Firstly, box kernels are used to approximate the first-order Gaussian derivative kernel. Second, gradient based integral images are borrowed to compute the cornerness responses in overlapping windows in a fast way. Finally, an efficient non-maximal suppression is proposed based on

the Quick-sort algorithm for further time saving.

Sub-pixel low-complexity corner detector (S-Lococo) [263] is a version of Lococo where the detected feature points are interpolated for sub-pixel accuracy of the feature point location.

*Template based corner detectors* find corners by comparing the intensity of surrounding pixels with that of center pixels. Templates are firstly defined and placed around the center pixels. The cornerness measurement function is devised from the relations of surrounding/centering pixel intensities.

Smallest Univalue Segment Assimilating Nucleus (SUSAN) [264] is a method introduced by Smith and Brady. It is a generic low-level image processing technique where every pixel inside the circular mask is compared with the center pixel and the intensity difference is recorded. USAN measurement is defined as the number of pixels with absolute intensity difference smaller than a threshold. Points with smallest USAN value are recorded as corners. The computational cost for template based corner detection is mainly caused by the multiple times of comparison, which is relatively lower than that of gradient based methods.

Features from Accelerated Segment Test (FAST) [265, 266] is a corner detector originally developed by Rosten and Drummond. It is based on a circular template of diameter 3.4 pixels contains 16 pixels. In this detection scheme, candidate points are detected by applying a segment test to every image pixel by considering the template around the corner candidate pixel as a base of computation. A point is considered as a corner only if there are at least  $S$  contiguous pixels in the circle which are brighter or darker than the value determined by the center pixel intensity and a threshold  $t$ . A decision tree (machine learning algorithm) is learnt to determine the order of pixels for comparison, to accelerate template based corner detection.

FAST-Enhanced Repeatability (FAST-ER) [267] is a corner detector based on FAST. It is a generalization which allows the detector to be optimized for repeatability. It increases the thickness of circular template in order to increase the stability of detected corners. In this case, the segment test detector is represented as a ternary decision tree, thus the detector can be generalized by defining a feature detector to be a ternary decision tree which detects points with high repeatability. The repeatability of such a detector is a non-convex function of the configuration of the tree, so the tree is optimized using simulated annealing.

Adaptive and Generic Accelerated Segment Test (AGAST) [268] is another kind of FAST derivations. It applies backward induction to construct an optimal decision tree for speed improvement. Besides, a collection of different decision trees are trained with different sets of specified train images. This collection makes AGAST more generic for different environments. Templates are quite important in determining corners and circular templates are chosen since they are isotropic. In addition, interpolation is needed to calculate pixel intensity in sub-pixel level for high accuracy.

*Contour based corner detectors* find corners as the intersecting points of two adjacent straight edge lines (corners based on contour/boundary detection). These methods aim to find the points with maximum curvature in the planar curves which compose of edges. Traditional kinds of contour based corners are specified in binary edge maps, which are the achieved results of edge detection and tracking. Curve smoothing and curvature estimation are two essential steps. Smoothing curves helps to reduce the noise caused by quantized point locations and Gaussian filters are the most widely used smoothing function. A recent survey on contour based corner detection with binary edge maps is presented in [269]. Contour based corners are more applied in shape analysis and shape based image compression, rather than wide baseline matching.

Difference of Gaussian - curve (DoG-curve) [270] is a technique developed for detecting and localizing corners of planar curves. DoG filters are convolved with curves points to find curvature corner points [271]. The detector uses the contour's gradient vectors. The gradient correlation matrix (GCM) formulated using Lagrange multipliers only requires calculation of the first derivative of the planar curves. The eigen-structure and determinant of the GCMs encode the geometric features of these curves, such as curvature features and the dominant points. This is advantageous as avoiding the higher order derivatives reduces the effect of noise. In addition, the determinant of the GCMs has a strong corner response, and it is used as a "cornerness" measure of planar curves. A small Region of Support (RoS) radius may then be used to improve corner localization and to prevent nearby features from merging. Consequently, the detector offers a high detection rate along with good localization performance.

Anisotropic directional derivative (ANDD) filter [272] is based on Gaussian kernels that are applied to preserve the curves while reduce curve representation noise. The ANDD representation at a pixel is a function of the oriented angle and characterizes the local directional grayscale variation around the pixel. The anisotropic Gaussian directional derivative kernels [273, 274] are used to extract fine multidirectional intensity variations of a grayscale image for corner detection and classification. A universal corner model (UCM) is used to describe simple (L-type) corners and composite or high-order corners, including Y-type, X-type, and star-like corners. The anisotropic directional derivative (ANDD) representation of the UCM is derived analytically. It is shown that the ANDD representation of a corner is expressed as the sum of several basic components and whose number indicates the type of the corner. Each ANDD filter is an anisotropic Gaussian convolution kernel followed by a directional derivative operator, which can conciliate the conflict between noise robustness and detail preservation.

Hyperbola fitting [275] is a technique based on fitting algebraic shape models to contours in the edge image. Edge responses are computed and used to fit the hyperbolic polynomial curves. Fast corner detection is accomplished by thresholding the algebraic features from the

fitted curve based shape models. The algorithm computes the edge image, then for each  $(x, y)$  point in the edge image, compute the shape model, i.e., the coefficients of the hyperbolic curve fit a local region about  $(x, y)$  that approximates the local shape of the image contours. After that, extract estimates of the corner location and the lines that best approximate the image contours. Then, compute a vector of features which are functions of the edge image data and extracted curve coefficients. Finally, apply a threshold on the feature vectors to identify the set of salient features in the image. According to the authors, the main benefit of this approach is the performance (in space and time) because no image pyramid (space) and no edge-linking (time) is required.

A contrario junction detection (ACJ) [276] is a procedure involving an automatic decision step and permitting a geometrically accurate description of junction properties, including type, localization and scale. The normalized gradients and discrete orientations are used to compute the edge strength in branches. Probabilistic principles are set up for robust junction detection. The accuracy of corner localization relies on edge detection. Along with the improvement in edge detection performance caused by multi-cue integration and machine learning, the contour based corner detection has been promoted.

### 5.1.4.3 Blob detection

Blob is defined as a region inside which the pixels are considered to be similar to each other, meanwhile be different from the surrounding neighborhoods. The definition of blobs is based on constancy of interest properties, thus blob detection is further referred as the identification of interest point (keypoint)/ interest region (keyregion). Blobs are represented by regions of regular/irregular shapes [76].

Interest points as local extremas in scale-location spaces, which furthermore denote regular circular or square regions. Interest regions are referred as segmented regions (in most cases irregular) with defined constancy. Interest point detection aims to find local extremas in pyramid spaces and interest region detection aims to identify regions with constancy by segmentation techniques. As shown in Figure 5.1, blob detection methods can be divided in interest point detection and interest region detection [76].

**Interest point detection** Interest point can provide informative representation for digital images. It refers to local extrema in 3D scale spaces with locations and scale as axes. Thus interest point can be mathematically denoted as  $(x, y, \sigma)$ . Here  $(x, y)$  indicates the location and  $\sigma$  indicates the scale. Corner can be viewed as interest point at the fixed scale. Furthermore, feature descriptors can be obtained inside square or circular regions centered at  $(x, y)$  with the size determined by  $\sigma$  [277, 278]. A variety of interest point detection methods are proposed

and the existing literature for evaluation are given in [229, 232]. Below are briefly described some of them.

Laplacian of Gaussian (LoG) detector is a method of finding the derivative or maximum rate of change in a pixel area. Commonly, the Laplacian is approximated using standard convolution kernels that add up to zero. LoG is simply the Laplacian performed over a region that has been processed using a Gaussian smoothing kernel to focus edge energy. Given an input image  $I(x, y)$ , the scale space representation of the image defined by  $L(x, y, \sigma)$  is obtained by convolving the image by a variable scale Gaussian kernel  $G(x, y, \sigma)$  where

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (5.3)$$

and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (5.4)$$

For computing the Laplacian operator, the following formula is used

$$\nabla^2 L(x, y, \sigma) = L_{xx}(x, y, \sigma) + L_{yy}(x, y, \sigma) \quad (5.5)$$

where  $L_{xx}$ ,  $L_{yy}$  are the second partial derivatives.

Difference of Gaussians (DoG) detector is an approximation of the Laplacian of Gaussians, but computed in a simpler and faster manner using the difference of two smoothed or Gaussian filtered images to detect local extrema features. The idea with Gaussian smoothing is to remove noise artifacts that are not relevant at the given scale, which would otherwise be amplified and result in false DoG features. The DoG function  $D(x, y, \sigma)$  can be computed without convolution by subtracting adjacent scale levels of a Gaussian pyramid separated by a factor  $k$ .

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \quad (5.6)$$

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (5.7)$$

Feature types extracted by DoG can be classified in the same way as for the LoG operator.

Determinant of Hessian (DoH) method, also referred to as Hessian Matrix method, is used to detect interest objects from a multi-scale image. The determinant of the Hessian matrix is at a maxima and the Hessian matrix operator is calculated using the convolution of the second order partial derivative of the Gaussian to yield a gradient maxima. The DoH method uses integral images to calculate the Gaussian partial derivatives very quickly. Performance

for calculating the Hessian Matrix is therefore very good, and accuracy is better than many methods. The Hessian matrix for the Gaussian smoothed images is

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix} \quad (5.8)$$

The scale-normalized determinant of Hessian matrix  $\sigma^4 \det(H)$  is the measurement function for interest point detection.

Harris-Laplace method [279] is a scale invariant corner detector that relies on a combination of Harris corner detector and a Gaussian scale space representation. Although Harris-corner points have been shown to be invariant to rotation and illumination changes, the points are not invariant to the scale. Therefore, the second-moment matrix utilized in that detector is modified to make it independent of the image resolution. The scale adapted second-moment matrix is used in the Harris-Laplace detector and it is represented as

$$M(x, y, \sigma_I, \sigma_D) = \sigma_D^2 g(\sigma_I) \begin{bmatrix} I_x^2(x, y, \sigma_D) & I_x I_y(x, y, \sigma_D) \\ I_x I_y(x, y, \sigma_D) & I_y^2(x, y, \sigma_D) \end{bmatrix} \quad (5.9)$$

where  $I_x$  and  $I_y$  are the image derivatives calculated in their respective direction using a Gaussian kernel of scale  $\sigma_D$ . The parameter  $\sigma_I$  determines the current scale at which the Harris corner points are detected in the Gaussian scale-space.

Hessian-Laplace method [280] also operates on local extrema, using the determinant of the Hessian at multiple scales for spatial localization, and the Laplacian at multiple scales for scale localization. Hessian-Laplacian [232, 279] combines LoG and DoH for interest point detection.

Scale Invariant Feature Transform (SIFT) [281] is the most well-known method for finding interest points and feature descriptors, providing invariance to scale, rotation, illumination, affine distortion, perspective and similarity transforms, and noise. It locates interest points with DoG pyramid and Hessian matrix. The local extremas in DoG pyramid are recorded as potential keypoints and a 3D quadratic function is to approximately locate the interpolated location of candidate keypoints. The measurement function computed with the trace and determinant of Hessian matrix is used to eliminate keypoints with strong edge responses and sub-pixel localization. Histograms of gradient orientation are the feature description. SIFT has been widely used in wide baseline matching, structure from motion, visual tracking and object recognition.

The Speeded-Up Robust Features method (SURF) [282, 283] operates in a scale space and uses a fast Hessian detector based on the determinant maxima points of the Hessian matrix.



SURF uses a scale space over a box neighborhood to localize bloblike interest point features. To find feature orientation, a set of HAAR-like feature responses are computed in the local region surrounding each interest point within a circular radius, computed at the matching pyramid scale for the interest point.

Center Surround Extrema method (CenSurE) [284] provides a true multi-scale descriptor, creating a feature vector using full spatial resolution at all scales in the pyramid, in contrast to SIFT and SURF, which find extrema at subsampled pixels that compromises accuracy at larger scales. CenSurE defines an optimized approach to find extrema by first using the Laplacian at all scales, followed by a filtering step using the Harris method to discard corners with weak responses. The major innovation of CenSurE over SIFT and SURF is the use of bilevel center-surround filters, including Difference of Boxes (DoB), Difference of Octagons (DoO) and Difference of Hexagons (DoH) filters, octagons and hexagons are more rotationally invariant than boxes.

DART [285] is a method for scale-space extrema detection that is composed of three steps: (i) an efficient computation of the approximated determinant of Hessian matrix at each scale. DART uses weighted triangle responses to approximate the second derivative of Gaussian function, which corresponds to the elements of Hessian; (ii) extrema search in scale-space; and (iii) finding the keypoint with sub-pixel and sub-scale accuracy as in [286].

Rank-SIFT method [287] applies supervised learning as RankSVM to select stable interest SIFT points. The score to measure stability is modeled and RankSVM is used to solve the ranking function for interest point detection. Rank-SIFT use a set of differential features, and based on them it adopts a data-driven approach to learn a ranking function to sort local interest points according to their stabilities across images containing the same visual objects.

Rank Order Laplacian of Gaussian (ROLG) [288] is based on weighted rank order filter and LoG. It is used to detect the image local structures where a significant majority of pixels are brighter or darker than a significant majority of pixels in their corresponding surroundings. LoG filter can be expressed as the subtraction between weighted averages. Weighted median responses are used to replace the weighted average and generate the interest point detection.

Classical interest points such as LoG, DoG, DoH are based on the partial differentiation on Gaussian scale spaces. However, in the last years, interest points detectors based on nonlinear partial differential equations (PDEs) have emerged. The most representative are KAZE, AKAZE and WADE.

KAZE method [289] is a multiscale 2D feature detection and description algorithm in nonlinear scale spaces. It finds local extremas by nonlinear diffusion filtering. The nonlinear scale space is built using efficient Additive Operator Splitting (AOS) techniques to solve the nonlinear partial differential equations for diffusion function. Based on the consideration that



Gaussian smoothing leads to image blurring, diffusion filtering provide multi-scale image spaces mean-while preserve natural image boundaries. Based on the derivation, Scharr filters are used to approximate the first and second order derivatives of diffusion function. The main drawback of KAZE feature is of high computational cost.

Accelerated-KAZE method (A-KAZE) [290] uses a recent numerical schemes called Fast Explicit Diffusion (FED) for building a nonlinear scale space considering anisotropic diffusion. To speed-up the construction of the nonlinear scale space, FED scheme is embeded into a fine to coarse pyramidal framework. The pyramidal strategy and FED schemes allow for fast nonlinear scale space construction, suitable for robust feature detection and description. Key-points are located by finding the extrema of the second-order derivatives of the image over the nonlinear multiscale pyramid built from the principle of image diffusion. A-KAZE deploys a technique similar to SURF to estimate the direction of a patch [291].

WADE [292] interest point detection framework is based on wave propagation. Wave equation is prior to highlight and isolate salient symmetries, therefore WADE is priori to detect interest point with symmetries.

Inspired by the computational speed of decision trees based corner detector, interest point detection based on binary comparison and decision tree classification has arised. In addition, with the proliferation of camera-enabled mobile devices that have limited computational resources, new features have appeared that aim to reduce computational complexity while keeping up to the performance of methods such as SIFT and SURF [290]. According to Li et. al. [76], these methods are defined as templated based methods.

Oriented FAST and Rotated BRIEF (ORB) [293] is a very fast binary descriptor based on BRIEF, which is rotation invariant and resistant to noise. It uses FAST corner detector at each scale of image pyramid. Harris cornerness measurement is borrowed to suppress non-maximal potential interest points. ORB detector is widely adopted, especially by embedded robotics systems and for real-time applications, due to greatly reduced computational requirements [291].

Binary Robust Invariant Scalable Keypoints (BRISK) [294] is a local binary method using a circular-symmetric pattern region shape and a total of 60 point-pairs as line segments arranged in four concentric rings. The method uses point-pairs of both short segments and long segments, and this provides a measure of scale invariance, since short segments may map better for fine resolution and long segments may map better at coarse resolution. The BRISK algorithm uses a novel FAST detector adapted to use scale space, reportedly achieving an order of magnitude performance increase over SURF with comparable accuracy [295].

Fast Retina Keypoint (FREAK) [296] uses a foveal-inspired multiresolution pixel pair sampling shape with trained pixel pairs to mimic the design of the human eye as a coarse-to-

fine descriptor, with resolution highest in the center and decreasing further into the periphery [295]. FREAK uses the same interest point detector as BRISK, with binary descriptors motivated from human retina. Binary features are quite favored from the engineering point of view, since they are less time consuming and storage saving.

**Interest region detection** Image is represented based on its local structures by a set of local feature descriptors extracted from a set of image regions called interest regions. They refer to the regions segmented from neighboring area by exploiting the constancy of image properties. The definition of pixel constancy can be on pixel intensity, zero gradient, etc. Regions which remain stable along a large threshold range are chosen as interest regions. Ellipses or parallelograms are used to fit the segmented interest regions. The mathematical representation of interest regions depends on the parameters of fitted ellipses or parallelograms. Different from interest point detection, interest region detection often do not need extra multiscale pyramid construction. Below are briefly described some methods of interest region detection.

Maximally Stable Extremal Region (MSER) [297] is essentially an efficient variant of the watershed algorithm, except that the goal of MSER is to find a range of thresholds that leave the watershed basin unchanged in size. MSER obtains interest regions, which are connected components based on thresholding the pixel intensities. The word “extremal” refers to the property that all pixels inside the MSER have either higher (bright extremal regions) or lower (dark extremal regions) intensity than all the pixels on its outer boundary. The “maximally stable” in MSER describes the property optimized in the threshold selection process [218, 295].

Intensity Extrema-Based Region (IBR) [298] starts from points of local intensity extrema and detects region boundaries by tracing lines out from these points and finding extrema of a function of intensity differences along the lines [229]. The intensity profiles along rays emanating from a local extremum are investigated. A marker is placed on each ray in the place, where the intensity profile significantly changes. Finally, an ellipse is fitted to the region determined by the markers.

Salient region [299] is based on probability distribution of pixel intensities. Essentially, Salient Regions are regions that locally assume maximal signal complexity and at the same time exhibit self-dissimilarity in scale space [300]. The signal complexity is measured by the Shannon entropy (denoted by  $H$ ) of the local intensity histogram. The self-dissimilarity is approximated by the change of the probability density function (pdf) in scale space (denoted by  $W$ ). A region’s scale saliency  $Y$  is defined as the product of the two factors  $H$  and  $W$ , all of which are functions of scale  $s$  and position  $\mathbf{x}$  [300].

Principal Curvature Based Region (PCBR) [215] detects stable watershed regions within

the multi-scale principal curvature image. It is based on MSER operating in watershed regions of principal curvature images. The principal curvature image is extracted from eigenvalues of Hessian matrices. To detect robust watershed regions, enhanced watershed segmentation is borrowed and used in cleaned binary principal curvature images. Cleaned principal curvature image is obtained combining a grayscale morphological close with “eigenvector flow” hysteresis threshold. Robustness across scales is achieved by selecting the maximally stable regions across consecutive scales. PCBR typically detects distinctive patterns distributed evenly on the objects and it shows significant robustness to local intensity perturbations and intra-class variations. In general, the PCBR detector employs the first steps of Steger’s curvilinear detector algorithm [301]. It forms an image of the maximum or minimum eigenvalue of the Hessian matrix at each pixel. This is called the principal curvature image, as it measures the principal curvature of the image intensity surface. This process generates a single response for both lines and edges, producing a clearer structural sketch of an image than is usually provided by the gradient magnitude image.

Beta-Stable Features ( $\beta$ -stable features) [302] are based on a Laplacian scale-space description of the image. The concept of  $\beta$ -stability is a variation on the theme of a feature’s lifetime, and it is built on the notion of convexity: rather than selecting features that persist over a wide interval of scales. Features are computed a scale chosen so that the number of convex and concave regions of the image brightness function remains constant within a scale interval of length  $\beta$ . This shift in selection criterion leads to robustness to high-frequency perturbations of the image, in addition to the invariance advantages deriving from the use of the Laplacian.

Maximally Stable Colour Regions (MSCR) [303] is an extension of MSER with color information. The colorful distance is derived by the Poisson statistics of image pixels. Agglomerative clustering is applied to successively grouping neighboring pixels with similar colors. The selection of time-steps is stabilised against intensity scalings and image blur by modelling the distribution of edge magnitudes.

Medial Feature Detector (MFD) [304] is a boundary based framework. It is able to detect regions of arbitrary scale and shape, without scale space construction. Firstly, a weighted distance map is generated with image gradients. After that, a global weighted medial axis is computed in the weighted distance map. Regions are segmented by medial axis decomposition. Finally, a shape fragmentation factor is defined to select the interest regions.

Boundaries of image segments tend to align with (a portion of) an object’s boundaries and help reveal its shape. Boundary information becomes more integrated into interest region detection recently. However, leveraging segments for local features is not straightforward: despite their boundary-preserving nature, bottom-up segmentation algorithms are typically

sensitive to global image variations, such that images with similar content can produce a dramatically different set of image segments [305]. According to Li et. al. [76] the most important algorithms to detect boundaries are MFD, FLOG and BPLR.

Fan Laplacian of Gaussian (FLOG) [306] is used for automatic scale selection and final interest region localization. Cui et. al. propose a unified framework to extract and match both the edge junctions and the salient points along the edges for general structured scenes. The keypoints are selected from the edges that are efficiently and carefully detected to favor accurate surface boundaries. The repeatability of keypoints is guaranteed by a multiscale selection scheme. The point neighborhood is divided into multiple fan-shaped subregions, namely Fan features, by a method of edge association which does not rely on the continuity and completeness of edges. To achieve scale invariance for each Fan feature, a Fan Laplacian of Gaussian (FLOG) filter to select its characteristic scales is proposed. In particular, scale and affine invariant fan feature is based on the extracted edge fragments. It firstly applies Harris measurement to selecting salient edge points as candidate interest points along boundaries. Edge fragments are associated with candidate interest points to determine the shape of subregions.

Boundary Preserving Local Region (BPLR) [305] is based on the learning based Pb edge detector. The distance transform (DT) is computed for each boundary segment. Candidate circular regions are generated by densely sampled with maximal DT values. The minimum spanning tree algorithm is used to group the neighboring circular regions and generate the densely placed boundary preserving local regions. Compared with interest point detection, more parameters like rotation angles, aspect ratio are obtained by interest region detection. Feature descriptors extracted with segmented regions can be normalized with more geometrical parameters therefore are mostly affine-invariant.

### 5.1.5 Shape-based features

Most image analysis methods require, at some point, the detection of various structures in images. These structures may be complex, like objects or passerby in a scene, or simpler, like round cells in microscopy images. In any case, suitable prior knowledge is required to characterize the structure of interest, normally, it permits to describe the shape, color or pose of the structure.

The first characterization of structures that was found corresponds to the proposal of Shapiro et. al. [4], who identified three kinds of three-dimensional parts: sticks, plates and blobs. Sticks are long, thin parts having only one significant dimension. Plates are flatish wide parts consisting of two nearly flat surfaces connected by a thin edge between them. Plates have two significant dimensions. Blobs are neither thin nor flat. They have three significant dimensions. All three kinds of parts are “near convex”. That is a stick can not bend very much, the

surfaces of a plate can not fold too much, and a blob can be bumpy, but can not have large concavities. Figure 5.2 shows examples of sticks, plates, and blobs shapes.

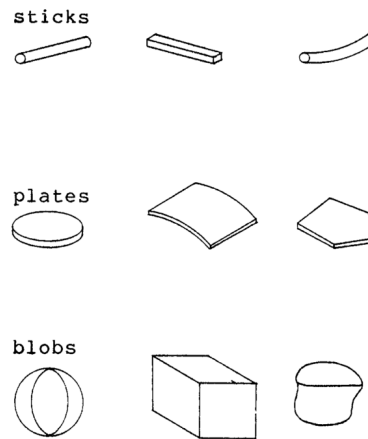


Figure 5.2: Examples of sticks, plates, and blobs shapes (taken from [4]).

In addition, Shapiro et. al. indicated that to describe an object it is important to have into account a list the parts, their types (stick, plate, or blob), their relative sizes, and how the parts fit together. The type of connection can be end-end, end-interior, end-center, end-edge, interior-center, or center-center where "end" refers to an end of a stick, "interior" refers to the interior of a stick or surface of a plate or blob, "edge" refers to the edge of a plate, and "center" refers to the center of mass of any part.

A most recent description of types of three-dimensional parts is presented by Merveille [5], who included the relation between the dimensions. Figure 5.3 shows the different type of structures in the 2D and 3D space, according to their sizes in the different dimensions. The reader should keep in mind that these illustrations are simplified drawings based on the geometric properties of each structure. In real applications, curvilinear structures and plane-like structure are usually not straight.

In general, the structures can be classified as blob-like, line-like (curvilinear, vessel-like, tube-like), and plate-like (sheet-like, plane-like) structures.

Based on the eigenvalues of the Hessian matrix, a local pattern is classified as plate-like, line-like or blob-like structures. The method was originated by Koller et al. [105] and was further developed by Lorenz et al. [307] and Frangi et al. [107] for vessel enhancement purpose. Sato et al. [73, 308] also developed a line enhancement filter based on the eigenvalues of the Hessian matrix, and generalized their filter for blob enhancement. One known problem of this kind of method is too much blurring can occur during the multiscale smoothing lead to false detections, especially for close-by structures.

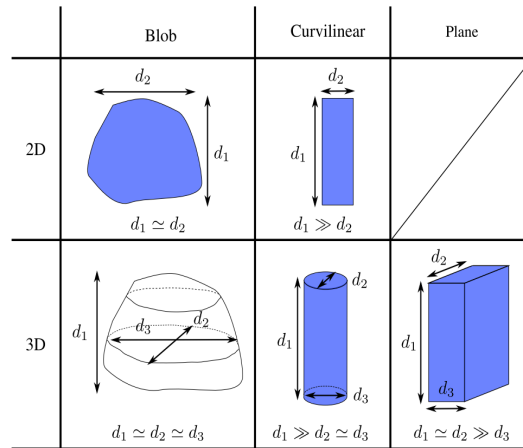


Figure 5.3: Examples of blobs, planes, and line (curvilinear) structures in 2D and 3D space (taken from [5]).

According to Sato et. al. [308], Frangi et. al. [107], Danielsson et. al. [309], and others, after deriving the eigenvalues, their relative signs and magnitudes are exploited to distinguish between sheet-like, line-like or blob-like structures. Relations between eigenvalues and local structures are:

Blob-like :  $\lambda_1 \approx \lambda_2 \approx \lambda_3$

Line-like:  $\lambda_1 \gg \lambda_2 \approx \lambda_3$

Sheet-like :  $\lambda_1 \approx \lambda_2 \gg \lambda_3$

The orientation of the eigenvectors can be used as a robust estimate of the local orientation.

### 5.1.5.1 Blob-like detection

According to Lindeberg [310], blobs are bright regions on dark backgrounds or vice versa. In addition, blobs are parts that have all three significant dimensions.

Automated detection of blob-like structures is desirable in many applications. Examples include, lung nodule detection in chest radiographs and thoracic CT scans [311, 312], lymph node detection in chest/abdominal CT images, and cell counting or tracking in biological images [313, 314]. The localization and distribution of these blob-like structures are required for further analysis in these applications [315, 316].

As shown in Section 5.1.4.3, there are many blob-like detectors defined for different purposes, most of which are compared in [76, 232, 279, 317]. Scale normalized Laplacian (SNL) based on the Laplacian of Gaussian (LOG) [318], Salient regions based on entropy [299], shift invariant feature transform (SIFT) based on the difference of Gaussians (DOG) [281], and speeded-up robust features (SURF) based on the determinant of Hessian (DOH) [283]

have attracted the most attention in the literature.

### 5.1.5.2 Sheet-like detection

3-D sheet model consists of the medial surfaces of sheets and the width associated with each point on these surfaces (see Figures 5.2 and 5.3). Examples of sheet-like structures in medical images are airway walls, temporal muscle, knee muscle, articular cartilage, thin bones such as sternum, ribs, most of the skull bones, and the vertebral cortical shell [106, 319–321].

Thickness measurement of sheet-like or plate-like thin anatomical structures is an important procedure in clinical practice. Accurate thickness measurement of sheet-like structures has become increasingly important in clinical applications as well as in fundamental research. Several methods for thickness quantification have been proposed [322–325].

Westin et al. [322] suggested using the structure tensor and adaptive thresholding to segment thin structures. Given  $\lambda_1, \lambda_2, \lambda_3$  ( $0 \leq \lambda_1 \leq \lambda_2 \leq \lambda_3$ ) as the eigenvalues of the structure tensor, the planar measure was defined as

$$C_{plane} = \frac{\lambda_3 - \alpha\lambda_2}{\lambda_3} \quad (5.10)$$

and used in adaptive thresholding. In theory, it has a value of 1 for plane structures and 0 for others. A challenge is that adaptive thresholding is usually sensitive to noise and leakages, and is not well-suited to inhomogeneous intensity [319, 323].

Descoteaux et al. [323] introduced a multi-scale sheetness measure that was inspired by the Frangi filter [107]. Their measure combined three ratios  $R_{sheet}$ ,  $R_{blob}$ ,  $R_{noise}$  to enhance sheet-like structures and eliminate noise and blobs. The multi-scale sheetness measure is used as the energy function in the level set method. Being a variational approach, their method is sensitive to initialization, and consequently the segmentation result could be trapped in local minima.

Krcak et al. [325] proposed to use the sheetness measure [323] as the boundary term in the graph cut method for the segmentation of the femur bone. Using only this measure may produce oversegmentation due to the strong response to the sheetness measure at step edges.

Reinbacher et al. [324] presented an interactive method based on total variation to segment thin volumetric structures. Edge direction and volume constraint were introduced to the convex functional which can be efficiently solved to obtain the global optimal segmentation result. However, the interactive segmentation of complex 3D structures, which is usually spread over a large volume, may require extensive user interaction.



### 5.1.5.3 Vessel-like detection

Another useful feature is called a line-like, vessel-like or curvilinear structure. This represents a line or a curve with some width, and it differs from conventional line or curve features, which are usually extracted based on edges. Curvilinear structures are more structured features and contain more information than edges. They can be found in most natural images, and their detection is particularly useful, for example, when trying to find roads or rivers in aerial images, blood vessels or bones in medical images, and characters in text images [208].

Humans have a well developed ability to detect curvilinear structures in noisy images. Good algorithms for performing this process would be very useful in machine vision for image segmentation and object recognition [326]. These algorithms typically require dedicated processing techniques. One is the ability to enhance, track, and detect low-contrast curvilinear features in a variety of image analysis applications [327, 328].

According to [329], [330], and [331], in life sciences and health care, a number of image processing solutions have been proposed to extract curvilinear structures, mostly within the context of blood vessel detection [107, 332]. Other solutions proposed including airways in lung images [43], neurons [333, 334], dendritic spines [335, 336], microtubules [337, 338], milk ducts, fibrous tissue [339], spiculations [340, 341], and many other similar entities [342].

There are several papers that present a review of the main algorithms to detect curvilinear structures, mainly focused on the detection of tubular structures. In this case, Steger [301], Jang et. al. [209], Kirbas et. al. [70, 343] and Lesage et al. [68] will be considered as basic elements to present a brief classification of the existing methods to detect curvilinear structures.

The first one is presented by Steger [301], who groups the different approaches to extract curvilinear structures from 2D and 3D images in four categories.

- Gray-value based approaches only use the gray values of the original images or some sort of local differences to extract line points [344, 345].
- Differential geometric approaches regard the image as a surface in 3D or 4D space, and try to extract lines as ridges by using various differential geometric features [301].
- Special filters enhance linear structures, so that lines can easily be obtained by thresholding.
- Some approaches use an explicit line model to detect lines and their width.

Second, according to Jang et. al. [209], methods are based on one of following three approaches:



- Locally parallel edge based approach [105, 346, 347].
- Ridge based differential geometric approach [10, 301, 318, 348–350].
- Active contour model based approach [351, 352].

Third, Kirbas et. al. [70, 343] presented an extensive vessel segmentation review classifying the various methods according to which approach they belong to: pattern recognition, model-based, tracking, artificial intelligence-based, neural network-based, and miscellaneous tube-like object detection approaches. In the vessel extraction domain, pattern recognition techniques are concerned with the detection of vessel structures and the vessel features automatically. Kirbas and Quek divide pattern recognition techniques into seven categories (see references inside [70]): multiscale approaches, skeleton-based (centerline detection) approaches, region growing approaches, ridge-based approaches, differential Geometry-based approaches, matching filters approaches, and mathematical morphology schemes. Model-based approaches are divided into deformable models (parametric deformable models-active contours (Snakes), geometric deformables models and deformable template matching approaches), parametric models and generalized cylinders approaches.

Finally, a more exhaustive review can be found in Lesage et al. [68], who classified various vessel lumen segmentation methods in contrast-enhanced imaging modalities (such as magnetic resonance angiography and computed tomography angiography), analyzing the different models, features and extraction schemes. In addition, recent approaches are described in Rudyanto et. al. [45], Lo et. al. [43, 87] and Novoa et. al. [353].

A brief summary of models and features is provided of related vessel/airway segmentation techniques following closely the classification and work of Lesage et. al. [68].

**Models** Models correspond to the prior assumptions made on the target vessels, e.g. elongation and hyper-intensity. Figure 5.4 shows the classification of vessel models. The main difference is between appearance (photometric) and geometric information.

Methods utilizing pure photometric models use prior knowledge about the expected luminance of the vessels [354–356], and/or the background [355, 357–360] as well as the image noise [357, 361] to distinguish between vessel structures and non-vessel structures.

Methods utilizing geometric models have into account the shape of the vessel. A key characteristic is the elongation. Depending on the application, additional knowledge (specific vessel radius ranges and bifurcations for instance) might be incorporated. Therefore, the methods may encode information about their elongation, properties of their centerline [362–364], and/or their cross-section [108, 362, 365–367].

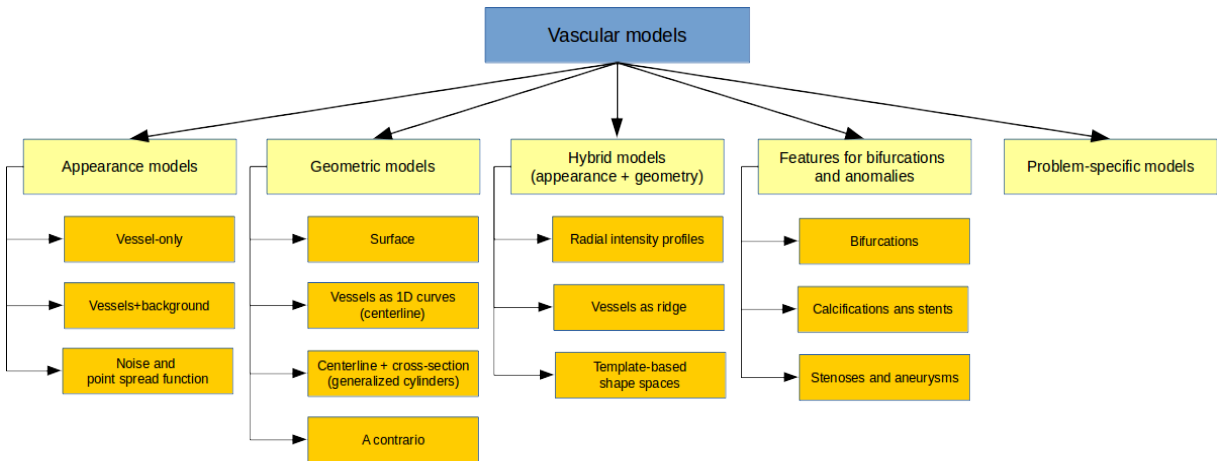


Figure 5.4: Vascular models classification.

Photometric and geometric information are often combined into hybrid models, incorporating assumptions about the spatial appearance of vessels such as Gaussian- or bar-like cross section profiles [105, 301, 368, 369], appearance as a ridge in scale space [[370? , 371], or template-based approaches [[107, 372, 373]. The models typically focus solely on regular vessel segments, while models about bifurcations or anomalies can only be found rarely, although exceptions – typically adapted to specific kinds of disease – can be found [354, 362, 373–375].

**Features** Features are the vessel- dedicated image measures used to estimate the models on the image, e.g. local intensity curvatures. In general, features are the detectors/filters used to evaluate a vascular model on the image data. Figure 5.5 shows a vascular features classification.

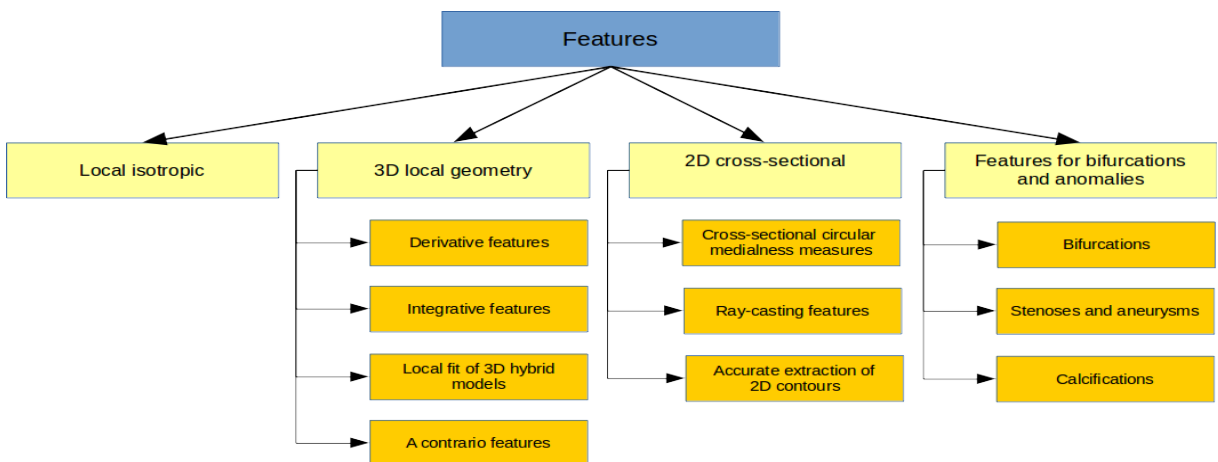


Figure 5.5: Vascular features classification.

Methods using information about the basic image intensity, first order, and/or second order image derivatives to match against the expected model, often embedded in a Gaussian scale-space framework [[71, 376] to account for tubular structures of varying size. Features used in the literature are isotropic features where no assumption about the directionality of the vessel is made [212, 377–379], features making use of the local geometry typically based on derivative features [[107, 214, 307, 354, 373, 380], or model fitting [[361, 364, 372, 381] with few other techniques.

Another approach is utilizing features based on 2D cross-sectional measurements that require the orientation as an input parameter [[105, 355, 375, 382? –384]. Similarly as with the different models, bifurcations or anomalies are often not considered explicitly. However, exceptions – typically adapted to specific diseases – can be found in [354, 371, 373].

## 5.2 Proposed method

The proposed method to detect the characteristics of the upper airways is based mainly on the work done by Haralick et. al. [10], Monga et. al. [104, 385, 386], Thirion et. al. [387, 388], Koenderink [142, 389], Steger [301, 390], and Deng et. al. [215].

Haralick et. al. [10] proposed the definition of the types of structures obtained when using the curvatures. The distinction can be made as follows:

Slope if  $\lambda_1 = \lambda_2 = 0$

Convex if  $\lambda_1 \geq \lambda_2 \geq 0, \lambda_1 \neq 0$

Concave if  $\lambda_1 \leq \lambda_2 \leq 0, \lambda_1 \neq 0$

Saddle hill if  $\lambda_1 * \lambda_2 < 0$

The complete table is shown in Table 2.2, where is presented the topographic classes according to gradient magnitude, eigenvalues and first directional derivatives taken in the directions which extremize second directional derivatives.

Monga et. al. [104, 385, 386] proposed a technique to extract crest lines in a 3D image. This technique have five steps:

- Computation of the first, second and third order partial derivatives of the image  $I(x, y, z)$
- Extraction of the 3D edge points using the gradient.
- For each point of the 3D edge map, computation of:
  - the two principal curvatures and the corresponding principal curvature directions.
  - the extremality criterion (derivative of the maximum curvature along the corresponding principal direction).

- Building of an extremality criterion image  $C(x, y, z)$ .
- Determination of an image  $Z(x, y, z)$  set using the extremality criterion (to 1 at each edge point and to 0 otherwise).

Thirion et. al. [387, 388] indicated that a pre-filtering is required to smooth the image because third order differentials are very sensitive to noise. In addition, the principle of the crest lines characterization is to compute a criterion called extremality, whose zero-crossings exactly correspond to the crest lines position on the iso-surface. Furthermore, they have designed a new algorithm, called Marching Lines, which can extract very precisely (with sub-voxel precision) the position of the crest lines. Finally, they define based on Monga and Benayoun a crest line as a locus of points whose maximal curvature (i.e. the maximum absolute value of the two principal curvature), is a local maximum in the corresponding principal direction. Thirion and Gourdon [391] indicated that depending on what principal curvature is used (largest or second), and if it is a local maximum or minimum, four different kinds of extremal lines can be found [392]:

- lines of maximum largest principal curvature (these are called crest lines).
- lines of minimum largest principal curvature.
- lines of maximum smallest principal curvature.
- lines of minimum smallest principal curvature.

Koenderink [142, 389, 393] define the *curvedness* as the distance from the origin in the  $(\lambda_1, \lambda_2)$ -plane. They have proposed:

$$c = \sqrt{\frac{\lambda_1 + \lambda_2}{2}} \quad (5.11)$$

positive values of  $c$  for describing the magnitude of the curvedness at a point. It is a measure of how highly or gently curved a point is. At a point that has no curvedness the value becomes zero. Therefore, this variable can be used to recognise a plane surface. In addition, Koenderink et. al. [389], indicated that a positive curvature indicates a concave shape and a negative curvature indicates a convex shape.

Steger [301, 390] indicated that the direction of the line is determined from the Hessian matrix. Line points are then found by selecting pixels that have a high second directional derivative perpendicular to the line direction. The advantage of this approach is that lines can be detected with sub-pixel accuracy without having to construct specialized directional filters. Furthermore, the approach only uses the first and second directional derivatives of an image for the extraction of the line points. This makes the approach computationally very efficient.

Deng et. al. [215] proposed a PCBR algorithm, which detects stable watershed regions within the multi-scale principal curvature image. To detect robust watershed regions, they “clean” a principal curvature image by combining a grayscale morphological close with a new “eigenvector flow” hysteresis threshold. The PCBR detector employs the first steps of Steger’s curvilinear detector algorithm. It forms an image of the maximum or minimum eigenvalue of the Hessian matrix at each pixel. They call this the principal curvature image, as it measures the principal curvature of the image intensity surface. This process generates a single response for both lines and edges, producing a clearer structural sketch of an image than is usually provided by the gradient magnitude image. In the examples presented by Deng et. al. [215], the curvatures are obtained using the surface generated by 2D images.

In this case, the proposed algorithm uses as input the image containing the gradient magnitude information. This technique to detect edges have two steps:

- Computation of the principal curvatures (eigenvalues) and the corresponding principal curvature directions (eigenvectors) using the Hessian matrix for each point of the edge map.
- A criterion is used to discriminate the curvature that detects the ridge lines.

In the first step, the Hessian matrix is defined as

$$H(x, y) = \begin{bmatrix} I_{xx}(x, y) & I_{xy}(x, y) \\ I_{xy}(x, y) & I_{yy}(x, y) \end{bmatrix} \quad (5.12)$$

where  $I$  is the gradient magnitude image.

In the second step, the criterion used to discriminate the curvature is based in Deng et. al. [215] where positive curvatures are considered as high response only for dark lines on a light background (Equation 5.13) and negative curvatures to detect light lines in a darker background (Equation 5.14), the principal curvature in two-dimensional images is given by either

$$P(x, y) = \max(\lambda_1(x, y), 0) \quad (5.13)$$

or

$$P(x, y) = \min(\lambda_2(x, y), 0) \quad (5.14)$$

where  $\lambda_1(x, y)$  and  $\lambda_2(x, y)$  are the maximum and minimum eigenvalues, respectively, of Hessian matrix  $H$  at  $(x, y)$ . Deng et. al. [215] do not specify which is the ordering criterion of

the eigenvalues used in their PCBR algorithm. Tests performed seem to indicate that it is an ordering by value.

Eberly et. al. [394] indicated that if the eigenvalues of the Hessian are sorted according to their magnitude, a  $d$ -dimensional ridge or ravine must have the first  $d$  eigenvalues of the same sign and must have a local maximum or minimum in the direction given by the  $d$  corresponding eigenvectors.

Considering the above elements and using  $|\lambda_1| \geq |\lambda_2|$ , equations 5.15 and 5.16 are proposed to use as a criterion to discriminate the features based on the curvature.

$$P(x, y) = \min(\lambda_1(x, y), 0) \quad (5.15)$$

or

$$P(x, y) = \max(\lambda_1(x, y), 0) \quad (5.16)$$

The absolute values in Equation 5.15 correspond to the ridges when light lines are detected and the values of Equation 5.16 permit to detect valleys corresponding to dark lines.

## 5.3 Experimental results

The experimentation process was divided into three parts. In the first part, tests are performed on 2D images to determine which is the best ordering criterion of the eigenvalues of the Hessian matrix. In the second part, the selected criterion is applied to synthetic 3D images. Finally, in the third part, the selection criterion for 3D medical images is applied.

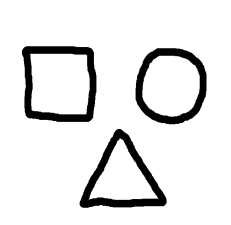
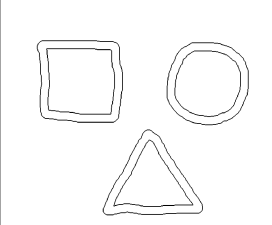
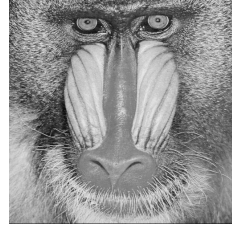
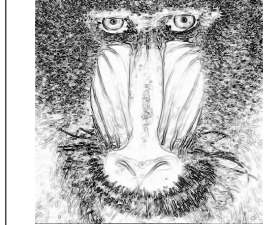



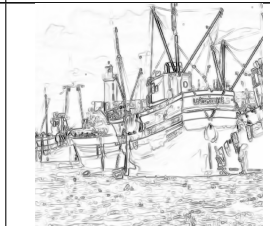


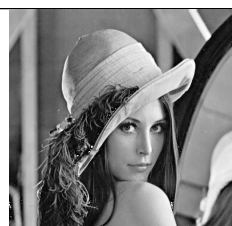

### 5.3.1 Ordering criterion of Hessian eigenvalues

To determining the ordering criterion of the Hessian matrix eigenvalues and hence the criterion to discriminate curvatures, six test images were selected as shown in Table 5.1. The gradient and its magnitude were computed using the algorithm proposed by Savitzky-Golay, since it presents a better response than the traditional techniques [395] (Gaussian filters and Gaussian recursive filters). The polynomial order is of degree 2 and the window size is of radius 1.

Table 5.1: Test 2D images. First and third columns - original images, second and fourth columns - gradient magnitude.

Image	Gradient magnitude	Image	Gradient magnitude
-------	--------------------	-------	--------------------

Table 5.1: Test 2D images. First and third columns - original images, second and fourth columns - gradient magnitude.

Image	Gradient magnitude	Image	Gradient magnitude
			
			
			

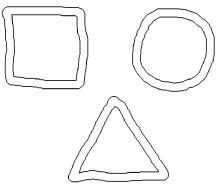
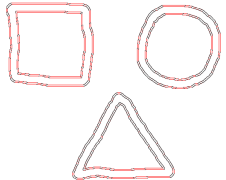
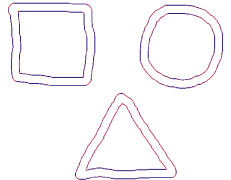
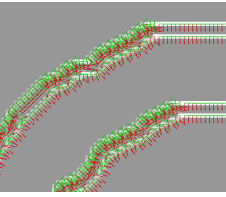
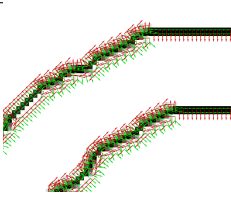



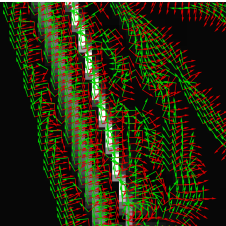
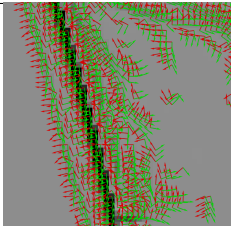
Using  $\lambda_1 \geq \lambda_2$  as a criterion for ordering the eigenvalues, tests were performed for each of the images. The results obtained for the geometric figures and lena images are shown in table 5.2. The gradient magnitude is found in the first column. In the second column, the behavior of the maximum value defined in Equation 5.13 can be observed. In red color the highest values of the curvatures are presented and in black the smaller ones. In addition, the zoom of the curvatures with their direction vectors is shown. In the third column, the behavior of the curvatures using equation 5.14 is presented. The highest values of the curvatures are shown in red color while the lowest values are in black.

Table 5.2: Curvatures using  $\lambda_1 \geq \lambda_2$  as ordering criterion.

$\nabla I$	$\lambda_1, [v_1, v_2]$	$\lambda_2, [v_2, v_1]$
------------	-------------------------	-------------------------



Table 5.2: Curvatures using  $\lambda_1 \geq \lambda_2$  as ordering criterion.

$\nabla I$	$\lambda_1, [v_1, v_2]$	$\lambda_2, [v_2, v_1]$
		
		
		
		

Based on the results presented in Table 5.2 (see zoom images), it can be concluded that when the ordering criterion is  $\lambda_1 \geq \lambda_2$ , the curvatures obtained using Equation 5.14 show a better behavior.

Now using the ordering criterion  $|\lambda_1| \geq |\lambda_2|$  and the equations 5.13 and 5.14 proposed by Deng et. al. [215] the curvatures are selected. The results obtained are shown in Table 5.3.

Table 5.3: Curvatures using  $|\lambda_1| \geq |\lambda_2|$  as ordering criterion.

$\nabla I$	$\lambda_1, [v_1, v_2]$	$\lambda_2, [v_2, v_1]$
------------	-------------------------	-------------------------



Table 5.3: Curvatures using  $|\lambda_1| \geq |\lambda_2|$  as ordering criterion.

$\nabla I$	$\lambda_1, [v_1, v_2]$	$\lambda_2, [v_2, v_1]$
		
		
		
		

The second column of Table 5.3 shows the results using the equation 5.13. The red features correspond to the higher values of curvatures while the black ones are the lowest. The curvatures present good location, however compared with the previous results generate a greater amount of information around the edges. The third column presents the results using Equation 5.14. As can be seen, an incomplete edge detection is generated. It also detects a double border around the gradients of the original image.

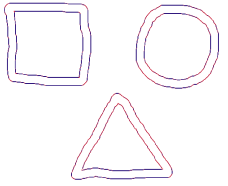
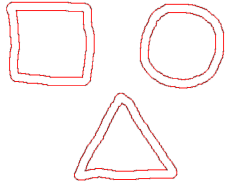
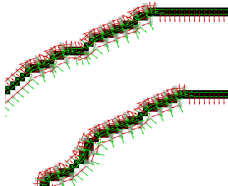
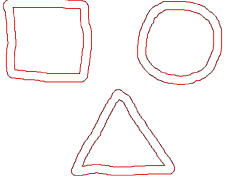
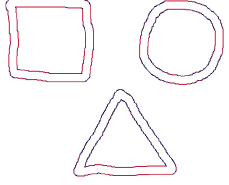
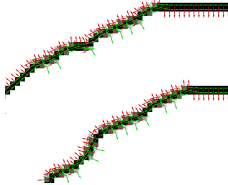


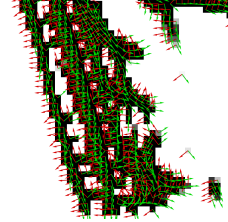


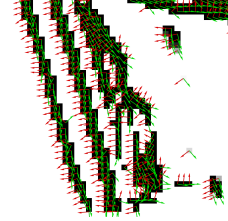
In conclusion, using the equations proposed by Deng et. al. [215] give better results when the ordering criterion of the eigenvalues is  $\lambda_1 \geq \lambda_2$ . Furthermore, Equation 5.14 presents a better behavior to detect the curvatures.

On the other hand, the proposed criterion to detects features based on principal curvatures must be evaluated. To meet this aim, tests with defined images are performed. The obtained

results are compared with those generated using the Deng et. al. equation.

Table 5.4 shows the results obtained. The first and third row present the results obtained using the criterion defined by Deng et. al.[215]. The second and fourth row present the results generated by the proposed criterion. The first column of Table 5.4 shows the images that include all eigenvalues ( $\lambda_1$  or  $\lambda_2$  according to the case). The second column corresponds to the images of the curvatures where eigenvalues ( $\lambda_1$  or  $\lambda_2$  according to the case) are less than zero. The third column shows a zoom of the curvatures and directions. The fourth column corresponds to the number of pixels selected when the discrimination criterion is used.

Table 5.4: Ordering and discrimination criteria.

	$\lambda_i$	$\lambda_i < 0$	Zoom in	N. points
$\lambda_2, \lambda_1 \geq \lambda_2$				11206
$\lambda_1,  \lambda_1  \geq  \lambda_2 $				7805
$\lambda_2, \lambda_1 \geq \lambda_2$				114383
$\lambda_1,  \lambda_1  \geq  \lambda_2 $				74329


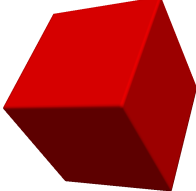
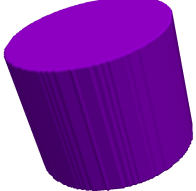
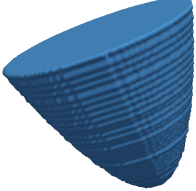
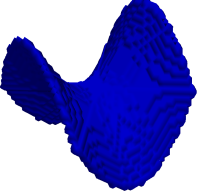
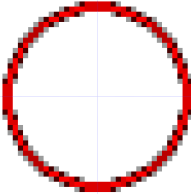
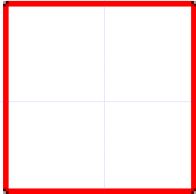
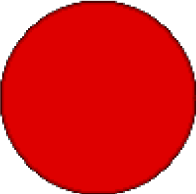
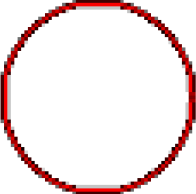
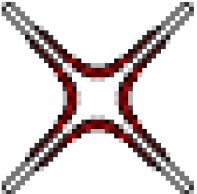
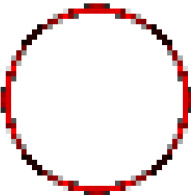
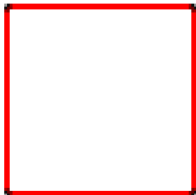
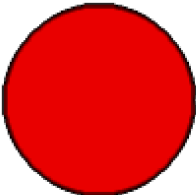
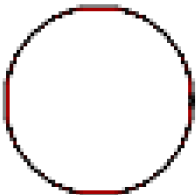
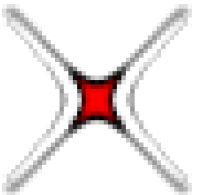
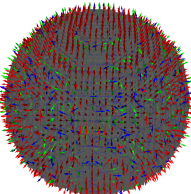
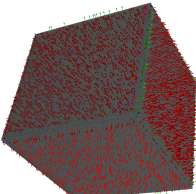
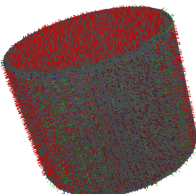
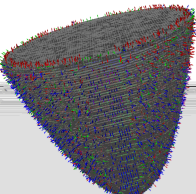
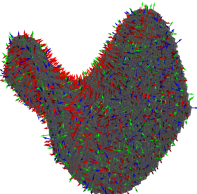
As can be seen in the fourth column, the number of pixels detected is lower when the proposed criterion is used. This is related to the precision in the pixels location belonging to the ridges, as shown in the third column.

In conclusion, the proposed criterion to detect curvatures presents a better behavior than the criterion proposed by Deng et. al. [215]. Curvatures corresponding to the crests of the gradient magnitude are better located with the proposed criterion.

### 5.3.2 3D synthetic images

For this second part, a test set of five 3D images representing different types of geometric objects were constructed. Based on the results obtained in the first part, the ordering criterion and the feature detection criterion (discrimination criterion) to each image is applied. The results obtained are presented in Table 5.5.

Table 5.5: 3D synthetic images curvatures.

Ellipsoid	Cube	Cylinder	Elliptic	Hyperbolic
				
				
				
				

It can be observed that the detected curvatures have a good location. The curvature is

affected in the contours that have corners, as can be seen in the image of cube, ellipsoid, cylinder and elliptic. The value of the curvatures is reduced in areas with corners, so in these images the curvature looks black or gray, while in areas with high curvature it looks red.

### 5.3.3 3D medical images

In this part, tests are performed with 3D medical images. Ten images were selected for testing. Each test includes the computation of gradient magnitude and curvatures using the Hessian matrix. For the calculation of both the gradient and the curvatures, the Savitzky-Golay algorithm was used with a polynomial of degree 2 and a radius mask 1. Then, the proposed criterion was used to detect the curvatures.

Table 5.6 presents the results obtained for five images (slice 110). The first column corresponds to the original image. The second column shows the gradient magnitude map. Finally, the third column presents the curvatures obtained when applying the proposed criterion.

Table 5.6: Gradients and curvatures of 3D medical images.

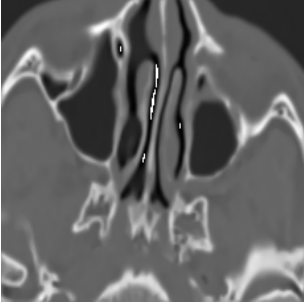
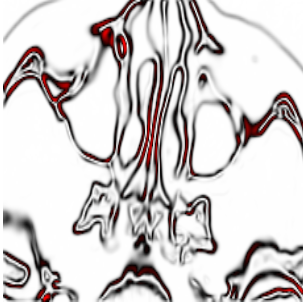

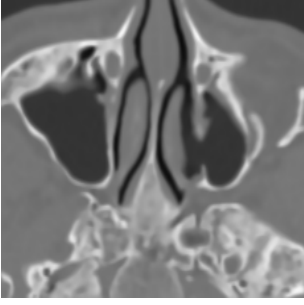
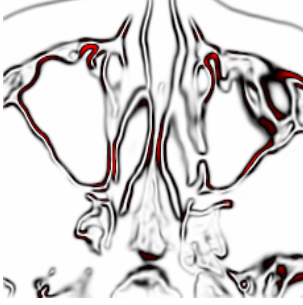
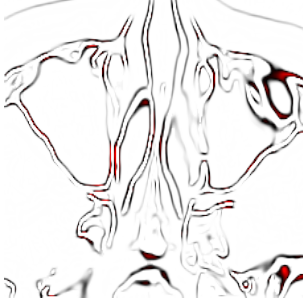
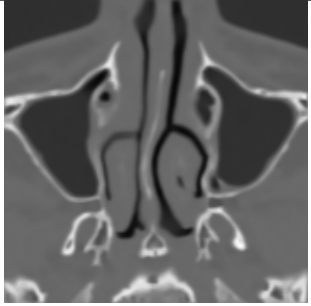
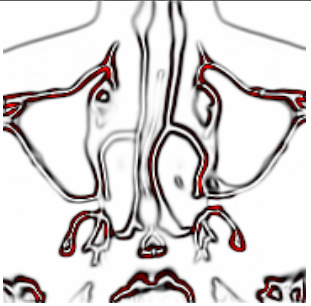

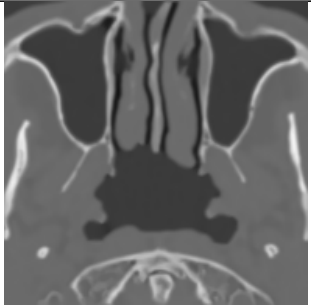
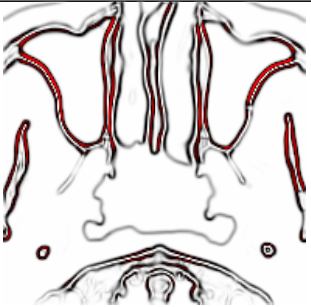
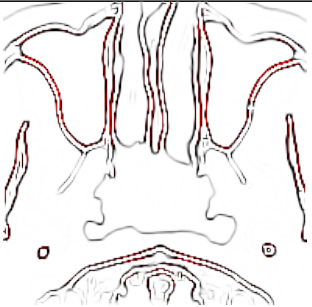
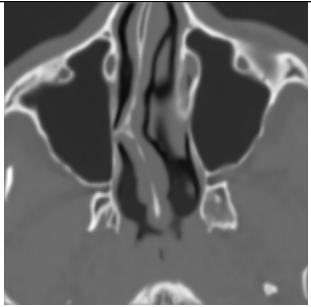
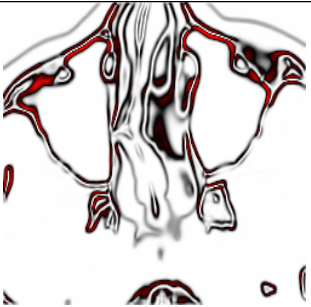
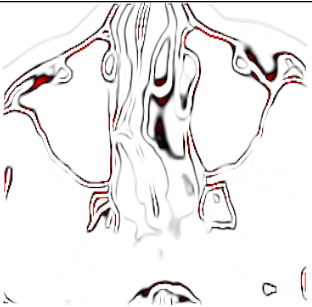
	original (slice 110)	$\ \nabla\ $	Curvatures
image1			
image3			

Table 5.6: Gradients and curvatures of 3D medical images.

	original (slice 110)	$\ \nabla\ $	Curvatures
image5			
image7			
image9			

Based on the obtained results, it is possible to see that the ordering criterion of the eigenvalues and the proposed selection criterion to detect the curvatures maintain a behavior as in the previous cases (2D images and 3D synthetic images). To observe with a greater level of detail, Table 5.7 shows the zoom in of a region of interest (blue box) within the image of gradient magnitude.



Table 5.7: Zoom in of gradients and curvatures of 3D medical images.

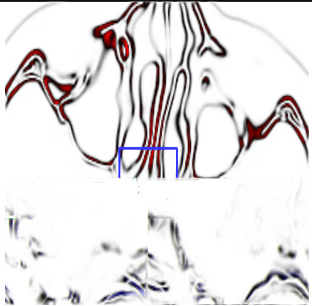

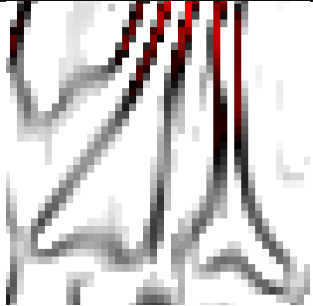
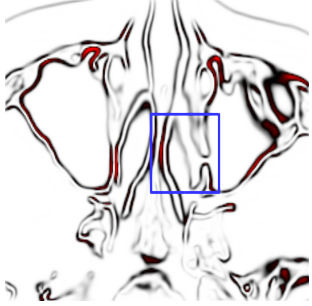





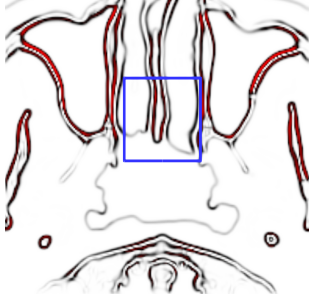


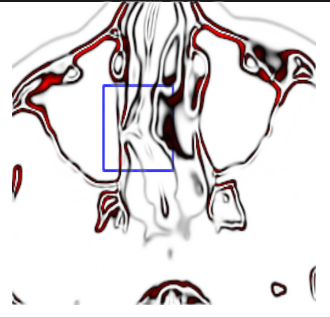
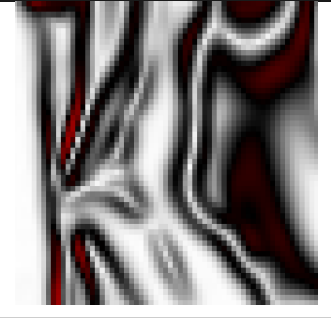

	$\ \nabla\ $	$\ \nabla\ $ - Zoom in	Curvatures - Zoom in
image1			
image3			
image5			
image7			

Table 5.7: Zoom in of gradients and curvatures of 3D medical images.

	$\ \nabla\ $	$\ \nabla\ $ - Zoom in	Curvatures - Zoom in
image9			





# Chapter 6

## Features and shapes extraction

“Geometry has two great treasures; one is the Theorem of Pythagoras; the other, the division of a line into extreme and mean ratio. The first we may compare to a measure of gold; the second we may name a precious jewel.”  
– Johannes Kepler.

From a given edge or curvature map the most direct high-level representation consists in computing closed contours, linking edge points by proximity, similarity, continuation, closure and symmetry. Apparently, this is a very simple and almost a trivial action for the human being, however, it becomes a difficult task when it is automatically performed. In general, ridges, edges or curvatures extraction is mainly a two-stage process: non-maximum suppression followed by edge linking. This often results in broken edges where the object's boundaries are not well defined.

Non-maximum suppression thins the ridges of curvature map by suppressing all values along the line of curvature that are not peak values of a ridge. While the linking process associates the maximum points to obtain the contour of the objects that belong to the image.

However, common problems of thinning algorithms are that in general they distort the shape of the objects, as well as big gaps can not be properly closed. In the case of linking algorithms, they do not guarantee to produce closed contours and they are sensitive to user-defined parameters, for example, if hysteresis algorithm is used, its threshold values are hard to define.

To solve above problems, in this chapter a non-maximum suppression algorithm and a hysteresis algorithm are proposed. Each of the algorithms is defined for both 2D and 3D images. In the case of the NMS algorithm, it is sought that no gaps in the suppression are generated using not only the curvature magnitude and the normal direction to the surface but also using the tangent direction. In the case of the linking algorithm, the previous information

is used and a distance criterion is added between the neighbors that make up the surface contour.

The chapter is organized as follows. In section 6.1 a brief state of the art of the edge detection and extraction techniques is presented, the most relevant algorithms for non-maximum suppression are explored and edges linking algorithms are reviewed. In section 6.2 the proposed algorithms are presented. Finally, section 6.3 shows the tests and the obtained results by applying the algorithms proposed, both for 2D and 3D images.

## 6.1 State of the art

One of the important elements in image processing is the features extraction. In this case, the extraction is made from the curvatures found in the previous chapter. Normally, the most used techniques refer to the edges extraction or the ridges extraction using gradient magnitude images. Therefore, this section presents in the first instance some definitions of a ridge and the characteristics or geometrical properties that it has. Then, the principal techniques found in the literature to edges and ridges extraction are addressed. As shown, two important steps in most techniques are the non-maximum suppression and the edges or ridges linking.

### 6.1.1 Ridge definition

In an intuitive way, a ridge is to be thought of as the path you follow on a mountain, where there's always a drop both to your left and to your right [100, 396]. However, various mathematical definitions of ridge exist [100, 394].

In one of the oldest studies of this kind, Saint-Venant identified ridges as loci of minimum gradient magnitude along level curve of a relief. This approach arises from a deep mathematical study, but its immediate application to images produces discontinuous ridges [396, 397].

According to Haralick [100], the first intuitive notion is that a digital ridge (valley) occurs when there is a simply connected sequence of pixels having gray-tone intensity values which are significantly higher (lower) in the sequence than those neighboring the sequence. Significantly higher or lower may depend on the distribution of brightness values surrounding the sequence as well as the length of the sequence. In addition, to determine ridges and valleys is needed to use the neighborhood of a pixel to estimate a continuous surface whose direction derivatives can be computed analytically.

## 6.1.2 Geometrical properties of the surface of a ridge point

According to Eberly [394], each ridge definition has its advantages and disadvantages, but it is desirable that the ridge satisfy certain properties.

First, the process which identifies ridges should be local. That is, a ridge point should be determined solely by the information in a local neighborhood of the point.

Second, the ridges should be invariant under the following transformations:

- translations in the spatial variables
- rotations in the spatial variables
- uniform magnification in the spatial variables
- monotonic transformations of the intensity function.

Geometrical properties of the local surface associated to a point  $(x, y)$  in an image are characterized by two main curvatures  $\lambda_1, \lambda_2$ , according to two main directions  $v_1, v_2$ . These curvatures are in fact the eigenvalues of the Hessian matrix, two corresponding directions are its eigenvectors. The Hessian matrix approximates locally the surface to a quadric. The relations of sign and greatness of the curvatures decide whether the point  $(x, y)$  is a ridge point, a peak or a saddle point, etc [308, 396].

According to feature detection criterion proposed in chapter 5, the curvatures along a ridge can not change sign because ridges were only detected (saddle point were not considered) using Equations 5.15 and 5.16.

## 6.1.3 Extraction schemes

In this section, schemes proposed in the literature to effectively perform edges or ridges detection and extraction are reviewed in a chronological way.

### 6.1.3.1 Nevatia and Babu feature extractor

Nevatia and Babu [398] proposed a technique of edge detection and line finding for linear feature extraction. Edge detection is by convolution with small edge-like masks. The resulting output is thinned and linked by using edge positions and orientations and approximated by piecewise linear segments. The technique proposed by Nevatia and Babu contains the following steps:

1. Determine edge magnitude and direction by convolution of an image with a number of edge masks.

## 2. Thinning and thresholding these edge magnitudes

- Using the magnitudes and the directions of neighbors during thinning retaining only edges whose magnitude is a local maximum
- The presence of an edge at a pixel is decided by comparing the edge data with some of the eight neighboring pixels. An edge element is said to be present at a pixel if:
  - (a) The output edge magnitude at the pixel is larger than the edge magnitudes of its two neighbors in a direction normal to the direction of this edge. (The normal to a  $30^\circ$  edge is approximated by the diagonals on a  $3 \times 3$  grid)
  - (b) The edge directions of the two neighboring pixels are within one unit ( $30^\circ$ ) of that of the central pixel.
  - (c) The edge magnitude of the central pixel exceeds a fixed threshold.

Further, if conditions a) and b) are satisfied, the two neighboring pixels are disqualified from being candidates for edges.

## 3. Linking the edge elements based on proximity and orientation.

## 4. Tracing boundaries using linking information.

## 5. Approximating the linked elements by piecewise linear segments.

In conclusion, to extract ridges or edges, Nevatia and Babu use thinning and thresholding, linking, and tracing boundaries.

### 6.1.3.2 Canny edge detector

It is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images [136, 241, 399]. Canny edge detector should have a good detection because it responds to edge, not noise. Then, it has a good localization because edges are detected near true edges. Finally, it has a single response because only one detector response per edge is obtained. The algorithm includes the following steps:

#### 1. Smoothing

- Blurring of the image to remove noise by convolving the image with the Gaussian filter.

#### 2. Finding derivatives, magnitude and orientation of gradient

- Filter image with  $x, y$  derivatives of Gaussian
  - Find magnitude and orientation of gradient
  - The edges should be marked where the gradients of the image has large magnitudes.
3. Non -maximum suppression
    - Thin multi-pixel wide “ridges” down to single pixel width.
    - Only local maxima should be marked as edges. Finds the local maxima in the direction of the gradient, and suppresses all others, minimizing false edges.
  4. Double thresholding and edge tracking (linking) by hysteresis
    - Potential edges are determined by thresholding (two thresholds: low and high are used). Canny recommended an upper:lower ratio between 2:1 and 3:1
    - Final edges are determined by suppressing all edges that are not connected to a very certain (strong) edge. Use the high threshold to start edge curves and the low threshold to continue them.

In conclusion, Canny uses non-maximum suppression followed by edge linking (tracking) using hysteresis to features extraction.

### 6.1.3.3 Förstner features extractor

Förstner [400] proposed a framework for low level features extraction. Its main goal is to explicitly exploit the information content of the image as far as possible. Feature extraction is based on local statistics of the image function. The main steps of the extractor proposed by Förstner are:

1. Estimation of Noise Characteristics

The noise characteristics are decisive for thresholding. Thresholding always is performing a hypothesis test of some kind. Therefore, thresholds should depend on the distribution of the test statistic and the significance level.

2. Information Preserving Restoration

Restoration in general aims at recovering the original signal from a given signal, undoing the effects of blur and noise.

### 3. Feature Detection

The classification of all image pixels into the three classes (point, line or region) can be interpreted as feature detection in the sense that only the existence and approximate location of the features is of primary concern.

### 4. Feature Location

The precise location of the point- and line-type features requires generic models which in principle allow to estimate the real-valued position of points and edges of the ideal image using the appropriate local scale of the feature. Location of edges can be seen to be special junctions with two edges of the same orientation meeting. Förstner use four steps to find features location. First, locating point-type features, second, classifying junctions and circular symmetric features, third, locating edges, and fourth, classifying edge- and line-type features.

#### 6.1.3.4 Armande, Montesinos and Monga 3D thin net detector

Armande, Montesinos and Monga [401] proposed an algorithm to extract 3D thin nets that can be written as follows:

1. Computation of the first, second and third order partial derivatives of the 3D volumetric image. They estimate these derivatives using recursive Gaussian filter and its derivatives [402].
2. Computation of the three principal curvatures  $k_1, k_2, k_3$  as well as the three principal curvature directions  $t_1, t_2, t_3$ .
3. Identification of the maximum, medium and minimum curvatures  $k_{max}, k_{med}, k_{min}$ .
4. Computation of the DMC.
5. Extraction of the zero-crossings of the DMC.
6. Computation of the DmC.
7. Extraction of the zero-crossings of the DmC.
8. Intersection of the two zero-crossings DMC, DmC images where  $k_{max}$  and  $k_{med}$  are greater than a given positive threshold.

Where DMC is the directional derivative of the maximum principal curvature and DmC is the directional derivative of the medium principal curvature.

### 6.1.3.5 Steger line detector

Steger [403] indicated that the extraction of lines with asymmetrical profiles usually results in biased line positions and widths, as was shown for lines with equal polarity in [301, 404]. Bias can be removed from the extracted line positions and widths for lines with equal polarity how it is shown in [301, 404]. Steger [403] proposed a technique to extract edges. Edges are considered as bright lines in the gradient image. Furthermore, by a scale-space analysis it is shown why line and edge junctions often cannot be extracted. From this analysis, a method to extract complete junction information is derived. The steps proposed in the Steger algorithm are:

1. Modeling lines as curves  $s(t)$  that exhibit a characteristic 1D profile in the direction perpendicular to the line, i.e., perpendicular to  $n(t) = s'(t)$ .
2. Extracting lines points in 2D using the first directional derivative in the direction  $n(t)$ . direction  $n(t)$  can be obtained for each pixel from the eigenvector corresponding to the eigenvalue of largest magnitude of the Hessian matrix of the smoothed image.
3. Extracting the line position with subpixel accuracy using the Hessian and the gradient result in a second-degree Taylor polynomial in each pixel
4. Extracting the line width
  - the edges on the right and left side of the line are extracted by extracting edge points on a search line of length  $2.5\sigma$  in the direction  $\pm n(t)$ .
  - The length of the search line is restricted.
5. Linking the individual line points into lines by an extension of Canny's hysteresis thresholding algorithm [136, 241] which takes the direction of the lines into account and correctly handles junctions [301].

Edge extraction only need two modifications to be made to the linking algorithm. First, the responses of the line detector are substituted by the response of the edge operator at the corresponding point. Second, how to implement the algorithm efficiently. The line detector only needs to be applied to the region of the image in which the response of the edge detector is higher than the lower hysteresis threshold. With these modifications, the resulting edge detector is only about 15% slower than the typical edge detection cycle of running the edge operator, doing a combined non-maximum suppression and hysteresis threshold operation, and linking the edge points into contours.

### 6.1.3.6 Aylward and Bullitt ridge detector

Aylward and Bullitt [405] proposed a ridge traversal method that uses an iterative step-maximize procedure. It can be written as follows:

1. Given an initial ridge point  $x_0$ , the ridge will extend in the positive  $\vec{t}_0$  and negative  $-\vec{t}_0$  tangent directions. Each direction is traversed independently.

- For the default implementation, at the  $i - th$  point  $x_i$  traversed on a ridge the approximate tangent direction  $\vec{t}_i$  is defined as  $\vec{v}_3$  the maximum eigenvalued eigenvector of the Hessian at  $x_i$ . The direction of ridge traversal is maintained by multiplying  $\vec{v}_3$  by the sign of the dot-product of  $\vec{v}_3$  and the previous tangent direction  $t_{i-1}$

$$\vec{t}_i = \text{sign}(\vec{v}_3 \cdot \vec{t}_{i-1}) \vec{v}_3 \quad (6.1)$$

2. The approximate normal directions at  $x_i$  (defined as  $\vec{v}_1$  and  $\vec{v}_2$  in the default implementation) specify an  $(N - 1) - D$  plane that the local ridge passes through. Under the assumption of smoothness, if that normal plane is shifted by a small amount  $\beta$  in the tangent direction, the ridge should continue to pass through that shifted normal plane—the ridge will exist as a local maximum in that shifted normal plane. The ridge criteria are tested at that  $(N - 1) - D$  maximum, and if the criteria are met, that maximum becomes the next ridge point  $x_{i+1}$ . Otherwise, the ridge traversal process terminates in that direction and, if not previously performed, the ridge extending in the direction  $-\vec{t}_0$  from  $x_0$  is traversed.
3. To assure the ridge continuity they define two additional criteria that must be met at each shifted normal plane's local maximum for that maximum to be accepted as the ridge point  $x_{i+1}$ .
  - One, to terminate traversal when T-junctions are encounter, the next tangent direction  $t_{i+1}$  must point in the “same direction” as the current tangent direction  $\vec{t}_i$ . They have chosen 0.7 as the minimum acceptable  $\vec{t}_i \cdot \vec{t}_{i+1}$  value indicating the “same direction”.
  - Two, to terminate traversal when a large spatial discontinuity is encountered,  $x_{i+1}$  must be “close” to  $x_i$ . They assume that centerline resolution at the inner scale of the data is desired. They, therefore, do not allow the Euclidean distance between any two continuous ridge points to be greater than one voxel.



They use Brendt's line-search method [406] to find local extremes. They use cubic splines to calculate subvoxel values and first and second derivatives. Also, the image is not blurred as a preprocessing step. Scaled calculations are made as necessary during the traversal process—this is more efficient if tubes are relatively sparse in the image, and this is required if the dynamic-scale implementation is used.

### 6.1.3.7 PCBR detector

Deng, Zhang, Mortensen, Dietterich and Shapiro [215] proposed a structure-based interest region detector called Principal Curvature-Based Regions (PCBR) (see Chapter 5). The PCBR detector employs the first steps of Steger's curvilinear detector algorithm [390]. It forms an image of the maximum or minimum eigenvalue of the Hessian matrix at each pixel. They called this the principal curvature image, as it measures the principal curvature of the image intensity surface. This process generates a single response for both lines and edges, producing a clearer structural sketch of an image than is usually provided by the gradient magnitude image. PCBR detector can be written as follows:

1. Smooth each image using an incremental Gaussian scale.
  - They first double the size of the original image to produce their initial image, and then produce increasingly Gaussian smoothed images.
2. Compute principal curvatures in scale space using Hessian matrix
3. Select principal curvature image using Equation 5.13 or 5.14.
4. "Clean" the principal curvature image with a grayscale morphological close operation
5. Hysteresis thresholding method based on local eigenvector flow
  - Eigenvector-flow hysteresis thresholding requires two thresholds (high and low) just as in traditional hysteresis thresholding.
6. The watershed transform is applied to the cleaned principal curvature image and the resulting watershed regions define the PCBR regions. PCBR utilizes line and edge features to construct structural interest regions.

PCBR detector uses hysteresis thresholding based on local eigenvector flow to extract the principal curvatures.

### 6.1.4 Non-Maximum suppression (NMS)

Based on the algorithms presented in the previous section, it is possible to observe that there are two elements in common to extract the maximum points (in this case ridges or edges). The first corresponds to non-maximum suppression and the second to the hysteresis. Afterwards, most detectors follow the same basic process: non-maximum suppression that eliminates pixels that are not local maxima, and a thresholding step that obtains the final set of points.

According to ter Haar Romeny [144] a natural way to define edges from a continuous grey-level image  $I : \mathbb{R}^2 \rightarrow \mathbb{R}$  is as the union of the points for which the gradient magnitude assumes a maximum in the gradient direction. This method is usually referred to as non-maximum suppression (see e.g. Canny [136, 241] or Korn [6]). Weickert [153] indicates that applying sophisticated thinning and linking mechanisms (non-maximum suppression and hysteresis thresholding), edges are identified as locations where the gradient magnitude has a maximum. According to Lindeberg [318], an edge point is defined as a point at which the gradient magnitude assumes a maximum in the gradient direction (see e.g. Canny [136, 241] or Korn [6]). In conclusion, NMS can be positively formulated as local maximum search, where a local maximum is greater than all its neighbors (excluding itself) [407]. In this section, some NMS algorithms are reviewed in more detail.

#### 6.1.4.1 Canny's NMS algorithm

Nixon et. al. [408] indicated that Canny edge detection operator [136, 241] is perhaps the most popular edge detection technique at present. It was formulated with three main objectives:

- optimal detection with no spurious responses
- good localization with minimal distance between detected and true edge position
- single response to eliminate multiple responses to a single edge.

The first requirement aims to reduce the response to noise. This can be effected by optimal smoothing; Canny was the first to demonstrate that Gaussian filtering is optimal for edge detection (within his criteria). The second criterion aims for accuracy: edges are to be detected, in the right place. This can be achieved by a process of non-maximum suppression (which is equivalent to peak detection). Non-maximum suppression retains only those points at the top of a ridge of edge data, while suppressing all others. This results in thinning: the output of non-maximum suppression is thin lines of edge points, in the right place. The third constraint concerns location of a single edge point in response to a change in brightness. This is because more than one edge can be denoted to be present, consistent with the output obtained by earlier edge operators.

Canny [136, 241] uses the edge direction estimated from gradient of a Gaussian-smoothed

image surface using a simply differentiating in the  $x$  and  $y$  directions. The gradient magnitude is then non-maximum suppressed in the gradient direction. It uses a nine-pixel neighbourhood as shown in Figure 6.1.

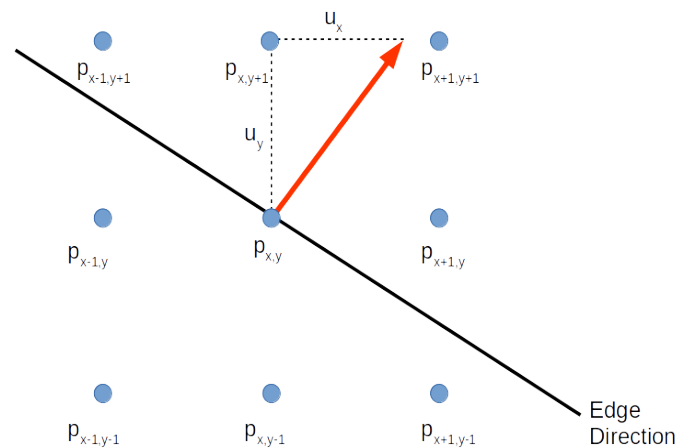


Figure 6.1: Kernel for non-maximum suppression used by Canny.

The normal to the edge direction is shown as an red arrow, and it has components  $(u_x, u_y)$ . Canny uses three points for non-maximum suppression, one of which is  $p_{x,y}$  and the other two are estimates of the gradient magnitude at points displaced from  $p_{x,y}$  by the vector  $u = (u_x, u_y)$ . Then, Canny considers for any vector  $u$  the two points in the 8-pixel neighbourhood of  $p_{x,y}$  which lie closest to the line through  $p_{x,y}$  in direction  $u$ . The gradient magnitude at these two points together with the gradient at the point  $p_{x,y}$  are used to estimate the value at a point on the line. Canny uses the value of the interpolated gradient in both sides of the line. The point  $p_{x,y}$  is marked as a maximum if his gradient magnitude is greater than the magnitude of its neighbors.

#### 6.1.4.2 Korn's NMS algorithm

Korn [6] proposes the use of confidence intervals to determine the neighbours of the analyzed pixel. Korn considers the use of four major regions, as shown in Figure 6.2. Each region has its opposite in the direction contrary to the gradient. This allows consider all the possible options to find the two neighbours to the central pixel.

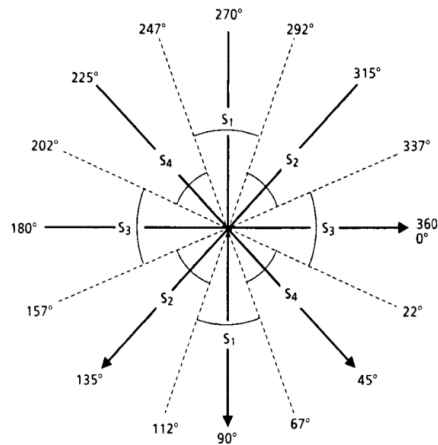


Figure 6.2: Search paths for the detection of maxima[6].

The search for extrema in the image must be performed in the direction  $\alpha$ , which is the direction of maximal gray value change (gradient magnitude). The angles  $\alpha = 0$  and  $\alpha = 90$  degrees denote the directions of the  $x$ - and  $y$ -axis, respectively. For applications in machine vision, Korn defines four search paths  $s_k$ , which are marked in Figure 6.2 by heavy arrows. The search for extrema is performed in the matrix of the magnitudes of the gray value gradient. With the exception of boundary pixels, the following differences are computed for every point  $P(i, j)$  in this matrix (see Table 6.1):

1.  $P(i, j) - P(i, j - 1)$ , search path  $s_1$
2.  $P(i, j) - P(i + 1, j - 1)$ , search path  $s_2$
3.  $P(i, j) - P(i - 1, j)$ , search path  $s_3$
4.  $P(i, j) - P(i - 1, j - 1)$ , search path  $s_4$

searching for maxima on the search path  $s_k$ , ( $k = 1, 2, 3, 4$ ).

$(i - 1, j - 1)$	$(i, j - 1)$	$(i + 1, j - 1)$
$(i - 1, j)$	$(i, j)$	$(i + 1, j)$

Table 6.1: Pixel coordinates.

If a maximum  $M_{s_k}(i, j)$  of the gradient magnitude has been detected on search path  $s_k$ , the corresponding angle  $\alpha$  of the gradient at pixel  $(i, j)$  is checked. The angle  $\alpha$  and the search path  $s_k$  must be *compatible using the ranges of tolerance* for the gradient direction  $\alpha$ . For example, if a maximum of the gradient magnitude is found for search path  $s_1$ , at position

$(i, j)$ , then the condition  $67 < \alpha \leq 112$  degrees or  $247 < \alpha \leq 292$  degrees must be satisfied for the gradient direction  $\alpha$  at position  $(i, j)$ .

For large changes of the direction  $\alpha$  of the gray value gradient (e.g., corners, branching of edges), the accuracy of  $\alpha$  is often very bad. As a consequence, a maximum can be ignored because the corresponding angle  $\alpha$  does not lie inside the prescribed range of tolerance, causing a gap. Thus, a connected chain of maxima can be terminated by such a gap.

According to Korn [6], gaps can be detected using the following procedure:

- Beginning with a central pixel  $P(i, j) \neq 0$  in a  $3 \times 3$  window the neighbors which differ from zero are counted. The result is a number  $N$  with  $0 \leq N \leq 8$ . The two interesting cases are  $N = 1$  and  $N = 2$ .
  - If  $N = 1$ , there must be a gap.
  - If  $N = 2$ , there are two neighbors  $P_1 \neq 0$  and  $P_2 \neq 0$  of  $P(i, j)$ .

The absolute value of the difference  $D_C$  of the columns and the difference  $D_R$  of the rows of  $P_1$  and  $P_2$ , is considered to indicate a gap, provided the following condition is true:  $(D_C \neq 2 \text{ and } D_R \neq 2)$  and  $((D_C + D_R) < 2)$ .

#### 6.1.4.3 Devernay's NMS algorithm

Non-maximum suppression method proposed by Devernay [7] is based on one of the two methods commonly used for edge detection, the suppression of the local non-maxima of the magnitude of the gradient of image intensity in the direction of this gradient, the other one being to consider edges as the zero-crossings of the Laplacian of image intensity. NMS algorithm consists of:

1. Let a point  $(x, y)$ , where  $x$  and  $y$  are integers and  $I(x, y)$  the intensity of pixel  $(x, y)$ .
2. Calculate the gradient of image intensity and its magnitude in  $(x, y)$ .
3. Estimate the magnitude of the gradient along the direction of the gradient in some neighborhood around  $(x, y)$ .
4. If  $(x, y)$  is a local maximum then estimate the position of the edge point in the direction of the gradient as the maximum of an interpolation on the values of gradient norm at  $(x, y)$  and the neighboring points.
5. If  $(x, y)$  is not a local maximum of the magnitude of the gradient along the direction of the gradient then it is not an edge point.

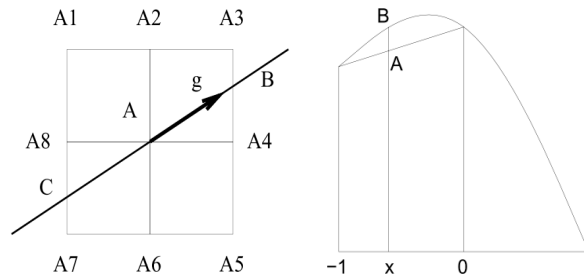


Figure 6.3: Examples of linear and quadratic interpolation[7].

Devernavy uses for step (4) a typical  $3 \times 3$  neighborhood and the values of the magnitude are linearly interpolated between the closest points in the neighborhood. For example, in Figure 6.3 (left), the value at  $C$  is interpolated between the values at  $A_7$ ,  $A_8$  and the values at  $B$  between those at  $A_3$  and  $A_4$ . Devernavy has also tried to use quadratic interpolation to compute these (the value at  $A$  would be interpolated between those at  $A_7$ ,  $A_8$ , and  $A_1$  as in Figure 6.3 (right)) and compared the results with the linear interpolation. After this edge detection process one usually does hysteresis thresholding [241] on the gradient norm and linking to get chains of pixels.

#### 6.1.4.4 Multiple Directional Non-Maximum Suppression

Sun and Vallotton [8] proposed an algorithm for linear feature detection using multiple directional non-maximum suppression (NMS). NMS is a process for marking all pixels whose intensity is not maximal as zero within a certain local neighborhood. This local neighborhood can be a linear window at different directions. Figure 6.4 shows four examples of linear windows at angles of  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$ , and  $135^\circ$ .

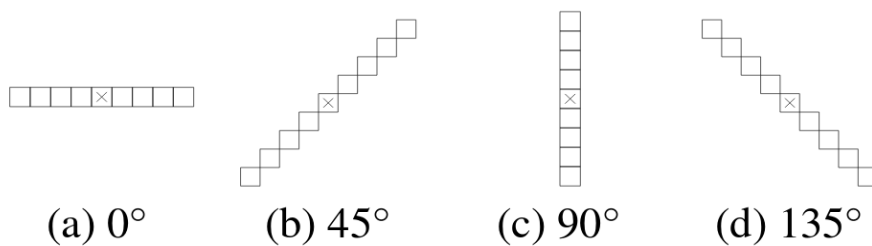


Figure 6.4: 2D linear windows at different directions [8].

Additional directions, such as those at  $22.5^\circ$ ,  $67.5^\circ$ ,  $112.5^\circ$ , and  $157.5^\circ$ , could also be used for obtaining the multiple directional non-maximum suppression. Furthermore, “x” in Figure 6.4 indicates the center of the linear windows. The length of the linear window shown is 9.

For 3D images, the linear windows used need to be oriented in 3D space. Figure 6.5 shows examples of the union of 3 and the union of 9 linear windows. Additional directions could also be used for obtaining the multiple directional non-maximum suppression for 3D images.

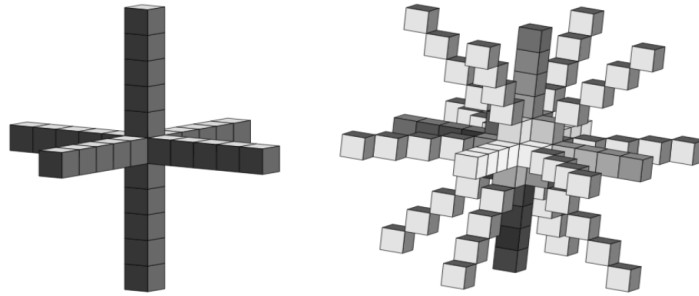


Figure 6.5: 3D linear window at different directions [8].

The steps of the linear feature detection algorithm proposed by Sun and Valotton [8] are the following:

1. Carry out 4 or 8 (3 or 9 for 3D case) directional non-maximum suppression on the images, and combine the multiple NMS outputs into one image. The symmetry check is performed during the process of NMS. Multiple local maxima can also be found within a linear window.
2. Remove small objects.
3. Obtain and link the end points of different linear objects when the end points are close to each other.
4. Thin the obtained image if necessary.

### 6.1.5 Linking and thinning algorithms

Virtually all edge detection techniques use some form of thresholding with the exception of the Marr-Hildreth technique, where edges are marked at any zero-crossing in the output of a LoG filter. Normally, the edge detection algorithms refer to the algorithm proposed by Canny, which includes a thresholding algorithm using hysteresis (also known as “double thresholding”) in the edge detection to link edge points.

In thresholding with hysteresis, two thresholds  $T_{low}$  and  $T_{high}$  are used, and starts from pixels or voxels  $x$  with  $f(x) \geq T_{high}$ . Values below  $T_{low}$  are discarded, those above  $T_{high}$  are preserved. The interesting points are those of intermediate value between  $T_{low}$  and  $T_{high}$ . These

points are preserved if they form a connected segment which is linked to a point above  $T_{high}$ . The use of hysteresis reduces the probability of streaking, i.e., the break-up of edges because of fluctuations in the operator response above and below a single threshold. The values of  $T_{low}$  and  $T_{high}$  can be determined by a global histogram estimate.

This procedure ensures segmentation of connected regions, since a number of “certain” foreground elements are selected while its neighbors may have a lower value. At the same time, noisy background voxels are suppressed by the higher threshold  $T_{high}$ .

In general, as previously mentioned, the Canny’s hysteresis algorithm is the most used to linking edges, however, there are other algorithms that will be presented below.

### 6.1.5.1 Canny’s hysteresis algorithm

The algorithm proposed by Canny [241] is based on local estimates of image noise and therefore falls into adaptive thresholding algorithms. Edge tracking is done by hysteresis thresholding. Two thresholds,  $T_{low}$  and  $T_{high}$  with  $T_{low} < T_{high}$ , are defined and applied to the gradient magnitude  $|g|$ . If  $|g| > T_{high}$  for some pixel, this pixel always belongs to an edge. Pixels with  $|g| > T_{low}$  are edge pixels if they are adjacent to other edge pixels. The algorithm proceeds as follows:

- Select the next pixel with  $|g| > T_{high}$  that is not yet assigned to an edge and assign it to a new edge.
- Track the edge as long as the adjacent pixels are found with  $|g| > T_{low}$ .

This process is repeated until no further pixels with  $|g| > T_{high}$  are found. The method finds connected edge segments. At intersections, it will track only one of the continuing curves. The other curve will be found as well if at least one of its edge pixels has a gradient larger than  $T_{high}$ . The value for  $T_{high}$  should be high enough to make sure that none of the starting pixels is a noise pixel. However, since the continuation of an edge is only found between neighboring pixels, the threshold  $T_{low}$  should be low so as not to hinder tracking.

The Canny edge detector (or similar tracking approaches) will create edges as a subset of all edge locations with a certain strength. Since the locations to be considered for an edge are local maxima of the gradient magnitude, edges may be false if noise has distorted the course of the edge. Low-pass filtering before computing the gradient prevents some of these erroneous responses, but it may also lead to false local maxima locations if smoothing causes the nearby edges to fuse.



### 6.1.5.2 Monga, Deriche, Malandain, and Cocquerez's hysteresis algorithm

They have extended the hysteresis thresholding introduced by Canny for 2D edge detection [241] to 3D and slightly improved it by adding a constraint [385, 409, 410].

Canny's algorithm is improved in the following way. The expansion in connected components is performed in any direction (by using 8-connectivity in 2D or 26-connectivity in 3D). The idea is to move along a direction orthogonal to the gradient i.e. within the hyperplane tangent to the contour. Let  $M_0$  be the edge point involved,  $\vec{G}(M_0)$  be its gradient direction,  $V$  be the set of its neighbourhoods (in 26-connectivity for instance). The expansion is performed by examination of all points  $M \in V$  such that the distance between  $M$  and the hyperplane tangent at point  $M_0$  is less than a threshold  $s$ :

$$\frac{|M\vec{M}_0 \times \vec{G}(M_0)|}{\|\vec{G}(M_0)\|} < s \quad (6.2)$$

The choice of  $s$  is related to the curvature of the contour to be obtained. They noticed that when choosing  $s$  enough high, the expansion in connected components is done for all neighbors of  $M_0$ .

This improvement is essentially useful in the case where the images are very noisy i.e. many 3D medical images. Particularly it allows to push down the low threshold without introducing too much false edge points.

This thresholding strategy is particularly efficient in the 3D case because it enables to get good connected edge points. This is of great interest to regroup these points in order to built surfaces.

### 6.1.5.3 Steger's hysteresis algorithm

Steger [390] proposed an algorithm to link the individual line points into lines. In the algorithm two hysteresis thresholds are applied to the eigenvalues of the dark lines in the gradient image, which are equivalent to the third directional derivative in the direction perpendicular to the original line. The algorithm to line and edge extraction in 2D images proposed by Steger can be described as follows.

Starting from the pixel with maximum second derivative, lines will be constructed by adding the appropriate neighbor to the current line. Because the maximum point typically does not lie at the endpoints of the line, this is done for both directions of the line (the orientation of the line is  $(n_x, n_y) = (\cos \alpha, \sin \alpha)$ ), that the line point detection algorithm will yield a fairly accurate estimate for the local direction. Since it can be assumed of the line, only three

neighboring pixels that are compatible with this direction are examined. Figure 6.6 shows two examples of pixels analyzed in the linking process. If the current pixel is  $(c_x, c_y)$  and the current orientation of the line is in the interval  $[-22.5^\circ, 22.5^\circ]$ , only the points  $(c_x + 1, c_y)$ ,  $(c_x + 1, c_y - 1)$ , and  $(c_x + 1, c_y + 1)$  are examined. Two of the eight possible cases given in Figure 6.6 show where pixels are checked for possible line continuation and they are marked as light gray squares. The choice regarding the appropriate neighbor to add to the line is based on the distance between the respective sub-pixel line locations and the angle difference of the two points.

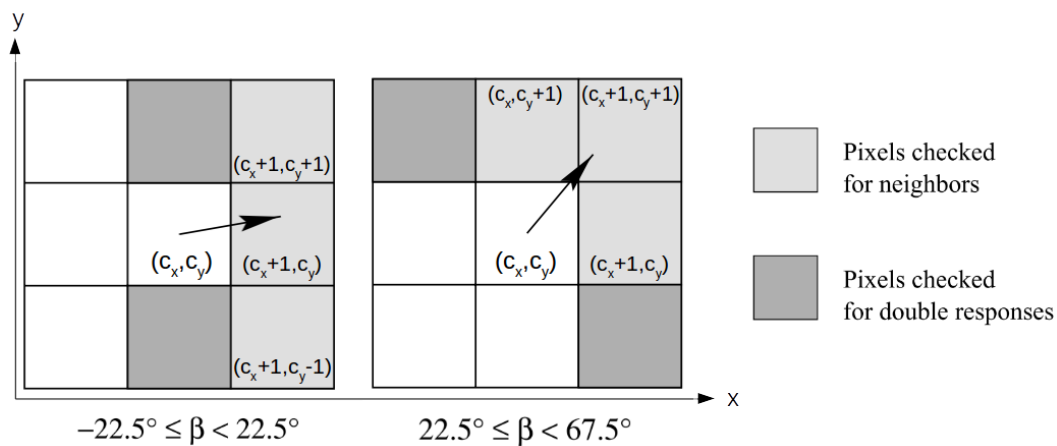


Figure 6.6: Two of the eight possible neighborhoods examined during the Steger's linking process.

In addition, if there are multiple responses to the line in the direction perpendicular to the line, e.g., the pixels  $(c_x, c_y + 1)$  and  $(c_x, c_y - 1)$ , they are marked as processed if they have roughly the same orientation as  $(c_x, c_y)$ .

In sum up, new lines will be created as long as the starting point has a second directional derivative that lies above a certain, user-selectable upper threshold. Points are added to the current line as long as their second directional derivative is greater than another user-selectable lower threshold. This is similar to a hysteresis threshold operation proposed by Canny [241]. However, the algorithm sometimes fails to extract junction points, for this reason, Steger proposes an algorithm to extract missed junctions.

In 3D case, linking of surfaces points into surfaces is a much more complicated problem than linking line points into lines because each surface point has more than two neighbors. However, the fundamentals of the algorithm, i.e., the hysteresis threshold operation, can be adapted to the case of 3D surfaces.

To add appropriate neighbors to the current surface, all 26 neighboring voxels of the cur-

rent surface point are examined whether they are compatible with the current point. Two criteria are used to determine compatibility: First, the surface normals of the neighboring point must not form an angle larger than a certain threshold,  $30^\circ$ , with the normal of the current surface point. This heuristic removes most, but not all, of the extraneous surface points extracted perpendicularly to the surface. Therefore, another criterion is also checked: the distance of the neighboring point to the plane spanned by the two smaller eigenvectors of the current surface point. If  $p$  denotes the current surface point,  $n$  the surface normal, with  $\|n\|_2 = 1$ , and  $q$  the neighboring point, this distance is given by

$$d = |(q - p) \cdot n| \quad (6.3)$$

He define that value of  $d$  must be closer than  $\sqrt{2}$  to the plane. This second criterion was also used in [411] to link 3D edges.

By combining the two criteria, almost all of the extraneous surface points are eliminated. After determining the appropriate neighbors, edges connecting the current surface point and the neighbors are created, the neighbors are inserted into a queue of points waiting to be processed if they are not marked as belonging to the current surface, and the current point is marked as processed and belonging to the current surface. This allows junction points to lie on multiple surfaces, and thus ensures that no gaps occur at junction areas.

#### 6.1.5.4 PCBR's hysteresis algorithm

Deng, Zhang, Mortensen, Dietterich and Shapiro [215] proposed a hysteresis thresholding method for 2D images based on local eigenvector flow called eigenvector-flow hysteresis thresholding. They use two thresholds (high and low) just as in traditional hysteresis method. The high threshold (set at 0.04) indicates a strong principal curvature response. Pixels with a strong response act as seeds that expand to include connected pixels that are above the low threshold. Unlike traditional hysteresis thresholding, the low threshold is a function of the support that each pixel's major eigenvector receives from neighboring pixels. Each pixel's low threshold is set by comparing the direction of the major (or minor) eigenvector to the direction of the 8 adjacent pixels' major (or minor) eigenvectors. This can be done by taking the absolute value of the inner product of a pixel's normalized eigenvector with that of each neighbor. If the average dot product over all neighbors is high enough, we set the low-to-high threshold ratio to 0.2 (for a low threshold of  $0.04 \cdot 0.2 = 0.008$ ); otherwise the low-to-high ratio is set to 0.7 (giving a low threshold of 0.028). The threshold values are based on visual inspection of detection results on many images.

This eigenvector-based active thresholding process yields better performance in building

continuous ridges and in handling perturbations, which results in more stable regions.

## 6.2 Proposed methods

The proposed method for extracting surfaces includes non-maximum suppression and hysteresis thresholding. Therefore, the algorithms proposed for NMS and hysteresis are presented below.

### 6.2.1 Non-maximum suppression

Non-maximum suppression methods have some problems. First, in Canny's method there are discontinuities between edges in the contour because NMS does not take into account the edge direction during suppression, it takes only into account the gradient direction (to locate neighbors) and gradient magnitude (to suppress neighbors). These contours can be closed by domain experts who have knowledge of anatomy. Second, in Korn's method is included a procedure to detect gaps, however, this criterion is based on masks that do not solve all cases. Third, multiple directional non-maximum suppression proposed by Sun and Valloton include more direction in the analysis, however, they do not specify additional criteria to suppress neighbors (it is similar to Canny's method).

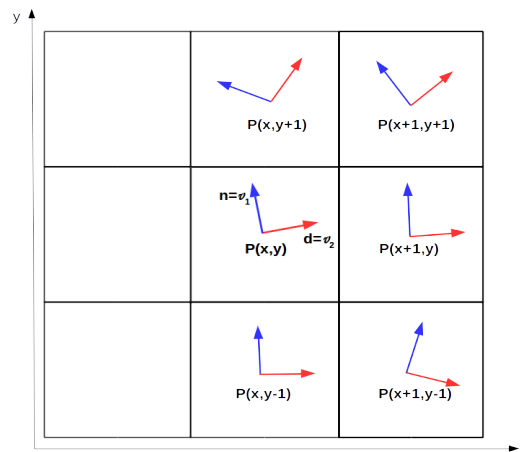


Figure 6.7: Non-maximum suppression neighbors.

As mentioned by Lindeberg [318], a natural extension of the notion of non-maximum suppression is to define an edge as a curve on the edge surface such that some suitably selected measure of edge strength is locally maximal with respect to this curve. Normally, this happens when the curve does not have bifurcations or corners that affect the maxima analysis,

because the curve has a maximal at some point when this point has the same direction that his neighbors. For this reason, the most important analysis occurs when there are neighbors in different directions to the analyzed point. Figure 6.7 shows a possible situation of neighbors to which the NMS algorithm should be applied. If the point analyzed is  $p(x,y)$ , according to Canny the two neighbors that are orthogonal to the direction of the normal are analyzed, in this case,  $p(x,y+1)$  and  $p(x,y-1)$ . Then, if and only if the magnitude of  $p(x,y)$  is greater than the magnitude of  $p(x,y+1)$  and  $p(x,y-1)$ , it is selected as maximum. As a result, the bifurcations and corners are eliminated generating gaps in the line.

For 3D images, Monga et. al. include the gradient direction of the surface in each point analyzed in order to increase the connectivity and obtain a surface with fewer holes. However, the expansion is executed for all points belonging to the observation window.

In this case, an NMS algorithm is proposed that allows the use of the Canny and Monga et. al. results, generating two conditions for non-maximum suppression. Figure 6.8 shows an example of a possible situation of neighbors analysis in the NMS algorithm. If the point analyzed is  $p(x,y)$  and its normal direction to the surface is  $n = v_1$ , the two orthogonal neighbors  $p(x,y+1)$  and  $p(x,y-1)$  are analyzed. The difference between the normal of the analyzed point and the normal ones of the neighbors is evaluated. In this case, the values  $\beta_1$  and  $\beta_2$  are obtained. It is determined if the angle is less than a threshold ( $30^\circ$ ). To calculate the difference, the dot product between the unit vectors is used. The absolute value that is in the range  $[0, 1]$  is taken, being the value of 1 when they present the same normal direction and 0 when they are orthogonal. For this case, since the maximum difference allowed between angles is  $30^\circ$ , a threshold of 0.866 is taken.

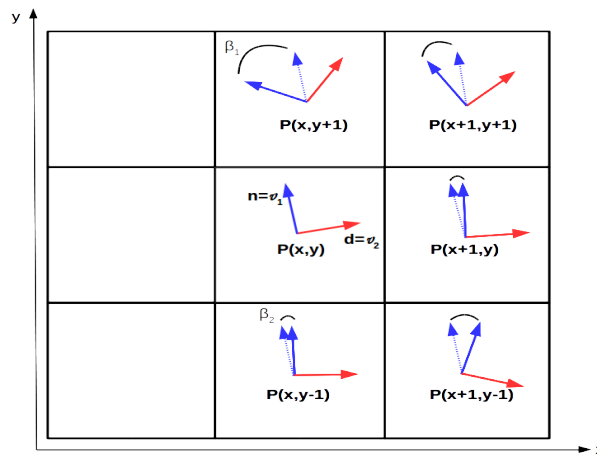


Figure 6.8: Non-maximum suppression using normal directions.

Therefore, NMS algorithm contains two restrictions. The first corresponds to the similar-

ity of the normal direction of the neighbors. The second is the previously described condition, that is, the neighbors magnitude is less than that of the evaluated point. This allows eliminating only the neighbors that are in the same normal direction and that have a smaller value of magnitude. In conclusion, the above conditions allow reducing the surface thickness by eliminating neighbors that are not maxima. In addition, it does not eliminate possible bifurcations or corners reducing the presence of holes in the surface.

Additionally, the proposed algorithm allows another restriction to be included. This restriction corresponds to the tangent direction of the surface, which can be evaluated in the same way as the normal direction to the surface. Figure 6.9 shows an example of the possible neighbors that are analyzed in the previous example. At first glance, this may seem redundant, but it is not. Mainly, when processing 3D images that present singularities (umbilical points).

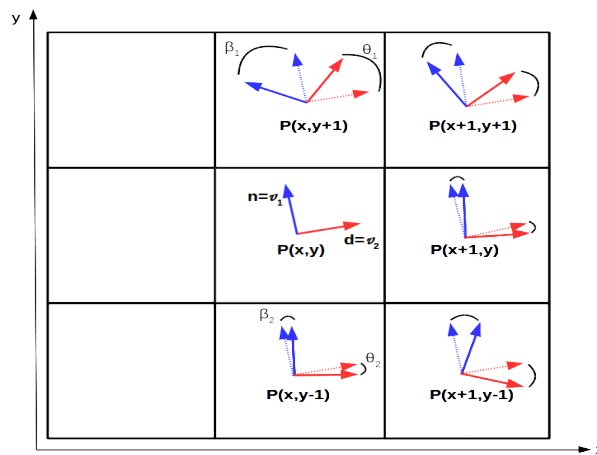


Figure 6.9: Non-maximum suppression using normal directions and tangent directions.

## 6.2.2 Hysteresis thresholding

According to Steger [390], the first thing that strikes the eye is that because Canny's original approach to hysteresis thresholding [136, 241] does not take into account the edge direction during the linking of edge points, it gets sidetracked into a maze of edges at the top of the container and in the shadow at the bottom of the container. This is undesirable since these edges have to be weeded out before or during the model matching.

For the above reason, the model of Canny and Monga et. al. do not completely comply with the linking of the curves that are to be extracted from the image. In the case of 2D images, the line linking algorithms proposed by Steger and by PCBR take into account the direction of the edge. These two algorithms present some differences. The first corresponds to that in Steger's algorithm the minimum and maximum thresholds for the hysteresis are obtained

through the angles that form the directions of the edges, while in PCBR they are defined dynamically according to the strength of the principal curvature and surface directions. The second refers to the fact that the Steger algorithm includes a metric of the distance between pixels or voxels analyzed while the PCBR algorithm does not take it into account.

The proposed hysteresis algorithm includes characteristics of the Steger and PCBR proposals. In the 2D case, the proposed algorithm includes the evaluation of the three neighbors that are in the curve direction. The pixel with the greatest curvature and whose direction is in the confidence interval is selected. The double responses proposed by Steger are not taken into account initially. The average point product of all the neighbors is not considered to define the thresholds, simply the upper threshold is defined in such a way that sufficient seed points are included to cover the curves that make up the image. One way to obtain this threshold automatically is by means of the histogram or by applying the Otsu thresholding algorithm or by simply inspecting the image and identifying the maximum curvatures and defining a value based on this exploration.

Figure 6.10 shows one of the possible configurations. Based on the tangent direction on the curve  $d = v_2$ , the neighboring pixels of the point  $p(x,y)$  are selected, in this case  $d$  lies between  $0^\circ$  and  $22.5^\circ$  which corresponds to the point  $p(x+1,y)$ , by therefore, its two neighbors  $p(x+1,y+1)$  and  $p(x+1,y-1)$  are selected. Then, the closest direction for each pixel is evaluated using the dot product between the direction of the curve at that point and the point analyzed. The selected pixel will be the one that is in the confidence interval and has the greatest curvature magnitude. If a next point is not found, the algorithm stops at that direction.

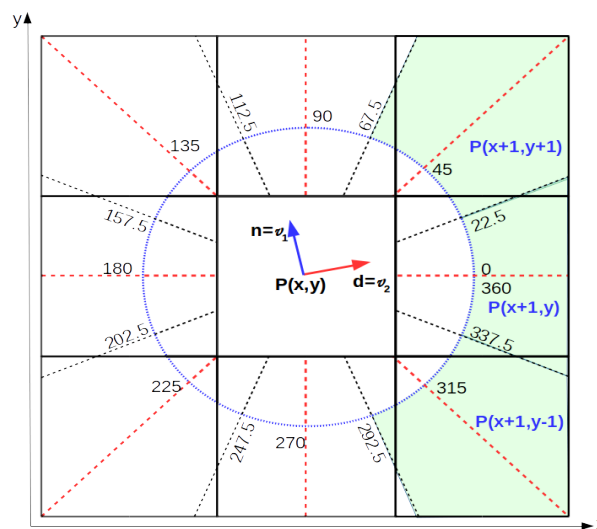


Figure 6.10: Possible configuration in hysteresis flow.

The two pixels that do not correspond to the line are analyzed to determine if they are

possible bifurcations or corners of the selected pixel. To do this, the direction of the selected pixel is evaluated with respect to its neighbors, if they are in the same direction they are eliminated, otherwise, they are marked as possible bifurcations or corners (this step is similar to the double responses defined by Steger).

When a pixel is found that corresponds to the end of the line, the line is traversed in the opposite direction from the starting point. To facilitate this process, the storage structure of the line (queue) is inverted and the last node is taken to start the expansion in the other direction. Finally, all the lines found and that have a minimum length conform the image edges.

In 3D case, linking surface points into surfaces is a much more complicated problem than linking line points into lines because each surface point has more than two neighbors. This means that the surface linking algorithm should link all points into a surface that have a response larger than a lower threshold and are connected to at least one point having a response larger than an upper threshold.

In the first instance, using the upper threshold, the seed points are selected to initiate the process of linking voxels to the surface. The plane tangent to the surface is defined from each seed point. The tangent plane defines the neighbors that must be explored to determine whether or not they belong to the surface. In 3D there is a 26-neighbourhood system instead of having a 8-neighbourhood system. There are 13 configurations that depend on the orientation of the normal vector to the surface. Each of the configurations is determined by the angles formed by the eigenvector  $n = v_1$  associated with the eigenvalue of greater magnitude. Figure 6.11 presents three possible tangent planes to the surface in the hysteresis process. In each example, the pixel analyzed in yellow and its neighbors in gray are shown. In the image on the left, the normal direction  $n = v_1$  is  $(90^\circ, 0^\circ)$ . In the central image the normal direction is  $(90^\circ, 45^\circ)$  and in the image on the right the direction is  $(45^\circ, 45^\circ)$ . In each case, the normal direction is given by  $n = v_1$ , where  $v_1$  is the eigenvector of the eigenvalue of greater magnitude and  $d = v_2$  corresponds to the tangent direction to the surface given by the eigenvector of the eigenvalue of smaller magnitude.

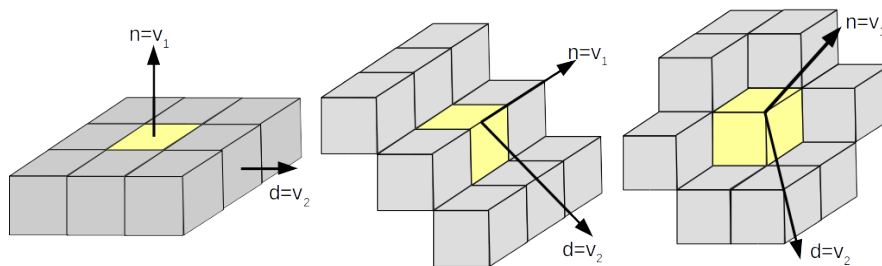


Figure 6.11: Different hysteresis configurations in 3D.



Based on the 8-neighbors found in the previous step, the orthogonal neighbors to each of them are evaluated. The same criteria defined in the non-maximum suppression algorithm are used. Figure 6.12 presents two examples of analyzed orthogonal neighbors. The voxel that is in red is the one found in the 8-neighbors and the blue voxels correspond to its orthogonal neighbors. The image on the left corresponds to the normal direction  $(90^\circ, 0^\circ)$  while the image on the right corresponds to the normal direction  $(90^\circ, 45^\circ)$ .

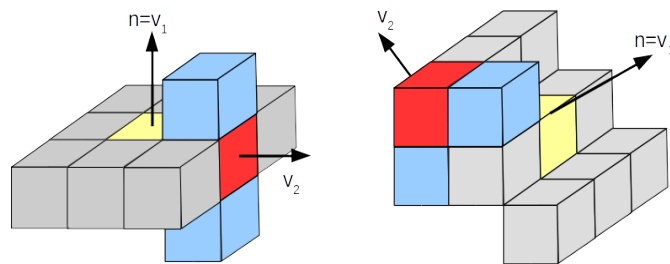


Figure 6.12: Examples of neighbors analyzed in hysteresis algorithm.

In some cases it was identified that the neighbors found as part of the surface have a distance greater than  $\sqrt{2}$ , which implies that they generate holes in the extracted surface. To solve this problem, proceed as follows. First of all, the 8-neighbors must be traversed in a circular and contiguous way finding the voxel that belongs to the surface to be extracted. Second, the distance between each of the voxels found must be evaluated to identify possible gaps using the distance between them. If the distance is greater than  $\sqrt{2}$  the two voxels closest to the tangent of the surface in each of the directions analyzed are marked as part of the surface. Figure 6.13 shows an example of this possible situation. The image on the left side shows two contiguous directions of the 8-neighbors analyzed. The voxels marked in red correspond to the voxels that are identified as part of the surface whose distance is greater than  $\sqrt{2}$ . In the image on the right side corresponds to mark the two voxels closest to the plane tangent to the surface, which are in yellow. It is clear that this procedure generates a redundant voxel in the contour of the surface, but this cost reduces in a simple way the possibility of holes in the surface. There are other alternatives to solve this problem, for example, a more time-consuming computation solution is to use an interpolation algorithm between the three points, i.e. the two voxels found as part of the surface (red voxels) and the voxel analyzed (central voxel of the 8-neighbors), this permits identify which voxel (yellow voxels) is closest to the plane or surface formed by these three points and add it to the surface.

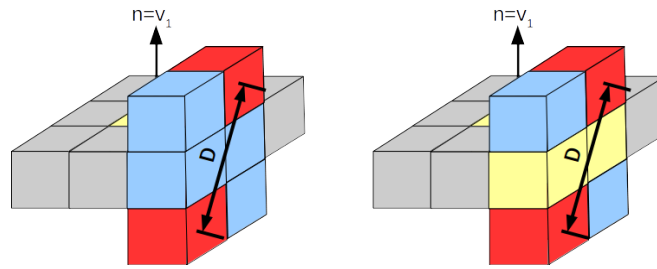


Figure 6.13: Example of possible holes in the surface.

Additionally, in order to apply the criteria defined in the non-maximum suppression algorithm in the hysteresis algorithm, the vectors must be reoriented using a rotation matrix that corresponds to a rotation around the  $y$ -axis of  $180^\circ$  and a rotation around the  $z$ -axis of  $90^\circ$ .

## 6.3 Experimental results

To test the proposed algorithms for the non-maximum suppression and hysteresis several tests are performed for both 2D images and 3D images. In 2D case, constructed images (synthetic images) and images traditionally used in computer vision and image processing are used. In 3D case, synthetic images were also constructed and medical images of the upper airways are used. In the first place, the tests with 2D images are presented and in second place the tests with 3D images.

### 6.3.1 Non-maximum suppression tests

In the non-maximum suppression tests both for 2D images and for 3D images the three conditions of suppression were considered. First, when the magnitude of the curvature (CM) is only used. Second, when the normal direction to the surface (CM + CN) is added. Finally, when the tangent direction of the surface (CM + CN + CD) is also included. The tests begin with 2D images and then the tests with 3D images are presented.

#### 6.3.1.1 2D images

Tests with 2D images start with the synthetic image that presents three geometrical figures, the square, the triangle and the circle. These figures were drawn by hand, therefore, their delineation is not smooth. This allows to evaluate what happens with "irregular" edges.

Table 6.2 presents the results obtained by applying the three criteria for NMS. The second column corresponds to the principal curvatures image. The third column presents the NMS

taking into account only the curvatures magnitude of the orthogonal neighbors. The fourth column shows the NMS using the curvatures magnitude and the normal direction to the surface. The fifth column presents the results by using the magnitude, the normal direction and the direction of the tangent plane to the curvatures. With respect to the rows, the second shows the curvatures magnitude and the region of interest (blue box) to make a zoom that allows to appreciate with a greater level of detail the results of applying the NMS algorithm. The third row presents both the magnitude and the direction vectors (normal and tangent). The fourth row corresponds to the zoom of the region of interest defined in the second row. The fifth row shows the zoom of both the magnitude and the direction vectors. Finally, in the sixth row, the number of non-zero pixels that make up the image is assigned.

Table 6.2: Non-maximum suppression results for 2D synthetic image.

	Original	CM	CM+CN	CM+CN+CD
Image				
Directions				
Image+Zoom				
Curvatures+Zoom				
N. Pixels	7805	4847	5416	5416

As can be seen in Table 6.2, using only the curvatures magnitude in a similar way to the NMS proposed by Canny, gaps are obtained. While in the other two cases the gap is eliminated. The main problem when taking into account the normal direction and the tangent direction is that a greater number of pixels is maintained that must be processed by the linking algorithm. However, this problem becomes an advantage, because it is not initially required an algorithm to fill gaps. In sum up, this allows junction points to lie on multiple lines, and thus ensures that no gaps occur at junction areas.

Next, tests are carried out with 2D images, which are traditionally used in computer vision. These images were used in the previous chapters. Tests that are going to be performed are the same as in the case of the synthetic image. From the image of the curvatures and the directions obtained when using the Hessian matrix, the NMS process is performed taking into account the three previously mentioned criteria. Table 6.3 shows the results obtained using the lena image. The columns and rows of Table 6.3 are the same as for Table 6.2. Selected region to zoom corresponds to Lena's left eye.

Table 6.3: Non-maximum suppression results for Lena's image.






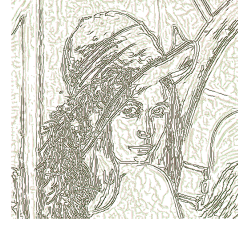






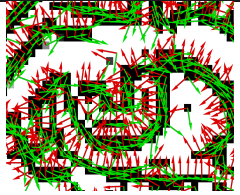
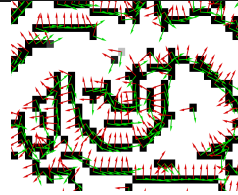
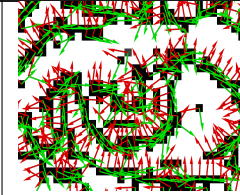
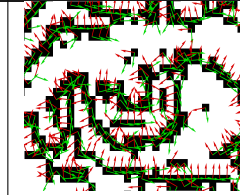
	Original	CM	CM+CN	CM+CN+CD
Image				
Directions				
Image+Zoom				

Table 6.3: Non-maximum suppression results for Lena's image.

	Original	CM	CM+CN	CM+CN+CD
Curvatures+Zoom				
N. Pixels	119518	61677	72090	72328

In the row of images with zoom (see Table 6.3), if only the curvature magnitude is used in NMS algorithm, gaps appear. While the resulting images when using the normal direction and the tangent address do not generate those gaps. However, when considering the normal direction and the tangent direction a greater number of pixels (in this case 72328) is generated.

Table 6.4 shows the number of pixels that each 2D image has before applying the NMS algorithms and after it is applied using the different suppression criteria. As can be seen, the variation between the algorithm that only considers the curvature magnitude and the other two is approximately 11% on average. While the difference between the algorithms that consider (CM + CN) and (CM + CN + CD) is 0.2% on average.

Table 6.4: Number of pixels before and after applying the NMS algorithms to 2D images.

	Original	CM	CM+CN	CM+CN+CD
geometric	7805	4847	5416	5416
baboon	120310	83763	100273	100524
barbara	118779	73852	87923	88183
boat	115407	64910	80207	80436
cameraman	31177	16477	20224	20273
lena	119518	61677	72090	72328
<b>average</b>		<b>40.9%</b>	<b>29.8%</b>	<b>29.6%</b>

It can be concluded at this point, that there is no significant difference between the last two

NMS algorithms for the analyzed 2D images. In addition, it can be stated that when curvature magnitude is only used, gaps appear at the edges, while for the other two NMS algorithms, no.

### 6.3.1.2 3D images

The tests performed with 3D images are divided into two sets. The first corresponds to the synthetic images used in the previous chapters. The second corresponds to the head-neck medical images defined in the previous chapters. In the same way as in the previous section, the three criteria of voxels selection not to be suppressed is applied, that is, the curvature magnitude (CM), the normal direction to the surface (CN) and the tangent direction to the surface (CD).

Table 6.5 presents the results obtained for the ellipsoid and the hyperbolic paraboloid. For each of the images, the three NMS algorithms are applied. The results obtained are presented in the rows of Table 6.5. Additionally, a region of interest was selected for each of the images in order to see the behavior of the algorithms with a greater level of detail. For both the ellipsoid and the hyperbolic paraboloid, the upper left quadrant was selected. In addition, the slice 16 of the ellipsoid is shown in the axial plane and for the hyperbolic paraboloid, the slice 129 was selected.

Table 6.5: Non-maximum suppression results for 3D synthetic images.


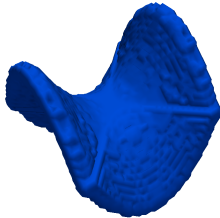
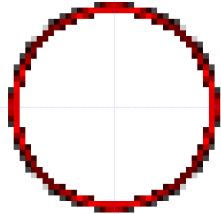
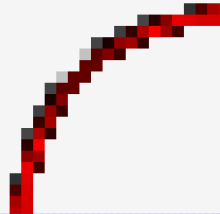
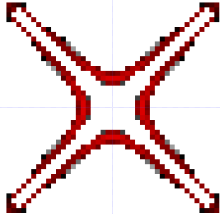
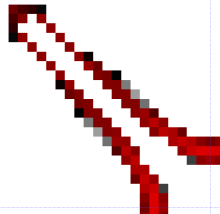
	ellipsoid	zoom	hyperbolic paraboloid	zoom
Image				
original				

Table 6.5: Non-maximum suppression results for 3D synthetic images.

	ellipsoid	zoom	hyperbolic paraboloid	zoom
CM				
CM+CN				
CM+CN+CD				

The reduction in the number of voxels using the curvature magnitude is greater. In the case of the ellipsoid, the edges are much more defined and with fewer voxels around the curvatures with high magnitude (red color). The lowest voxel reduction is presented when using the three criteria to select the voxels to be eliminated. The ellipsoid in all three cases does not have gaps, this is because the contours do not have bifurcations or significant corners. In the case of the hyperbolic paraboloid, the results are different, the use only of the curvature magnitude generates a loss of edges while the other two cases generate a complete contour.

Table 6.6 presents the number of voxels of original images and images obtained when applying the NMS algorithms. Comparing the averages of reduction with that of the 2D images, this is reduced for all cases. In addition, the difference between the algorithm that only uses the curvature magnitude and the other two algorithms is reduced in percentage. This difference changes from 11% to 9%. Additionally, the difference between the algorithms that use as criteria (CM + CN) and (CM + CN + CD) is increased from 0.2% to 5% approximately.



Table 6.6: Number of pixels before and after applying the NMS algorithms to 3D synthetic images.

	Original	CM	CM+CN	CM+CN+CD
ellipsoide	6632	4278	4870	5588
cube	51216	30228	31000	38704
cylinder	80794	54959	61195	63499
elliptic paraboloid	44611	27494	31921	33510
hyperbolic paraboloid	10234	7793	9582	9766
<b>average</b>		<b>34.13%</b>	<b>25.02%</b>	<b>18.21%</b>

In conclusion, as in the case of 2D, the results when applying the NMS algorithm using only the curvature magnitude generates gaps in the surface, while when adding direction information this does not happen.

Table 6.7 presents the results of applying the NMS algorithms in medical images. Table shows the results obtained for image1 and image3. In the original images, a region of interest is marked in a blue box. In these tests it can be observed that the NMS algorithm that uses only the curvature magnitude as the one that uses the curvature magnitude and the normal direction to the surface generate discontinuities in the edges of the surface. This means that the algorithm that considers the three criteria for eliminating non-maximum voxels allows obtaining better results in the continuity of the surface.

Table 6.7: Non-maximum suppression results for 3D medical images.

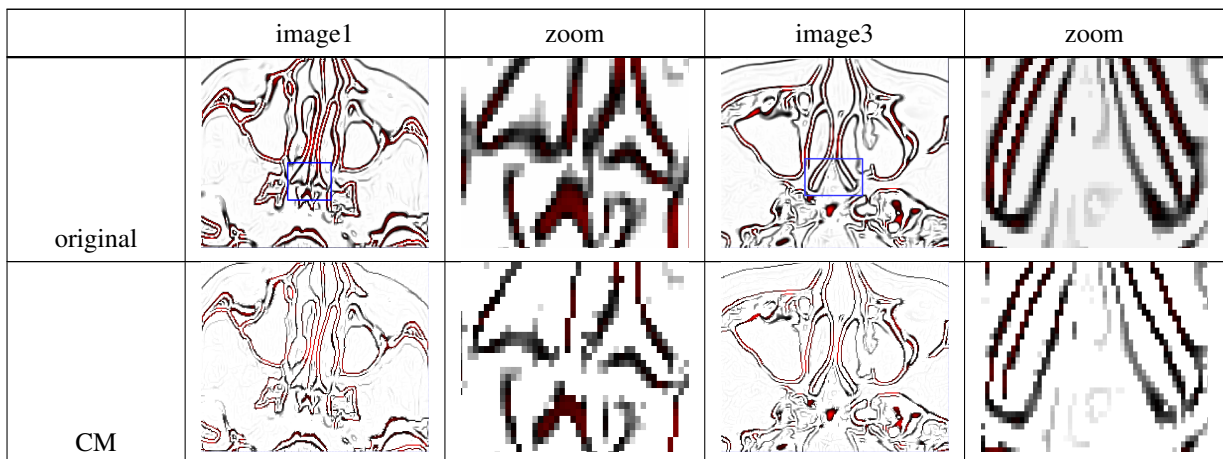




Table 6.7: Non-maximum suppression results for 3D medical images.

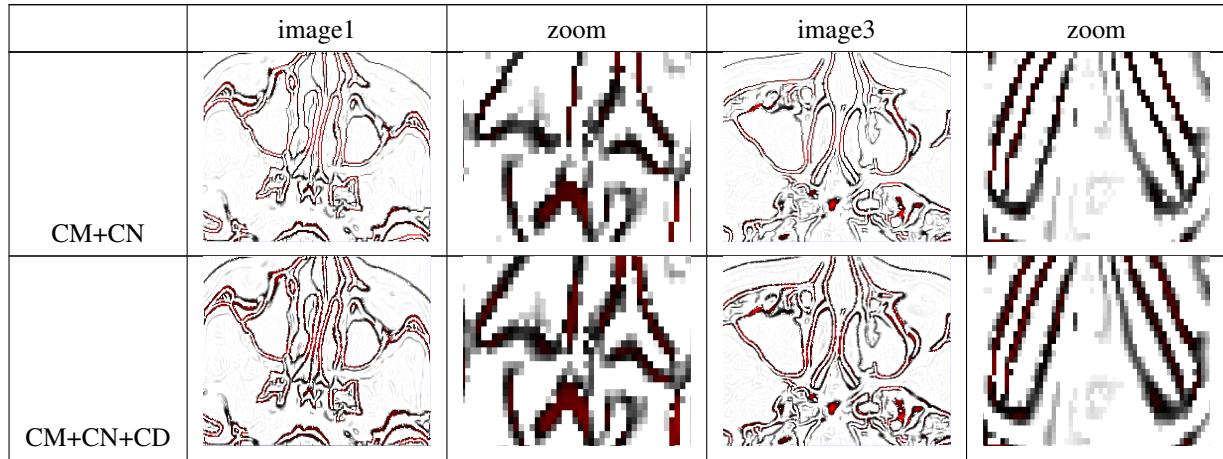


Table 6.8 quantitatively shows the reduction of voxels when applying the NMS algorithms. As you can see the behavior in the reduction is similar to the 3D synthetic images.

Table 6.8: Number of pixels before and after applying the NMS algorithms to 3D medical images.

	Original	CM	CM+CN	CM+CN+CD
image1	3.78634e+06	2.22882e+06	2.72729e+06	3.13605e+06
image3	4.17945e+06	2.37147e+06	2.89012e+06	3.39652e+06
image5	2.27787e+06	1.31016e+06	1.58116e+06	1.84454e+06
image7	3.19246e+06	1.93176e+06	2.34403e+06	2.66253e+06
image9	4.46121e+06	2.49975e+06	3.03941e+06	3.58097e+06
<b>average</b>		<b>42.1%</b>	<b>30.1%</b>	<b>18.3%</b>

NMS strategy using the three selection criteria enables to get good connected edge points in the 3D case. The probability of gaps is greatly reduced since a greater number of voxels is maintained. The hysteresis algorithm to link edges will allow discarding the points that do not belong to the surface and leave a thinner contour that will be processed by the segmentation algorithm.

### 6.3.2 Hysteresis tests

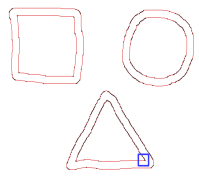
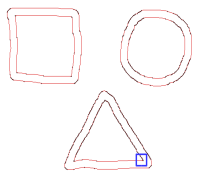
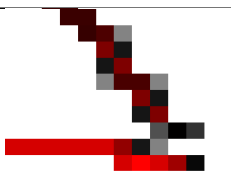
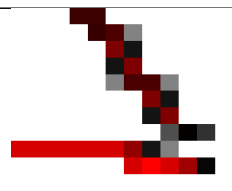
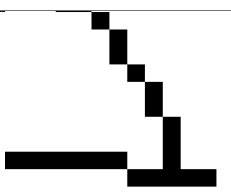
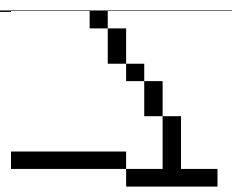
Tests of the hysteresis algorithm proposed are divided into two parts. The first one allows to evaluate the behavior in 2D images and the second part evaluates the 3D images.

#### 6.3.2.1 2D images

The first part of the tests with 2D images is done with the synthetic image. This allows seeing the behavior of the algorithm in the zones that have an acute form and that can generate ruptures because the normal direction has great variations. In the second part, traditional images are used which present multiple forms that will allow evaluating the behavior of the hysteresis algorithm.

Table 6.9 shows the results of applying the hysteresis algorithm to the images generated by the non-maximum suppression algorithms that generated best results (CM + CN and CM + CN + CD). The first row shows the original image (image obtained using NMS). The second row shows the zoom of the region marked in the blue box for each image. The third row presents the result after applying the hysteresis algorithm. The fourth row contains the number of pixels that belong to the detected region.







Table 6.9: Hysteresis results for a 2D synthetic image.

	CM+CN	CM+CN+CD
Image		
Image+Zoom		
Hysteresis		
N. Pixels	3623	3623

As you can see, the results present regions without gaps. This indicates that the proposed algorithm works properly when there are regions whose angles are less than  $90^\circ$ . Also in the corners of the square, there are no gaps. In conclusion, there are no gaps in the corners or bifurcations in the analyzed synthetic image.

Proceeding in the same way for traditional 2D images, Table 6.10 presents the results obtained for the Lena image. In this case, the number of pixels that form the regions obtained differs slightly. For the image obtained by using the NMS algorithm that considers the CM + CN, a pixel total of 27035 is obtained while 27074 pixels are obtained for the other image. However, in the region of interest that is expanded, no difference can be seen. What is important to note is that double responses are presented in one of the locations that appear expanded. This is because the algorithm detects these regions as possible bifurcations of the edge and maintains them as part of the region. This can be corrected by using an additional algorithm to detect the length of the possible bifurcations and if it is less than two pixels, it can be eliminated from the detected region.

Table 6.10: Non-maximum suppression results for Lena's image.

	CM+CN	CM+CN+CD
Image		
Image+Zoom		
Hysteresis		
N. Pixels	27035	27074

As it happened in the synthetic image, the hysteresis algorithm presents a good follow-up of the curvatures to detect the image edges.

Table 6.11 shows the number of pixels detected by the hysteresis algorithm for each of the processed images. As it is appreciated, there is no great variation between the results obtained when applying the algorithm to the two images resulting from applying the NMS algorithms. Even for the synthetic image and the cameraman image, the results are equals. The above implies that closing image edges permits to obtain accurate results when applying hysteresis algorithm proposed.

Table 6.11: Number of pixels after applying the hysteresis algorithm to 2D images.

	Original	CM+CN	CM+CN+CD
geometric	7805	3623	3623
baboon	120310	67169	67239
barbara	118779	48378	48437
boat	115407	42118	42167
cameraman	31177	7761	7761
lena	119518	27035	27074
<b>Average</b>		<b>62.17%</b>	<b>62.14%</b>

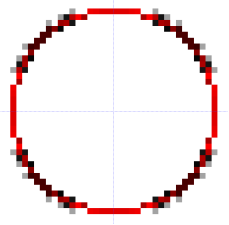
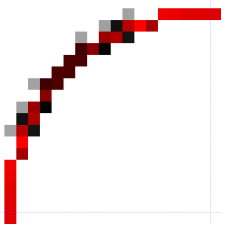
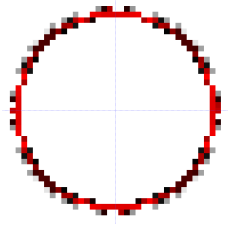
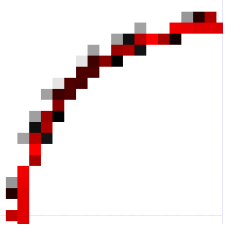
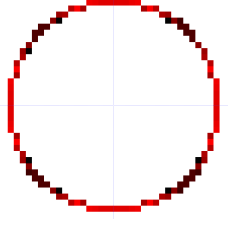
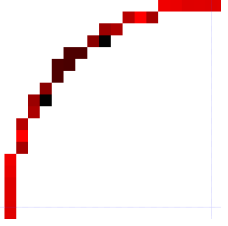
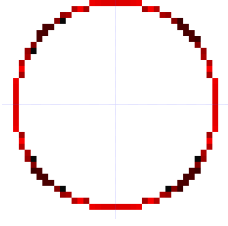
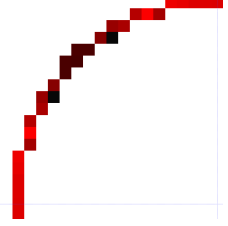
### 6.3.2.2 3D images

As mentioned before, the extraction of edges in 3D images is more complex. The proposed hysteresis algorithm allows detecting the bifurcation points and the corners present in the surface contour. To prove these characteristics, two tests were carried out. The first part with the synthetic images and the second part with the medical images.

Table 6.12 shows the results obtained by applying the proposed hysteresis algorithm to the image containing an ellipsoid. As input images to the proposed algorithm, the images obtained when applying the NMS algorithms are used using two and three suppression criteria respectively. The first column shows the result of the hysteresis algorithm for the first image and the second column shows a zoom to a region of interest to see the results. The region of interest corresponds to the upper left quadrant of the ellipsoid's image. The third column

shows the results of the hysteresis algorithm when using the second image. Similarly, the fourth column shows a zoom to the region of interest as mentioned before.

Table 6.12: Non-maximum suppression results for 3D synthetic image.

	CM+CN	zoom (CM+CN)	CM+CN+CD	zoom (CM+CN+CD)
ellipsoid				
Hysteresis				

As can be seen, the results in the region of interest are practically the same for the two input images. However, the total number of pixels is different. In the first image, 3348 pixels are obtained, while the number of pixels for the second image is 3459. This is due to the fact that in some regions of the ellipsoid for the second image, a greater number of double responses was generated that reduce the possibility of gaps in the surface contour.

Table 6.13 shows the number of pixels obtained for the surfaces of the synthetic images used in the tests. It can be seen that except for the cube, the number of pixels obtained using as input images input that use the three suppression criteria is greater.

Table 6.13: Number of pixels after applying the hysteresis algorithm to 3D synthetic images.

	Original	CM+CN	CM+CN+CD
ellipsoide	6632	3348	3459
cube	51216	25795	25795
cylinder	80794	43116	43120

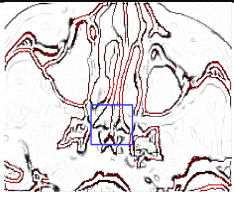

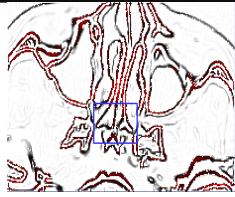

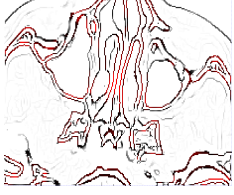

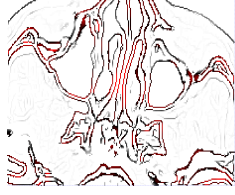

Table 6.13: Number of pixels after applying the hysteresis algorithm to 3D synthetic images.

	Original	CM+CN	CM+CN+CD
elliptic paraboloid	44611	22752	23613
hyperbolic paraboloid	10234	6664	6965

Looking in more detail at the images obtained, two elements were found to be taken into account. The first is that not in all cases the surfaces are completely closed. Particularly, for the cube a hole was detected in the region where the two planes of the surface are located, that is, in the area of the corners. The second element is the disappearance of contours in regions in which the edges are abrupt and close, that is, voxels are eliminated in areas with high concavity and the whose distance between edges is less than three voxels.

With respect to medical images, Table 6.14 shows the results obtained for image1. As can be seen, the best response obtained by the hysteresis algorithm to detect the edges of the surface was using the image that applies the three criteria for suppression.

Table 6.14: Hysteresis results for 3D medical image.

	CM+CN	zoom (CM+CN)	CM+CN+CD	zoom (CM+CN+CD)
image1				
Hysteresis				

However, although the results obtained are quite good, not all the edges of the surfaces are closed. It is recommended to use a closing technique to be able to complete the surface of the upper airways. It must be taken into account that the closing techniques use a window

of radius one, which implies that the minimum separation between the edges must be three voxels in order not to cause closure of the walls present in the upper airways.

Table 6.15 shows the number of voxels obtained for each of the medical images. As in 2D images and synthetic 3D images, the largest number is presented when NMS algorithm with the three selection criteria is used.

Table 6.15: Number of pixels after applying the hysteresis algorithm to 3D medical images.

	Original	CM+CN	CM+CN+CD
image1	3.78634e+06	1.89114e+06	1.98679e+06
image3	4.17945e+06	1.95456e+06	2.07518e+06
image5	2.27787e+06	1.03343e+06	1.08745e+06
image7	3.19246e+06	1.49914e+06	1.56956e+06
image9	4.46121e+06	1.91569e+06	2.02822e+06

In conclusion, the proposed hysteresis algorithm permits to obtain good results, however there are gaps in the contours. The gaps are reduced by the algorithm when using the distance criterion between the contiguous maxima found when visiting the 8-neighbors.





# Chapter 7

## Upper airways segmentation using principal curvatures

“Any sufficiently advanced technology is indistinguishable from magic.”

– *Arthur C. Clarke.*

This chapter briefly introduce the most common categories of image segmentation methods used for medical image segmentation to segment upper airways. Two inherent challenges in these problems are similar appearance of different tissues and the complex shapes of anatomical structures [412].

### 7.1 State of the art

The segmentation tasks have requirements that may vary for different applications regarding the type of extracted information, accuracy requirements, and the degree of automation. This has to be outlined for each application individually. However, a set of general requirements can be identified that typically have to be addressed by the methods in order to be clinically applicable:

- Deliver correct 3D reconstructions of the structures in a volume of interest.
- Have the ability to segment thin or small structures with low contrast.
- Differentiate structures from other adjacent structures with the same gray values.
- Robustly handle cases where parts of the structure are disturbed; e.g. due to imaging artifacts such as motion artifacts or beam hardening or due to disease such as tumors, calcifications, aneurysms, or stenosis where the appearance and geometry of the structures are disturbed.

- Require only minimal or no user interaction, which can usually be distinguished as automatic (unsupervised), interactive (semi-supervised), and supervised.

### 7.1.1 Image segmentation

Image segmentation is perhaps the most studied area in image analysis. A large number of papers on this topic is published annually in image analysis journals and conference proceedings[15, 19, 48, 72, 74, 77–80]. The developed methods often take into consideration various properties of images or objects, and when such properties deviate from the anticipated ones, errors occur. Even for a limited class of images, for instance MRI brain images, various methods have been developed, none of which is guaranteed to work correctly on a new image. This may be because there are sensor variations; variations in the brain's shape, size, and intensity distribution; and variations in intensities of tissues surrounding the brain. Since an error-proof image segmentation method cannot be developed, user assistance is needed to correct the obtained errors. At present, the best one can hope for is to have a segmentation method that can correctly find most areas of an object of interest, and in areas where it makes a mistake, allow the user to correct them.

The computerization of medical image segmentation plays an important role in medical imaging applications. It has found wide application in different areas such as diagnosis, localization of pathology, study of anatomical structure, treatment planning, and computer-integrated surgery. However, the variability and the complexity of the anatomical structures in the human body have resulted in medical image segmentation remaining a hard problem [78].

Segmentation of biomedical images in computer-assisted medical diagnosis is a preliminary operation aimed at detection and localization of regions of interest (ROI) in the images for their more detailed examination. The aim of image segmentation consists in reduction of total image examination time. Manually performed image segmentation is a tiresome operation itself; that is why it is reasonable to include computer-aided image segmentation into advanced image processing procedures as their integral part. In general, two basic types of image segmentation tasks in biomedical image processing arise[62]:

- segmentation of anatomical objects of known location and form, co-occurring with other anatomical objects in the given examination being to be neglected.
- detection and contouring of pathomorphological structures of a priori unknown localization and form, distinguished due to their specific biophysical or biochemical properties manifested by differences in color and/or in texture respecting the environment.

This section includes the most simple techniques typically referred to as thresholding and region growing. Then, more recent techniques are introduced where a segmentation is found by

means of optimizing an energy functional. In this context we talk about continuous variational and discrete combinatorial methods [413].

### 7.1.1.1 Thresholding and region growing

Early, and simple, techniques for segmentation mainly used the assumption that relevant objects in an image can be identified based on intensity values. The most simple approach identifies objects using a single threshold value, such that pixels above and below the threshold are object pixels and background pixels respectively. This works fine for high contrast objects with a sharp edge, but the method often fails as soon as the edges are smooth, of varying intensity and influenced by noise [48, 67, 68, 70, 72, 74, 414].

Global thresholding is the simplest segmentation technique. In upper airways, all regions below a chosen pixel intensity (threshold) are identified as air, and all above as tissue. The key decision to be made by the operator or algorithm is the value of the threshold intensity. The Otsu algorithm [415] or variants are often used. Global thresholding performs satisfactorily for many segmentation tasks in conventional CT and MRI images [414]. However the nasal cavity, having a number of intricate and narrow passages and adjacent paranasal sinuses, is susceptible to either under-segmentation (real boundaries between tissue and air are ignored by the segmentation algorithm, and distinct air-filled cavities are merged into one) or over-segmentation (the algorithm returns spurious boundaries where anatomically none exist)[416].

## 7.1.2 Upper airways segmentation methods

Upper airways segmentation is of significant importance because one of the most prevalent chronic disorders, airway disease is a major cause of morbidity and mortality worldwide [16, 417]. In order to understand its underlying mechanisms and to enable assessment of therapeutic efficacy of a variety of possible interventions, noninvasive investigation of the airways in a large number of subjects is of great research interest. Due to its high resolution in temporal and spatial domains, Computed Axial Tomography (CT) has been widely used in clinical practices for studying the normal and abnormal manifestations of airways diseases[18]. For example, evaluation of the upper respiratory airway obstruction (URAO) is of significant importance, as its physiological effects on ventilation rapidly evolve to secondary body malfunctions. Given the nature and location of the lesions, the invasive handling of the condition with direct bronchoscopy is undesirable; an imaging approach based on CT is preferred. In addition, objective and quantitative evaluation of the obstruction therefore requires the application of image processing and analysis techniques[19, 20].

Several automated segmentation methods have been proposed to segment the airways from CT

images. Evaluation of these methods has been problematic because, manual segmentation of airways is a difficult and very time consuming task due to the complexity of the 3D structure of the airways, specially, upper airways and airway tree. In addition, problem of automatic upper airways (nose, nasal cavity, paranasal sinuses, nasopharynx and pharynx) segmentation is hard, because low contrast exists between the interior of the airway and its surrounding tissue making very difficult the segmentation process[47, 48]. Without a prior model to constrain the segmentation, most algorithms (including intensity and edge based techniques) fail. Similar problems arise in other imaging applications as well, hindering the segmentation of the image.

Proposed algorithms to segment the upper airways are quite scarce, despite the need for a complete segmentation of the entire airway. Some of them have no formal specification what technique was used for segmentation, it seems that it was done manually. The result was used for executing simulation of airflow and thus able to determine where the airflow resistance affects the normal breathing process. The techniques found are presented below.

#### 7.1.2.1 Vos et al. segmentation

Vos et al. [418] indicate that all patients underwent a low-radiation dose CT scan to evaluate the upper airway geometry. Scanning was performed with patients awake, in supine and neutral position during one breath hold at the end of a normal inspiration. The dataset contain around 350–400 images with an in-slice pixel size of around 0.4mm and an inter-slice distance of 0.5mm. The segmentation of the upper airway is done using a commercial software package (Mimics 9.1, Materialise). It is based on the Hounsfield unit (HU) assigned to each pixel in the series of DICOM images. The HU is a value for the electron density of the tissue and reaches from -1024 to 3071. Characteristic values on the Hounsfield scale are -1024 and 1000 HU, respectively corresponding with air and bone.

Mimics is made up of several modules that allow semi-automatic segmentation of muscles, bones, airways, blood system and heart chambers among others. In addition, it allows to find centerline of tubular organs. Mimics's segmentation is based on a set of atlases and a crude input mask (see Figure 7.1). The crude input mask is obtained using thresholding and the segmentation is mainly performed using region growing or dynamic region growing.

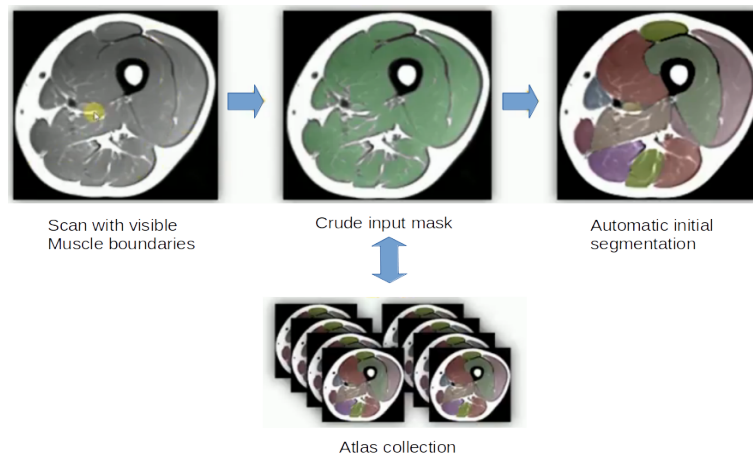


Figure 7.1: Example of Mimics's muscle segmentation [adapted from <http://www.materialise.com>].

However, as shown in Figure 7.2, they do not segment all of the upper airways, they initiate segmentation from the oropharynx to the lungs.

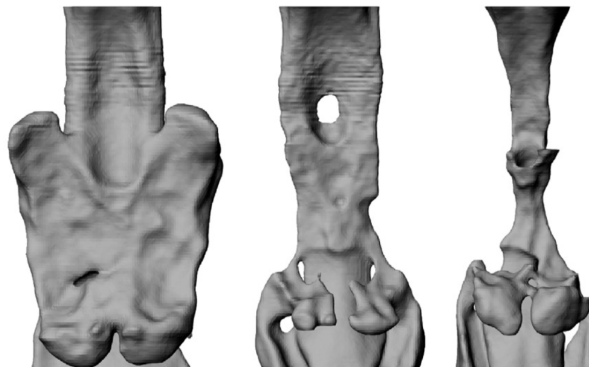


Figure 7.2: Models of upper airway for three different patients.

Vos et al. perform an analysis of the 3D models obtained from the segmentation. They define three anatomical parameters that can be measured in the upper airways: the volume, the length and the minimal cross-sectional area. From these measurements they calculate the mean cross-sectional area of the upper airways.

#### 7.1.2.2 Van Holsbeke et. al. segmentation

In the same way that Vos et. al., Van Holsbeke *et al.* [59] indicate that the acquired DICOM images were processed using a commercial software package (Mimics 15.0, Materialise). Subsequently, a segmentation of the upper airway was done using the Hounsfield Unit (HU) of

each voxel in the DICOM images as a discriminator parameter, making a binary distinction between air and solid structures (in other words using thresholding segmentation). Segmentation was performed from the nares to the first thoracic vertebra. The segmented region was then converted to a 3D model using a contour interpolation algorithm to reduce the staircasing effect. Using an appropriate smoothing algorithm with volume compensation, the 3D model was converted to a smooth realistic model without loss of patient-specific morphology of the upper airway. This model was then used for detailed analysis of the anatomical parameters, volume meshing, and simulation.

Figure 7.3 shows an example of the upper airways segmentation results using Mimics. Studied regions of the upper airway. The first segment (Zone 1) is formed between nostril to bottom of inferior turbinate. Second segment (Zone 2) between bottom of inferior turbinate to choanae. Third segment (Zone 3) between choanae to tip of uvula. Fourth segment (Zone 4) between uvula to epiglottis. Finally, fifth segment (Zone 5) between epiglottis to the first thoracic vertebra.

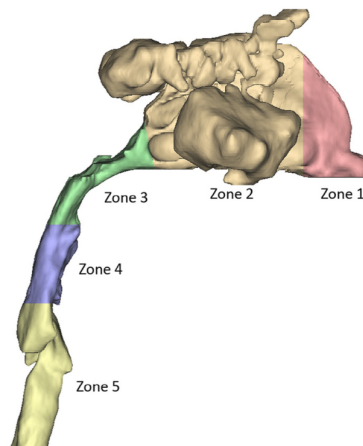


Figure 7.3: Upper airway segmented and divided into regions.

Van Holsbeke *et al.* [59] indicate that the dataset used contain an average of 350 to 400 DICOM images, all images having an in-plane spatial resolution of 0.3 mm and reconstructed with a slice increment of 0.5 mm.

### 7.1.2.3 Xu *et. al.* segmentation

Xu *et. al.* [58] made the upper airway segmentation using a software package based on 3DVIEWNIX. This software was developed by the Medical Image Processing Group of the University of Pennsylvania and used for various operations of visualization, segmentation,

registration, prefiltering, interpolation, standardization, and quantitative analysis of the MR images.

After field inhomogeneity correction, the airway is segmented by manual identification of seed pixels in selected slices, followed by semi-automated identification of airway voxels based on a fuzzy connectedness algorithm. The segmented and interpolated 1-bit axial slices are exported from 3DVIEWNIX for computational fluid dynamic (CFD) model preprocessing at a voxel size of  $0.2 \times 0.2 \times 0.2 \text{ mm}^3$ . Figure 7.4 shows an example of upper airway segmented model. Anatomical locations along the airway are shown for reference, including the overlap region (OL), where tonsils below and lateral to the oropharynx and nasopharynx overlap the adenoids superior and posterior to the nasopharynx, causing an area minimum.

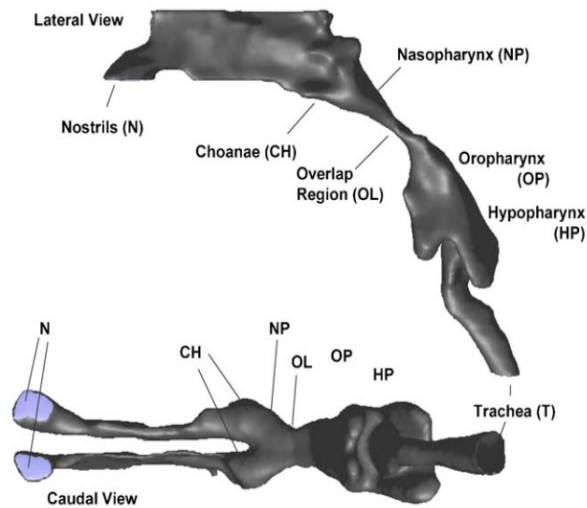


Figure 7.4: Upper airway segmented model.

#### 7.1.2.4 Jeong *et. al.* segmentation

Jeong *et. al.* [419] indicate that the scanned images are transferred to Bionix Body Builder software (Version 3.0, CANTIBio Inc., Suwon, Korea). The segments of interest are reconstructed into 3D images. The airway region is isolated from other structures by the built-in threshold and segmentation function. A threshold level range of 0–800 is applied to the area consisting of air. The frontal, ethmoidal and paranasal sinuses and oral cavity are erased by manual editing.

Figure 7.5 shows an example of upper airway segmented. In addition, the regions of the nasal airway are designated following the nomenclature proposed by Proctor [18]. Section 1 is located between the end of the nasal vestibule and the beginning of the main nasal passage. The middle and inferior turbinate begin at Section 2. Section 3 is located in the middle of

the nasal passage and shows the most convoluted cross-section. The two nasal cavities merge posterior to Section 4, and Section 5 constitutes the end of the nasal passage and the beginning of the nasopharynx. Section 6 is located at the velopharynx and Section 7 at the oropharynx. Sections 8 and 9 are located up and downstream of the epiglottis, respectively.

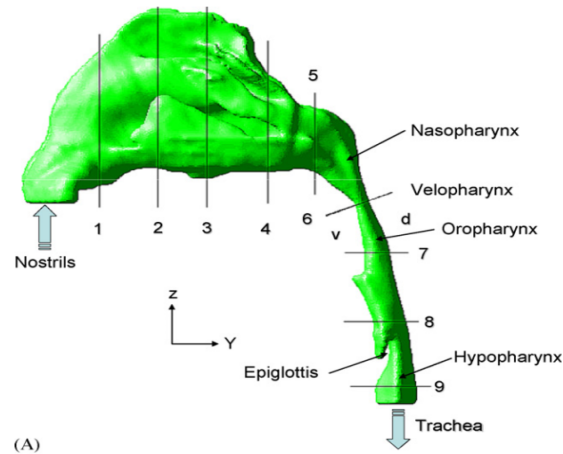


Figure 7.5: Upper airway segmented example.

## 7.2 Dataset

Information provided by San Ignacio Hospital about 109 patients is used for the testing process. For each patient, two sets of CT images were taken (awake and with simulated sleep). The main characteristics are presented below.

### 7.2.1 Patient data

The information of the patients is presented in Table 7.1 and Figure 7.6.

Characteristics of the 109 patients			
<b>Age</b>	Average	57	$\pm 34$ years
<b>Gender</b>	Female	47	43.12%
	Male	62	56.88%

Table 7.1: Patient's characterization

The age's patient distribution shown in Figure 7.6 present a wide range, between 23 and 81 years. The maximum frequency correspond to 65 years for six patients.



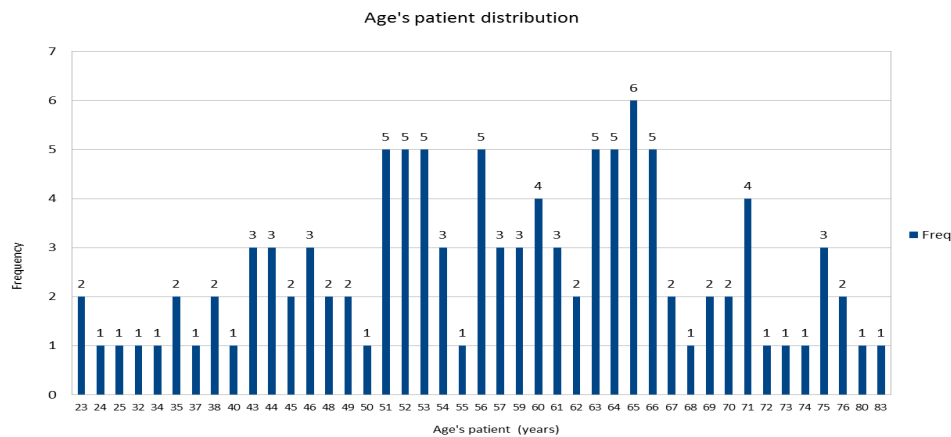


Figure 7.6: Age's patients

## 7.2.2 CT device specifications

The device used and its configuration is presented in Table 7.2.

Parameter	Value
Manufacturer	TOSHIBA
Modality	CT
Institution name	Hospital Universitario San Ignacio
Series description	Neck 0.5 CE
Manufacturer's model name	Aquilion
Scan Options	HELICAL_CT

Table 7.2: CT Scanner specifications

## 7.2.3 CT Metadata

The image characterization is done by means of the information present in the metadata of the DICOM images. The metadata fields that were considered relevant were DICOM code, DICOM parameter name, type of parameter (variable or constant) and value or interval related to the parameter for each of the captures made. Table 7.3 presents this characterization (blue values are variables and black values are constants).

Table 7.3: CT scanner configuration parameters

DICOM code	Name	Type	Value
0018 0050	Slice Thickness	constant	0.5 mm
0018 0060	KVP	constant	120
0018 0090	Data Collection Diameter	variable	[320, 500]
0018 1100	Reconstruction Diameter	variable	[178, 300]
0018 1120	Gantry/Detector Tilt	constant	+0.0
0018 1130	Table Height	variable	[40, 137]
0018 1150	Exposure Time (ms)	constant	500
0018 1151	X-ray Tube Current	variable	[141, 428]
0018 1152	Exposure (mAs)	variable	[70, 214]
0018 1160	Filter Type	constant	EC
0018 1190	Focal Spot(s)	constant	1.6/1.4
0018 1210	Convolution Kernel	constant	FC04
0020 0037	Image Orientation (Patient)	constant	LAP
0028 0002	Samples per Pixel	constant	1
0028 0010	Rows	constant	512
0028 0011	Columns	constant	512
0028 0030	Pixel Spacing (deltax, deltay)	variable	[0.349, 0.586]
0018 1170	Generator Power	variable	[16, 51]
0018 9324	Estimated Dose Saving	variable	[10, 54]
0018 9345	CTDI vol	variable	[24, 42]
0028 0100	Bits Allocated	constant	16
0028 1050	Window Center	constant	40
0028 1051	Window Width	constant	350

#### 7.2.4 Slices number

The slices number obtained was variable among the patients. The range of slices number was between 571 and 1001, the most usual value being 821 slices. This variation also affects the spacing between slices (see Table 7.7), which was between 0.349 mm and 0.586 mm.

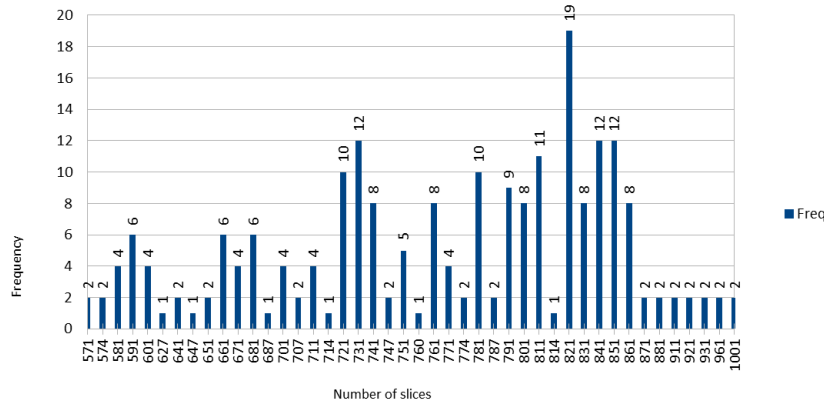


Figure 7.7: Number of slices obtained per patient.

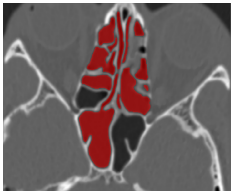
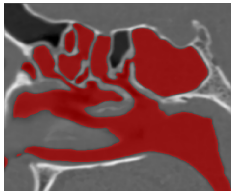
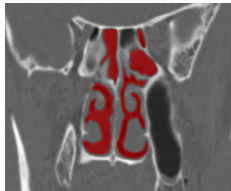
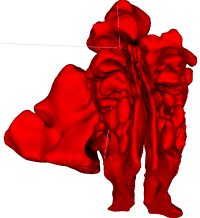
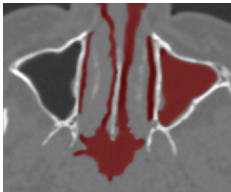
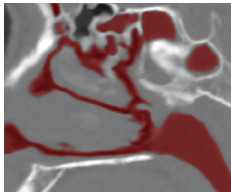
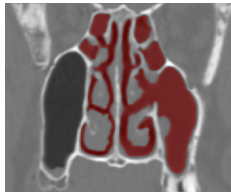
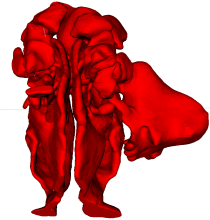
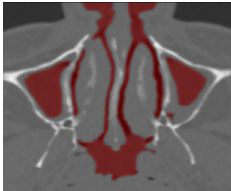
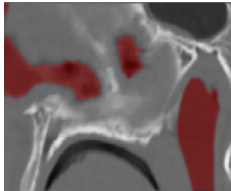
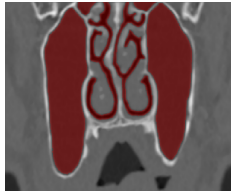
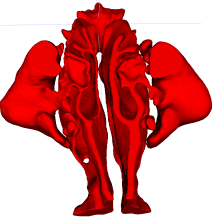
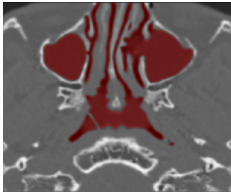
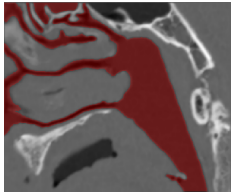
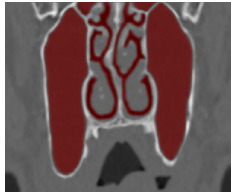
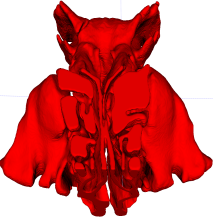
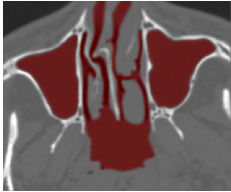
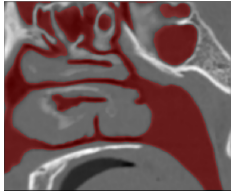

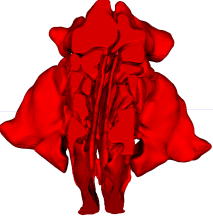
### 7.2.5 Ground truth (semi-automatic) segmentation

Based on the set of images, 10 images were selected to perform their segmentation in a semi-automatic way and thus conform the reference set (ground truth). The segmented 3D model of the upper airways was stored in metafile (.mha) format. The disconnected regions were removed from the upper airways geometry. The ITK-SNAP software was used to perform the segmentation. ITK-SNAP is an open-source semi-automatic 3D medical image segmentation package. This provides semi-automatic segmentation using active contour methods, as well as manual tracing and delineation of regions [420]. First, this software provides four pre-segmentation modes: Threshold: Voxels with intensity within a range specified by the user are assigned to positive velocity values. This is the simplest method of pre-segmentation suitable for easy tasks. Classification: the user draws examples of two or more kinds of tissue in the image, and an automatic learning algorithm uses these examples to assign velocity values to the rest of the image. Clustering: the image is automatically divided into two or more kinds of tissue by clustering voxels with similar intensities. This approach requires the least amount of information from the user, but is not always reliable. Edge Attraction: This approach finds and accentuates the edges in the image. The speed image is close to 1 away from the edges and close to 0 at the edges. Edge attraction is useful when the intensity of the image varies within the structure of interest [416, 420]. Second, ITK-SNAP requires an initialization step: in this stage, one or more seeds are placed inside of the structure of interest. These seeds are spheres that will be grown to form the segmentation of the structure of interest in the next step. Finally, evolution step: in this stage, the seeds evolve, expanding over the regions of the positive portions of the speed image, and contracting over the negative regions [416, 420].

Table 7.4 presents the axial, sagittal, coronal view and the volume of ground truth for the

images 1,3, 5, 7 and 9 respectively.

Table 7.4: Ground truth images

img	Axial	Sagittal	Coronal	Volume
1				
3				
5				
7				
9				

### 7.3 Segmentation method proposed

According to the upper airways segmentation summary presented in section 7.1.2, all segmentation methods are carried out manually and/or semi-automatically. In addition, in some cases

the segmented regions not include all parts that composed the upper airways.

Based on the results obtained in the previous chapters, the segmentation method proposed to segment the upper airways is presented in Figure 8. First, the original image is enhanced using a multi-scale top-hat morphological operator. This process is done in a non-supervised way. Second, the enhanced image is smoothed using an anisotropic diffusion algorithm. In the same way, this algorithm is non-supervised and the parameters suggested by Mirebeau et. al. [157] are used. Third, based on the smoothed image, principal curvatures using the Hessian matrix are computed. The matrix is defined for each image point using the gradients map. Fourth, similar to Canny's proposal, non-maximum suppression is used to thin the curvatures edge. The algorithm takes into account the curvature value and the curvature direction defined by its eigenvectors. In addition, to reduce the number of holes, possible points of junctions and/or bifurcations are taken into account. Fifth, a process of surface edges linking is performed using an algorithm that evaluates the surface direction of the analyzed point and its neighbors, as well as its intensity. Sixth, when performing the previous steps, a surface without holes is not always obtained. Therefore a holes filling algorithm is applied to fill the holes of the obtained surfaces. Finally, a region growing algorithm is used to obtain the upper airways segmentation. It requires a set of seed points. This step was carried out in this way to guarantee that the desired region of interest is obtained.

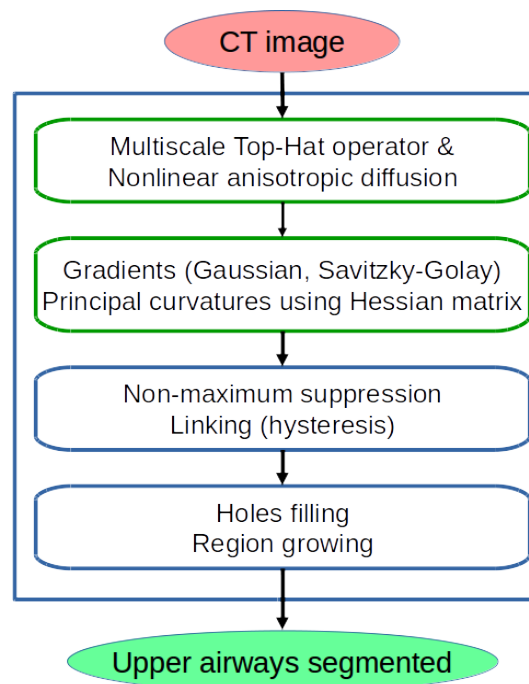


Figure 7.8: Upper airways segmentation method proposed.

## 7.4 Evaluation measures

There are measurements that are useful for evaluating results derived from image segmentation tasks. In this case, four overlap agreement measures and two overlap error measures are used, each quantifying some fraction of source  $S$  (segmented image) and target  $T$  (ground truth image) volumes where their regions agree or disagree.

The first overlap agreement measure is the “target overlap”  $TO$ . It is a measure of sensitivity.  $TO$  is the intersection between two similarly regions  $r$  in  $S$  and  $T$  divided by the volume of the region in  $T$ , where  $||$  indicates volume computed as the number of voxels:

$$TO_r = \frac{|S_r \cap T_r|}{|T_r|} \quad (7.1)$$

The second overlap agreement measure is the “union overlap”  $UO$ , or Jaccard coefficient. It is the intersection over the union:

$$UO_r = 2 \frac{|S_r \cap T_r|}{|S_r \cup T_r|} \quad (7.2)$$

The third overlap agreement measure is the “mean overlap”  $MO$ . It is called the Dice coefficient. It is computed as the intersection divided by the mean volume of the two regions:

$$MO_r = 2 \frac{|S_r \cap T_r|}{|S_r| + |T_r|} \quad (7.3)$$

The fourth overlap agreement measure is the “volume similarity coefficient”  $VS$ . It is equal to the differences between two volumes divided by their mean volume:

$$VS_r = 2 \frac{|S_r| - |T_r|}{|S_r| + |T_r|} \quad (7.4)$$

To complement the above agreement measures, false negative (FN) and false positive (FP) error measures are also computed. The range of these error measures is between zero and one; a value of zero means a perfect overlap.

A false negative error for a given region is the measure of how much of that region is incorrectly segmented. It is computed as the volume of a target region outside the corresponding source region divided by the volume of the target region:

$$FN_r = \frac{|T_r \setminus S_r|}{|T_r|} \quad (7.5)$$

Finally, a false positive error for a given region is the measure of how much of the volume outside that region is incorrectly assigned that region’s label. It is computed as the volume of

a source region outside the corresponding target region divided by the volume of the source region:

$$FP_r = \frac{|S_r \setminus T_r|}{|S_r|} \quad (7.6)$$

## 7.5 Experimental results

In order to evaluate the performance of the proposed segmentation method, several experiments are carried out. In addition, a comparison is performed using two classical segmentation methods: Active Contours (AC) and Chan-Vese (CV).


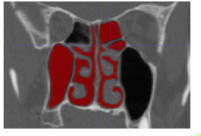
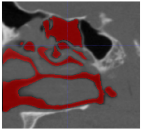

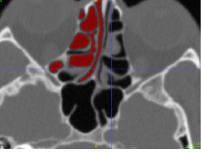

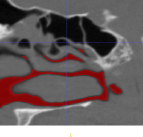
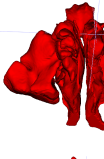
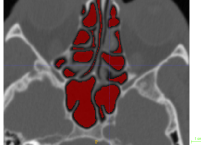

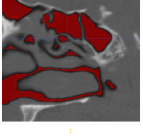
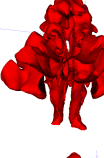


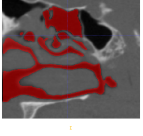
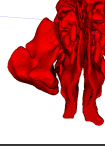
First, active contours method, or snakes has shown to be an efficient technique for image segmentation. The fundamental idea of active contour method is to start with a curve around the object to be detected, and the curve moves toward its interior normal and stops on the true boundary of the object based on an energy-minimizing model. The main drawbacks of this method are its sensitivity to initial conditions and the difficulties associated with topological changes like the merging and splitting of the evolving curve.

Second, to reduce the active contour drawbacks, Chan-Vese method is used. CV permits to detect objects whose boundaries are not necessarily detected by the gradient. CV successfully solved the minimization problem by using level set functions, which utilized the global image statistics inside and outside the evolving curve rather than the gradients on the boundaries. CV method has achieved good performance in image segmentation task due to its ability of obtaining a larger convergence range and handling topological changes naturally.

The first tests were carried out using five images. Each of them was segmented in a semi-automatically way, then it was manually corrected to obtain the reference image (see 7.2.5 section). The three previously mentioned methods were used: Active Contours (AC), Chan-Vese (CV) and the proposed algorithm (BF). Table 7.5 shows the results for the first image (image1).

The first row in Table 7.5 presents the axial, coronal and sagittal views, as well as a volume view of the reference image. The second row presents the results obtained using the AC method. The third row shows the results when using the CV method and the last row shows the results using the proposed method (BF). As can be seen, the AC method does not completely segment the regions corresponding to the nasal concha, inferior concha, ethmoidal sinuses and nasal cavity. The CV method segments regions that are not connected, for example, the right maxillary sinus. In addition, it does not completely segment the nasal cavity. The BF method also presents problems in the nasal cavity, however, it correctly segments the other upper airways regions.

Table 7.5: Upper airways segmentation of image1 using AC, CV and BF methods.

Img	Meth.	Axial	Coronal	Sagittal	Volume
1	Ref				
	AC				
	CV				
	BF				

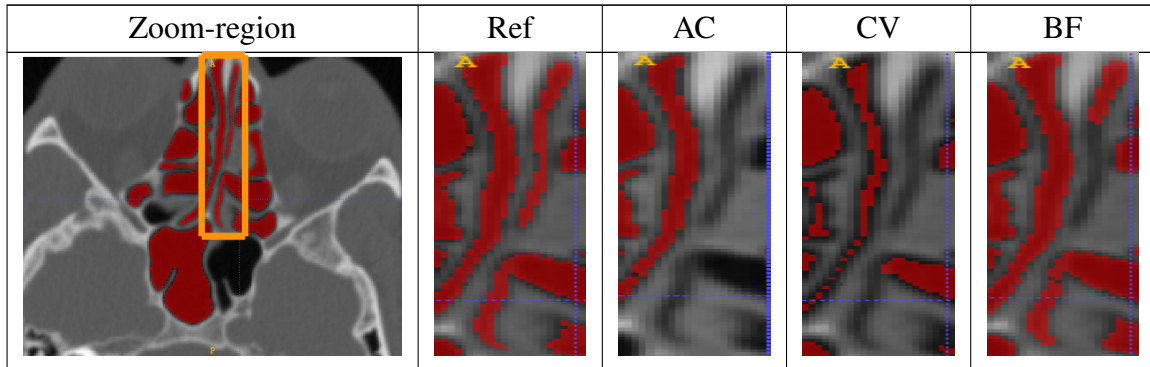
To observe the behavior in the nasal cavity region with a greater level of detail, Table 7.6 shows the images corresponding to this region of interest. The view used is the axial one. The first column corresponds to the reference image with the region of interest marked by a box. The second column corresponds to the zoom of the region of interest associated with the reference image. The third, fourth and fifth column correspond to the region of interest obtained by using the segmentation methods AC, CV and BF respectively. As can be seen, using the CV method, incomplete results in the left nasal cavity are obtained while with the proposed BF method, better results are achieved.

Table 7.6: Nasal cavity segmentation using AC, CV and BF methods.

Zoom-region	Ref	AC	CV	BF
-------------	-----	----	----	----

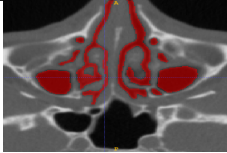
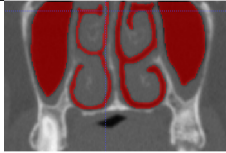
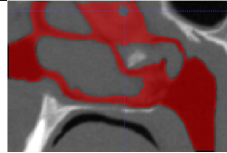
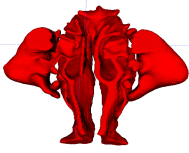
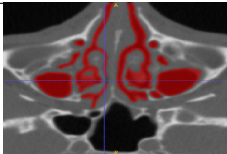
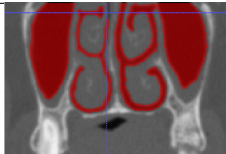
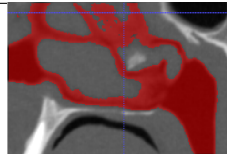
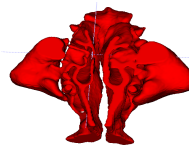
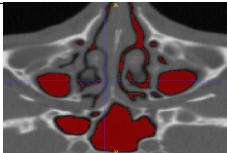
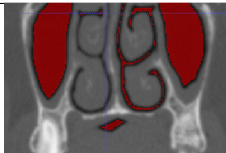
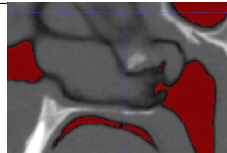


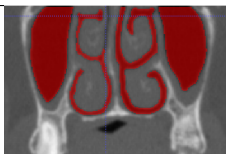
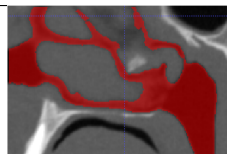
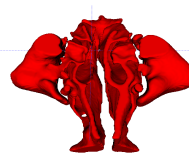


Table 7.6: Nasal cavity segmentation using AC, CV and BF methods.



To verify the behavior of the segmentation methods, Table 7.7 presents the results obtained for the image5. In this case, the visual results show a better performance using the AC method. This is because the left nasal cavity is segmented in a better way with respect to the BF method. The CV method generates unsatisfactory results.

Table 7.7: Upper airways segmentation of image5 using AC, CV and BF methods.

Img	Meth.	Axial	Coronal	Sagittal	Volume
5	Ref				
	AC				
	CV				
	BF				

To perform a quantitative analysis, the evaluation measures previously presented were

computed. The five test images were used and the results obtained are presented in Table 7.8. The best results according to each measure for each segmented image are highlighted in red. The BF method according to the Jaccard and Dice measures presents the best results, that is, the overlap of the obtained segmentation is better than that obtained using AC and CV. With respect to the similarity of the volume, the proposed algorithm presents a greater similarity except for the image7. The values obtained for false negatives and false positives are close to zero for the proposed algorithm, therefore the number of voxels that are out of the location is small.

Table 7.8: Evaluation measures results.

Image	Method	Target	Jaccard	Dice	Vol. sim.	False neg.	False pos.
image1	AC	<b>0,9885</b>	0,7383	0,8494	0,2814	<b>0,0115</b>	0,2554
	CV	0,6398	0,5135	0,6786	-0,1212	0,3602	0,2777
	BF	0,9506	<b>0,9460</b>	<b>0,9723</b>	<b>-0,0455</b>	0,0494	<b>0,0051</b>
image3	AC	0,9292	0,9106	0,9532	-0,0515	0,0708	0,0216
	CV	0,6891	0,5037	0,6700	0,0556	0,3109	0,3481
	BF	<b>0,9724</b>	<b>0,9584</b>	<b>0,9788</b>	<b>-0,0130</b>	<b>0,0276</b>	<b>0,0149</b>
image5	AC	0,8577	0,8563	0,9226	-0,1514	0,1423	<b>0,0019</b>
	CV	0,8595	0,5482	0,7081	0,3521	0,1405	0,3979
	BF	<b>0,9847</b>	<b>0,9763</b>	<b>0,9880</b>	<b>-0,0066</b>	<b>0,0153</b>	0,0087
image7	AC	<b>0,9682</b>	0,9305	0,9640	<b>0,0087</b>	<b>0,0318</b>	0,0402
	CV	0,9386	0,7083	0,8293	0,2330	0,0614	0,2573
	BF	0,9577	<b>0,9561</b>	<b>0,9776</b>	-0,0416	0,0423	<b>0,0017</b>
image9	AC	0,9151	0,9103	0,9530	-0,0829	0,0849	<b>0,0057</b>
	CV	0,9630	0,7554	0,8607	0,2125	0,0370	0,2220
	BF	<b>0,9690</b>	<b>0,9618</b>	<b>0,9805</b>	<b>-0,0239</b>	<b>0,0310</b>	0,0076
<b>Avg</b>	AC	0,9318	0,8692	0,9284	0,1152	0,0683	0,0650
	CV	0,8180	0,6058	0,7493	0,1949	0,1820	0,3006
	<b>BF</b>	<b>0,9669</b>	<b>0,9597</b>	<b>0,9794</b>	<b>0,0261</b>	<b>0,0331</b>	<b>0,0076</b>

In conclusion, the BF method proposed for image segmentation using principal curvatures presents excellent results. It is necessary to improve the branches and junctions detection in narrow regions, for example, in the nasal cavity, to increase the accuracy in the segmentation results.

# Chapter 8

## Conclusions

In this dissertation, a novel approach to segment upper airways using principal curvatures is proposed. To achieve this goal it was necessary to improve several elements within the set of algorithms required for this purpose. In addition, it was identified that the resolution of the CT images is fundamental for the algorithms of contrast enhancement, smoothing and closure of gaps are not affected significantly.

Image enhancement is an important technique in the field of image processing, which is useful for different applications. To efficiently enhance images, an algorithm using multi-scale top-hat was applied showing very good results in CT images. In addition, the inclusion of the proposed stopping criterion allowed this process to be carried out automatically. In comparison with the existing criteria, it presents a greater stability since the curve is attenuated consistently looking for its stable state. Therefore, it will always stop regardless of the image characteristics (noise or blur). It was tested for both 2D images and 3D images and always generated better results compared to Joao, Gambaruto, Tiago, and Sequeira proposal. A disadvantage that the stopping algorithm presents is the time required for its calculation, but this can be improved by using a region of interest to determine the image quality. Another disadvantage is the space required for processing, so a possible solution is the inclusion of parallel processing or including a pixelwise computation to all steps defined in the criterion.

The selected nonlinear diffusion algorithms allowed to define that the edge information is preserved in a better way using the cEED algorithm. Isotropic algorithm also preserves the edges but in the internal regions of the structures does not perform a good smoothing. The CED and cCED algorithms do not properly preserve the edges and generate edges continuity incorrectly.

NMS proposed algorithm that use the three selection criteria enables to get good connected edge points in the 3D case. The probability of gaps is greatly reduced since a greater number of voxels is maintained. The linking algorithm allows to discard the points that do not belong

to the surface and leave a thinner contour that will be processed by the segmentation algorithm.

Proposed linking algorithm generates good results, however there are gaps in the contours. The gaps are reduced by the algorithm when using the distance criterion between the contiguous maxima founded.

A novel approach to image segmentation using principal curvatures in two-dimensional and three-dimensional images is proposed. Individual points are extracted by using its gradient magnitude. Shape discrimination using Hessian analysis is considered into the detection response measure. The partial derivatives used in the Hessian matrix are obtained by convolutions with a polynomial approximation using Savitzky-Golay technique. This result is a single scale-space description of the ridge position. For 3D ridges, the determination of the ridge position can easily be extended from the 2D algorithm.

The generated results by the proposed algorithm to segment the upper airways are better than those obtained with the active contour and Chan-Vese methods.

The principal curvatures generate more information than that obtained when using the gradient, allowing a better analysis of the behavior of the surfaces. Specifically, the eigenvalues and eigenvectors allow a more precise characterization of the surface.

# References

- [1] R. Wolk, A. S. Shamsuzzaman, and V. K. Somers, “Obesity, Sleep Apnea, and Hypertension,” *Hypertension*, vol. 42, no. 6, pp. 1067–1074, 2003.
- [2] Rice-University, *Anatomy and Physiology*. OpenStax College - Rice University, 2013.
- [3] M. B. Singh and R. Kant, “Shape- and size-dependent electronic capacitance in nanostructured materials,” *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 469, no. 2158, p. 20130163, 2013. [Online]. Available: <http://rspa.royalsocietypublishing.org/content/469/2158/20130163>
- [4] L. G. Shapiro, J. D. Moriarty, P. G. Mulgaonkar, and R. M. Haralick, “Sticks, Plates, and Blobs: A Three-Dimensional Object Representation for Scene Analysis,” in *American Association for Artificial Intelligence*, 1980, pp. 28–30. [Online]. Available: <http://www.aaai.org/Papers/AAAI/1980/AAAI80-008.pdf>
- [5] O. Merveille, “RORPO: A morphological framework for curvilinear structure analysis,” Ph.D. dissertation, Université Paris, 2016.
- [6] A. F. Korn, “Toward a Symbolic Representation of Intensity Changes in Images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 5, pp. 610–625, 1988.
- [7] F. Devernay, “A Non-Maxima Suppression Method for Edge Detection with Sub-Pixel Accuracy,” institut National De Recherche En Informatique Et En Automatique, Sophia Antipolis Cedex (France), Tech. Rep., 1995. [Online]. Available: <http://hal.archives-ouvertes.fr/inria-00073970/>
- [8] C. Sun and P. Vallotton, “Fast linear feature detection using multiple directional non-maximum suppression,” *Journal of Microscopy*, vol. 234, no. 2, pp. 147–157, 2009.

- [9] M. Hafner, M. Stepanek, J. Taylor, W. M. Troxel, and C. V. Stolk, "Why sleep matters-the economic costs of insufficient sleep: A cross-country comparative analysis," RAND Corporation, Tech. Rep., 2016. [Online]. Available: [http://www.rand.org/pubs/research\[\\_\]reports/RR1791.html](http://www.rand.org/pubs/research[_]reports/RR1791.html)
- [10] R. M. Haralick, L. T. Watson, and T. J. Laffey, "The Topographic Primal Sketch," *The International Journal of Robotics Research*, vol. 2, no. 1, pp. 50–72, 1983.
- [11] J. Toriwaki and H. Yoshida, *Fundamentals of Three-Dimensional Digital Image Processing*. London: Springer-Verlag, 2009.
- [12] Y. Hirano, A. Shimizu, J.-I. Hasegawa, and J.-I. Toriwaki, "A tracking algorithm for extracting ridge lines in three-dimensional gray images using curvature of four-dimensional hypersurface," *Systems and Computers in Japan*, vol. 32, no. 12, pp. 25–37, 2001.
- [13] T. McInerney and D. Terzopoulos, "Deformable models in medical image analysis: A survey," *Medical Image Analysis*, vol. 1, no. 2, pp. 91–108, 1996. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1361841596800077>
- [14] V. Pekar, J. Kim, and D. A. Jaffray, "Head and Neck Auto-segmentation Challenge," *The MIDAS Journal - Head and Neck Auto-Segmentation Challenge.*, pp. 1–9, 2009.
- [15] J. S. Suri, D. L. Wilson, and S. Laxminarayan, *Handbook of Biomedical Image Analysis, Vol I: Segmentation Models*. Kluwer Academic / Plenum Publishers, 2005.
- [16] S. K. Chhabra and P. Chhabra, "Estimating prevalence of chronic obstructive pulmonary disease: from questionnaires to spirometry." *The Indian journal of chest diseases & allied sciences*, vol. 54, no. 3, pp. 155–8, 2012. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/23008920>
- [17] Global Initiative for Chronic Obstructive Lung Disease, "Global strategy for the diagnosis, management, and prevention of chronic obstructive pulmonary disease (Revised 2011)," Tech. Rep., 2011.
- [18] P. M. Boiselle and D. A. Lynch, *CT of the Airways*. Humana Press, 2008.
- [19] J. S. Suri, S. K. Setarehdan, and S. Singh, *Advanced Algorithmic Approaches to Medical Image Segmentation*. Springer, mar 2002, vol. 26, no. 4.

- [20] P. Lenkiewicz, M. Pereira, M. Freire, and J. Fernandes, "Techniques for Medical Image Segmentation: Review of the Most Popular Approaches," in *Biomedical Diagnostics and Clinical Technologies: Applying High-Performance Cluster and Grid Computing*, M. Pereira and M. Freire, Eds., 2011, ch. 1, pp. 1–3.
- [21] C. A. Kushida, *Handbook of sleep disorders*, second ed. ed. Informa healthcare USA, Inc., 2009, vol. 55.
- [22] ———, *Obstructive Sleep Apnea: Diagnosis and Treatment*, C. A. Kushida, Ed. New York: Informa healthcare USA, Inc., 2007.
- [23] H. Shi, W. C. Scarfe, and A. G. Farman, "Upper airway segmentation and dimensions estimation from cone-beam CT image datasets," *International Journal of Computer Assisted Radiology and Surgery*, vol. 1, no. 3, pp. 177–186, 2006.
- [24] H. R. Colten and B. M. Altevogt, *Sleep Disorders and Sleep Deprivation: An Unmet Public Health Problem*, H. R. Colten and B. M. Altevogt, Eds. National Academy of Sciences, 2006. [Online]. Available: <http://books.google.com/books?hl=en{&}lr={&}id=KexmhhRu84MC{&}oi=fnd{&}pg=PA1{&}dq=Sleep+disorders+and+sleep+deprivation:+an+unmet+public+health+problem{&}ots=w3UIttU4bn{&}sig=V6XaggAZKg{&}Viu7ZGTGF9Sf{&}SCA{&}5Cnhttp://linkinghub.elsevier.com/ret>
- [25] J. Rivlin, V. Hoffstein, J. Kalbfleisch, W. McNichols, N. Zamel, and A. Bryan, "Upper airway morphology in patients with idiopathic obstructive sleep apnea," *American Review Respiratory Diseases*, vol. 129, pp. 355–360, 1984.
- [26] D. R. Hillman, A. S. Murphy, and L. Pezzullo, "The economic cost of sleep disorders." *Sleep*, vol. 29, no. 3, pp. 299–305, 2006.
- [27] G. C. Barkdull, C. A. Kohl, M. Patel, and T. M. Davidson, "Computed Tomography Imaging of Patients With Obstructive Sleep Apnea," *The Laryngoscope*, vol. 118, no. 8, pp. 1486–1492, 2008. [Online]. Available: <http://doi.wiley.com/10.1097/MLG.0b013e3181782706>
- [28] S. A. Little, M. W. Sproule, M. D. Cowan, K. J. Macleod, M. Robertson, J. G. Love, G. W. Chalmers, C. P. McSharry, and N. C. Thomson, "High resolution computed tomographic assessment of airway wall thickness in chronic asthma: reproducibility and relationship with lung function and severity." *Thorax*, vol. 57, no. 3, pp. 247–253, 2002.
- [29] J. C. Hogg, F. Chu, S. Utokaparch, R. Woods, W. M. Elliott, L. Buzatu, R. M. Cherniack, R. M. Rogers, F. C. Sciurba, H. O. Coxson, and P. D. Paré, "The Nature of

- Small-Airway Obstruction in Chronic Obstructive Pulmonary Disease,” *The New England journal of medicine*, vol. 350, no. 26, pp. 2645–2653, 2004.
- [30] F. R. Long, R. S. Williams, and R. G. Castile, “Structural airway abnormalities in infants and young children with cystic fibrosis.” *The Journal of Pediatrics*, vol. 144, no. 2, pp. 154–161, 2004.
- [31] D. Aykac, E. A. Hoffman, G. McLennan, and J. M. Reinhardt, “Segmentation and analysis of the human airway tree from three-dimensional X-ray CT images.” *IEEE transactions on medical imaging*, vol. 22, no. 8, pp. 940–950, aug 2003. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/12906248>
- [32] D. Bartz, D. Mayer, J. Fischer, S. Ley, A. Del Río, S. Thust, C. P. Heussel, H. U. Kauczor, and W. Straßer, “Hybrid Segmentation and Exploration of the Human Lungs,” *Proceedings of the IEEE Visualization Conference*, pp. 177–184, 2003.
- [33] M. W. Graham, J. D. Gibbs, D. C. Cornish, and W. E. Higgins, “Robust 3-D airway tree segmentation for image-guided peripheral bronchoscopy,” *IEEE Transactions on Medical Imaging*, vol. 29, no. 4, pp. 982–997, 2010.
- [34] W. Park, E. Hoffman, and M. Sonka, “Segmentation of intrathoracic airway trees: a fuzzy logic approach.” *IEEE transactions on medical imaging*, vol. 17, no. 4, pp. 489–97, aug 1998. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/9845305>
- [35] A. P. Kiraly, W. E. Higgins, G. McLennan, E. a. Hoffman, and J. M. Reinhardt, “Three-dimensional Human Airway Segmentation Methods for Clinical Virtual Bronchoscopy,” *Academic Radiology*, vol. 9, pp. 1153–1169, 2002.
- [36] A. P. Kiraly, J. P. Helferty, E. a. Hoffman, G. McLennan, and W. E. Higgins, “Three-dimensional path planning for virtual bronchoscopy,” *IEEE Transactions on Medical Imaging*, vol. 23, no. 11, pp. 1365–1379, 2004.
- [37] A. Fabijańska, “Two-pass region growing algorithm for segmenting airway tree from MDCT chest scans,” *Computerized Medical Imaging and Graphics*, vol. 33, no. 7, pp. 537–546, 2009.
- [38] C. I. Fetita, F. Prêteux, C. Beigelman-Aubry, and P. Grenier, “Pulmonary airways: 3-D reconstruction from multislice CT and clinical investigation,” *IEEE Transactions on Medical Imaging*, vol. 23, no. 11, pp. 1353–1364, 2004.



- [39] T. Kitasaka, K. Mori, J.-i. Hasegawa, and J.-i. Toriwaki, "A Method for Extraction of Bronchus Regions from 3D Chest X-ray CT Images by Analyzing Structural Features of the Bronchus," *Chest*, vol. 17, pp. 321–338, 2002. [Online]. Available: <http://www.scipress.org/journals/forma/pdf/1704/17040321.pdf>
- [40] T. Kitasaka, H. Yano, M. Feuerstein, and K. Mori, "Bronchial region extraction from 3D chest CT image by voxel classification based on local intensity structure," in *Third International Workshop on Pulmonary Image Analysis*, 2010, pp. 21–29.
- [41] D. Mayer, D. Bartz, S. Ley, S. Thust, C. P. Heussel, H.-U. Kauczor, and W. Straßer, "Segmentation and virtual exploration of tracheobronchial trees," *International Congress Series*, vol. 1256, pp. 35–40, 2003.
- [42] J. Tschirren, E. a. Huffman, G. McLennan, and M. Sonka, "Intrathoracic airway trees: Segmentation and airway morphology analysis from low-dose CT scans," *IEEE Transactions on Medical Imaging*, vol. 24, no. 12, pp. 1529–1539, 2005.
- [43] P. Lo, "Segmentation of Lung Structures in CT," Ph.D. dissertation, University of Copenhagen, 2010. [Online]. Available: <http://diku.dk/forskning/phd-studiet/phd/Thesis.pdf{ }Pechin/>
- [44] J. Pu, C. Fuhrman, W. F. Good, F. C. Sciurba, and D. Gur, "A Differential Geometric Approach to Automated Segmentation of Human Airway Tree," *IEEE transactions on medical imaging*, vol. 30, no. 2, pp. 266–278, 2011.
- [45] R. D. Rudyanto, S. Kerkstra, E. M. van Rikxoort, C. Fetita, P.-Y. Brillet, C. Lefevre, W. Xue, X. Zhu, J. Liang, Ä. Öksüz, D. Ünay, K. Kadipasaoglu, R. S. José Estépar, J. C. Ross, G. R. Washko, J.-C. Prieto, M. Hernández Hoyos, M. Orkisz, H. Meine, M. Hüllebrand, C. Stöcker, F. L. Mir, V. Naranjo, E. Villanueva, M. Staring, C. Xiao, B. C. Stoel, A. Fabijanska, E. Smistad, A. C. Elster, F. Lindseth, A. H. Foruzan, R. Kiros, K. Popuri, D. Cobzas, D. Jimenez-Carretero, A. Santos, M. J. Ledesma-Carbayo, M. Helmberger, M. Urschler, M. Pienn, D. G. Bosboom, A. Campo, M. Prokop, P. a. de Jong, C. Ortiz-de Solorzano, A. Muñoz-Barrutia, and B. V. Ginneken, "Comparing algorithms for automated vessel segmentation in computed tomography scans of the lung: The VESSEL12 study," *Medical Image Analysis*, vol. 18, pp. 1217–1232, 2014. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S136184151400111X>
- [46] A. Huang, G. M. Nielson, A. Razdan, G. E. Farin, D. P. Baluch, and D. G. Capco, "Thin structure segmentation and visualization in three-

- dimensional biomedical images: a shape-based approach,” *IEEE Trans Vis Comput Graph*, vol. 12, no. 1, pp. 93–102, 2006. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/16382611>{%}5Cn<http://ieeexplore.ieee.org/ielx5/2945/32930/01542003.pdf?tp={&}arnumber=1542003{&}isnumber=32930>
- [47] J. S. Suri, D. L. Wilson, and S. Laxminarayan, *Handbook of Biomedical Image Analysis Vol2*, J. S. Suri, D. L. Wilson, and S. Laxminarayan, Eds. Kluwer Academic / Plenum Publishers, 2005.
- [48] D. L. Pham, C. Xu, and J. L. Prince, “Current Methods in Medical Image Segmentation,” *Annual Review of Biomedical Engineering*, vol. 2, pp. 315–337, 2000.
- [49] Y. Katou and T. Shinkai, “Respiratory system,” *Cancer & chemotherapy*, vol. 35, no. 13, pp. 2311–2315, 2008.
- [50] J. Tu, K. Inthavong, and G. Ahmadi, “The Human Respiratory System,” in *Computational Fluid and Particle Dynamics in the Human Respiratory System*. Springer Science Business Media, 2013. [Online]. Available: <http://link.springer.com/10.1007/978-94-007-4488-2>
- [51] A. Person and M. Mintz, “Anatomy and physiology of the respiratory tract,” in *Disorders of the Respiratory Tract*. Springer-Verlag, 2006, pp. 11–15.
- [52] L. Barnes, J. W. Eveson, P. Reichart, and D. Sidransky, *Pathology and Genetics of Head and Neck Tumours*. World Health Organization, 2005.
- [53] R. M.R. and J. Paul, “Applied anatomy and physiology of the airway and breathing,” *Indian J. Anaesth*, vol. 49, no. 4, pp. 251–256, 2005. [Online]. Available: <http://discovery.ucl.ac.uk/33955/>
- [54] A. Sahin-Yilmaz and R. M. Naclerio, “Anatomy and physiology of the upper airway,” *Proceedings of the American Thoracic Society*, vol. 8, no. 1, pp. 31–39, 2011.
- [55] S. Grützenmacher, D. M. Robinson, K. Gräfe, C. Lang, and G. Mlynski, “First findings concerning airflow in noses with septal deviation and compensatory turbinate hypertrophy - A model study,” *Orl*, vol. 68, no. 4, pp. 199–205, 2006.
- [56] R. Mladina, E. Cujčić, M. Subarić, and K. Vuković, “Nasal septal deformities in ear, nose, and throat patients: an international study,” *American Journal of Otolaryngology*, vol. 29, no. 2, pp. 75–82, 2008.

- [57] J. Yataco and A. Mehta, "Upper Airway Obstruction," in *ACP manual of critical care*, S. Raof, Ed. McGraw-Hill Medical, 2009, ch. 37, pp. 388–397. [Online]. Available: <https://store.acponline.org/ebizatpro/images/ProductImages/books/samplechapters/CCM37.pdf>
- [58] C. Xu, S. Sin, J. M. McDonough, J. K. Udupa, A. Guez, R. Arens, and D. M. Wootton, "Computational fluid dynamics modeling of the upper airway of children with obstructive sleep apnea syndrome in steady flow," *Journal of Biomechanics*, vol. 39, no. 11, pp. 2043–2054, 2006.
- [59] C. Van Holsbeke, W. Vos, K. Van Hoorenbeeck, A. Boudewyns, R. Salgado, P. R. Verdonck, J. Ramet, J. De Backer, W. De Backer, and S. L. Verhulst, "Functional respiratory imaging as a tool to assess upper airway patency in children with obstructive sleep apnea," *Sleep Medicine*, vol. 14, no. 5, pp. 433–439, 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.sleep.2012.12.005>
- [60] Y. Li, N. Lin, J. Ye, Q. Chang, D. Han, and A. Sperry, "Upper Airway Fat Tissue Distribution in Subjects With Obstructive Sleep Apnea and Its Effect on Retropalatal Mechanical Loads," *Respiratory Care*, vol. 57, no. 7, pp. 1098–1105, 2012.
- [61] S. Jindal, *Textbook of pulmonary and critical care medicine Voll.* Jaypee Brothers Medical Publishers, 2011.
- [62] V. Burdzuis, "Effect of Orthognathic Surgery on the Upper Airway System," Ph.D. dissertation, Tampere University of Technology, 2013.
- [63] M. Kohler and J. R. Stradling, "Mechanisms of vascular damage in obstructive sleep apnea." *Nature reviews. Cardiology*, vol. 7, no. 12, pp. 677–685, 2010. [Online]. Available: <http://dx.doi.org/10.1038/nrcardio.2010.145>
- [64] F. H. S. Kuniyoshi, S. Pusalavidyasagar, P. Singh, and V. K. Somers, "Cardiovascular consequences of obstructive sleep apnoea," *Indian Journal of Medical Research*, vol. 131, no. 2, pp. 196–205, 2010.
- [65] P. J. Strollo, R. J. Soose, J. T. Maurer, N. de Vries, J. Cornelius, O. Froymovich, R. D. Hanson, T. a. Padhya, D. L. Steward, M. B. Gillespie, B. T. Woodson, P. H. Van de Heyning, M. G. Goetting, O. M. Vanderveken, N. Feldman, L. Knaack, and K. P. Strohl, "Upper-airway stimulation for obstructive sleep apnea." *The New England journal of medicine*, vol. 370, no. 2, pp. 139–49, 2014. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/24401051>

- [66] F. Zhao and X. Xie, “An Overview of Interactive Medical Image Segmentation,” *Annals of the BMVA*, vol. 2013, no. 7, pp. 1–22, 2013. [Online]. Available: <http://www.bmva.org/annals/2013/2013-0007.pdf>
- [67] R. Gonzalez and R. Woods, *Digital image processing*, 3rd ed. Pearson Education, 2002. [Online]. Available: <http://mirror.klaus-uwe.me/ctan/biblio/bibtex/contrib/persian-bib/Persian-bib-userguide.pdf>{%}5Cn<http://ftp.neu6.edu.cn/mirrors/CTAN/biblio/bibtex/contrib/persian-bib/Persian-bib-userguide.pdf>
- [68] D. Lesage, E. D. Angelini, I. Bloch, and G. Funka-Lea, “A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes,” *Medical Image Analysis*, vol. 13, no. 6, pp. 819–845, 2009. [Online]. Available: <http://dx.doi.org/10.1016/j.media.2009.07.011>
- [69] C. T. Leondes, *Medical Imaging Systems Technology: Methods in general anatomy*, C. Leondes, Ed. World Scientific Publishing Co., 2005.
- [70] C. Kirbas and F. Quek, “A Review of Vessel Extraction Techniques and Algorithms,” *ACM Computing Surveys*, vol. 36, no. 2, pp. 81–121, 2004.
- [71] B. M. ter Haar Romeny, *Front-End Vision and Multi-Scale Image Analysis*. Springer, 2003.
- [72] W. K. Pratt, *Digital Image Processing*, third edit ed. John Wiley & Sons Inc., 2001, vol. 5.
- [73] Y. Sato, S. Nakajima, N. Shiraga, H. Atsumi, S. Yoshida, T. Koller, G. Gerig, and R. Kikinis, “Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images,” *Medical Image Analysis*, vol. 2, no. 2, pp. 143–168, 1998. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S1361841598800091>
- [74] A. Rosenfeld and A. C. Kak, “Segmentation,” in *Digital Picture Processing*, second-vol ed. Academic Press, 1982, ch. 10-Segment, pp. 55–190. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/B9780125973021500098>
- [75] W. E. Higgins, J. P. Helferty, K. Lu, S. a. Merritt, L. Rai, and K. C. Yu, “3D CT-Video Fusion for Image-Guided Bronchoscopy,” *Computerized Medical Imaging and Graphics*, vol. 32, no. 3, pp. 159–173, 2008.

- [76] Y. Li, S. Wang, Q. Tian, and X. Ding, "A survey of recent advances in visual feature detection," *Neurocomputing*, vol. 149, no. PB, pp. 736–751, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0925231214010121>
- [77] S. Tripathi, K. Kumar, B. Singh, and R. Singh, "Image Segmentation: A Review," *International Journal of Computer Science and Management Research*, vol. 1, no. 4, pp. 838–843, 2012.
- [78] X. Huang and G. Tsechpenakis, "Medical image segmentation," in *Information Discovery on Electronic Health Records*, V. Hristidis, Ed. Chapman & Hall, 2009, ch. 10, pp. 1–35. [Online]. Available: <http://journals.indexcopernicus.com/abstracted.php?level=5&icid=936830>
- [79] Z. Ma, J. M. Tavares, and R. Natal Jorge, "A Review on the Current Segmentation Algorithms for Medical Images," in *Proceedings of the First International Conference on Computer Imaging Theory and Applications (IMAGAPP 2009)*, Lisboa, Portugal, 2009. [Online]. Available: <http://repositorio-aberto.up.pt/handle/10216/7125>
- [80] R. Whitaker, D. Breen, K. Museth, and N. Soni, "A framework for level set segmentation of volume datasets," *Volume Graphics*, vol. D, pp. 159–68, 2001. [Online]. Available: [http://www.cs.drexel.edu/~david/Papers/david\[\\_\]VG01.pdfhttp://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:A+Framework+for+Level+Set+Segmentation+of+Volume+Datasets{#}0](http://www.cs.drexel.edu/~david/Papers/david[_]VG01.pdfhttp://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:A+Framework+for+Level+Set+Segmentation+of+Volume+Datasets{#}0)
- [81] D. L. Pham, C. Xu, and J. L. Prince, "A Survey of Current Methods in Medical Image Segmentation," John Hopkins University, Baltimore, Tech. Rep., 1998.
- [82] R. Malladi, J. A. Sethian, and B. C. Vemuri, "A Fast Level Set Based Algorithm for Topology-Independent shape Modeling," *Journal of Mathematical Imaging and Vision*, vol. 01, no. 0, pp. 269–290, 1996. [Online]. Available: <http://link.springer.com/article/10.1007/BF00119843>
- [83] I. Cheng, S. Nilufar, C. Flores-Mir, and A. Basu, "Airway segmentation and measurement in CT images," *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, pp. 795–799, 2007.
- [84] G. R. Vineetha and G. Darshan, "Level Set Method for Image Segmentation: A Survey," *IOSR Journal of Computer Engineering*, vol. 8, no. 6, pp. 74–78, 2013.
- [85] K. Mori, J. Hasegawa, J. Toriwaki, H. Anno, and K. Katada, "Recognition of bronchus in three-dimensional X-ray CT images with applications to virtualized bronchoscopy

- system,” *Proceedings - International Conference on Pattern Recognition*, vol. 3, pp. 528–532, 1996.
- [86] M. Nakamura, S. Wada, T. Miki, Y. Shimada, Y. Suda, and G. Tamura, “Automated segmentation and morphometric analysis of the human airway tree from multidetector CT images.” *The journal of physiological sciences*, vol. 58, no. 7, pp. 493–498, 2008.
- [87] P. Lo, B. van Ginneken, J. M. Reinhardt, T. Yavarna, P. a. de Jong, B. Irving, C. Fetita, M. Ortner, R. Pinho, J. Sijbers, M. Feuerstein, A. Fabijańska, C. Bauer, R. Beichel, C. S. Mendoza, R. Wiemker, J. Lee, A. P. Reeves, S. Born, O. Weinheimer, E. M. van Rikxoort, J. Tschirren, K. Mori, B. Odry, D. P. Naidich, I. Hartmann, E. a. Hoffman, M. Prokop, J. H. Pedersen, and M. de Bruijne, “Extraction of airways from CT (EXACT’09).” *IEEE transactions on medical imaging*, vol. 1, no. 1, pp. 2093–2107, nov 2009. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/22855226>
- [88] J. Pu, S. Gu, S. Liu, S. Zhu, D. Wilson, J. M. Siegfried, and D. Gur, “CT based computerized identification and analysis of human airways: A review,” *Medical Physics*, vol. 39, no. 5, p. 2603, 2012.
- [89] M. do Carmo, *Differential Geometry of Curves and Surfaces*. Prentice Hall, 1976.
- [90] O. Monga and S. Benayoun, “Using Differential Geometry in R4 to extract typical features in 3D images,” pp. 684–685, 1993.
- [91] B. Avants and J. Gee, “The shape operator for differential analysis of images.” *Information processing in medical imaging : proceedings of the ... conference*, vol. 18, pp. 101–113, 2003. [Online]. Available: [http://link.springer.com/chapter/10.1007/978-3-540-45087-0\\_9](http://link.springer.com/chapter/10.1007/978-3-540-45087-0_9)  
[https://static.aminer.org/pdf/PDF/000/407/121/the\\_shape\\_operator\\_for\\_differential\\_analysis\\_of\\_images.pdf](https://static.aminer.org/pdf/PDF/000/407/121/the_shape_operator_for_differential_analysis_of_images.pdf)
- [92] A. Gray, *Modern Differential Geometry of Curves and Surfaces with Mathematica*, 3rd ed. Chapman & Hall / CRC Press, 2006.
- [93] X. Yang and J. Zheng, “Curvature tensor computation by piecewise surface interpolation,” *CAD Computer Aided Design*, vol. 45, pp. 1639–1650, 2013.
- [94] P. J. Besl and R. C. Jain, “Invariant surface characteristics for 3D object recognition in range images,” *Computer Vision, Graphics, and Image Processing*, vol. 33, no. 1, pp. 33–80, 1986.

- [95] ———, “Three-dimensional object recognition,” *Computing Survey*, vol. 17, no. 1, pp. 75–145, 1985.
- [96] L. R. Seidenberg, R. B. Jerard, D. Nh, and J. Magewick, “Surface Curvature Analysis Using Color Body and Chassis Engineering Ford Motor Company,” *Proceedings of IEEE Conference on Visualization, 1992*, pp. 260–269, 1992.
- [97] D. N. Dovencioglu, M. W. Wijntjes, O. Ben-Shahar, and K. Doerschner, “Effects of surface reflectance on local second order shape estimation in dynamic scenes,” *Vision Research*, vol. 115, pp. 218–230, 2011. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0042698915000231>
- [98] B. Rieger, F. J. Timmermans, and L. J. V. Vliet, “Estimation of curvature on surfaces in 3D grey-value images,” in *ASCI 2002, 8th annual conf. of the advanced school for computing and imaging*, 2002, pp. 1–8.
- [99] A. Roberts, “Curvature attributes and their application to 3D interpreted horizons,” *First Break*, vol. 19, no. 2, pp. 85–100, 2001.
- [100] R. M. Haralick, “Ridges and Valleys on Digital Images,” *Computer Vision, Graphics, and Image Processing*, vol. 22, pp. 28–38, 1983.
- [101] G. Taubin, “Estimating the tensor of curvature of a surface from a polyhedral approximation,” *Proceedings of IEEE International Conference on Computer Vision*, pp. 902–907, 1995. [Online]. Available: <http://ieeexplore.ieee.org/document/466840/>
- [102] O. Monga and S. Benayoun, “Using partial Derivatives of 3D images to extract typical surface features,” Tech. Rep., 1992.
- [103] O. Monga, S. Benayoun, and O. D. Faugeras, “From Partial Derivatives of 3D density images to ridge lines,” *IEEE*, pp. 354–359, 1992.
- [104] O. Monga and S. Benayoun, “Using partial derivatives Of 3D images to extract typical surface features,” *Computer Vision and Image Understanding*, vol. 61, no. 2, pp. 171–189, 1995.
- [105] T. Koller, G. Gerig, G. Szekely, and D. Dettwiler, “Multiscale detection of curvilinear structures in 2-D and 3-D image data,” in *Proceedings of IEEE International Conference on Computer Vision*, 1995, pp. 864–869.



- [106] Y. Sato, C. F. Westin, A. Bhalerao, S. Nakajima, N. Shiraga, S. Tamura, and R. Kikinis, "Tissue classification based on 3D local intensity structures for volume rendering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, no. 2, pp. 160–180, 2000.
- [107] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever, "Multiscale vessel enhancement filtering," *Medical Image Computing and Computer-Assisted Intervention - MICCAI'98. Lecture Notes in Computer Science, vol 1496*, vol. 1496, pp. 130–137, 1998.
- [108] A. F. Frangi, W. J. Niessen, R. M. Hoogeveen, T. van Walsum, and M. A. Viergever, "Model-based quantitation of 3-D magnetic resonance angiographic images." *IEEE transactions on medical imaging*, vol. 18, no. 10, pp. 946–56, oct 1999. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/10628954>
- [109] R. Firoz, M. Shahjahan Ali, M. Nasir Uddin Khan, M. Khalid Hosain, M. Khairul Islam, and M. Shahinuzzaman, "Medical Image Enhancement Using Morphological Transformation," *Journal of Data Analysis and Information Processing*, vol. 4, no. 4, pp. 1–12, 2016. [Online]. Available: <http://www.scirp.org/journal/jdaip>{%}5Cn<http://dx.doi.org/10.4236/jdaip.2016.41001>{%}5Cn<http://creativecommons.org/licenses/by/4.0/>
- [110] K. K. Rajkumar and G. Raju, "Enhancement of Mammograms Using Tophat Filtering and Wavelet Decomposition," *J. Comp. & Math. Sci.*, vol. 2, no. 6, pp. 812–818, 2011.
- [111] P. Maragos, "Morphological Filtering for image enhancement and feature detection," in *The Image and Video Processing Handbook (2nd ed.)*, 2nd ed., A. C. Bovik, Ed. Elsevier Academic Press, 2005, ch. 3, pp. 135–156.
- [112] H. Hassanpour, N. Samadiani, and S. M. Mahdi Salehi, "Using morphological transforms to enhance the contrast of medical images," *Egyptian Journal of Radiology and Nuclear Medicine*, vol. 46, no. 2, pp. 481–489, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.ejrn.2015.01.004>
- [113] J. Wang and Y. Tan, "Morphological image enhancement procedure design by using genetic programming," in *GECCO '11: Proceedings of the 13th annual conference on Genetic and evolutionary computation*, 2011, pp. 1435–1442. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2001769>



- [114] Sos S. Aghaian, K. Panetta, and A. M. Grigoryan, "A new measure of image enhancement," in *IASTED International Conference on Signal Processing & Communication*, 2000, pp. 19–22.
- [115] S. S. Aghaian, B. Silver, and K. A. Panetta, "Transform coefficient histogram-based image enhancement algorithms using contrast entropy," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 741–758, 2007.
- [116] G. Gilboa, N. Sochen, and Y. Y. Zeevi, "Image enhancement and denoising by complex diffusion processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1020–1036, 2004.
- [117] S. Mukhopadhyay and B. Chanda, "Local Contrast Enhancement of Grayscale Images Using Multiscale Morphology," in *ICVGIP*, 2000, pp. 1–8.
- [118] Ritika, "A Novel Approach for Local Contrast Enhancement of Medical Images using Mathematical Morphology," *International Journal of Computer Science and Information Technology & Security*, vol. 2, no. 2, pp. 392–397, 2012.
- [119] Y. Kimori, "Morphological image processing for quantitative shape analysis of biomedical structures: Effective contrast enhancement," *Journal of Synchrotron Radiation*, vol. 20, no. 6, pp. 848–853, 2013.
- [120] A. Shinkar and P. Devale, "Contrast Enhancement Technique for Medical Images," in *Proceedings NCSPA-07*, 2007, pp. 303–306.
- [121] D. J. Jobson, Z. U. Rahman, and G. A. Woodell, "Properties and performance of a center/surround retinex," *IEEE Transactions on Image Processing*, vol. 6, no. 3, pp. 451–462, 1997.
- [122] ———, "A multiscale retinex for bridging the gap between color images and the human observation of scenes," *IEEE Transactions on Image Processing*, vol. 6, no. 7, pp. 965–976, 1997.
- [123] G. Matheron, "Random sets theory and its applications to stereology," *J. Microscopy*, vol. 95, no. 1, pp. 15–23, 1972.
- [124] J. P. Serra, *Image Analysis and Mathematical Morphology*. Academic Press, 1982, vol. 1, no. 2.
- [125] K. Preston and M. J. B. Duff, *Modern cellular automata: theory and applications*. Springer Science and Business Media, 1984.

- [126] J. P. Serra, "A lattice approach to image segmentation," *Journal of Mathematical Imaging and Vision*, vol. 24, no. 1, pp. 83–130, 2006.
- [127] T. A. Mahmoud and S. Marshall, "Medical image enhancement using threshold decomposition driven adaptive morphological filter," in *European Signal Processing Conference*, 2008, pp. 2–6.
- [128] P. Soille, *Morphological image analysis: Principles and applications*, 2nd ed. Berlin Heidelberg: Springer-Verlag Berlin Heidelberg, 2004. [Online]. Available: [http://www.springerlink.com/index/10.1007/978-3-662-05088-0\\_{\\_}1](http://www.springerlink.com/index/10.1007/978-3-662-05088-0_{_}1)
- [129] M. Pesaresi and J. A. Benediktsson, "A new approach for the morphological segmentation of high-resolution satellite imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 2, pp. 309–320, 2001.
- [130] A. Kamra, V. K. Jain, and Pragya, "Contrast Enhancement of Masses in Mammograms Using Multiscale Morphology," *International Journal of Medical, Health, Biomedical, Bioengineering and Pharmaceutical Engineering*, vol. 9, no. 7, pp. 546–549, 2015.
- [131] T. K. Le, "Segmentation of Lung Vessels Together With Nodules in CT Images Using Morphological Operations and Level Set," *Journal of Medical and Bioengineering*, vol. 2, no. 1, pp. 5–10, 2013. [Online]. Available: <http://www.jomb.org/index.php?m=content{&}c=index{&}a=show{&}catid=34{&}id=31>
- [132] H. S. Jagannath, J. Virmani, and V. Kumar, "Morphological Enhancement of Microcalcifications in Digital Mammograms," *Journal of The Institution of Engineers (India): Series B*, vol. 93, no. 3, pp. 163–172, 2012. [Online]. Available: <http://link.springer.com/10.1007/s40031-012-0020-1>
- [133] X. Bai and F. Zhou, "Multi Scale Top-hat Transform Based Algorithm for Image Enhancement," in *ICSP2010 Proceedings*, 2010, pp. 797–800.
- [134] Y.-P. Wang, Q. Wu, K. R. Castleman, and Z. Xiong, "Chromosome image enhancement using multiscale differential operators," *IEEE Transactions on Medical Imaging*, vol. 22, no. 5, pp. 685–693, 2003. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1207403>
- [135] A. João, A. M. Gambaruto, J. Tiago, and A. Sequeira, "Computational advances applied to medical image processing: an update," *Open Access Bioinformatics*, vol. 8, pp. 1–15, 2016.

- [136] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.
- [137] T. Barbu, "Robust anisotropic diffusion scheme for image noise removal," *Procedia Computer Science*, vol. 35, no. C, pp. 522–530, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.procs.2014.08.133>
- [138] P. Perona and J. Malik, "Scale-Space and Edge Detection Using Anisotropic Diffusion," pp. 629–639, 1990.
- [139] S. L. Keeling and R. Stollberger, "Nonlinear anisotropic diffusion filtering for multi-scale edge enhancement," *Inverse Problems*, vol. 18, no. 1, pp. 175–190, 2002.
- [140] G. Leitmann, "Partial Differential Equations and Finite-Difference Methods in Image Processing, Part 1: Image Representation," *Optimization*, vol. 23, no. September, pp. 65–91, 1977.
- [141] M. Lindenbaum, M. Fischer, and A. Bruckstein, "On Gabor's contribution to image enhancement," *Pattern Recognition*, vol. 27, no. 1, pp. 1–8, 1994.
- [142] J. J. Koenderink, "The structure of images," *Biological Cybernetics*, vol. 50, no. 5, pp. 363–370, 1984.
- [143] A. Witkin, "Scale-space filtering: A new approach to multi-scale description," in *ICASSP '84. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 9, 1983, pp. 150–153. [Online]. Available: <http://ieeexplore.ieee.org/document/1172729/>
- [144] B. M. ter Haar Romeny, *Geometry-Driven Diffusion in Computer Vision*, B. M. t. H. Romeny, Ed. Springer-Science and Business Media, 1994, vol. 1. [Online]. Available: <http://www.springer.com/computer/image+processing/book/978-0-7923-3087-5>
- [145] V. Caselles, J.-M. Morel, G. Sapiro, and A. Tannenbaum, "Introduction to the Special Issue on Partial Differential Equations and Geometry-Driven Diffusion in Image Processing and Analysis," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 269–273, 1998.
- [146] G. Sapiro, *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press, 2006. [Online]. Available: <http://ebooks.cambridge.org/ref/id/CBO9780511626319{%}5Cnhttp://www.cambridge.org/us/academic/subjects/mathematics/differential-and-integral-equations-dynamical-systems-and-co/>

- [geometric-partial-differential-equations-and-image-analysis{% }5Cnfiles/941/Sapiro-2](#)
- [147] G. Aubert and P. Kornprobst, *Editors' Picks*, 2nd ed. Springer Science+Business Media, LLC, 2011, vol. 131, no. 9. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0022202X15354178>
- [148] F. Cao, *Geometric Curve Evolution and Image Processing*, J.-M. Morel, F. Takens, and B. Teissier, Eds. Berlin Heidelberg: Springer-Verlag, 2003. [Online]. Available: <http://books.google.at/books?id=ij6jAjpzSDUC>
- [149] L. Rudin and S. Osher, "Feature-Oriented Image Enhancement with Shock filters, I," California Institute of Technology, Tech. Rep., 1989.
- [150] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," *Proceedings of Conference on Computer Graphics and Interactive Techniques*, pp. 417–424, 2000. [Online]. Available: <http://portal.acm.org/citation.cfm?doid=344779.344972>
- [151] Chunming Li, Chenyang Xu, Changfeng Gui, and M. Fox, "Level Set Evolution without Re-Initialization: A New Variational Formulation," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 430–436. [Online]. Available: <http://ieeexplore.ieee.org/document/1467299/>
- [152] J. Weickert, "Theoretical Foundations of Anisotropic Diffusion in Image Processing," *Computing, Suppl.*, vol. 11, pp. 221–236, 1996.
- [153] ———, *Anisotropic diffusion in image processing*. B.G. Teubner Stuttgart, 1998, vol. 256, no. 3. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.11.7513{% }rep=rep1{% }type=pdf{% }5Cnhttp://www.lpi.tel.uva.es/muitic/pim/docus/anisotropic{% }diffusion.pdf>
- [154] ———, "Coherence-Enhancing Diffusion Filtering," *International Journal of Computer Vision*, vol. 31, no. 2, pp. 111–127, 1999. [Online]. Available: <http://www.springerlink.com/content/q7kr7v15n8437021/>
- [155] C. Bazán, M. Miller, and P. Blomgren, "Structure enhancement diffusion and contour extraction for electron tomography of mitochondria," *Journal of Structural Biology*, vol. 166, no. 2, pp. 144–155, 2010.
- [156] G. Dong, Z. Guo, Z. Zhou, D. Zhang, and B. Wo, "Coherence-enhancing diffusion with the source term," *Applied Mathematical Modelling*, vol. 39, no. 19, pp. 6060–6072, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.apm.2015.01.041>

- [157] J.-M. Mirebeau, J. Fehrenbach, L. Risser, and S. Tobji, “Anisotropic Diffusion in ITK,” *Insight Journal*, pp. 1–9, 2014.
- [158] F. Guichard, L. Moisan, and J.-M. Morel, “A review of P.D.E. models in image processing and image analysis,” *Journal of Physics*, vol. 12, no. 1, pp. 137–154, 2002.
- [159] K. D. Toennies, *Guide to Medical Image Analysis*. Springer, 2012. [Online]. Available: <http://link.springer.com/10.1007/978-1-4471-2751-2>
- [160] J. Weickert and T. Brox, “Diffusion and regularization of vector- and matrix-valued images,” in *Inverse problems, image analysis, and medical imaging*, M. Z. Nashed and O. Scherzer, Eds. American Mathematical Society, 2002, vol. 313, pp. 251–268. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.12.195>
- [161] J. Suri, Dee Wu, J. Gao, S. Singh, and S. Laxminarayan, “A comparison of state-of-the-art diffusion imaging techniques for smoothing medical/non-medical image data,” in *Object recognition supported by user interaction for service robots*, vol. 1, 2002, pp. 508–511. [Online]. Available: <http://ieeexplore.ieee.org/document/1044779/>
- [162] G. Gerig, O. Kubler, R. Kikinis, and F. a. Jolesz, “Nonlinear anisotropic filtering of MRI data,” *IEEE Transactions on Medical Imaging*, vol. 11, no. 2, pp. 221–232, 1992.
- [163] R. A. Redner and H. F. Walker, “Society for Industrial and Applied Mathematics,” *Society for Industrial and Applied Mathematics Review*, vol. 26, no. 2, pp. 195–239, 1984.
- [164] B. B. Kimia and K. Siddiqi, “Geometric Heat Equation and Nonlinear Diffusion of Shapes and Images,” *Ieee*, pp. 113–120, 1994.
- [165] M. J. Black and G. Sapiro, “Edges as Outliers: Anisotropic Smoothing using Local Image Statistics,” *Scale-Space Theories in Computer Vision*, vol. 1682, pp. 259–270, 1999.
- [166] S. R. Arridge and A. Simmons, “Multi-spectral probabilistic diffusion using Bayesian classification,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, B. t. H. Romeny, L. M. J. Florack, J. J. Koenderink, and M. a. Viergever, Eds., vol. 1252. Springer, 1997, pp. 225–235.
- [167] I. Bajla and I. Hollander, “Nonlinear filtering of magnetic resonance tomograms by geometry-driven diffusion,” *Machine Vision and Applications*, vol. 10, pp. 243–255, 1998.

- [168] P. J. Olver, “Generating differential invariants,” *J. Math. Anal. Appl.*, vol. 333, no. 1, pp. 450–471, sep 2007. [Online]. Available: [www.elsevier.com/locate/jmaa](http://www.elsevier.com/locate/jmaa)
- [169] O. Scherzer and J. Weickert, “Relations between regularization and diffusion filtering,” *Journal of Mathematical Imaging and Vision*, vol. 12, no. 1, pp. 43–63, 2000.
- [170] M. Bart and H. Romeny, “Scale-Space Theory for Multiscale Geometric Image Analysis,” *Conf on Visualization in Biomedical Computing VBC’96*, no. September, 1985. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.9204{&}rep=rep1{&}type=pdf>
- [171] M. Nielsen, L. Florack, and R. Deriche, “Regularization, scale-space, and edge detection filters,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1065, no. 1997, pp. 70–81, 1996. [Online]. Available: <http://www.springerlink.com/content/v3q545t67804w645/>
- [172] C. Guo and S. A. Morris, *Engineering cell identity: establishing new gene regulatory and chromatin landscapes*. SIAM: Society for Industrial and Applied Mathematics, 2017, vol. 46, no. 9.
- [173] D. Tschumperlé and R. Deriche, “Vector Valued Image Regularization with PDE’s: A Common Framework for Different Applications,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. April, pp. 506–517, 2005.
- [174] L. Alvarez, P.-L. Lions, and J.-M. Morel, “Image Selective Smoothing and Edge Detection by Nonlinear Diffusion. II,” *SIAM Journal on Numerical Analysis*, vol. 29, no. 3, pp. 845–866, 1992. [Online]. Available: <http://epubs.siam.org/doi/abs/10.1137/0729012{&}5Cnhttp://epubs.siam.org/doi/abs/10.1137/0729052>
- [175] W. Joachin, “A Review of Nonlinear Diffusion Filtering,” in *Scale-Space Theory in Computer Vision Lecture Notes in Computer Science*, B. t. H. Romeny, L. Florack, J. J. Koenderink, and M. Viergever, Eds., 1997, vol. 1252, pp. 3–28.
- [176] W. J. Niessen, K. L. Vincken, J. A. Weickert, and M. A. Viergever, “Nonlinear Multiscale Representations for Image Segmentation,” *Computer Vision and Image Understanding*, vol. 66, no. 2, pp. 233–245, 1997. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S1077314297906147>
- [177] S. M. Chao and D. M. Tsai, “Anisotropic diffusion with generalized diffusion coefficient function for defect detection in low-contrast surface images,” *Pattern*

- Recognition*, vol. 43, no. 5, pp. 1917–1931, 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2009.12.005>
- [178] D. S. Fritsch, “Medial description of gray-scale image structure by gradient-limited diffusion,” *Proceedings of SPIE*, vol. 1808, pp. 105–117, 1992. [Online]. Available: <http://link.aip.org/link/?PSI/1808/105/1{%&}Agg=doi>
- [179] M. Z. Nashed and O. Scherzer, “Least squares and bounded variation regularization with nondifferentiable functionals,” *Numerical Functional Analysis and Optimization*, vol. 19, no. 7-8, pp. 873–901, 1998. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/01630569808816863>
- [180] T. Barbu, “Novel Linear Image Denoising Approach Based on a Modified Gaussian Filter Kernel,” *Numerical Functional Analysis and Optimization*, vol. 33, no. September 2011, pp. 1269–1279, 2012. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/01630563.2012.676588>
- [181] M. Cytowski, “Partial Differential Equations in Image Processing,” in *Signal Processing Conference (EUSIPCO 1998), 9th European*, no. May, 2007, pp. 61–64.
- [182] D. Marr and E. Hildreth, “Theory of Edge Detection,” *Proceedings of the Royal Society B: Biological Sciences*, vol. 207, no. 1167, pp. 187–217, feb 1980. [Online]. Available: <http://rspb.royalsocietypublishing.org/cgi/doi/10.1098/rspb.1980.0020>
- [183] J. Calder and A. Mansouri, “Anisotropic Image Sharpening via Well-Posed Sobolev Gradient Flows,” *SIAM Journal on Mathematical Analysis*, vol. 43, no. 4, pp. 1536–1556, 2011. [Online]. Available: <http://epubs.siam.org/doi/10.1137/100810654>
- [184] G. Cong, “Nonlinear Diffusion,” pp. 403–406, 1996.
- [185] T. Brox, J. Weickert, B. Burgeth, and P. Mr??zek, “Nonlinear structure tensors,” *Image and Vision Computing*, vol. 24, no. 1, pp. 41–55, 2006.
- [186] Y. L. You, W. Xu, A. Tannenbaum, and M. Kaveh, “Behavioral analysis of anisotropic diffusion in image processing,” *IEEE Trans. Im. Proc.*, vol. 5, no. 11, pp. 1539–1553, 1996.
- [187] G. H. Cottet and L. Germain, “Image-Processing Through Reaction Combined With Nonlinear Diffusion,” *Mathematics Of Computation*, vol. 61, no. 204, pp. 659–673, 1993.



- [188] F. Torkamani-Azar and K. E. Tait, "Image recovery using the anisotropic diffusion equation," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 5, no. 11, pp. 1573–8, 1996. [Online]. Available: <http://ieeexplore.ieee.org/xpl/freeabs{ }all.jsp?arnumber=541427>
- [189] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 421–432, 1998.
- [190] J. Monteil and A. Beghdadi, "A new interpretation and improvement of the nonlinear anisotropic diffusion for image enhancement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 9, pp. 940–946, 1999.
- [191] P. Charbonnier, L. Blanc-Fraud, G. Aubert, and M. Barlaud, "Two deterministic half-quadratic regularization algorithms for computed imaging," *Proceedings - International Conference on Image Processing, ICIP*, vol. 2, no. 4, pp. 168–172, 1994.
- [192] S. M. Chao and D. M. Tsai, "An anisotropic diffusion-based defect detection for low-contrast glass substrates," *Image and Vision Computing*, vol. 26, no. 2, pp. 187–200, 2008.
- [193] W. Förstner and E. Gülch, "A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features," *Proceedings of ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, no. October, pp. 281–305, 1987.
- [194] J. Bigun and G. H. Granlund, "Optimal Orientation Detection of Linear Symmetry," in *Proceedings of the IEEE First International Conference on Computer Vision*, vol. 54, 1987, pp. 433–438.
- [195] T. M. Khan, M. A. U. Khan, Y. Kong, and O. Kittaneh, "Stopping criterion for linear anisotropic image diffusion: a fingerprint image enhancement case," *EURASIP Journal on Image and Video Processing*, vol. 2016, no. 1, p. 6, 2016. [Online]. Available: <http://www.scopus.com/inward/record.url?eid=2-s2.0-84957894427{&}partnerID=tZOtx3y1>
- [196] M. Nitzberg and T. Shiota, "Nonlinear Image Filtering with Edge and Corner Enhancement," pp. 826–833, 1992.
- [197] A. M. Mendrik, E. J. Vonken, A. Rutten, M. A. Viergever, and B. Van Ginneken, "Noise reduction in computed tomography scans using 3-D anisotropic hybrid diffusion with



- continuous switch,” *IEEE Transactions on Medical Imaging*, vol. 28, no. 10, pp. 1585–1594, 2009. [Online]. Available: <http://ieeexplore.ieee.org/document/4967958/>
- [198] E. Meijering, W. Niessen, J. Weickert, and M. Viergever, “Diffusion-enhanced visualization and quantification of vascular anomalies in three-dimensional rotational angiography: Results of an in-vitro evaluation,” *Medical Image Analysis*, vol. 6, no. 3, pp. 215–233, 2002.
- [199] J. Weickert and H. Scharr, “A Scheme for Coherence-Enhancing Diffusion Filtering with Optimized Rotation Invariance,” *Journal of Visual Communication and Image Representation*, vol. 13, no. 1-2, pp. 103–118, 2002. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S104732030190495X>
- [200] K. Krissian, G. Malandain, and N. Ayache, “Directional anisotropic diffusion applied to segmentation of vessels in 3D images,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1252, pp. 344–348, 1997.
- [201] F. Beekman and M. Viergever, “Evaluation of fully 3D iterative scatter compensation and post-reconstruction filtering in SPECT,” in *Computational Imaging and Vision 4: Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, P. Grangeat and J.-L. Amans, Eds. Kluwer Academic Publishers, 1996, vol. 4, pp. 163–175.
- [202] C. Bazán and P. Blomgren, “Image Smoothing and Edge Detection by Nonlinear Diffusion and Bilateral Filter,” *Research Report CSRCR*, vol. 21, 2007. [Online]. Available: <http://www.csrc.sdsu.edu/csrc/acsess/reports/ACSESS200806.pdf>
- [203] A. Ilyevsky and E. Turkel, “Stopping criteria for anisotropic PDEs in image processing,” *Journal of Scientific Computing*, vol. 45, no. 1-3, pp. 333–347, 2010.
- [204] J. Sporring and J. Weickert, “Information Measures in Scale Spaces,” *IEEE Trans. Information Theory*, vol. 45, no. 3, pp. 1051–1058, 1999.
- [205] I. Capuzzo Dolcetta and R. Ferretti, “Optimal stopping time formulation of adaptive image filtering,” *Applied Mathematics and Optimization*, vol. 43, no. 3, pp. 245–258, 2001.
- [206] P. Mrázek and M. Navara, “Selection of optimal stopping time for nonlinear diffusion,” *International Journal of Computer Vision*, vol. 52, no. 2–3, pp. 189–203, 2003.

- [207] C. Tsotsios and M. Petrou, "On the choice of the parameters for anisotropic diffusion in image processing," *Pattern Recognition*, vol. 46, no. 5, pp. 1369–1381, 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2012.11.012>
- [208] F. Y. Shih, *Image Processing and Pattern Recognition: Fundamentals and Techniques*. IEEE & Wiley, 2010.
- [209] J. H. Jang and K. S. Hong, "Detection of curvilinear structures and reconstruction of their regions in gray-scale images," *Pattern Recognition*, vol. 35, no. 4, pp. 807–824, 2002.
- [210] H. Knutsson, C. F. Westin, and M. Andersson, "Representing local structure using tensors II," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6688 LNCS, 2011, pp. 545–556.
- [211] C.-F. Westin, "A Tensor Framework for Multidimensional Signal Processing," Ph.D. dissertation, Linköping University, 1994. [Online]. Available: <http://www.diva-portal.org/smash/get/diva2:302457/FULLTEXT01.pdf>
- [212] S. Aylward, E. Bullitt, S. Pizer, and D. Eberly, "Intensity ridge and widths for tubular object segmentation and description," in *Workshop on Mathematical Methods in Biomedical Image Analysis*, 1996, pp. 131–138.
- [213] C.-f. Westin, S. Warfield, A. Bhalerao, L. Mui, J. Richolt, and R. Kikinis, "Tensor Controlled Local Structure Enhancement of CT Images for Bone Segmentation," in *Medical Image Computing and Computer-Assisted Intervention-MICCAI'98*, vol. 1496, 1998, pp. 1205–1212. [Online]. Available: <http://dx.doi.org/>
- [214] M. W. Law and A. C. Chung, "Three dimensional curvilinear structure detection using optimally oriented flux," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5305 LNCS, no. PART 4, pp. 368–382, 2008.
- [215] H. Deng, W. Zhang, E. Mortensen, T. Dietterich, and L. Shapiro, "Principal curvature-based region detector for object recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.
- [216] C. Tomasi and T. Kanade, "Detection and Tracking of Point Features," Carnegie Mellon University, Pittsburgh, Tech. Rep., 1991. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.131.5899{%&}rep=rep1{%&}type=pdf>

- [217] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors," *International Journal of Computer Vision*, vol. 37, no. 2, pp. 151–172, 2000.
- [218] T. Tuytelaars and K. Mikolajczyk, "Local Invariant Feature Detectors: A Survey," *Foundations and Trends® in Computer Graphics and Vision*, vol. 3, no. 3, pp. 177–280, 2007. [Online]. Available: <http://www.nowpublishers.com/article/Details/CGV-017>
- [219] D. Song and D. Tao, "Biologically inspired feature manifold for scene classification," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 174–184, 2010.
- [220] Y. Huang, K. Huang, D. Tao, T. Tan, and X. Li, "Enhanced biologically inspired model for object recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 6, pp. 1668–1680, 2011. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/21768049>
- [221] T. Dickscheid, F. Schindler, and W. Förstner, "Coding images with local features," *International Journal of Computer Vision*, vol. 94, no. 2, pp. 154–174, 2011.
- [222] H. P. Moravec, "Obstacle avoidance and navigation in the real world by a seeing robot rover." Ph.D. dissertation, Carnegie-Mellon University, 1980. [Online]. Available: [https://www.ri.cmu.edu/publication\[\\_\]view.html?pub\[\\_\]id=22](https://www.ri.cmu.edu/publication[_]view.html?pub[_]id=22)
- [223] D. Marr, S. Ullman, and T. Poggio, "Bandpass channels, zero-crossings, and early visual information processing." *Journal of the Optical Society of America*, vol. 69, no. 6, pp. 914–6, 1979. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/490228>
- [224] L. Kitchen and A. Rosenfeld, "Gray-level corner detection," *Pattern Recognition Letters*, vol. 1, no. 2, pp. 95–102, 1982.
- [225] L. Dreschler and H. H. Nagel, "Volumetric model and 3D trajectory of a moving car derived from monocular TV frame sequences of a street scene," *Computer Graphics and Image Processing*, vol. 20, no. 3, pp. 199–228, 1982.
- [226] M. Hassaballah, A. A. Abdelmgeid, and H. A. Alshazly, *Image Feature Detections, Descriptions and Matching*, 2016, vol. 630. [Online]. Available: <http://www.scopus.com/inward/record.url?eid=2-s2.0-84959386702{&}partnerID=tZOtx3y1>
- [227] D. L. Giokas, L. P. Eksperiandova, A. B. Blank, and M. I. Karayannis, "Comparison and evaluation of cloud point extraction and low-temperature directed crystallization as

- preconcentration tools for the determination of trace elements in environmental samples,” in *Analytica Chimica Acta*, vol. 505, no. 1, 2004, pp. 51–58.
- [228] S. Gauglitz, T. Höllerer, and M. Turk, “Evaluation of interest point detectors and feature descriptors for visual tracking,” *International Journal of Computer Vision*, vol. 94, no. 3, pp. 335–360, 2011.
- [229] H. Aanas, A. L. Dahl, and K. S. Pedersen, “Interesting interest points: A comparative study of interest point performance on a unique data set,” *International Journal of Computer Vision*, vol. 97, no. 1, pp. 18–35, 2012.
- [230] D. Ziou and S. Tabbone, “Edge detection techniques-an overview,” *International Journal of Pattern Recognition and Image Analysis*, vol. 8, no. 4, pp. 537–559, 1998. [Online]. Available: <http://nyx-www.informatik.uni-bremen.de/1044/1/ziou{ }pria{ }98.pdf>
- [231] M. Zuliani, C. Kenney, and B. Manjunath, “A Mathematical Comparison of Point Detectors,” in *2004 Conference on Computer Vision and Pattern Recognition Workshop*, 2004, pp. 1–22.
- [232] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, “A comparison of affine region detectors,” *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, 2005.
- [233] P. Moreels and P. Perona, “Evaluation of features detectors and descriptors based on 3D objects,” *International Journal of Computer Vision*, vol. 73, no. 3, pp. 263–284, 2007. [Online]. Available: <http://link.springer.com/10.1007/s11263-006-9967-1>
- [234] O. Miksik and K. Mikolajczyk, “Evaluation of Local Detectors and Descriptors for Fast Feature Matching,” *Pattern Recognition (ICPR), 2012 21st International Conference on*, no. Icpr, pp. 2681–2684, 2012. [Online]. Available: <http://ieeexplore.ieee.org/xpls/abs{ }all.jsp?arnumber=6460718>
- [235] L. Kitchen and A. Rosenfeld, “Using Local Edge,” *IEEE Transactions on Systems, Man , And Cybernetics*, vol. SMC-11, no. 9, pp. 597–605, 1981.
- [236] F. A. Pellegrino, W. Vanzella, and V. Torre, “Edge detection revisited,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 3, pp. 1500–1518, 2004.

- [237] T. B. Nguyen and D. Ziou, "Contextual and non-contextual performance evaluation of edge detectors," *Pattern Recognition Letters*, vol. 21, no. 9, pp. 805–816, 2000.
- [238] M. Basu, "Gaussian-based edge-detection methods - A survey," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 32, no. 3, pp. 252–260, 2002.
- [239] S. Saluja, A. K. Singh, and S. Agrawal, "A Study of Edge-Detection Methods," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 1, pp. 994–999, 2013.
- [240] P. Perona and J. Malik, "Detecting and localizing edges composed of steps, peaks and roofs," in *Proceedings Third International Conference on Computer Vision*, 1990, pp. 52–57.
- [241] J. Canny, "Finding edges and lines in images," M.S. thesis, Massachusetts Institute of Technology, 1983.
- [242] W. McIlhagga, "The canny edge detector revisited," *International Journal of Computer Vision*, vol. 91, no. 3, pp. 251–261, 2011.
- [243] J. Shen and S. Castan, "An optimal linear operator for step edge detection," *CVGIP: Graphical Models and Image Processing*, vol. 54, no. 2, pp. 112–133, 1992.
- [244] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 530–549, 2004. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/15460277>
- [245] X. Ren, "Multi-scale improves boundary detection in natural images," in *Computer Vision-ECCV 2008*, 2008, pp. 533–545.
- [246] M. Maire, P. Arbeláez, C. Fowlkes, and J. Malik, "Using contours to detect and localize junctions in natural images," *26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2008.
- [247] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to Detect Natural Image Boundaries Using Local Brightness and Texture Cues," in *NIPS Vancouver*, 2002, pp. 1–8. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/15460277>
- [248] N. Payet and S. Todorovic, "SLEDGE: Sequential labeling of image edges for boundary detection," *International Journal of Computer Vision*, vol. 104, no. 1, pp. 15–37, 2013.

- [249] I. Kokkinos, “Highly accurate boundary detection and grouping,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2520–2527.
- [250] J. J. Lim, C. L. Zitnick, and P. Dollar, “Sketch tokens: A learned mid-level representation for contour and object detection,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3158–3165.
- [251] P. Dollár, Z. Tu, and S. Belongie, “Supervised learning of edges and object boundaries,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006, pp. 1964–1971.
- [252] P. Dollár and C. L. Zitnick, “Structured forests for fast edge detection,” in *International Conference on Computer Vision (ICCV)*, 2013, pp. 1841–1848.
- [253] P. Dollar and C. L. Zitnick, “Fast Edge Detection Using Structured Forests,” *Pattern Analysis and Machine Intelligence*, vol. 37, no. 8, pp. 1558–1570, 2015. [Online]. Available: <http://www.mendeley.com/catalog/structured-forests-fast-edge-detection/%5Cnhttp://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6975234>
- [254] J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce, “Discriminative sparse image models for class-specific edge detection and image interpretation,” in *European Conference on Pattern Recognition (ECCV)*, 2008, pp. 43–56.
- [255] X. Ren and L. Bo, “Discriminatively trained sparse code gradients for contour detection,” *Advances in Neural Information Processing Systems (NIPS)*, pp. 593–601, 2012.
- [256] R. Deriche and G. Giraudon, “A computational approach for corner and vertex detection,” *International Journal of Computer Vision*, vol. 10, no. 2, pp. 101–124, 1993.
- [257] F. Mokhtarian and F. Mohanna, “Performance evaluation of corner detectors using consistency and accuracy measures,” *Computer Vision and Image Understanding*, vol. 102, no. 1, pp. 81–94, 2006.
- [258] J. Chen, L. hui Zou, J. Zhang, and L. hua Dou, “The comparison and application of corner detection algorithms,” *Journal of Multimedia*, vol. 4, no. 6, pp. 435–441, 2009.
- [259] J. A. Noble, “Finding corners,” *Image and Vision Computing*, vol. 6, no. 2, pp. 121–128, 1988.
- [260] C. Harris and M. Stephens, “A Combined Corner and Edge Detector,” *Proceedings of the Alvey Vision Conference 1988*, pp. 147–151, 1988. [Online]. Available: <http://www.bmva.org/bmvc/1988/avc-88-023.html>

- [261] J. Shi and C. Tomasi, "Good features to track," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600.
- [262] P. Mainali, Q. Yang, G. Lafruit, R. Lauwereins, and L. Van Gool, "Lococo: low complexity corner detector," in *International conference on Acoustics, Speech, and Signal Processing*, 2010, pp. 810–813.
- [263] P. Mainali, Q. Yang, G. Lafruit, L. Van Gool, and R. Lauwereins, "Robust low complexity corner detector," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 4, pp. 435–445, 2011.
- [264] S. Smith and M. Brady, "SUSAN- A New Approach to Low Level Image Processing," *International Journal of Computer Vision*, vol. 23, pp. 45–78, 1997. [Online]. Available: <http://www.springerlink.com/index/m904105936xr3323.pdf>
- [265] E. Rosten and T. Drummond, "Fusing points and lines for high performance real-time tracking Ed Rosten, Tom Drummond University of Cambridge," 2005.
- [266] ———, "Machine Learning for High-Speed Corner Detection," in *European Conference on Computer Vision*, 2006, pp. 430–443.
- [267] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.
- [268] E. Mair, G. D. Hager, D. Burschka, M. Suppa, and G. Hirzinger, "Adaptive and generic corner detection based on the accelerated segment test," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6312, no. PART 2, pp. 183–196, 2010.
- [269] M. Awrangjeb, G. Lu, and C. S. Fraser, "Performance comparisons of contour-based corner detectors," *IEEE Transactions on Image Processing*, vol. 21, no. 9, pp. 4167–4179, 2012.
- [270] X. Zhang, H. Wang, A. W. B. Smith, X. Ling, B. C. Lovell, and D. Yang, "Corner detection based on gradient correlation matrices of planar curves," *Pattern Recognition*, vol. 43, no. 4, pp. 1207–1223, 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2009.10.017>
- [271] X. Zhang, H. Wang, M. Hong, L. Xu, D. Yang, and B. C. Lovell, "Robust image corner detection based on scale evolution difference of planar curves,"



- Pattern Recognition Letters*, vol. 30, no. 4, pp. 449–455, 2009. [Online]. Available: <http://dx.doi.org/10.1016/j.patrec.2008.11.002>
- [272] P.-L. Shui and W.-C. Zhang, “Corner detection and classification using anisotropic directional derivative representations,” *IEEE Transactions of Image Processing*, vol. 22, no. 8, pp. 3204–3218, 2013. [Online]. Available: <http://ovidsp.ovid.com/ovidweb.cgi?T=JS{&}CSC=Y{&}NEWS=N{&}PAGE=fulltext{&}D=medl{&}AN=23743776{&}5Cnhttp://sfx.bibl.ulaval.ca:9003/sfx{&}local?sid=OVID:medline{&}id=pmid:23743776{&}id=doi:10.1109/TIP.2013.2259834{&}issn=1057-7149{&}isbn={&}volume=22{&}issue=8{&}spage=3204{&}pages=320>
- [273] J.-M. Geusebroek, A. W. M. Smeulders, and J. Van de Weijer, “Fast Anisotropic Gauss Filtering,” *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 938–943, 2003.
- [274] P.-L. Shui and W.-C. Zhang, “Noise-robust edge detector combining isotropic and anisotropic Gaussian kernels,” *Pattern Recognition*, vol. 45, no. 2, pp. 806–820, 2012. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2011.07.020>
- [275] A. Willis and Y. Sui, “An algebraic model for fast corner detection,” *Proceedings of the IEEE International Conference on Computer Vision*, no. Iccv, pp. 2296–2302, 2009.
- [276] G.-S. Xia, J. Delon, and Y. Gousseau, “Accurate Junction Detection and Characterization in Natural Images,” *International Journal of Computer Vision*, vol. 106, no. 1, pp. 31–56, 2013. [Online]. Available: <http://link.springer.com/10.1007/s11263-013-0640-1>
- [277] C. Ding, J. Choi, D. Tao, and L. S. Davis, “Multi-Directional Multi-Level Dual-Cross Patterns for Robust Face Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 3, pp. 518–531, 2016.
- [278] T. Lindeberg, “Edge detection and ridge detection with automatic scale selection,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1998, pp. 465–470. [Online]. Available: <http://ieeexplore.ieee.org/document/517113/>
- [279] K. Mikolajczyk and C. Schmid, “Scale & affine invariant interest point detectors,” *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.
- [280] ———, “Indexing based on scale invariant interest points,” in *Eighth IEEE International Conference on Computer Vision. ICCV 2001*, 2001, pp. 525–531.



- [281] D. G. Lowe, “Distinctive Image Features from Scale-invariant Keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [282] H. Bay, T. Tuytelaars, and L. Van Gool, “SURF: Speeded up robust features,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3951 LNCS, pp. 404–417, 2006.
- [283] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-Up Robust Features (SURF),” *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.
- [284] M. Agrawal, K. Konolige, and M. R. Blas, “CenSurE: Center surround extremas for realtime feature detection and matching,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5305 LNCS, no. PART 4, pp. 102–115, 2008.
- [285] D. Marimon, A. Bonnin, T. Adamek, and R. Gimeno, “DARTs: Efficient scale-space extraction of DAISY keypoints,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2416–2423.
- [286] M. Brown and D. Lowe, “Invariant Features from Interest Point Groups,” in *British Machine Vision Conference (BMVC)*, 2002, pp. 253–262.
- [287] B. Li, R. Xiao, Z. Li, R. Cai, B.-L. Lu, and L. Zhang, “Rank-SIFT: Learning to rank repeatable local interest points,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1737–1744.
- [288] Z. Miao and X. Jiang, “Interest point detection using rank order LoG filter,” *Pattern Recognition*, vol. 46, no. 11, pp. 2890–2901, 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2013.03.024>
- [289] P. Fernández Alcantarilla, A. Bartoli, and A. J. Davison, “KAZE features,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7577, no. PART 6, pp. 214–227, 2012.
- [290] P. Fernandez Alcantarilla, J. Nuevo, and A. Bartoli, “Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces,” in *British Machine Vision Conference (BMVC)*, 2013, pp. 1–11. [Online]. Available: <http://www.bmva.org/bmvc/2013/Papers/paper0013/index.html>
- [291] H. J. Chien, C. C. Chuang, C. Y. Chen, and R. Klette, “When to use what feature? SIFT, SURF, ORB, or A-KAZE features for monocular visual odometry,” in *International Conference Image and Vision Computing*, 2016.

- [292] S. Salti, A. Lanza, and L. Di Stefano, “Keypoints from symmetries by wave propagation,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2898–2905.
- [293] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB : an efficient alternative to SIFT or SURF,” in *IEEE International Conference on Computer Vision*, 2011, pp. 2564–2571.
- [294] S. Leutenegger, M. Chli, and R. Y. Siegwart, “BRISK : Binary Robust Invariant Scalable Keypoints,” in *IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2548–2555.
- [295] S. Krig, *Computer vision metrics: Survey, taxonomy and analysis*. Apress open, 2014. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/15003161> { % } 5Cnhttp://cid.oxfordjournals.org/lookup/doi/10.1093/cid/cir991 { % } 5Cnhttp://www.scielo.cl/pdf/udecada/v15n26/art06.pdf { % } 5Cnhttp://www.scopus.com/inward/record.url?eid=2-s2.0-84861150233 { & } partnerID=tZOtx3y1
- [296] A. Alahi, R. Ortiz, and P. Vanderghenst, “FREAK: Fast Retina Keypoint,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 510–517. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2354409.2354903>
- [297] J. Matas, O. Chum, M. Urban, and T. Pajdla, “Robust wide-baseline stereo from maximally stable extremal regions,” *Image and Vision Computing*, vol. 22, no. 10, pp. 761–767, 2004.
- [298] T. Tuytelaars and L. Van Gool, “Matching widely separated views based on affine invariant regions,” *International Journal of Computer Vision*, vol. 59, no. 1, pp. 61–85, 2004.
- [299] T. Kadir, A. Zisserman, and M. Brady, “An affine invariant salient region detector,” in *European Conference on Pattern Recognition (ECCV)*, 2004, pp. 345–357. [Online]. Available: [http://link.springer.com/10.1007/978-3-540-24670-1\\_{\\_}18 { % } 5Cnhttp://www.springerlink.com/index/AHJRHQDX3UQRVDXU.pdf](http://link.springer.com/10.1007/978-3-540-24670-1_{_}18 { % } 5Cnhttp://www.springerlink.com/index/AHJRHQDX3UQRVDXU.pdf)
- [300] S. Fan and F. Ferrie, “Structure Guided Salient Region Detector,” *Proceedings of the British Machine Vision Conference 2008*, pp. 43.1–43.10, 2008. [Online]. Available: <http://www.bmva.org/bmvc/2008/papers/228.html>
- [301] C. Steger, “An unbiased detector of curvilinear structures,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 2, pp. 113–125, 1998.

- [302] S. Gu, Y. Zheng, and C. Tomasi, "Critical nets and beta-stable features for image matching," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6313, no. PART 3, pp. 663–676, 2010.
- [303] P. E. Forssén, "Maximally stable colour regions for recognition and matching," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.
- [304] Y. Avrithis and K. Rapantzikos, "The medial feature detector: Stable regions from image boundaries," in *IEEE International Conference on Computer Vision*, 2011, pp. 1724–1731.
- [305] J. Kim and K. Grauman, "Boundary preserving dense local regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 5, pp. 931–943, 2015.
- [306] C. Cui and K. N. Ngan, "Scale- and affine-invariant fan feature," *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1627–1640, 2011.
- [307] C. Lorenz, I. Carlsen, T. Buzug, and J. Weese, "Multi-scale line segmentation with automatic estimation of width, contrast and tangential direction in 2D and 3D medical images," in *First Joint Conference on Computer Vision, Virtual Reality and Robotics in Medicine, and Medical Robotics and Computer-Assisted Surgery*, vol. 1205, 1997, pp. 233–242. [Online]. Available: <http://www.springerlink.com/index/w3465411151ju2w4.pdf>
- [308] Y. Sato, C. F. Westin, A. Bhalerao, S. Nakajima, N. Shiraga, S. Tamura, and R. Kikinis, "Tissue classification based on 3D local intensity structures for volume rendering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, no. 2, pp. 160–180, 2000.
- [309] P.-e. Danielsson, Q. Lin, and Q.-z. Ye, "Efficient detection of second-degree variations in 2D and 3D images," *Journal of Visual Communication and Image Representation*, vol. 12, no. 3, pp. 255–305, 2001.
- [310] T. Lindeberg, "Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention," *International Journal of Computer Vision*, vol. 11, no. 3, pp. 283–318, 1993.
- [311] M. L. Giger, K. Doi, and H. MacMahon, "Image feature analysis and computer-aided diagnosis in digital radiography. 3. Automated detection of

- nodules in peripheral lung fields,” p. 158, 1988. [Online]. Available: <http://scitation.aip.org/content/aapm/journal/medphys/15/2/10.1118/1.596247>
- [312] T. Miwa, J.-i. Kako, S. Yamamoto, M. Matsumoto, Y. Tateno, T. Iinuma, and T. Mastsumoto, “Automatic Detection of Lung Cancers in Chest CT Images by the Variable N-Quoit Filter,” *Systems and Computers in Japan*, vol. J60-D, no. 1, pp. 1–63, 2002.
- [313] M. Usaj, D. Torkar, and D. Miklavcic, “Automatic cell detection in phase-contrast images for evaluation of electroporation efficiency in vitro,” in *Mediterranean Conference on Medical and Biomedical Engineering and Computing*, 2007, pp. 851–855. [Online]. Available: <http://link.springer.com/10.1007/978-3-540-73044-6>
- [314] M. Usaj, D. Torkar, M. Kanduser, and D. Miklavcic, “Cell counting tool parameters optimization approach for electroporation efficiency determination of attached cells in phase contrast images,” *Journal of Microscopy*, vol. 241, no. 3, pp. 303–314, 2011.
- [315] J. Liu, J. M. White, and R. M. Summers, “Automated detection of blob structures by hessian analysis and object scale,” *Proceedings - International Conference on Image Processing, ICIP*, pp. 841–844, 2010.
- [316] Q. Razlighi and Y. Stern, “Blob-like feature extraction and matching for brain MR images,” in *IEEE (IEMBS)*, 2011, pp. 116–125.
- [317] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005. [Online]. Available: <http://doi.ieeecomputersociety.org/10.1109/1109/TPAMI.2005.188>
- [318] T. Lindeberg, “Edge detection and ridge detection with automatic scale selection,” *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 30, no. 2, pp. 465–470, 1998. [Online]. Available: <http://ieeexplore.ieee.org/document/517113/>
- [319] B. N. Linh, “Modeling of human upper airways from multimodal 3D dentofacial images,” Ph.D. dissertation, National University of Singapore, 2014.
- [320] R. S. J. Estépar, J. J. Reilly, E. K. Silverman, and G. R. Washko, “Three-dimensional airway measurements and algorithms.” *Proceedings of the American Thoracic Society*, vol. 5, no. 9, pp. 905–909, 2008.

- [321] Y. Sato, H. Tanaka, T. Nishii, K. Nakanishi, N. Sugano, T. Kubota, H. Nakamura, H. Yoshikawa, T. Ochi, and S. Tamura, "Limits on the Accuracy of 3-D Thickness Measurement in Magnetic Resonance Images - Effects of Voxel Anisotropy," *IEEE Transactions on Medical Imaging*, vol. 22, no. 9, pp. 1076–1088, 2003.
- [322] C.-F. Westin, A. Bhalerao, H. Knutsson, and R. Kikinis, "Using local 3D structure for segmentation of bone from computer tomography images," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Comput. Soc, 1997, pp. 794–800. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=609418>
- [323] M. Descoteaux, M. Audette, K. Chinzei, and K. Siddiqi, "Bone Enhancement Filtering: Application to Sinus Bone Segmentation and Simulation of Pituitary Surgery," *Computer Aided Surgery*, vol. 11, no. 5, pp. 247–255, 2006. [Online]. Available: <http://ovidsp.ovid.com/ovidweb.cgi?T=JS{&}PAGE=reference{&}D=emed7{&}NEWS=N{&}AN=2006594285>
- [324] C. Reinbacher, T. Pock, C. Bauer, and H. Bischof, "Variational segmentation of elongated volumetric structures," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3177–3184.
- [325] M. Krcah, "Segmentation of Bones In 3D CT Images," Ph.D. dissertation, Czech Technical University, 2011.
- [326] Z. Gigus and J. Malik, "Detecting curvilinear structure in images," University of California at Berkeley, Tech. Rep., 1991.
- [327] C. Panagiotakis, E. Kokinou, and A. Sarris, "Curvilinear structure enhancement and detection in geophysical images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 6 PART 1, pp. 2040–2048, 2011.
- [328] V. Bismuth, R. Vaillant, H. Talbot, and L. Najman, "Curvilinear structure enhancement with the polygonal path image—application to guide-wire segmentation in X-ray fluoroscopy." *Medical image computing and computer-assisted intervention (MICCAI)*, vol. 15, pp. 9–16, 2012. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/23286026>
- [329] B. Obara, M. Fricker, D. Gavaghan, and V. Grau, "Contrast-independent curvilinear structure detection in biomedical images," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2572–2581, 2012.

- [330] D. Rivest-Hénault and M. Cheriet, “3-D curvilinear structure detection filter via structure-ball analysis,” *IEEE Transactions on Image Processing*, vol. 22, no. 7, pp. 2849–2863, 2013.
- [331] S.-g. Jeong, “Curvilinear Structure Modeling and Its Applications in Computer Vision,” 2016.
- [332] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, and M. Goldbaum, “Detection of blood vessels in retinal images using two-dimensional matched filters.” *IEEE transactions on medical imaging*, vol. 8, no. 3, pp. 263–269, 1989.
- [333] E. Meijering, M. Jacob, J.-C. F. Sarria, P. Steiner, H. Hirling, and M. Unser, “Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images.” *Cytometry. Part A : The journal of the International Society for Analytical Cytology*, vol. 58, no. 2, pp. 167–176, 2004. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/15057970>
- [334] G. Xiong, X. Zhou, A. Degterev, L. Ji, and S. T. C. Wong, “Automated Neurite Labeling and Analysis in Fluorescence Microscopy Images,” *Journal of the International Society for Analytical Cytology*, vol. 69, no. A, pp. 494–505, 2006.
- [335] Y. Zhang, X. Zhou, R. M. Witt, B. L. Sabatini, D. Adjeroh, and S. T. C. Wong, “Dendritic spine detection using curvilinear structure detector and LDA classifier,” *NeuroImage*, vol. 36, pp. 346–360, 2007.
- [336] A. Rodriguez, D. B. Ehlenberger, D. L. Dickstein, P. R. Hof, and L. Susan, “Automated Three-Dimensional Detection and Shape Classification of Dendritic Spines from Fluorescence Microscopy Images,” *PLoS ONE*, vol. 3, no. 4, pp. 1–12, 2008.
- [337] S. Hadjidemetriou, J. S. Duncan, D. Toomre, and D. Tuck, “Automatic quantification of microtubule dynamics,” in *IEEE International Symposium on Biomedical Imaging*, 2004, pp. 656–659.
- [338] A. Altinok, E. Kiris, A. J. Peck, S. C. Feinstein, L. Wilson, B. S. Manjunath, and K. Rose, “Model based dynamics analysis in live cell microtubule images.” *BMC cell biology*, vol. 8, no. s4, pp. 1–16, 2007.
- [339] N. Cerneaz and M. Brady, “Finding curvilinear structures in mammograms,” in *International Conference on Computer Vision, Virtual Reality and Robotics in Medicine*, 1995, pp. 372–382.

- [340] R. Zwiggelaar, T. C. Parr, J. E. Schumm, I. W. Hutt, C. J. Taylor, S. M. Astley, and C. R. M. Boggis, "Model-based detection of spiculated lesions in mammograms," *Medical Image Analysis*, vol. 3, no. 1, pp. 39–62, 1999.
- [341] L. C. Wai, M. Mellor, and M. Brady, "A multi-resolution CLS detection algorithm for mammographic image analysis," *Medical Image Computing and Computer-Assisted Intervention*, vol. 3217, pp. 865–872, 2004.
- [342] M. Barva, J. Kybic, J.-M. Mari, C. Cachard, and V. Hlavac, "Automatic Localization of Curvilinear Object in 3D Ultrasound Images," *Medical Imaging: Ultrasonic Imaging and Signal Processing*, vol. 5750, pp. 455–462, 2005.
- [343] C. Kirbas and F. K. H. Quek, "Vessel extraction techniques and algorithms : A survey," in *IEEE Symposium on BioInformatics and BioEngineering, BIBE 2003*, 2003, pp. 238–245.
- [344] D. Geman and B. Jedynek, "An active testing model for tracking roads in satellite images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 1, pp. 1–14, 1996.
- [345] ———, "Tracking Roads in Satellite Images by Playing Twenty Questions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. xx, no. xx, pp. xx–xx, 1996.
- [346] A. Filbois and D. Gemmerlé, "From Step Edge to Line Edge: Combining Geometric and Photometric Information," in *MVA '94 IAPR Workshop on Machine Vision Applications*, 1994, pp. 87–90.
- [347] A. Baumgartner, C. Steger, H. Mayer, and W. Eckstein, "Multi-resolution, semantic objects, and context for road extraction," *Semantic Modeling for ...*, pp. 1–17, 1997. [Online]. Available: <http://books.google.com/books?hl=en&lr=&id=u9g3GtWaF7UC&oi=fnd&pg=PA140&dq=Multi-resolution,+semantic+objects,+and+context+for+road+extraction&ots=5OAZp4-zPn&sig=8GZuqeE2r-0bl4FvO2XWT{ }5cVpQ{ }5Cnhttp://books.google.com/books?hl=en&lr=&id=u9g3GtWaF7UC&oi=fnd>
- [348] S.-W. Lee and Y. J. Kim, "Direct Extraction of Topographic Features for Gray-Scale Character-Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 7, pp. 724–729, 1995.



- [349] J. B. A. Maintz, P. A. Van Den Elsen, and M. A. Viergever, "Evaluation of ridge seeking operators for multimodality medical image matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 353–365, 1996.
- [350] L. Wang and T. Pavlidis, "Direct Gray-Scale Extraction of Features for Character Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1053–1067, 1993.
- [351] N. Merlet and J. Zerubia, "New prospects in line detection by dynamic programming," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 426–431, 1996.
- [352] A. Zlotnick and P. D. Carnine, "Finding road seeds in aerial images," *CVGIP: Image Understanding*, vol. 57, no. 2, pp. 243–260, 1993.
- [353] F. J. Novoa, N. Ezquerro, L. Traba, M. Villar, J. Pereira, J. M. Vazquez-Rodriguez, N. Vazquez, M. Martinez-Romero, and V. Mato, "Extraction of Quantitative Anatomical Information from Coronary Angiographies," *Current Bioinformatics*, vol. 6, pp. 233–250, 2011.
- [354] G. Agam, S. G. Armato, and C. Wu, "Vessel tree reconstruction in thoracic CT scans with application to nodule detection," *IEEE Transactions on Medical Imaging*, vol. 24, no. 4, pp. 486–499, 2005.
- [355] C. Florin, N. Paragios, and J. Williams, "Particle Filters, a Quasi-Monte Carlo Solution for Segmentation of Coronaries," in *Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, vol. 3749, 2005, pp. 246–253.
- [356] F. K. H. Quek and C. Kirbas, "Vessel extraction in medical images by wave-propagation and traceback," *IEEE Transactions on Medical Imaging*, vol. 20, no. 2, pp. 117–131, 2001.
- [357] M. S. Hassouna, A. A. Farag, S. Hushek, and T. Moriarty, "Statistical-Based Approach for Extracting 3D Blood Vessels from TOF-MyRA Data," in *Medical Image Computing and Computer Assisted Intervention*, 2003, pp. 680–687. [Online]. Available: [http://dx.doi.org/10.1007/978-3-540-39899-8\\_{\\_}83](http://dx.doi.org/10.1007/978-3-540-39899-8_{_}83)
- [358] R. Manniesing and W. Niessen, "Local speed functions in level set based vessel segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, 2004, pp. 475–482. [Online]. Available: <http://www.springerlink.com/index/5xt3ham0yu56p96m.pdf>



- [359] M. Schaap, R. Manniesing, I. Smal, T. van Walsum, A. van der Lugt, and W. Niessen, “Bayesian tracking of tubular structures and its application to carotid arteries in CTA.” in *Medical image computing and computer-assisted intervention*, 2007, pp. 562–570.
- [360] M. Schaap, I. Smal, C. Metz, T. van Walsum, and W. J. Niessen, “Bayesian Tracking of Elongated Structures in 3D Images,” in *Information Processing in Medical Imaging*, 2007, pp. 74–85.
- [361] J. A. Tyrrell, E. Di Tomaso, D. Fuja, R. Tong, K. Kozak, R. K. Jain, and B. Roysam, “Robust 3-D modeling of vasculature imagery using superellipsoids,” *IEEE Transactions on Medical Imaging*, vol. 26, no. 2, pp. 223–237, 2007.
- [362] A. F. Frangi, W. J. Niessen, P. J. Nederkoorn, O. E. Elgersma, and M. A. Viergever, “Three-dimensional model-based stenosis quantification of the carotid arteries from contrast-enhanced MR angiography,” in *Mathematical Methods in Biomedical Image Analysis*, 2000, pp. 110–118. [Online]. Available: <http://ieeexplore.ieee.org/ielx5/6872/18498/00852367.pdf?tp={&}arnumber=852367{&}isnumber=18498{&}5Cnhttp://ieeexplore.ieee.org/xpls/abs{&}all.jsp?arnumber=852367>
- [363] S. Bouix, K. Siddiqi, and A. Tannenbaum, “Flux driven automatic centerline extraction,” *Medical Image Analysis*, vol. 9, no. 3, pp. 209–221, 2005.
- [364] C. Lacoste, G. Finet, and I. Magnin, “Coronary Tree Extraction from X-Ray Angiograms Using Marked Point Processes,” in *IEEE International Symposium on Biomedical Imaging: Macro to Nano*, 2006, pp. 157–160. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1624876>
- [365] M. de Bruijne, B. van Ginneken, W. J. Niessen, M. Loog, and M. A. Viergever, “Model-based Segmentation of Abdominal Aortic Aneurysms in CTA Images,” in *Proceedings of the SPIE*, 2003, pp. 1560–1571.
- [366] M. de Bruijne, B. van Ginneken, M. A. Viergever, and W. J. Niessen, “Adapting active shape models for 3D segmentation of tubular structures in medical images,” in *Information Processing in Medical Imaging*, 2003, pp. 136 – 147. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/15344453{&}5Cnhttp://link.springer.com/chapter/10.1007/978-3-540-45087-0{&}12>
- [367] C. M. van Bommel, O. Wink, B. Verdonck, M. A. Viergever, and W. J. Niessen, “Blood pool contrast-enhanced MRA: Improved arterial visualization in the steady state,” *IEEE Transactions on Medical Imaging*, vol. 22, no. 5, pp. 645–652, 2003.

- [368] K. Krissian, G. Malandain, N. Ayache, R. Vaillant, and Y. Troussset, "Model-based detection of tubular structures in 3D images," *Computer Vision and Image Understanding*, vol. 80, pp. 130–171, 2000. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S107731420090866X>
- [369] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow." *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 359–369, jan 1998. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/18276256>
- [370] H. Blum, "A transformation for extracting new descriptors of shape," pp. 362–380, 1967. [Online]. Available: <papers2://publication/uuid/33A7D570-B63C-4E43-996A-4DE15D8EE75F>
- [371] Y. Fridman, S. M. Pizer, S. Aylward, and E. Bullitt, "Segmenting 3D Branching Tubular Structures Using Cores," in *Medical Image Computing and Computer Assisted Intervention*, 2003, pp. 570–577.
- [372] O. Friman, M. Hindennach, and H. O. Peitgen, "Template-based multiple hypotheses tracking of small vessels," in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 2008, pp. 1047–1050.
- [373] Q. Lin, "Enhancement, Extraction, and Visualization of 3D Volume Data," Ph.D. dissertation, Linkoping Universitet, 2003. [Online]. Available: <http://liu.diva-portal.org/smash/record.jsf?pid=diva2:302939{% }5Cnhttp://liu.diva-portal.org/smash/get/diva2:302939/FULLTEXT01.pdf>
- [374] S. Wesarg and E. A. Firlle, "Segmentation of vessels: the corkscrew algorithm," in *Medical imaging*, 2004, pp. 1609–1620. [Online]. Available: <http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.535125>
- [375] O. Wink, W. J. Niessen, and M. A. Viergever, "Fast delineation and visualization of vessels in 3-D angiographic images," *IEEE Transactions on Medical Imaging*, vol. 19, no. 4, pp. 337–346, 2000. [Online]. Available: [http://ieeexplore.ieee.org/ielx5/42/18449/00848184.pdf?tp={ & }arnumber=848184{ & }isnumber=18449{ % }5Cnhttp://ieeexplore.ieee.org/xpls/abs{ \\_ }all.jsp?arnumber=848184{ & }tag=1](http://ieeexplore.ieee.org/ielx5/42/18449/00848184.pdf?tp={ & }arnumber=848184{ & }isnumber=18449{ % }5Cnhttp://ieeexplore.ieee.org/xpls/abs{ _ }all.jsp?arnumber=848184{ & }tag=1)
- [376] T. Lindeberg, *Scale-Space Theory in Computer Vision*. Springer-Science+Business Media, 1994.

- [377] M. W. Law and A. C. Chung, "Efficient implementation for spherical flux computation and its application to vascular segmentation," *IEEE Transactions on Image Processing*, vol. 18, no. 3, pp. 596–612, 2009.
- [378] J. Lee, P. Beighley, E. Ritman, and N. Smith, "Automatic segmentation of 3D micro-CT coronary vascular images," *Medical Image Analysis*, vol. 11, no. 6, pp. 630–647, 2007.
- [379] A. Szymczak, A. Stillman, A. Tannenbaum, and K. Mischaikow, "Coronary vessel trees from 3D imagery: A topological approach," *Medical Image Analysis*, vol. 10, no. 4, pp. 548–559, 2006.
- [380] Y. Sato, T. Araki, M. Hanayama, H. Naito, and S. Tamura, "A viewpoint determination system for stenosis diagnosis and quantification in coronary angiographic image acquisition." *IEEE transactions on medical imaging*, vol. 17, no. 1, pp. 121–37, 1998. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/9617913>
- [381] S. Wörz and K. Rohr, "Cramér-Rao bounds for estimating the position and width of 3D tubular structures and analysis of thin structures with application to vascular images," *Journal of Mathematical Imaging and Vision*, vol. 30, no. 2, pp. 167–180, 2008.
- [382] C. Florin, N. Paragios, and J. Williams, "Globally optimal active contours, sequential Monte Carlo and on-line learning for vessel segmentation," in *European Conference on Computer Vision*, 2006, pp. 476–489. [Online]. Available: <http://www.springerlink.com/index/t38437n6xq8mt37x.pdf>
- [383] Y. Fridman, S. M. Pizer, S. Aylward, and E. Bullitt, "Extracting branching tubular object geometry via cores," *Medical Image Analysis*, vol. 8, no. 3, pp. 169–176, 2004.
- [384] H. Tek, D. Comaniciu, and J. Williams, "Vessel detection by mean shift based ray propagation," in *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, 2001, pp. 228–235.
- [385] O. Monga and R. Deriche, "3D Edge Detection Using Recursive Filtering: Application to scanner images," *Computer Vision and Pattern Recognition*, pp. 28–35, 1989.
- [386] O. Monga, R. Deriche, G. Malandin, and J.-P. Cocquerez, "Recursive Filtering and Edge Training: Two Primary Tools for Edge," *Image and Vision Computing*, vol. 9, no. 4, pp. 202–206, 1991.
- [387] J.-P. Thirion, O. Monga, S. Benayoun, A. P. Gueziec, and N. Ayache, "Automatic registration of 3D images using surface curvature," *Mathematical Methods in*

- Medical Imaging*, vol. 1768, no. 1992, pp. 206–216, 1992. [Online]. Available: <http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=999345>
- [388] J.-P. Thirion, “New feature points based on geometric invariants for 3D image registration,” *International Journal of Computer Vision*, vol. 18, no. 2, pp. 121–137, 1996. [Online]. Available: <http://www.springerlink.com/content/v440r80ur5154454/>
- [389] J. J. Koenderink, A. van Doorn, and J. Wagemans, “Local solid shape,” *i-Perception*, vol. 6, no. 5, pp. 1–15, 2015.
- [390] C. Steger, “Unbiased Extraction of Curvilinear Structures from 2D and 3D Images,” Ph.D. dissertation, Technischen Universitat Munchen, 1996.
- [391] J.-P. Thirion and A. Gourdon, “Computing the Differential Characteristics of Isointensity Surfaces,” pp. 190–202, 1995. [Online]. Available: <http://dx.doi.org/10.1006/cviu.1995.1015>
- [392] S. Vagnarelli, M. Vendola, and M. B. Valentini, “Differential structure of images,” *Image (Rochester, N.Y.)*, pp. 91–136, 2000.
- [393] J. J. Koenderink and A. J. van Doorn, “Surface shape and curvature scales,” *Image and Vision Computing*, vol. 10, no. 8, pp. 557–564, 1992.
- [394] D. Eberly, R. Gardner, B. Morse, S. Pizer, and C. Scharlach, “Ridges for image analysis,” *Journal of Mathematical Imaging and Vision*, vol. 4, no. 4, pp. 353–373, 1994. [Online]. Available: <http://link.springer.com/article/10.1007/BF01262402> { % } 5Cn<http://link.springer.com/content/pdf/10.1007/BF01262402.pdf>
- [395] C. Bustacara-Medina and L. Flórez-Valencia, “Comparison and evaluation of first derivatives estimation,” in *International Conference on Computer Vision and Graphics (ICCVG)*. Springer International Publishing, 2016, pp. 134–142.
- [396] T. H. Tran Thi and A. Lux, “A Method for ridge extraction,” in *Asian Conference on Computer Vision*, 2004, pp. 64–69.
- [397] A. M. Lopez, F. Lumbreras, J. Serrat, and J. J. Villanueva, “Evaluation of methods for ridge and valley detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 327–335, 1999.
- [398] R. Nevatia and R. K. Babu, “Linear feature extraction and description,” *Computer Graphics and Image Processing*, vol. 13, no. 3, pp. 257–269, 1980. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0146664X80900490>

- [399] J. A. Saif, A. A. M. Al-kubati, A. S. Hazaa, and M. Almoraish, "Image Segmentation Using Edge Detection and Thresholding," in *13th International Arab Conference on Information Technology*, 2012, pp. 473–476.
- [400] W. Förstner, "A Framework for Low Level Feature Extraction," in *European Conference on Computer Vision (ECCV)*, 1994, pp. 383–394. [Online]. Available: <http://www.springerlink.com/index/DG82U3637H211256.pdf>
- [401] N. Armande, P. Montesinos, and O. Monga, "A 3D thin nets extraction method for medical imaging," *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, vol. 1, pp. 642–646, 1996.
- [402] O. Monga, R. Lengagne, and R. Deriche, "Extraction of the zero-crossings of the curvature derivatives in volumic 3D medical images: a multi-scale approach," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 852–855, 1994.
- [403] C. Steger, "Subpixel-precise extraction of lines and edges," in *International Archives of Photogrammetry and Remote Sensing*, 2000, pp. 141–156. [Online]. Available: <http://ias.in.tum.de/people/steger/publications/2000/ISPRS-Congress-00-Steger.pdf>
- [404] ———, "Removing the bias from line detection," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, pp. 116–122. [Online]. Available: [http://ieeexplore.ieee.org/xpls/abs/\\_all.jsp?arnumber=609308](http://ieeexplore.ieee.org/xpls/abs/_all.jsp?arnumber=609308)
- [405] S. R. Aylward and E. Bullitt, "Initialization, noise, singularities, and scale in height ridge traversal for tubular object centerline extraction." *IEEE transactions on medical imaging*, vol. 21, no. 2, pp. 61–75, feb 2002. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/11929106>
- [406] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3rd ed. Cambridge University Press, 2007, vol. 1. [Online]. Available: <http://www.amazon.com/Numerical-Recipes-3rd-Edition-Scientific/dp/0521880688?SubscriptionId=0JYN1NVW651KCA56C102{&}tag=techkie-20{&}linkCode=xm2{&}camp=2025{&}creative=165953{&}creativeASIN=0521880688>
- [407] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *International Conference on Pattern Recognition*, 2006, pp. 850–855.

- [408] M. Nixon and A. Aguado, *Feature Extraction and Image processing*, 2nd ed. Academic Press, 2008. [Online]. Available: <http://medcontent.metapress.com/index/A65RM03P4874243N.pdf>
- [409] O. Monga, R. Deriche, G. Malandain, and J.-P. Cocquerez, "Recursive filtering and edge closing : two primary tools for 3d edge detection," INRIA, Tech. Rep., 1989.
- [410] ———, "3D edge detection by separable recursive filtering and edge closing," in *Conference on Pattern Recognition*, 1990, pp. 652–654.
- [411] O. Monga, R. Deriche, and J.-M. Rocchisani, "3D Edge Detection Using Recursive Filtering: Application to Scanner Images," *Computer Vision, Graphics, and Image Processing: Image Understanding*, vol. 53, no. 1, pp. 76–87, 1991.
- [412] S. Kurugol, "Machine learning and model based 3D segmentation algorithms for challenging medical imaging problems," Ph.D. dissertation, Northeastern University, 2011.
- [413] G. Lathén, "Segmentation Methods for Medical Image Analysis," Ph.D. dissertation, Campus Norrköping, Linköping University, 2010.
- [414] I. N. Bankman, *Handbook of Medical Imaging processing and analysis*. Academic Press, 2000.
- [415] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [416] N. Kabaliuk, A. Nejati, C. Loch, D. Schwass, J. E. Cater, and M. C. Jermy, "Strategies for Segmenting the Upper Airway in Cone-Beam Computed Tomography (CBCT) Data," *Open Journal of Medical Imaging*, vol. 07, no. 04, pp. 196–219, 2017. [Online]. Available: <http://www.scirp.org/journal/doi.aspx?DOI=10.4236/ojmi.2017.74019>
- [417] S. Chhabra, "Chronic Obstructive Pulmonary Disease : Evolving Definitions," *The Indian journal of chest diseases & allied sciences*, vol. 51, pp. 135–137, 2009.
- [418] W. Vos, J. De Backer, A. Devolder, O. Vanderveken, S. Verhulst, R. Salgado, P. Geronpre, B. Partoens, F. Wuyts, P. Parizel, and W. De Backer, "Correlation between severity of sleep apnea and upper airway morphology based on advanced anatomical and functional imaging," *Journal of Biomechanics*, vol. 40, no. 10, pp. 2207–2213, 2007.

- 
- [419] S.-J. Jeong, W.-S. Kim, and S.-J. Sung, “Numerical investigation on the flow characteristics and aerodynamic force of the upper airway of patient with obstructive sleep apnea using computational fluid dynamics.” *Medical engineering & physics*, vol. 29, no. 6, pp. 637–651, 2007.
- [420] P. A. Yushkevich, J. Piven, H. C. Hazlett, R. G. Smith, S. Ho, J. C. Gee, and G. Gerig, “User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability.” *NeuroImage*, vol. 31, no. 3, pp. 1116–1128, jul 2006. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/16545965>