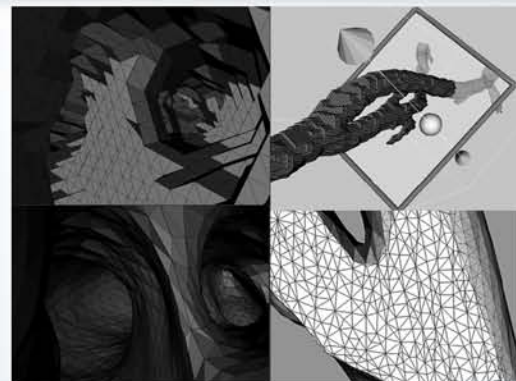
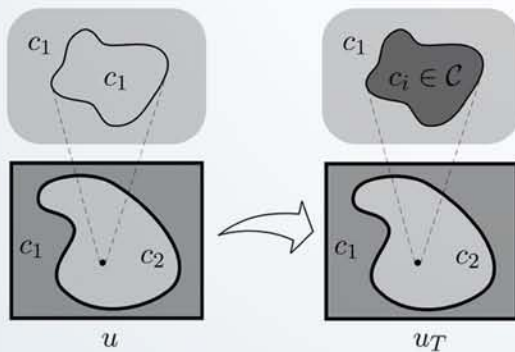
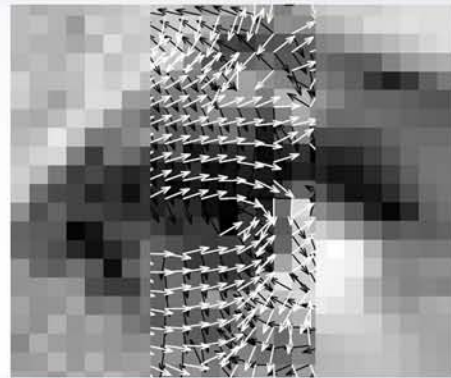
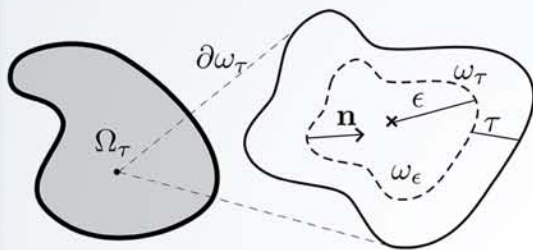


Processamento de imagens via Derivada Topológica e suas aplicações na modelagem e simulação computacional do Sistema Cardiovascular Humano

Tese de Doutorado | Ignacio Larrabide



Petrópolis - Março 2007

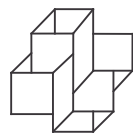


Laboratório
Nacional de
Computação
Científica

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.



Laboratório
Nacional de
Computação
Científica

Programa de Pós Graduação em Modelagem Computacional

**Processamento de imagens via Derivada
Topológica e suas aplicações na modelagem e
simulação computacional do Sistema
Cardiovascular Humano**

Por

Ignacio Larrabide

PETRÓPOLIS, RJ - BRASIL

MARÇO DE 2007

PROCESSAMENTO DE IMAGENS VIA DERIVADA TOPOLÓGICA E SUAS
APLICAÇÕES NA MODELAGEM E SIMULAÇÃO COMPUTACIONAL DO
SISTEMA CARDIOVASCULAR HUMANO

Ignacio Larrabide

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DE FORMAÇÃO
DE RECURSOS HUMANOS DO LABORATÓRIO NACIONAL DE COMPUTAÇÃO
CIENTÍFICA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA OBTENÇÃO
DO GRAU DE DOUTOR EM MODELAGEM COMPUTACIONAL.

DOUTOR EM CIÊNCIAS

Aprovado por:

Prof. Raúl Antonino Feijóo, DSc. (PRESIDENTE)

Prof. Gustavo Perla Menzala, PhD.

Prof. Abimael Fernando Dourado Loula, DSc.

Prof. Antonio Alberto Fernandes de Oliveira, DSc.

Prof. Nelson Albuquerque de Souza e Silva, DSc.

Prof. Marco Antonio Gutierrez, DSc.

Prof. Hae Yong Kim, DSc.

PETRÓPOLIS, RJ - BRASIL

MARÇO DE 2007

LARRABIDE, IGNACIO

Processamento de imagens via Derivada Topológica e suas aplicações na modelagem e simulação computacional do Sistema Cardiovascular Humano [Petrópolis] 2007

1, 218 p., 29.7 cm (LNCC/MCT, D.Sc., Modelagem Computacional, 2007)

Tese - Laboratório Nacional de Computação Científica, LNCC

1. Processamento de imagens 2. Derivada topológica 3. Modelagem do sistema cardiovascular humano

I. LNCC/MCT II. Título (série)

Resumo da tese submetida à Coordenação de Pós-Graduação do Laboratório Nacional de Computação Científica - LNCC/MCT como parte dos requisitos necessários para obtenção do grau de Doutor em Ciências

PROCESSAMENTO DE IMAGENS VIA DERIVADA TOPOLÓGICA E SUAS APLICAÇÕES NA MODELAGEM E SIMULAÇÃO COMPUTACIONAL DO SISTEMA CARDIOVASCULAR HUMANO

Ignacio Larrabide

Março/2007

Orientadores: Raúl Antonino Feijóo, Orientador
Antonio André Novotny, Co-Orientador

Este trabalho tem por objetivo estudar a restauração e segmentação de imagens médicas e sua aplicação na Modelagem do Sistema Cardiovascular Humano. O aumento da complexidade na ciência tem motivado o intercâmbio de conhecimento e técnicas entre diferentes disciplinas. Neste sentido a Derivada Topológica, originariamente concebida para tratar de maneira adequada problemas de otimização de topologia e problemas inversos, é utilizada no processamento de imagens. No caso de restauração, são estudadas duas alternativas baseadas na equação de difusão de calor, sendo uma abordagem estacionária e outra evolutiva. Para estudar a performance dos respectivos algoritmos propostos diversas imagens são restauradas e os resultados são quantitativamente comparados com resultados obtidos empregando outros algoritmos amplamente encontrados na literatura. Também é discutido neste trabalho o problema de segmentação que consiste em identificar objetos e regiões em uma imagem. Este problema também é abordado utilizando a derivada topológica, sendo apresentadas duas técnicas inovadoras. A primeira baseada na derivada topológica contínua e a outra em simplificações desta, chegando-se em uma versão discreta, de menor custo computacional. A performance dos respectivos algoritmos é testada segmentando diferentes imagens e comparando os resultados com os de outros métodos usualmente utilizados na literatura. Em ambas aplicações (restauração e segmentação), é realizada uma análise da influência dos parâmetros associados a cada algoritmo nos resultados do processamento. Por último, é estudado o uso de técnicas de modelagem na simulação computacional do Sistema Cardiovascular Humano (SCVH). O objetivo deste estudo é a integração das técnicas de processamento de imagens propostas com os modelos necessários na simulação computacional do SCVH. Cabe ressaltar que, na atualidade, modelos multidimensionais que integram representações das diversas partes do sistema cardiovascular com diferente nível de detalhe, são amplamente utilizados. Para criar estes tipos de modelos, informações de diversas origens e variada natureza devem ser tratadas, requerendo-se o desenvolvimento de ferramentas computacionais que permitam operar com grandes volumes de dados. No entanto, não existe na atualidade uma ferramenta deste tipo. Sendo assim, e com o intuito de suprir esta deficiência, foi desenvolvido um sistema chamado HeMoLab - Laboratório de Modelagem em Hemodinâmica, o qual é descrito neste trabalho. Por último, são apresentados alguns resultados obtidos com este sistema computacional.

Abstract of a Thesis Submitted to LNCC/MCT as a partial fulfilment of the requirements for the degree of Doctor of Science (D.Sc.)

IMAGE PROCESSING VIA TOPOLOGICAL DERIVATIVE AND ITS APPLICATIONS TO HUMAN CARDIOVASCULAR SYSTEM MODELLING AND SIMULATION

Ignacio Larrabide

March/2007

Advisors: Raúl Antonino Feijóo, Thesis Advisor
Antonio André Novotny, Thesis Co-Advisor

The objective of this work is to study the restoration and segmentation of medical images and its application to Modeling the Human Cardiovascular System (HCVS). The growing complexity in science has motivated the exchange of knowledge and techniques between different scientific disciplines. In this sense the Topological Derivative, originally conceived to treat in an adequate manner topology optimization and inverse problems, is applied to image processing. In the case of image restoration, two different alternatives based on the heat diffusion equation are studied, being a stationary and an evolutive approach. In order to study the performance of these algorithms, different images are restored and the results are quantitatively compared to other methods widely used on the literature. In this work we also discuss the segmentation problem, where the objective is to identify objects or regions in an image. This problem is also considered using the topological derivative, where two novel techniques are proposed. The first is based on the continuous topological derivative and the other, on a simplified discrete version with lower computational cost. The performance of the corresponding algorithms is tested segmenting different images and comparing the results to those obtained with other commonly used methods. In both cases (restoration and segmentation), is performed an analysis of the algorithm's parameters influence on the processing results. Finally, the use of computational modeling in the simulation of the HCVS is discussed. The aim of this study is the integration of the proposed image processing techniques with computational models of the HCVS. Nowadays, the use of multidimensional models, that merge representations of different parts of the HCVS with different degrees of detail, are widely used. In order to create this type of models, information of different kinds and nature has to be treated, requiring the development of computational tools that allow to handle large data sets. As such a tool does not exist at the time, a software called HeMoLab - Hemodynamics Modeling Laboratory was developed to fulfill this need. This computational tool is described and some results obtained with it are presented.

Prefácio

Esta tesis resume el trabajo que desarrollé durante mis estudios de doctorado en el LNCC - Laboratório Nacional de Computação Científica, Petrópolis - RJ - Brasil. El curso ofrecido por esta institución, llamado “Doutorado em Modelagem Computacional”, forma doctores con una visión amplia y de incumbencia en los diferentes aspectos envueltos en el modelaje computacional de fenómenos físicos y biológicos, abarcando el estudio y desarrollo de modelos matemáticos, implementación de soluciones numéricas, desarrollo de sistemas computacionales asociados y su aplicación a problemas concretos. El presente trabajo de tesis abarca los diferentes aspectos propuestos por el programa de pos-graduación mencionado. El foco de mi atención en esta tesis fue en técnicas de procesamiento de imágenes, en particular restauración y segmentación, utilizando el concepto de derivada topológica y su aplicación en el modelaje y simulación computacional de Sistema Cardiovascular Humano.

Para realizar este trabajo en el período estipulado de cuatro años, fue necesario el apoyo, la confianza y la ayuda de muchas personas, hacia quienes deseo expresar mi mas profundo agradecimiento. En primer lugar, y gracias a quien todo esto fue posible, deseo expresar mi completa gratitud al Prof. Raúl Antonino Feijóo, por haber confiado y apostado hace cuatro años atrás en un joven formado en Ingeniería de Sistemas con conocimientos muy básicos en matemática y nulos en mecánica. Gracias a esta confianza, a la paciencia, a los conocimientos transmitidos, a la orientación, a la amistad y libertad que el me otorgó fue posible desarrollar esta tesis. Esta confianza también me permitió participar de la manera mas activa en un proyecto de la importancia y dimensión del HeMoLab, otorgándome libertad para llevar adelante esta desafiante tarea.

Fue para mí, un individuo con formación en computación, un gran desafío comprender conceptos con fuertes raíces matemáticas y de ingeniería mecánica como los que forman la base teórica de este trabajo de tesis. Para esto fue indispensable la ayuda de varias personas a quienes quiero agradecer. Al Prof. Antonio André Novotny por su amistad y por contribuir de las mas variadas formas, sea aclarando dudas, respondiendo preguntas, haciendo cuentas, compartiendo su experiencia en elementos finitos o sencillamente revisando de forma minuciosa este y otros trabajos. Al Prof. Edgardo Taroco por la calma y por darme herramientas de mecánica del continuo, útiles que me permitieron encarar los mas desafiantes problemas. Al Prof. Claudio Padra - Centro Atómico Balseiro, Bariloche - Argentina, por amistad y su ayuda en cuestiones matemáticas. Al Prof. Mohamed Masmoudi - MIP Université Paul Sabatier - Toulouse, Francia, por haber contribuido con ideas que fueron las semillas que dieron origen a este trabajo y por haber trabajado conjuntamente en varias oportunidades con nuestro grupo en el LNCC.

Al grupo de trabajo del Pladema - UNICEN, Tandil - Argentina, en especial al Prof. Marcelo Vénere por ponerme en contacto con el LNCC, dando inicio a todo este trabajo, y por haber estado presente durante todo este tiempo. Al Prof. Enzo Dari por compartir generosamente todo su conocimiento y experiencia en desarrollo de *solvers* numéricos, paralelismo y sistemas operativos. Al Prof. Santiago Urquiza por contribuir con sus conocimientos en dinámica de los fluidos y métodos de resolución numérica.

A los Profs. Gilson Giraldi y Paulo Sergio Rogrigues por contribuir con su conocimiento y experiencia en procesamiento de imágenes en varias discusiones y reuniones de trabajo que en mucho mejoraron la calidad de esta tesis, y también por aportar datos e imágenes que fueron de mucha utilidad.

A mis profesores a lo largo de los cursos que realice en el LNCC durante los primeros dos años, especialmente al Prof. Abimael F. D. Loula, Prof. Gustavo Perla, Prof. Jaime

Rivera, Prof. Jauvane C. de Oliveira.

Al Prof. Hae Yong Kim - USP, São Paulo - Brasil y al Prof. Alejandro Frangi - CiLab UPF, Barcelona - España, por los conocimientos, consejos y sugerencias que hemos intercambiado, principalmente durante las últimas etapas de este trabajo.

Al los Profs. Nelson Albuquerque de Souza e Silva y Ronaldo Lima - UFRJ, Rio de Janeiro - Brasil, por contribuir desde el punto de vista médico y por los datos e imágenes gentilmente cedidos.

Por la excelente labor desarrollada a lo largo del proyecto HeMoLab, complementando el trabajo de planeamiento y diseño de arquitectura, agradezco al equipo de desarrollo integrado por DSc. Rodrigo Luis de Souza da Silva, MSc. Paulo Ziemer, Carlos Antonio Salgado, Claudia Valeria Messner, Igor Freitas, Jan Palach, Vinicius Pessôa, Glaucio Reynaud y al equipo de mecánica integrado por MSc. Claudio Pivello e Ing. Pablo Blanco.

Al CNPq por posibilitar, con su apoyo económico, mi estadía en el LNCC y el desarrollo de este trabajo. Al LNCC por haberme recibido y permitirme trabajar con la mayor libertad.

Por haberme dado todo, a mis padres Ani y Rulo, que desde siempre me apoyaron en todos mis emprendimientos, y por mas de 20 años pagaron mi primera beca de estudios. A mis hermanos Ceci, Joaco y Aixa, que aunque estemos distantes siempre siento que estamos cerca. A Mercedes, Gonzalo y Lautaro por acordarse siempre. A Elsi, Luis, Pauline y July por haberme recibido cuando era un recién llegado y por seguir ahí todo este tiempo. A Miguel y María por dejar a mis cuidados a la nena y por siempre estar pendientes de mis avances. Guardo también un lugar muy especial en estos agradecimientos para mi abuela la Prof. Elsa Marelli, quien fue una fuente de motivación e inspiración constante para continuar mis estudios y seguir mis ideales, siempre con una visión optimista de la vida.

Quien ha tenido la oportunidad de vivir en una cultura o país distinto del que nació y creció, entenderá la importancia de tener amigos que comparten las mismas costumbres. En mi estadía en Petrópolis cultivé amistades entrañables con otros argentinos que de otra forma no hubiera conocido y en mucho ayudaron a amenizar el duro trabajo que implica realizar un doctorado. Con ellos hemos compartido innumerables momentos, inolvidables eventos gastronómicos e importantes acontecimientos. Por orden de aparición, a Esteban, Silvana, Sebastian, Gabriela, Pablo, Cabeza, Maru y Ale les agradezco de corazón por compartir conmigo todos estos eventos. A Seba “El Fiore”, por haberle puesto garra al encarar este desafío. A Diego y Pablo por visitar el LNCC y trabajar con nuestro grupo. A todo los muchachos de la facu que siempre estuvieron ahí.

A los amigos y compañeros que conocí en el LNCC Demerson, Magno, Sidarta, Jean, Jairo, Honorio, Eduardo, Thiago, Flávio, Boness, Manuel, Marcos, Daniel, Maicom, Jaqueline, Paula, Ana Luisa, por compartir muchos buenos momentos. A Aline, por su amistad y por ayudarme en los mas variados asuntos burocráticos.

A los amigos que hice en Petrópolis y a la “galera” del triatlón, Bruno, Rodrigo, José Mauro, Marcos, André, Tadeu, Rodolfo, Fredy por las miles de brazadas, pedaleadas y trotes compartidos, y por las duras metas alcanzadas haciendo que todo parezca posible.

Finalmente pero en primer lugar, a Gia, el amor de mi vida, por el cariño, el apoyo, la garra, el compañerismo, la paciencia, la tranquilidad, la compañía, el humor y por animarse a pasar por esto juntos.

IGNACIO LARRABIDE
MARZO 2007, PETRÓPOLIS - RJ

Para Gia.

Sumário

Prefácio	ix
Tabela de símbolos	xvii
Introdução e Motivação	1
I Restauração de Imagens	7
1. Derivada Topológica	11
1.1 Trabalhos pioneiros na área	12
1.2 Motivação para o uso da Derivada Topológica em Processamento de Imagens	15
1.3 Derivada Topológica - Formulação Variacional	16
1.4 Equação de calor estacionária	22
1.A Propriedades de $a(\cdot, \cdot)$	31
1.B Algumas identidades úteis	32
2. Restauração de Imagens	33
2.1 Degradação de Imagens - Modelo Matemático	34
2.2 O Uso de EDP's na Restauração de Imagens	34
2.3 O Método da Energia	36
2.3.1 Regularização do Problema	37
2.4 Métodos Evolutivos	43
2.4.1 A Equação de Calor	44
2.4.2 Difusão Não-linear (Perona e Malik)	46
2.4.3 Suavizado Seletivo (Alvarez, Morel e Lions)	48
2.4.4 Difusão Anisotrópica Robusta (Black <i>et al.</i>)	50
2.5 Comentários	52
2.A Medidas de erro em imagens	54
3. Derivada Topológica em Restauração de Imagens	55
3.1 Abordagem Contínua	57
3.2 Abordagem Discreta	61
3.2.1 Cálculo da Derivada Topológica	64
3.3 Comentários	66
4. Análises e Comparações das Técnicas Propostas e Existentes	71
4.1 Teste 1 - Remoção de ruído uniforme	72
4.2 Teste 2 - Remoção de ruído Gaussiano	79
4.3 Teste 3 - Remoção de ruído sal e pimenta	81

4.4	Teste 4 - Remoção de ruído speckle	82
4.5	Teste 5 - Processamento de imagens médicas	83
4.6	Comentários	83
II Segmentação de Imagens		93
5.	Segmentação de Imagens	97
5.1	Estado da arte em Segmentação de Imagens via EDP's	97
5.2	Segmentação como um Problema de Minimização	99
5.2.1	Funcional de Mumford-Shah	100
5.2.2	Existência da Solução	100
5.A	Medida de Hausdorff	107
6.	Derivada Topológica em Segmentação de Imagens	109
6.1	Abordagem Contínua	109
6.1.1	Problema Perturbado	110
6.1.2	Cálculo da Derivada Topológica	111
6.2	Abordagem Discreta	115
6.3	Otimização dos Valores das Classes	118
6.A	Análise Assintótica para uma Perturbação na Fonte	120
6.A.1	Formulação do problema associado ao domínio original	120
6.A.2	Formulação do problema associado ao domínio perturbado	120
6.A.3	Análise Assintótica	121
6.B	Tensor Momento-Energia de Eshelby	122
7.	Análises e Comparações das Técnicas Propostas e Existentes	125
7.1	Teste 1 - Validação e comparação com outros métodos	125
7.2	Teste 2 - Robustez	130
7.3	Teste 3 - Dependência nos valores das classes	132
7.4	Teste 4 - Estimação de parâmetros	135
7.4.1	S_{D_T} -Continuous	137
7.4.2	S_{D_T} -Discrete	138
7.5	Teste 5 - Processamento de imagens médicas	138
7.6	Comentários	139
III Modelagem e Simulação Computacional do Sistema Cardiovascular Humano		143

8. Modelagem Computacional Multidimensional do Sistema Cardiovascular Humano	147
8.1 Modelos	149
8.1.1 Modelo 1D	149
8.1.2 Geração de arvores	150
8.1.3 Modelo 0D	150
8.1.4 Modelo 3D e Acoplamento	151
8.1.5 Leis constitutivas	153
8.1.6 Reconstrução de Geometrias	153
8.A Dedução do modelo unidimensional do Sistema Cardiovascular	155
8.A.1 Hipóteses geométricas e cinemáticas	155
8.A.2 Teorema de Transporte de Reynolds	156
8.A.3 Conservação da Massa	158
8.A.4 Conservação do Momento	159
9. HeMoLab - Laboratório de Modelagem em Hemodinâmica	167
9.1 Análise de requisitos	168
9.1.1 Módulo 1: Processamento de Imagens	169
9.1.2 Módulo 2: Modelo 1D	170
9.1.3 Módulo 3: Modelo 3D	171
9.1.4 Módulo 4: Acoplamento	171
9.1.5 Módulo 5: Simulação Numérica	172
9.1.6 Módulo 6: Visualização e Análise	172
9.2 O <i>framework</i> utilizado	172
9.2.1 Ambiente de trabalho	173
9.2.2 ParaView	173
9.2.3 Arquitetura do ParaView	174
9.2.4 <i>Client</i> - CL	175
9.2.5 <i>Server Manager</i> - SM	175
9.2.6 <i>Render Server</i> - RS	177
9.2.7 <i>Data Server</i> - DS	178
9.2.8 Linguagem Cliente-Servidor	178
9.2.8.1 <i>Streams</i> Cliente-Servidor	179
9.2.8.2 Configuração - Deployment	180
9.3 HeMoLab - Laboratório de Modelagem em Hemodinâmica	180
9.3.1 Pacotes e Módulos	181
9.3.2 Módulo de Processamento de Imagens - <i>hmImageProcessing</i>	182
9.3.3 Módulo 1D - <i>hm1DModule</i>	184
9.3.4 Módulo 3D - <i>hm3DModule</i>	189
9.3.5 Módulo de Acoplamento - <i>hmCoupling</i>	193
9.3.6 <i>Solver</i> numérico - <i>SolverGP</i>	194
9.3.7 Cenários	195

9.3.7.1	Cenário 1 - Modelo 1D	196
9.3.7.2	Cenário 2 - Modelo 3D	197
9.3.7.3	Cenário 3 - Modelo Acoplado	197
9.4	Comentários	197
9.A	Terminologia	199
10.	Uso de Modelos na Simulação do Sistema Cardiovascular Humano	201
10.1	Metodologia	201
10.1.1	Árvore unidimensional	201
10.1.2	Geometrias 3D	203
10.1.3	Geometria 3D padrão	203
10.1.4	Geometria 3D real	203
10.2	Teste 1 - Sensibilidade a geometria	204
10.2.1	Diferentes geometrias de estenose	205
10.2.2	Geometria real vs. geometria padrão	205
10.3	Teste 2 - Sensibilidade ao modelo constitutivo	205
10.3.1	Geometria padrão	206
10.3.2	Geometria real	206
10.4	Comentários	207
	Conclusões e comentários finais	215
A.	Discretização de Equações Diferenciais	I
A.1	Diferenças Finitas	I
A.1.1	Discretização no tempo e no espaço	II
A.1.2	Análise de estabilidade	IV
A.2	Discretização via Elementos Finitos	VI
A.3	Método de aproximação de Galerkin	VI
A.3.1	Forma algébrica	VII
A.3.2	Família de elementos	VIII
A.3.3	Funções de forma e matrizes elementares	IX

Tabela de símbolos

$*$	produto usual de escalares.	$(\cdot) _{\tau=0}$	(\cdot) avaliada em $\tau = 0$.
\cdot	produto escalar entre vetores.	$ \cdot $	módulo de (\cdot) , valor absoluto.
\otimes	produto tensorial.		
\otimes_s	parte simétrica do produto tensorial.	$\ \cdot\ $ $\ \cdot\ _{L^1(\Omega)}$	norma de (\cdot) . norma L^1 de (\cdot) em Ω .
\times	produto cartesiano.	\mathcal{H}^{N-1}	medida de Hausdorff de dimensão $N - 1$.
\star	produto de convolução.		
v, u, λ, k	escalares.	$\epsilon \rightarrow 0$	ϵ tende a zero.
$\mathbf{x}, \mathbf{n}, \mathbf{v}, \boldsymbol{\nu}$	vetores.	$\lim_{\epsilon \rightarrow 0}$	limite quando ϵ tende a zero.
$\mathcal{J}, \mathcal{F}, E$	funções custo.		
$\mathbb{I}, \mathbb{K}, \mathbb{P}$	tensores.	$\inf_{u \in \mathcal{U}}$	ínfimo de u em \mathcal{U} .
$\text{tr}(\cdot)$	traço de (\cdot) .	$\sup_{u \in \mathcal{U}}$	supremo de u em \mathcal{U} .
F^T	transposta da matriz F .	$\arg \min_{u \in \mathcal{U}} \mathcal{J}(u)$	argumento que minimize $\mathcal{J}(u)$ em $u \in \mathcal{U}$.
F^1	inversa de F .		
F^{-T}	inversa transposta de F .		
$\det(F)$	determinante de F .		
R^*	adjunta de R .	$(\cdot)'$	derivada usual de (\cdot) .
$a(\cdot, \cdot)$	uma forma bilinear.	$\frac{\partial(\cdot)}{\partial t} = (\cdot)_{,t}$	derivada parcial de (\cdot) respeito de t .
$l(\cdot)$	uma forma linear.	$\frac{d(\cdot)}{dt}$	derivada total de (\cdot) respeito de t .
\emptyset	conjunto vazio.	$\text{div}(\cdot)$	divergência de (\cdot) .
Ω, ω	domínios de forma arbitrária.	$\nabla(\cdot)$	gradiente de (\cdot) .
$\partial\Omega, \partial\omega$	contorno de Ω e ω respectivamente.	$\nabla\nabla(\cdot)$	gradiente de segunda ordem de (\cdot) .
γ	uma fissura de dimensão $N - 1$.	$\triangle(\cdot)$	Laplaceano de (\cdot) .
\mathbb{R}^n	espaço real de dimensão n .	$\delta\mathcal{J}(u; v)$	variação de \mathcal{J} em u na direção de v .
L^2	espaço de funções quadrado integráveis.	$\left\langle \frac{\partial\mathcal{J}}{\partial u_\epsilon}; \frac{\partial u_\epsilon}{\partial \epsilon} \right\rangle$	derivada do funcional $\mathcal{J}(u_\epsilon)$ em ϵ .
H^1	espaço de Hilbert de ordem um.		
$W^{m,p}$	espaço de Sobolev de ordem m, p .	rand	gerador de números aleatórios entre 0 e 1 com distribuição uniforme).
C^1	curva com primeira derivada contínua.		
\mathcal{C}	conjunto de classes.	dB	decibel.
S_u	subconjunto de salto da função u .		

SNR	Signal-to-Noise ratio.
PSNR	Peak Signal-to-Noise ratio.
LoG	Laplacian of the Gaussian.
bpp	bits-per-pixel.
MRI	Magnetic Resonance Imaging.
CT	Computed Tomography.
US	Ultrasom.
FCM	Fuzzy C-means.
EDP	Equações em Derivadas Parciais.
ALE	Arbitrary Lagrangian-Eulerian.
PPV	Princípio das Potências Virtuais.
IDE	Integrated Development Environment.
C++	Linguagem de programação, derivada da linguagem “B”.
TCL	Tool Command Language.
XML	Extensible Markup Language.
LTS	Long Time Support.
GUI	Graphical User Interface.
GCC	Generic C/C++ Compiler.
HeMoLab	Hemodynamics Modelling Laboratory.
OpenGL	Open Graphic Language.
VTK	Visualization ToolKit.
CMake	Cross-Platform Make.
CDT	C/C++ Development Tooling.

Introdução e Motivação

Nos últimos anos as doenças cardiovasculares têm sido a maior causa de morte não somente no país como também no mundo. De acordo com a Associação Americana do Coração [14] as doenças cardiovasculares são a principal causa de morte nos Estados Unidos, correspondendo aproximadamente a um terço do total. No ano 2002, devido às doenças cardiovasculares morreram no Brasil 396.000 pessoas, na Argentina 93.300, na Austrália 47.200, na China 3.000.000, nos Estados Unidos 922.700, na França 154.000, na Alemanha 387.900, na Índia 2.810.000, na Itália 247.700 e na Rússia 1.431.900 para citar somente alguns países desenvolvidos ou em desenvolvimento (dados obtidos da Organização Mundial da Saúde - OMS, [153]). Segundo a OMS, o Brasil aparece em 9º lugar na lista dos países cuja população morre mais, em números absolutos, de doenças cardíacas, e é o 6º de derrames. O número de casos de doenças cardiovasculares no país fica atrás somente de países como China, Índia, Rússia e Estados Unidos.

Um relatório publicado recentemente pela OMS, o mais abrangente já preparado sobre doenças cardiovasculares, aponta que estes males matarão mais de 24 milhões de pessoas por ano até 2030. Atualmente, 17 milhões de pessoas morrem anualmente desses males.

Estes fatos têm motivado o desenvolvimento de novas técnicas e procedimentos para diagnosticar, tratar e acompanhar a evolução deste males nos seres humanos. Técnicas como *bypass* coronários, implantação de *stents*, *coils* e *clips* em aneurismas são algumas das mais novas intervenções cirúrgicas relacionadas a doenças do Sistema Cardiovascular Humano (SCVH). O aumento na complexidade deste tipo de procedimentos exige a cada dia um maior uso de técnicas que permitam prever com precisão o resultado quando estes métodos cirúrgicos são aplicados nos pacientes.

Nas últimas décadas novas estratégias, baseadas no uso de modelos computacionais, têm sido muito utilizadas. O atual grau de desenvolvimento alcançado por estas técnicas de modelagem, conjuntamente com o rápido crescimento da performance de cálculo dos computadores, tem permitido o estudo, desenvolvimento e solução de modelos mecânico-biológicos altamente sofisticados capazes de antecipar, com aceitável grau de precisão, os resultados de diversos procedimentos médicos. No entanto, os benefícios de aplicar esta tecnologia à medicina cardiovascular tem como desafio superar algumas barreiras.

Primeiramente, esta tecnologia deve poder ser aplicada em pacientes reais, reque-rendo-se a incorporação de dados específicos de cada indivíduo que complementem os modelos matemáticos e físicos. Uma das formas mais efetivas de obter este tipo de dados é utilizando técnicas de aquisição de imagens médicas como *Magnetic Resonance Imaging* - MRI, *Computed Tomography* - CT, *Positron Emission Tomography* - PET, *Single Positron Emission Computed Tomography* - SPECT, Ultrassom - US etc. Estes dados devem ser processados para extrair as informações de interesse do paciente (e.g., informação geométrica da anatomia, propriedades mecânicas dos tecidos, etc.).

Outro ponto importante neste processo é o desenvolvimento de modelos matemáticos

e físicos que consigam aproximar com o suficiente nível de detalhe e complexidade os fenômenos envolvidos capturando em detalhe aspectos como pressão, fluxo, etc. Para obter as soluções destes modelos diferentes informações são necessárias

- propriedades mecânicas dos tecidos envolvidos,
- as geometrias das regiões de interesse obtidas a partir das imagens do paciente,
- condições iniciais das variáveis de interesse,
- condições de contorno consistentes com o resto do sistema.

A resolução das equações associadas aos modelos do SCVH completo é proibitivo devido à quantidade de graus de liberdade envolvidos, o que faz este problema computacionalmente muito custoso (sendo impraticável, ao menos com a capacidade de processamento dos computadores na atualidade). O trabalho pioneiro de Hughes e Lubliner [95], apresenta uma alternativa interessante para contornar estas dificuldades. Utilizando modelos de menor dimensão (e.g., 1D), é possível aproximar as características mais importantes do SCVH nos vasos de maior tamanho. Nestes modelos, hipóteses são utilizadas para permitir levar as equações de fluidos (originariamente tridimensionais), a formas simplificadas unidimensionais, nas quais somente valores médios das quantidades envolvidas (e.g., fluxo Q , pressão p , etc.) são considerados. Os modelos 1D levam em consideração o comprimento e o diâmetro dos vasos do sistema arterial permitindo que este último varie ao longo do distrito, diminuindo consideravelmente a quantidade de graus de liberdade. Acoplando modelos 3D detalhados das regiões de importância com modelos simplificados do resto do sistema [61, 193, 171, 199] é possível representar a árvore arterial como um todo obtendo-se uma resposta precisa nos locais de interesse. Desta maneira, a modelagem 1D da árvore arterial fornece naturalmente as condições de contorno necessárias para as regiões detalhadas com modelos 3D.

A aproximação destes modelos, utilizando por exemplo o Método dos Elementos Finitos (MEF), requer tanto a solução de sistemas de equações não-lineares (da ordem de milhões de equações para cada intervalo de tempo), como o uso de estimadores de erro que permitam a análise adaptativa do problema. Finalmente, é necessário o uso de técnicas de visualização científica que permitam interpretar estes resultados.

Os passos seguidos para criar estes modelos a partir dos dados do paciente são:

- Ler os dados, geralmente em formatos padrão (e.g., DICOM - *Digital Imaging and Communications in Medicine*) a partir de um estudo de imagens feito no paciente.
- Melhorar a qualidade da imagem e segmentá-la.
- Reconstruir a geometria em três dimensões.
- Gerar e otimizar as malhas de Elementos Finitos para análise dos modelos.
- Se for necessário, acoplar estes tipos de malhas com modelos simplificados do resto do sistema.

- Resolver o sistema de equações não-lineares dependentes do tempo.
- Visualizar os resultados.

Suponha-se, por exemplo, que é de interesse estudar a hemodinâmica na bifurcação da artéria carótida. A complexidade da geometria influencia fortemente as características hemodinâmicas e as ações mecânicas do fluido nas paredes dos vasos. Por esta razão, a geometria vascular tem um papel muito importante no desenvolvimento das diferentes patologias que afetam o SCVH. Diferentes publicações [13, 38, 72, 106] indicam que regiões de re-circulação estão ligadas ao desenvolvimento de placa de ateroma. Esta situação é analisada em [163], onde se observa que a presença e a localização de zonas de re-circulação é fortemente influenciada pelo ângulo de bifurcação (e.g., o ângulo entre a carótida interna e externa). Outro caso típico é dado pela região distal¹ em uma anastomose em um *bypass*. Dependendo do ângulo de anastomose, zonas de re-circulação podem ser criadas, provocando conseqüentemente o êxito ou fracasso da intervenção. Para nomear outras aplicações pode-se citar a avaliação de risco de ruptura de aneurismas, o estudo de implante de *stents* ou *clips* em aneurismas, estimação da presença de aterosclerose via caracterização de propriedades das paredes arteriais, dentre outras.

Assim, a aplicação de técnicas vindas da engenharia na medicina têm por objetivo complementar e tornar mais precisa a informação que os médicos dispõem para tomar decisões. Ferramentas como análise estrutural, identificação de propriedades materiais, dinâmica dos fluidos computacional (CFD - *Computational Fluid Dynamics*), simulação computacional, entre outras, estão se tornando a cada dia mais utilizadas na medicina.

Unido às técnicas de modelagem computacional, o processamento de imagens fornece ferramentas poderosas para análise de informações dos pacientes, diagnóstico de doenças, estudo e análise de diferentes alternativas de intervenção (planejamento cirúrgico) e pesquisa relacionada ao sistema cardiovascular.

As técnicas de aquisição de imagens médicas fornecem informação anatômica e funcional aos especialistas das diferentes áreas da medicina. Conseqüentemente, a demanda por ferramentas que permitam manipular estas imagens tem crescido drasticamente. Ao longo dos anos novos problemas têm aparecido na área, por exemplo restauração e segmentação de imagens, visualização de dados volumétricos, registro de imagens (*image registration*) e reconhecimento de padrões.

Os resultados fornecidos pelos modelos dependem fortemente da complexidade e da morfologia dos domínios sendo estudados. Por este motivo, a precisão das técnicas utilizadas no processo de restauração, segmentação e reconstrução das geometrias terá um impacto considerável nos resultados, tornando a reconstrução de geometrias a partir de imagens médicas (MRI, CT, PET, SPECT, Ultrassom, etc.) um ponto chave na modelagem do sistema arterial.

O uso de modelos computacionais tem mudado a maneira na qual problemas em diferentes áreas do conhecimento são tratados, por exemplo no processamento de imagens [16, 27, 65, 164]. O uso do Cálculo Variacional e Equações Diferenciais Parciais (EDP's)

¹i.e., ponto em que uma estrutura ou um órgão fica afastado de seu centro ou de sua origem.

nestes modelos tem crescido consideravelmente nos últimos anos. Na análise e processamento de imagens médicas, métodos que utilizam modelos matemáticos são usualmente mais precisos que aqueles que não aproveitam este tipo de informação. Por este motivo, neste trabalho serão estudadas duas áreas importantes no processamento de imagens:

- **Restauração de Imagens:** durante o processo de aquisição da imagem, esta pode ser alterada por diferentes fatores (lentes fora de foco, movimento do paciente, falta de sensibilidade dos dispositivos, limitações nos tempos de exposição dos pacientes, o próprio processo pelo qual a imagem é formada, entre outras). Neste caso o objetivo é melhorar a qualidade removendo o “ruído” de uma imagem que foi sujeita a alguma ou várias destas degradações.
- **Segmentação de Imagens:** no caso de imagens médicas, seja para fins de realidade virtual, realidade aumentada, reconstrução de geometrias, diagnóstico de doenças, ou outros, a segmentação é um passo fundamental quando é necessário identificar as diferentes estruturas presentes em uma imagem.

O crescimento desta interdisciplinariedade na ciência tem motivado o intercâmbio de conhecimento e técnicas entre diferentes disciplinas. Como se verá ao longo deste trabalho, a utilização da Derivada Topológica por exemplo, originariamente concebida para tratar de maneira adequada problemas de otimização de topologia e problemas inversos, tem mostrado resultados promissores ao ser aplicada no processamento de imagens.

Como mencionado, a motivação para o uso de técnicas de restauração e segmentação de imagens vem, nesta oportunidade, da necessidade de reconstruir regiões ou distritos arteriais para serem utilizados na modelagem do SCVH. Restauração e segmentação de imagens médicas, reconstrução de geometrias, geração de malhas de elementos finitos e edição de árvores arteriais são alguns dos passos necessários para gerar este tipo de modelos, requerendo cada um deles dados de diferente natureza (e.g., imagens médicas, malhas de elementos finitos, propriedades materiais, etc.). A complexidade destes dados torna o seu tratamento manual praticamente impossível. Por outro lado, não existe na atualidade um sistema computacional que reúna em um mesmo ambiente de trabalho as ferramentas necessária para criar estes modelos. Um sistema computacional destas características em muito contribuirá para melhorar a compreensão do SCVH, auxiliando na prevenção, diagnose, terapia e reabilitação (planejamento cirúrgico) das mais diversas patologias ou disfunções cardiovasculares.

Baseado nas considerações anteriores, o presente trabalho está organizado em três grandes partes, quais sejam:

Na Parte I, constituída de quatro capítulos, a atenção será focada na restauração de imagens. No Capítulo 1 será apresentado um breve resumo das origens da derivada topológica, assim como um levantamento bibliográfico dos trabalhos mais relevantes e as suas áreas de aplicação. Em particular, será realizado o cálculo da derivada topológica para o caso da equação de difusão-reação, o qual será posteriormente utilizado na restauração de imagens. No Capítulo 2 serão estudados alguns dos mais utilizados métodos de restauração de imagens existentes na literatura baseados no Cálculo Variacional e EDP's. Estes serão

divididos em dois grupos: evolutivos e estacionários. Os métodos apresentados foram implementados computacionalmente e resultados numéricos serão apresentados para cada um deles. O uso da derivada topológica na restauração de imagens será abordado no Capítulo 3. Neste caso serão estudadas duas alternativas, uma contínua e outra discreta. No caso contínuo, que tem a sua origem no método de restauração de imagens estacionário, foi estudado um algoritmo proposto por Belaid *et. al.* [22], mostrando alguns resultados numéricos e analisando algumas possíveis melhoras. No caso discreto, será proposto um algoritmo novo de restauração de imagens baseado em uma versão discreta da derivada topológica. Contrariamente ao caso contínuo, este baseia-se na equação de difusão de calor evolutiva. Finalizando a primeira parte, o Capítulo 4 será destinado a comparar quantitativamente os métodos clássicos (expostos no Capítulo 2) e os baseados na derivada topológica (Capítulo 3) para diferentes tipos e intensidades de ruído.

A Parte II, e seguindo uma estrutura semelhante à da primeira parte, abordará o problema de segmentação de imagens. O Capítulo 5 será direcionado à análise do funcional de Mumford-Shah, o qual tem sido a motivação inicial no estudo deste problema do ponto de vista do Cálculo Variacional e EDP's. Além disso, este funcional tem motivado o surgimento de outros utilizados na literatura e, em particular, neste trabalho. O Capítulo 6 apresentará uma nova técnica de segmentação de imagens baseada na derivada topológica. Esta técnica permite segmentar uma imagem em um conjunto finito de classes utilizando o conceito de derivada topológica contínua. Posteriormente, baseado nesta idéia será apresentado um algoritmo alternativo, mais simples na sua implementação e de menor custo computacional, utilizando uma versão discreta da derivada topológica. O Capítulo 7 terá por objetivo comparar quantitativamente os métodos propostos com outros freqüentemente utilizados mostrando-se a sua performance na presença de ruído. Também serão estudados os resultados dos métodos propostos para diferentes parâmetros e sua influência nos resultados da segmentação.

A Parte III, concentrar-se-á no estudo do uso de técnicas de modelagem na simulação computacional do SCVH. Esta última parte terá por objetivo integrar as técnicas propostas nas duas primeiras partes deste trabalho com outras técnicas e modelos necessários para a modelagem e simulação computacional do SCVH. Os modelos do SCVH utilizados na atualidade serão apresentados no Capítulo 8. Neste caso, será feita a dedução do modelo unidimensional (Modelo 1D), uma simplificação das equações de Navier-Stokes, utilizado para representar as maiores artérias do corpo humano e diminuindo assim o custo computacional do problema. Serão colocadas também as equações tridimensionais completas (Modelo 3D) utilizadas para obter informações detalhadas do fluxo sanguíneo em pequenos distritos, e finalmente as condições de acoplamento necessárias para utilizar estes modelos conjuntamente. Como mencionado, o uso destes modelos, chamados multidimensionais, apresenta um grande desafio no momento de criá-los, já que não existe na atualidade uma ferramenta computacional especializada que proporcione estas funcionalidades em um mesmo ambiente. Com o intuito de suprir esta deficiência, foi desenvolvido um sistema chamado HeMoLab - Laboratório de Modelagem em Hemodinâmica. Este sistema será descrito no Capítulo 9, onde será feita uma análise das suas necessidades assim como

uma descrição dos diferentes módulos e componentes que integram esta ferramenta. Por último, no Capítulo 10, serão apresentados alguns resultados obtidos com o HeMoLab.

Finalmente, parte dos resultados obtidos durante o desenvolvimento deste trabalho foram publicados e apresentados em periódicos e congressos internacionais [28, 108, 109, 110, 111, 113, 114, 115, 116, 117, 118].

Parte I

Restauração de Imagens

Introdução

Nesta primeira parte será estudado o problema de restaurar uma imagem degradada por algum fenômeno, utilizando a derivada topológica, cujo conceito tem suas origens na área de otimização estrutural e problemas inversos. Para fornecer ao leitor um melhor entendimento deste conceito, no Capítulo 1 é feita uma breve introdução mostrando as origens da derivada topológica e os respectivos trabalhos pioneiros. Por último, é feito o cálculo da derivada para o caso particular da equação de difusão-reação o qual será posteriormente utilizado com suas devidas adaptações, na restauração de imagens.

O Capítulo 2 apresenta uma breve descrição dos métodos existentes na literatura utilizados em restauração de imagens e que são baseados no uso do Cálculo Variacional e EDP's. Neste ponto são apresentados os fundamentos matemáticos que originaram essa classe de métodos. No Capítulo 3 são apresentados dois métodos de restauração de imagens baseados no conceito da derivada topológica, sendo um deles contínuo e o outro discreto. No caso contínuo, que tem a sua origem no método de restauração de imagens estacionário, será estudado um algoritmo proposto por Belaid *et. al.* [22], mostrando-se alguns resultados numéricos e discutindo-se algumas possíveis melhoras. No caso discreto, é proposto um algoritmo novo de restauração de imagens baseado em uma versão discreta da derivada topológica. Contrariamente ao caso contínuo, este baseia-se na equação de difusão de calor evolutiva. Finalizando a primeira parte, no Capítulo 4 é feita uma comparação quantitativa entre os métodos clássicos (expostos no Capítulo 2) e os baseados na derivada topológica (Capítulo 3) para diferentes tipos e intensidades de ruído.

Os resultados apresentados em esta parte foram parcialmente publicados nos seguintes trabalhos:

- I. Larrabide, A. A. Novotny, R. A. Feijóo, and E. Taroco. A medical image enhancement algorithm based on topological derivative and anisotropic diffusion. In *Proceedings of the XXVI Iberian Latin-American Congress on Computational Methods in Engineering - CILAMCE 2005* - Guarapari, Espírito Santo, Brazil, 2005.
- I. Larrabide, R. A. Feijóo, A. A. Novotny and E. A. Taroco. Topological Derivative: A Tool for Image Processing. *Computers & Structures - An International Journal*. Editors: K. J. Bathe & B. H. V. Topping. Guest Editor for the Special Issue: C. A. Mota Soares, M. Bendsoe, K. K. Choi and J. Herskovits. Accepted for publication - December 2006.

Capítulo 1

Derivada Topológica

Uma questão que tem motivado a atividade de pesquisa em modelagem computacional nos últimos anos é obter automaticamente, segundo um critério de desempenho ou função custo, a geometria ótima (forma e topologia) do domínio de um problema sob análise. Em geral, algum tipo de restrição encontra-se associada a esta função custo. Por exemplo, a própria equação de estado (na forma de uma equação variacional), ou a uma restrição de peso/tamanho da peça sendo desenhada, etc. A alternativa clássica para tratar este tipo de situação consiste em parametrizar o domínio de interesse e procurar os valores ótimos destes parâmetros. A limitação deste tipo de método é a impossibilidade de mudar a topologia no processo (i.e., criar novos furos ou aberturas artificiais no domínio do problema). A partir de 1994 começam a aparecer uma série de trabalhos que propõem um novo paradigma o qual irá mudar a maneira na qual este tipo de problema é tratado. Nos trabalhos de Eschenauer em 1994 [52], Schumacher em 1995 [177], Sokołowski e Żochowski em 1999 [155, 182], Cea *et al.* em 1998 [40] e Garreau *et al.* em 1998 [69] e 2001 [70] foi introduzida uma forma de obter tanto a forma quanto a topologia ótima utilizando a chamada Análise de Sensibilidade Topológica.

De maneira resumida e informal este novo conceito chamado de “expansão assintótica topológica” pode ser colocado da seguinte maneira. Seja $\mathcal{J}(\Omega) = \mathcal{F}(u(\Omega))$ uma função custo arbitrária que mede a “qualidade” associada a uma determinada topologia caracterizada pela “função de estado” $u(\Omega)$ solução de uma certa equação variacional definida no domínio Ω . Dado o número positivo ϵ suficientemente pequeno, a função positiva $f(\epsilon)$ que tende a zero com ϵ e denotando como Ω_ϵ o domínio perturbado com a introdução, no ponto $\hat{\mathbf{x}}$, de um furo de tamanho infinitesimal governado por ϵ , então a “expansão assintótica topológica”

$$\mathcal{J}(\Omega_\epsilon) = \mathcal{J}(\Omega) + f(\epsilon)D_T(\hat{\mathbf{x}}) + \mathcal{O}(f(\epsilon)) \quad (1.1)$$

fornece uma estimativa do valor da função custo no domínio perturbado para ϵ suficientemente pequeno, onde D_T é conhecida como “derivada topológica” (também designada como “gradiente topológico”). Assim, a derivada topológica pode ser definida da seguinte maneira:

*A **Derivada Topológica** é uma função escalar definida em Ω indicando em cada ponto $\hat{\mathbf{x}} \in \Omega$ a sensibilidade da função custo quando um furo de tamanho infinitesimal ϵ é introduzido nesse ponto.*

Este mesmo conceito pode ser empregado para considerar outros tipos de perturbações que não necessariamente correspondam a introduzir um furo. De fato, é possível considerar perturbações, em regiões de tamanho infinitesimal, nas propriedades do material

associado ao problema (inclusões), nos termos de carregamento, nas condições de contorno, etc. [126, 130, 140, 144, 182].

Por outro lado, da expansão assintótica topológica (Eq. (1.1)) e da positividade de $f(\epsilon)$ se segue que para diminuir o valor da função custo basta introduzir as perturbações em todo ponto \hat{x} para o qual a derivada topológica seja negativa. Assim, a derivada topológica fornece uma informação importante que permite o desenvolvimento de algoritmos eficientes e robustos. Tem-se assim uma nova e poderosa ferramenta para tratar diversos problemas: otimização de forma e topologia, caracterização de propriedades, problemas inversos e inclusive, como será visto mais adiante neste trabalho, processamento de imagens.

A organização deste capítulo é a seguinte. Na próxima seção são apresentadas as publicações relacionadas com a derivada topológica mencionando inclusive as áreas nas quais a mesma tem sido aplicada. Posteriormente a derivada topológica é apresentada de maneira formal assim como o método (*Topological-Shape Sensitivity Method* [58, 140, 141, 146]) que será utilizado para seu cálculo nas aplicações deste conceito no processamento de imagens propostas no presente trabalho. Em particular, a derivada topológica será calculada para uma função custo geral, considerando como restrição o problema de condução de calor estacionária escrito na forma variacional. Este resultado será posteriormente empregado, com as devidas adaptações, na restauração e segmentação de imagens. Finalmente, as imagens assim processadas serão utilizadas na modelagem e simulação computacional do sistema cardiovascular humano.

1.1 Trabalhos pioneiros na área

Como mencionado anteriormente, otimização de forma e topologia é um problema que tem sido motivo de muita atividade de pesquisa neste e no século passado. Nos últimos anos, diversas técnicas de otimização de topologia foram propostas [50, 51] sendo que em [53] são referenciados 425 trabalhos relacionados com este tema. Estas técnicas têm a vantagem de permitir obter a topologia ótima mesmo partindo de uma configuração “distante” dela. Dentre as mais populares, destacam-se as formulações baseadas em métodos de relaxação e homogeneização que, como resultado, fornecem uma distribuição de materiais compostos. Neste caso, algum critério (filtro) deve ser aplicado para se obter um resultado realista. Estas dificuldades: escolha dos filtros, interpretação dos resultados, impossibilidade de incorporar nas bordas das perturbações as condições de contorno de Dirichlet ou de Neumann não homogêneas, impossibilidade de extensão destas técnicas para outros problemas como os de caracterização de propriedades e problemas inversos, dentre outros, motivaram então o surgimento deste novo conceito de derivada topológica.

Em 1999, aparece o primeiro trabalho em Derivada Topológica intitulado *On topological derivative in shape optimization* por Sokołowski e Żochowsky [155]. Neste trabalho os autores calculam a derivada topológica usando o chamado “Método da derivada material” por Sokołowski e Zolésio [158], e dois exemplos são apresentados como aplicação. No primeiro os autores consideram uma equação elíptica geral apresentando resultados para variantes do problema de Laplace, e no segundo é considerada a equação de elasticidade

plana. Em ambos os casos consideram condições de contorno de Neumann homogêneas no furo.

Outra área de aplicação da derivada topológica é em problemas inversos [36]. Em 1999 Sokołowski e Żochowski [182] publicam o primeiro trabalho desta ferramenta na área, no qual a derivada é calculada para a equação de Laplace 3D com o intuito de identificar pequenas inclusões utilizando medidas realizadas no contorno do domínio. No mesmo ano, Lewiński e Sokołowski [119] calculam a derivada topológica para o caso de furos de forma arbitrária, para a função custo relacionada à energia do problema de elasticidade 2D com condição de Neumann.

Em 2000, Cea *et al.* [40] estendem o conceito de derivada topológica para problemas com condição de Dirichlet no furo, mostrando aplicações em electromecânica.

No trabalho de Garreau *et al.* [70] é obtida a expansão assintótica topológica do problema de elasticidade para funcionais gerais e furos de forma arbitrária utilizando uma adaptação do “Método Adjunto” e o “Método do domínio truncado” introduzido por Masmoudi [129]. No caso, os autores estudam o problema com condições de Neumann e Dirichlet homogêneas, mostrando que são problemas diferentes e devem ser analisados de maneira diferente. A propriedade atraente do Método Adjunto é a imediata generalização para uma vasta classe de funções custo, não apenas para o caso da energia. Além do mais, as expressões de sensibilidade são escritas explicitamente em termos das soluções das equações de estado e adjunta, ambas associadas ao domínio sem perturbação.

No ano 2002, Guillaume e Sid Idris [80] apresentam uma adaptação do método descrito por S. Garreau *et al.* [70] que permite estender o cálculo da derivada topológica para furos de forma arbitrária. Em particular esta derivada é calculada para o problema de Poisson com condições de Neumann e Dirichlet ambas homogêneas, considerando diversas funções custo. No caso de condição de contorno de Dirichlet no furo, é demonstrado que em 3D, a sensibilidade topológica depende da forma do furo mas não da sua orientação se a função custo depende somente da solução da equação de estado. No entanto, irá depender da orientação no caso em que a função custo dependa do gradiente da solução.

No mesmo ano, Novotny *et al.* [143, 144] estabelecem uma relação formal entre a derivada topológica e a análise de sensibilidade à mudança de forma, aplicando-a ao cálculo da derivada para o problema de elasticidade 2D. Este método fornece um marco para o cálculo da derivada topológica utilizando resultados clássicos de análise de sensibilidade a mudança de forma [146]. Esta alternativa, chamada *Topological-Shape Sensitivity Method*, fornece uma maneira simples e geral para o cálculo da derivada topológica. Feijóo *et al.* [57] apresentam uma comparação entre o método de domínio truncado e o *Topological-Shape Sensitivity Method* para o problema de Poisson, considerando um conjunto de condições de contorno mais geral e estendendo os resultados obtidos anteriormente por Garreau *et al.* [70] e Novotny *et al.* [143].

No ano 2003 são publicadas duas teses [7, 140] na área. Na tese de Novotny [140], é estudada a sensibilidade topológica para os problemas de condução de calor em sólidos rígidos com condições de contorno de Neumann, Robin e Dirichlet; de elasticidade linear plana; flexão elástica linear de placas de Kirchhoff; torção de barras sujeitas a fluência; e

no problema de Poisson é introduzido um novo tipo de perturbação: inclusão de materiais com propriedades diferentes neste caso a derivada topológica é chamada pelo autor de derivada configuracional já que neste tipo de perturbação a topologia não é modificada mas sim a configuração onde está definido o problema. Utilizando o Método Adjunto, na tese de Amstutz [7] a derivada topológica é empregada na localização de fissuras nos problemas de Laplace 2D; Helmholtz 2D com condição de Dirichlet em um furo circular, condição de Neumann em um furo de forma arbitrária ou fissura e inserção de inclusões com diferente propriedade material; e finalmente para problemas não lineares com condição de Dirichlet e furo de forma arbitrária, aplicando-o para a equação de Navier-Stokes.

Lewiński e Sokołowski [120] introduzem a noção de derivada topológica direcional na aparição de furos não-circulares (ou cavidades não esféricas em 3D) no caso de elasticidade. Neste trabalho, os autores provam que o método *bubble* anteriormente utilizado por Schumacher [177] coincide com a dissipação da energia elástica ao aparecer um furo (ou cavidade) no domínio, abrindo um novo campo de aplicação de derivada topológica: cavitação. Samet *et al.* [174] calculam a derivada topológica para o problema de Helmholtz com condição de Dirichlet homogênea no furo.

A idéia da derivada topológica é estendida por Sokołowski [154] para considerar um número finito de furos (utilizando a derivada topológica) e também variações no domínio (utilizando análise de sensibilidade à mudança de forma). Combinando estas duas técnicas são obtidas as condições de optimalidade¹ para o problema de otimização de forma.

O *Topological-Shape Sensitivity Method* é utilizado por Feijóo *et al.* [58] para calcular a derivada topológica para o problema de elasticidade 2D considerando a energia potencial total como função custo e a correspondente equação de equilíbrio como restrição. Resultados anteriormente obtidos por Guillaume *et al.* para o problema de Poisson são estendidos para o problema de Stokes por Guillaume e Sid Idris [81]. Os autores destacam que no caso 3D a sensibilidade topológica depende da forma do obstáculo (ou furo) mas não no caso 2D. Contudo, existe uma diferença com o caso de Poisson 3D em que a derivada topológica pode depender da orientação do obstáculo. Uma generalização da técnica de *level sets*, usualmente utilizada em otimização de topologia, incluindo derivada topológica é proposta por Burger *et al.* [35].

Nazarov e Sokołowski [136] introduzem a chamada derivada topológica exterior. Esta derivada é definida pela variação de topologia na forma de um ligamento delgado conectando duas pequenas partes do contorno, fora do domínio original. Segundo apontado pelos autores, o método de *level sets* descrito em [35] poderia ser melhorado pelo uso desta derivada ao criar furos adicionais.

Seguindo essa linha, Amstutz e Andrä [10, 11] propõem um novo algoritmo de otimização de topologia utilizando uma nova equação de evolução para a função de *level set* baseada em uma generalização do conceito da derivada topológica. Neste caso, para diminuir a quantidade de mínimos locais, é utilizada uma técnica de filtragem que atua como uma regularização. Diferentemente do feito por Burger *et al.* [35], os autores abandonam completamente a equação de Hamilton-Jacobi, evitando o uso de parâmetros

¹Condição que deve ser satisfeita pelo mínimo em um problema de otimização.

arbitrários para construir a correção em iterações sucessivas.

Em 2005 é publicado por Amstutz [8] o trabalho com o cálculo da derivada topológica para equações de Navier-Stokes no caso estacionário e fluido incompressível. O principal problema neste cálculo vem da não-linearidade do operador envolvido. A condição de contorno no furo (ou obstáculo) é de Dirichlet. Esta ferramenta pode ser utilizada para resolver problemas de otimização topológica em dinâmica dos fluidos. Uma versão geral deste trabalho é apresentada por Amstutz [9], no qual a derivada topológica é calculada para uma classe de problemas não-lineares em particular para Navier-Stokes e a versão não-linear da equação de Helmholtz.

A derivada topológica também tem sido utilizada na detecção de falhas. Amstutz *et al.* [12] realizam a análise de sensibilidade topológica para a equação de Laplace com respeito a introdução de uma fenda (ou talho) com condição de Neumann homogênea. Esta derivada é utilizada em um algoritmo para detecção de fissuras no problema geométrico inverso. Para este fim é utilizado o critério de Kohn-Vogelius [102] como função custo. A derivada topológica para problema de contato (problema de Signorini e contato sem fricção em elasticidade linear) é calculada por Sokołowski e Żochowski [156]. Sokołowski e Żochowski [157] obtêm a derivada topológica para o problema de obstáculo e é construída a aproximação assintótica da solução para uma perturbação do domínio pela criação de um furo.

Novotny *et al.* [142, 145, 147] utilizam o *Topological-Shape Sensitivity Method* para calcular a derivada topológica no problema de elasticidade linear 3D e no problema de p-Poisson (ou p-Laplaceano). Novotny *et al.* [141] utilizam este método para calcular a derivada no problema de placas elásticas no modelo cinemático de Kirchhoff. A derivada topológica é utilizada também no problema inverso de identificação de propriedades para equação de calor a partir de medidas na fronteira do domínio por Guinzani F. *et al.* [56]. Neste trabalho os autores apresentam um algoritmo iterativo baseado na informação da derivada topológica.

Masmoudi *et al.* [130] apresentam a expansão assintótica topológica para as equações de Maxwell. Também, esta informação é utilizada em um método de otimização topológica para resolver problemas inversos de identificação de propriedades eletromagnéticas.

Na Tabela 1.2 é apresentado um resumo das equações/problemas para os quais a derivada topológica tem sido calculada, assim como das publicações relacionadas.

1.2 Motivação para o uso da Derivada Topológica em Processamento de Imagens

Talvez a característica mais notável da derivada topológica é que permite calcular a variação de uma função custo com respeito a um parâmetro que pode variar de maneira não suave (e.g., função característica de um domínio, propriedades materiais e/ou mudança não contínua das forças que atuam no problema, etc.).

Assim, por exemplo, esta derivada pode ser utilizada para identificar a função característica de um domínio ótimo segundo algum critério (e.g., medida de performance,

função custo, etc.), para identificar propriedades materiais e sua distribuição em um certo domínio, ou até as forças atuando em um certo domínio e como elas estão distribuídas.

Por outro lado, este tipo de problema aparece freqüentemente no contexto de processamento de imagens:

- *Identificação de bordas*: a caracterização de bordas dentro de uma imagem é um processo importante na identificação de regiões e remoção de ruído dentre outras. As bordas dentro de uma imagem podem ser identificadas como um conjunto de descontinuidades.
- *Identificação de objetos*: o processo de segmentação identifica dentro de uma imagem diferentes objetos ou regiões. Identificar a região ocupada por um objeto é equivalente a identificar a função característica desta região.
- *Acompanhamento de objetos em seqüências de imagens (object tracking)*: como no caso de segmentação, acompanhar um objeto que aparece em uma seqüência de imagens é equivalente a achar a função característica da região que ele ocupa.
- *Decomposição em textura e geometria*: uma imagem pode ser interpretada como uma soma de textura e geometria. Neste caso, no processo de decomposição, é de interesse separar, na imagem tratada, a textura e a geometria. A geometria pode ser identificada pela sua função característica.
- *Reconstrução a partir de projeções*: no processo de reconstrução de uma imagem a partir das projeções objetiva-se identificar a propriedade material para cada região da imagem (ou objeto) projetada.

Em todos os casos o uso da derivada topológica aparece como uma alternativa natural para resolver estes problemas.

Em 2005 surgem os primeiros trabalhos de derivada topológica aplicada no processamento de imagens: restauração por Belaid *et al.* [22] e Larrabide *et al.* [113, 114], onde o objetivo é recuperar uma imagem que sofreu algum tipo de degradação; segmentação por Larrabide *et al.* [109, 110] e por Hintermüller [89], por exemplo em imagens médicas onde é de interesse identificar os diferentes órgãos para posterior reconstrução, diagnose, aplicação de algum tratamento ou simulação bio-mecânica do funcionamento do corpo humano; e classificação de imagens por Auroux *et al.* [18]. He e Osher [87] estabelecem uma relação entre a derivada topológica e outras técnicas amplamente utilizadas em processamento de imagens como *level sets*.

1.3 Derivada Topológica - Formulação Variacional

Como já mencionado a Análise de Sensibilidade Topológica permite caracterizar a sensibilidade do problema sob consideração quando o domínio Ω no qual este está definido é perturbado de alguma maneira. Esta perturbação poderá ser:

Segundo o tipo de perturbação		Condição de contorno no furo	
B	Furo circular	NH	Neumann Homogênea
E	Furo elíptico	N	Neumann
C	Fenda	DH	Dirichlet Homogênea
A	Furo arbitrário	D	Dirichlet
I	Inclusão	R	Robin

Problema de aplicação		Métodos de cálculo da derivada	
IP	Problemas Inversos	MD	Derivada Material
TO	Otimização de Topologia	AM	Método Adjunto
FI	Identificação de Fissuras	DTM	Método de domínio truncado
		TSSM	Topological-Shape Sensitivity Method

Tabela 1.1: Nomenclatura

- Mudança de topologia: Neste caso o domínio Ω é perturbado introduzindo em um ponto $\hat{\mathbf{x}} \in \Omega$ um furo de forma arbitrária, ω_ϵ e a derivada topológica fornece a sensibilidade da função custo quando $\epsilon \rightarrow 0$ (ver Eq. (1.1)).
- Mudança nas propriedades materiais: Neste caso é introduzida em um ponto $\hat{\mathbf{x}} \in \Omega$ uma perturbação nas propriedades do material definida na região de forma arbitrária ω_ϵ e a derivada topológica fornece a sensibilidade da função custo quando $\epsilon \rightarrow 0$ (ver Eq. (1.1)). As “propriedades do material” correspondem aos coeficientes que definem a equação variacional associada ao problema.
- Mudança nas forças/fontes que atuam em Ω : Similar ao caso anterior entretanto agora a perturbação é realizada sobre os valores das fontes.

A seguir, e por simplicidade, será tratado apenas o primeiro caso, isto é o da perturbação do domínio com a introdução de um furo. A extensão para os outros dois casos é similar e não apresenta nenhuma dificuldade.

Seja então um problema onde sua qualidade/performance esteja caracterizada por uma função custo $\mathcal{J}(\Omega) = \mathcal{F}(\Omega, u(\Omega))$ onde $\Omega \subset \mathbb{R}^n, n = 2, 3$, é um domínio aberto de contorno $\partial\Omega$ regular com normal exterior \mathbf{n} . Com a notação $(\Omega, u(\Omega))$ se deseja ressaltar que \mathcal{F} depende de Ω explicitamente e implicitamente através de $u(\Omega)$, solução da equação variacional (equação de estado), que pode ser escrita na forma abstrata como: determinar $u \in \mathcal{U} = \mathcal{U}(\Omega)$ tal que

$$a(u, \eta) = l(\eta) \quad \forall \eta \in \mathcal{V}, \quad (1.2)$$

onde \mathcal{U} caracteriza o conjunto (geralmente uma variedade linear de \mathcal{V}) de funções admissíveis definidas em Ω e $\mathcal{V} = \mathcal{V}(\Omega)$ o espaço vetorial de suas variações admissíveis. Por sua vez $a(., .) : \mathcal{U} \times \mathcal{V} \mapsto \mathbb{R}$ é uma forma bilinear simétrica e $l(.) : \mathcal{V} \mapsto \mathbb{R}$ uma forma linear (ver Apêndice 1.A). Admite-se ainda que estas formas satisfazem propriedades de continuidade e coercividade de maneira a garantir a existência e unicidade da solução u (ver Apêndice).

Seja ainda ω um aberto (de forma arbitrária) de contorno regular $\partial\omega$ contendo a origem. Dado $\epsilon > 0$ suficientemente pequeno pode-se definir para qualquer ponto $\hat{\mathbf{x}} \in \Omega$ o domínio ω_ϵ dado por $\omega_\epsilon = \hat{\mathbf{x}} + \epsilon\omega$. Desta maneira, a introdução de um furo ω_ϵ centrado

Problema	Características do Problema	Método utilizado no cálculo da derivada	Referências
Laplace 2D	NH, DH (TO) NH, DH, R (TO) NH (F,FI)	MD TSSM AM + DTM	[155], [137], [138] [140] [7], [12]
Laplace 3D	NH (IP)	MD	[182]
Elasticidade 2D/3D	NH NH, DH (A) NH NH (A)	MD AM + DTM TSSM MD	[119] [70] [140], [147], [142], [145], [57] [120]
Placas (Kirchhoff)	NH	TSS	[141]
Poisson 3D/2D	NH, DH I(IP) R	AM + DTM TSSM TSSM	[80] [140], [56] [57]
p-Poisson 2D	—	TSSM	[140], [145]
Maxwell	—	AM + DTM	[130]
Helmholtz 2D	DH(B), NH(A,C)	AM + DTM	[7], [9], [174]
Navier-Stokes	DH(A)	AM + DTM	[7], [8]
Stokes	DH(A)	AM + DTM	[81]
Contato (Signorini)	—	MD	[156]
Obstáculo	A	MD	[157]

Tabela 1.2: Problemas para os quais a Derivada Topológica já foi calculada. Ver Tabela 1.1 para nomenclatura.

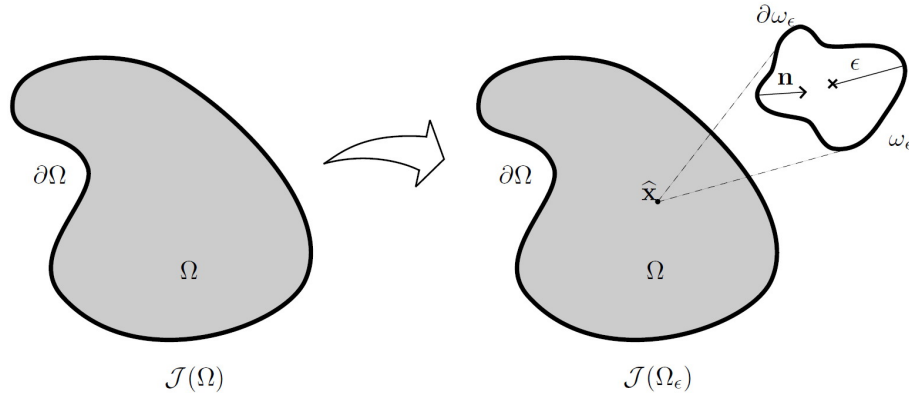


Figura 1.1: Derivada Topológica - Mudança na topologia

em $\hat{\mathbf{x}} \in \Omega$ permite caracterizar o domínio perturbado Ω_ϵ (Fig. 1.1) dado por

$$\Omega_\epsilon = \Omega \setminus \overline{\omega_\epsilon}.$$

Da Eq. (1.1), a derivada topológica em $\hat{\mathbf{x}} \in \Omega$ pode ser definida como

$$D_T(\hat{\mathbf{x}}) = \lim_{\epsilon \rightarrow 0} \frac{\mathcal{J}(\Omega_\epsilon) - \mathcal{J}(\Omega)}{f(\epsilon)} \quad (1.3)$$

onde $f(\epsilon)$ é uma função positiva monótona decrescente ($f(\epsilon) \rightarrow 0$ com $\epsilon \rightarrow 0$). Além do mais, $\mathcal{J}(\Omega_\epsilon) = \mathcal{F}(\Omega_\epsilon, u_\epsilon(\Omega_\epsilon))$, sendo que u_ϵ é a solução da mesma equação de estado entretanto agora definida no domínio perturbado, isto é em Ω_ϵ : Determinar $u_\epsilon \in \mathcal{U}_\epsilon = \mathcal{U}(\Omega_\epsilon)$ tal que

$$a_{\Omega_\epsilon}(u_\epsilon, \eta) = l_{\Omega_\epsilon}(\eta) \quad \forall \eta \in \mathcal{V}_\epsilon = \mathcal{V}(\Omega_\epsilon). \quad (1.4)$$

Em [146] é estabelecida uma relação entre a derivada topológica e a Análise de Sensibilidade a Mudança de Forma clássica [86, 134]. Este resultado permite utilizar ferramentas desenvolvidas em análise de sensibilidade clássica para o cálculo desta derivada. Esta nova abordagem para o cálculo da derivada topológica pode ser colocada no seguinte teorema:

Teorema 1. *Seja $f(\epsilon)$ uma função escolhida de forma que $0 < |D_T(\hat{\mathbf{x}})| < \infty$, logo o limite com $\epsilon \rightarrow 0$ que aparece em (1.3), pode ser escrito como*

$$D_T(\hat{\mathbf{x}}) = \lim_{\epsilon \rightarrow 0} \frac{1}{f'(\epsilon)} \left. \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \right|_{\tau=0} \quad (1.5)$$

onde $\frac{d\mathcal{J}(\Omega_\tau)}{d\tau}$ é a sensibilidade a mudança de forma clássica.

Prova 1. *O leitor interessado pode procurar em [146] a prova deste teorema.*

É importante notar que na expressão anterior está implícita uma transformação de domínio (deformação) $\chi_\tau : \mathbf{x}_\epsilon \in \Omega_\epsilon \rightarrow \mathbf{x}_\tau \in \Omega_\tau$ definida por

$$\mathbf{x}_\tau = \mathbf{x}_\epsilon + \tau \mathbf{v}(\mathbf{x}_\epsilon)$$

onde \mathbf{v} é o campo de velocidade que caracteriza a mudança de forma e $\Omega_\tau|_{\tau=0} = \Omega_\epsilon$. Por sua vez o campo \mathbf{v} esta caracterizado por

$$\mathbf{v}(\mathbf{x}) = -\mathbf{n} \quad \forall \mathbf{x} \in \partial\omega_\epsilon \text{ e } \mathbf{v}(\mathbf{x}) = 0 \quad \forall \mathbf{x} \in \partial\Omega_\epsilon \setminus \partial\omega_\epsilon. \quad (1.6)$$

Para maiores informações sobre este tipo de transformação ver Haug *et al.* [86] e Haug e Cea [85], Pironneau [166], Sokołowski *et al.* [158] e Zolezio [211].

Deste teorema é naturalmente deduzido o *Topological-Shape Sensitivity Method*, o qual será explorado a seguir. Formalmente, a derivada à mudança de forma da função custo em relação ao parâmetro τ pode ser escrita como

$$\begin{cases} \text{Calcular :} & \frac{d}{d\tau} \mathcal{J}_\tau(u_\tau) \\ \text{sujeito a :} & a_\tau(u_\tau, \eta) = l_\tau(\eta) \quad \forall \eta \in \mathcal{V}_\tau. \end{cases}$$

onde $a_{\Omega_\tau}(\cdot, \cdot)$ denota-se com $a_\tau(\cdot, \cdot)$, $l_{\Omega_\tau}(\cdot)$ com $l_\tau(\cdot)$ e onde com a notação $\mathcal{J}_\tau(u_\tau)$ se deseja colocar em evidência a dependência em u_τ da função custo.

Para calcular a derivada à mudança de forma considerando a equação de estado como restrição, pode ser utilizado o método Lagrangiano (i.e., relaxando a restrição pela introdução de um multiplicador de Lagrange). O Lagrangiano deste problema é escrito

como

$$\mathcal{L}_\tau(v, \eta) = \mathcal{J}_\tau(v) + a_\tau(v, \eta) - l_\tau(\eta) \quad \forall \eta \in \mathcal{V}_\tau \text{ e } v \in \mathcal{U}_\tau. \quad (1.7)$$

Verifica-se então que, para o caso $v = u_\tau$, tem-se que

$$\begin{aligned} \mathcal{L}_\tau(u_\tau, \eta) &= \mathcal{J}_\tau(u_\tau) + \underbrace{a_\tau(u_\tau, \eta) - l_\tau(\eta)}_{=0, \text{ solução equação de estado}} \quad \forall \eta \in \mathcal{V}_\tau \\ &= \mathcal{J}_\tau(u_\tau) \quad \forall \eta \in \mathcal{V}_\tau. \end{aligned}$$

Ao calcular a derivada com respeito de τ da Eq. (1.7) tem-se

$$\frac{d\mathcal{L}_\tau(v, \eta)}{d\tau} = \frac{\partial \mathcal{L}_\tau}{\partial \tau} + \left\langle \frac{\partial \mathcal{L}_\tau}{\partial v}; \frac{dv}{d\tau} \right\rangle + \left\langle \frac{\partial \mathcal{L}_\tau}{\partial \eta}; \frac{d\eta}{d\tau} \right\rangle. \quad (1.8)$$

Analisa-se então a expressão (1.8) termo a termo. Começando pelo terceiro termo do lado direito, tem-se

$$\left\langle \frac{\partial \mathcal{L}_\tau}{\partial \eta}; \frac{d\eta}{d\tau} \right\rangle = a_\tau \left(v, \frac{d\eta}{d\tau} \right) - l_\tau \left(\frac{d\eta}{d\tau} \right) \quad \forall \frac{d\eta}{d\tau} \in \mathcal{V}_\tau \quad (1.9)$$

Logo, para o caso particular $v = u_\tau$, Eq. (1.9) é zero. Considerando o segundo termo de (1.8), tem-se

$$\left\langle \frac{\partial \mathcal{L}_\tau}{\partial v}; \frac{dv}{d\tau} \right\rangle = \left\langle \frac{\partial \mathcal{J}_\tau}{\partial v}; \frac{dv}{d\tau} \right\rangle + a_\tau \left(\eta, \frac{dv}{d\tau} \right) \quad \forall \frac{dv}{d\tau} \in \mathcal{V}_\tau \quad (1.10)$$

onde foi utilizada a simetria de $a_\tau(\cdot, \cdot)$. Nesta expressão, η pode ser escolhido de maneira arbitrária. Em particular, será escolhido $\eta = q_\tau$, sendo $q_\tau \in \mathcal{V}_\tau$ a solução da equação adjunta dada por

$$\left\langle \frac{\partial \mathcal{J}_\tau}{\partial v}; \frac{dv}{d\tau} \right\rangle \Big|_{v=u_\tau} + a_\tau \left(q_\tau, \frac{dv}{d\tau} \right) = 0 \quad \forall \frac{dv}{d\tau} \in \mathcal{V}_\tau. \quad (1.11)$$

A equação anterior é conhecida como “equação (variacional) adjunta”, e sua solução q_τ (ou q_ϵ e q se a equação adjunta estiver definida no domínio Ω_ϵ e Ω respectivamente) como “solução adjunta”. É importante observar aqui que, pelas propriedades da forma $a(\cdot, \cdot)$, a equação adjunta é do mesmo tipo que a equação de estado Eq. (1.2) (ou a equação no domínio perturbado Eq. (1.4)). Do ponto de vista computacional, o anterior significa que o mesmo sistema computacional empregado para calcular a solução da equação de estado u (ou u_ϵ) poderá ser empregado para calcular q (ou q_ϵ).

Finalmente, a derivada total do Lagrangiano com respeito ao parâmetro τ é dada por

$$\begin{aligned} \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \Big|_{\tau=0} &= \frac{\partial \mathcal{L}_\tau(v, \eta)}{\partial \tau} \Big|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} \\ &= \left[\frac{\partial \mathcal{J}_\tau(v)}{\partial \tau} + \frac{\partial a_\tau(v, \eta)}{\partial \tau} - \frac{\partial l_\tau(\eta)}{\partial \tau} \right] \Big|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0}. \end{aligned} \quad (1.12)$$

Deve-se notar que $u_\tau|_{\tau=0} = u_\epsilon$ e $q_\tau|_{\tau=0} = q_\epsilon$. Desta maneira a expressão anterior resulta uma função de u_ϵ e q_ϵ e de suas derivadas. Como já observado, somente o contorno $\partial\omega_\epsilon$ é perturbado por uma expansão uniforme (Eq. (1.6)). Logo, a derivada à mudança de forma resulta em uma integral definida somente na borda $\partial\omega_\epsilon$. Assim, a derivada topológica é dada por uma expressão da forma

$$D_T(\hat{\mathbf{x}}) = -\lim_{\epsilon \rightarrow 0} \frac{1}{f'(\epsilon)} \int_{\partial\omega_\epsilon} \boldsymbol{\Sigma}_\epsilon \mathbf{n} \cdot \mathbf{n} \, d\omega_\epsilon \quad (1.13)$$

onde $\boldsymbol{\Sigma}_\epsilon$ depende de u_ϵ e q_ϵ , e pode ser interpretado como uma generalização do tensor momento-energia de Eshelby [54]. O tensor $\boldsymbol{\Sigma}_\epsilon$ deve ser identificado para cada problema particular sendo tratado, dependendo da função custo adotada e da equação de estado associada a u_ϵ .

Por último é necessário calcular o limite quando $\epsilon \rightarrow 0$ para obter a expressão da derivada topológica. Para isto é necessário estudar o comportamento das soluções u_ϵ e q_ϵ quando $\epsilon \rightarrow 0$. Este comportamento pode ser obtido realizando a análise assintótica ao redor de uma vizinhança do furo. Diferentes alternativas podem ser utilizadas neste ponto dependendo do problema sob consideração (i.e., condições de contorno, tipo de perturbação, etc.). Em todos os casos esta análise assintótica permite expressar u_ϵ e q_ϵ como uma função de ϵ , $u(\hat{\mathbf{x}})$, $q(\hat{\mathbf{x}})$ e de suas derivadas em $\hat{\mathbf{x}}$ respectivamente. Isto é como funções das soluções das equações de estado e adjunta definidas no domínio **não** perturbado fornecendo também a função $f(\epsilon)$. Assim, da substituição destas expansões na Eq. (1.13) e do cálculo do limite para $\epsilon \rightarrow 0$ obtém-se a expressão final da derivada topológica no ponto $\hat{\mathbf{x}}$ que, como mencionado, dependerá somente do valor de u e q e de suas derivadas nesse ponto. O anterior tem conseqüências importantíssimas do ponto de vista computacional. De fato, uma vez obtidos u e q o cálculo da derivada topológica corresponde computacionalmente a um simples pós-processamento.

Então, para uma dada função custo $\mathcal{F}(\Omega, u(\Omega))$ o *Topological-Shape Sensitivity Method* pode ser resumido nos seguintes passos:

1. Calcular a derivada à mudança de forma para a função custo $\mathcal{F}(\Omega_\epsilon, u_\epsilon(\Omega_\epsilon))$, usando o método Lagrangiano.
2. Identificar o tensor de momento-energia de Eshelby $\boldsymbol{\Sigma}_\epsilon$ e escrever a expressão de sensibilidade como uma integral definida na borda $\partial\omega_\epsilon$.
3. Fazer a análise assintótica para estudar o comportamento das soluções u_ϵ e q_ϵ quando $\epsilon \rightarrow 0$.
4. Da análise assintótica escolher a função $f(\epsilon)$.
5. Calcular a derivada topológica usando a Eq. (1.13).

Tem-se então uma forma geral e sistemática de calcular a derivada topológica para uma função custo qualquer. Para dar ao leitor uma idéia mais clara de como é feito este cálculo, a seguir será estudado um problema particular onde a equação de estado

está caracterizada pelo problema de condução estacionária de calor. Como se verá mais adiante, esta equação desempenha um papel importante no processamento de imagens.

1.4 Equação de calor estacionária

Como mostrado ao longo deste trabalho, a equação de difusão tem sido amplamente empregada para resolver problemas de restauração e segmentação de imagens. Por este motivo é apresentado nesta seção o cálculo detalhado da sensibilidade a mudança de forma para esta equação e sua posterior utilização no cálculo da correspondente derivada topológica. Os resultados aqui obtidos serão úteis nos próximos capítulos e permitirão ao leitor ter uma visão mais clara dos passos seguidos no cálculo desta derivada.

Considera-se neste caso uma função custo da forma

$$\mathcal{J}(\Omega) = \int_{\Omega} J(u, \nabla u) \, d\Omega,$$

com $u \in \mathcal{U}$ solução da equação (1.2) para

$$a(u, \eta) = \int_{\Omega} (k(\mathbf{x}) \nabla u \cdot \nabla \eta + u \eta) \, d\Omega \quad \text{e} \quad l(\eta) = \int_{\Omega} b(\mathbf{x}) \eta \, d\Omega \quad \eta \in \mathcal{V}, \quad (1.14)$$

com

$$\mathcal{U} = \{u : u \in H^1(\Omega)\} \quad \text{e} \quad \mathcal{V} = \{\eta : \eta \in H^1(\Omega)\}.$$

A equação de Euler-Lagrange associada a este problema é dada por

$$\begin{cases} -\operatorname{div}(k \nabla u) + u = b & \mathbf{x} \in \Omega \\ \frac{\partial u}{\partial \mathbf{n}} = 0 & \mathbf{x} \in \partial\Omega. \end{cases}$$

Ao se considerar o domínio perturbado Ω_ϵ a função custo é dada por

$$\mathcal{J}(\Omega_\epsilon) = \int_{\Omega_\epsilon} J(u_\epsilon, \nabla u_\epsilon) \, d\Omega_\epsilon,$$

sendo $u_\epsilon \in \mathcal{U}_\epsilon$ a solução da Eq. (1.4), para

$$a_\epsilon(u_\epsilon, \eta) = \int_{\Omega_\epsilon} (k(\mathbf{x}) \nabla u_\epsilon \cdot \nabla \eta + u_\epsilon \eta) \, d\Omega_\epsilon \quad \text{e} \quad l_\epsilon(\eta) = \int_{\Omega_\epsilon} b(\mathbf{x}) \eta \, d\Omega_\epsilon \quad \eta \in \mathcal{V}_\epsilon,$$

com

$$\mathcal{U}_\epsilon = \{u : u \in H^1(\Omega_\epsilon)\} \quad \text{e} \quad \mathcal{V}_\epsilon = \{\eta : \eta \in H^1(\Omega_\epsilon)\}.$$

A equação de Euler-Lagrange associada a este problema é dada por

$$\begin{cases} -\operatorname{div}(k \nabla u_\epsilon) + u_\epsilon = b & \mathbf{x} \in \Omega_\epsilon \\ \frac{\partial u_\epsilon}{\partial \mathbf{n}} = 0 & \mathbf{x} \in \partial\Omega_\epsilon. \end{cases} \quad (1.15)$$

Da mesma maneira, a equação adjunta definida no domínio não perturbado está

dada por: Achar $q \in \mathcal{V}$ tal que

$$\int_{\Omega} (k(\mathbf{x}) \nabla q \cdot \nabla \eta + q \eta) d\Omega = - \int_{\Omega} \left[\frac{\partial J}{\partial v}(v, \nabla v) \eta + \frac{\partial J}{\partial \nabla v}(v, \nabla v) \cdot \nabla \eta \right] d\Omega \Big|_{v=u} \quad \forall \eta \in \mathcal{V}_{\epsilon}.$$

A equação de Euler-Lagrange associada a este problema é dada por

$$-div(k \nabla q) + q = - \left[\frac{\partial J}{\partial v}(v, \nabla v) - div \left(\frac{\partial J}{\partial \nabla v}(v, \nabla v) \right) \right]_{v=u}. \quad (1.16)$$

Da mesma maneira, a equação adjunta definida no domínio perturbado é a solução de: Achar $q_{\epsilon} \in \mathcal{V}_{\epsilon}$ tal que

$$\int_{\Omega_{\epsilon}} (k(\mathbf{x}) \nabla q_{\epsilon} \cdot \nabla \eta + q_{\epsilon} \eta) d\Omega_{\epsilon} = - \int_{\Omega_{\epsilon}} \left[\frac{\partial J}{\partial v}(v, \nabla v) \eta + \frac{\partial J}{\partial \nabla v}(v, \nabla v) \cdot \nabla \eta \right] d\Omega_{\epsilon} \Big|_{v=u_{\epsilon}} \quad \forall \eta \in \mathcal{V}_{\epsilon},$$

e a equação de Euler-Lagrange associada é dada por

$$-div(k \nabla q_{\epsilon}) + q_{\epsilon} = - \left[\frac{\partial J}{\partial v}(v, \nabla v) - div \left(\frac{\partial J}{\partial \nabla v}(v, \nabla v) \right) \right]_{v=u_{\epsilon}}. \quad (1.17)$$

Em seguida, a sensibilidade à mudança de forma da função custo, dada pela expressão (1.12), será obtida termo a termo. Considerando o primeiro termo e as identidades (1.39), (1.41) e (1.40) do Apêndice 1.B, tem-se

$$\begin{aligned} \frac{\partial \mathcal{J}_{\tau}(v)}{\partial \tau} \Big|_{\substack{v=u_{\tau} \\ \eta=q_{\tau}}} \Big|_{\tau=0} &= \frac{\partial}{\partial \tau} \int_{\Omega_{\tau}} J(v, \nabla v) d\Omega_{\tau} \Big|_{\substack{v=u_{\tau} \\ \eta=q_{\tau}}} \Big|_{\tau=0} \\ &= \frac{\partial}{\partial \tau} \int_{\Omega_{\epsilon}} J(v, \mathbb{F}_{\tau}^{-T} \nabla_m v) \det \mathbb{F}_{\tau} d\Omega_{\epsilon} \Big|_{\substack{v=u_{\tau} \\ \eta=q_{\tau}}} \Big|_{\tau=0} \\ &= \int_{\Omega_{\epsilon}} \frac{\partial}{\partial \tau} (J(v, \mathbb{F}_{\tau}^{-T} \nabla_m v) \det \mathbb{F}_{\tau}) d\Omega_{\epsilon} \Big|_{\substack{v=u_{\tau} \\ \eta=q_{\tau}}} \Big|_{\tau=0}. \end{aligned} \quad (1.18)$$

Calculando a derivada dentro da integral tem-se

$$\begin{aligned} \frac{\partial}{\partial \tau} (J(v, \mathbb{F}_{\tau}^{-T} \nabla_m v) \det \mathbb{F}_{\tau}) &= \frac{\partial J}{\partial (\mathbb{F}_{\tau}^{-T} \nabla_m v)}(v, \mathbb{F}_{\tau}^{-T} \nabla_m v) \cdot \frac{\partial}{\partial \tau} (\mathbb{F}_{\tau}^{-T} \nabla_m v) \det \mathbb{F}_{\tau} \\ &\quad + J(v, \mathbb{F}_{\tau}^{-T} \nabla_m v) \frac{\partial \det \mathbb{F}_{\tau}}{\partial \tau}. \end{aligned}$$

Substituindo em Eq. (1.18), e utilizando as identidades (1.45), (1.46), (1.41), (1.42) e (1.43), tem-se

$$\begin{aligned} \frac{\partial \mathcal{J}_{\tau}(v)}{\partial \tau} \Big|_{\substack{v=u_{\tau} \\ \eta=q_{\tau}}} \Big|_{\tau=0} &= \int_{\Omega_{\epsilon}} \left[-\nabla u_{\epsilon} \cdot \nabla \mathbf{v} \frac{\partial J}{\partial \nabla u_{\epsilon}}(u_{\epsilon}, \nabla u_{\epsilon}) + J(u_{\epsilon}, \nabla u_{\epsilon}) \mathbb{I} \cdot \nabla \mathbf{v} \right] d\Omega_{\epsilon} \\ &= \int_{\Omega_{\epsilon}} \left[J(u_{\epsilon}, \nabla u_{\epsilon}) \mathbb{I} - \left(\nabla u_{\epsilon} \otimes \frac{\partial J}{\partial \nabla u_{\epsilon}}(u_{\epsilon}, \nabla u_{\epsilon}) \right) \right] \cdot \nabla \mathbf{v} d\Omega_{\epsilon}. \end{aligned}$$

Considera-se agora o segundo termo da Eq. (1.12) a descrição material de $k = k(\mathbf{x}_{\tau})$

será chamada $k_m = k(\mathbf{x} + \tau \mathbf{v})$ para simplificar a notação. Utilizando (1.39) e (1.40) para obter a descrição material, e derivando dentro da integral, tem-se

$$\begin{aligned}
\left. \frac{\partial a_\tau}{\partial \tau}(v, \eta) \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} &= \left. \frac{\partial}{\partial \tau} \int_{\Omega_\tau} (k \nabla v \cdot \nabla \eta + v \eta) d\Omega_\tau \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} \\
&= \left. \frac{\partial}{\partial \tau} \int_{\Omega_\epsilon} (k_m \mathbb{F}_\tau^{-T} \nabla_m v \cdot \mathbb{F}_\tau^{-T} \nabla_m \eta \det \mathbb{F}_\tau + v \eta \det \mathbb{F}_\tau) d\Omega_\epsilon \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} \\
&= \int_{\Omega_\epsilon} \left. \frac{\partial}{\partial \tau} (k_m \mathbb{F}_\tau^{-T} \nabla_m v \cdot \mathbb{F}_\tau^{-T} \nabla_m \eta \det \mathbb{F}_\tau + v \eta \det \mathbb{F}_\tau) \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} d\Omega_\epsilon \\
&= \int_{\Omega_\epsilon} \left(\frac{\partial k_m}{\partial \tau} \mathbb{F}_\tau^{-T} \nabla_m v \cdot \mathbb{F}_\tau^{-T} \nabla_m \eta \det \mathbb{F}_\tau + k_m \frac{\partial \mathbb{F}_\tau^{-T}}{\partial \tau} \nabla_m v \cdot \mathbb{F}_\tau^{-T} \nabla_m \eta \det \mathbb{F}_\tau \right. \\
&\quad + k_m \mathbb{F}_\tau^{-T} \nabla_m v \cdot \frac{\partial \mathbb{F}_\tau^{-T}}{\partial \tau} \nabla_m \eta \det \mathbb{F}_\tau + k_m \mathbb{F}_\tau^{-T} \nabla_m v \cdot \mathbb{F}_\tau^{-T} \nabla_m \eta \frac{\partial \det \mathbb{F}_\tau}{\partial \tau} \\
&\quad \left. + v \eta \frac{\partial \det \mathbb{F}_\tau}{\partial \tau} \right) d\Omega_\epsilon \Big|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0}.
\end{aligned}$$

Usando as identidades (1.45), (1.46), (1.42) e (1.43), tem-se

$$\begin{aligned}
\left. \frac{\partial a_\tau}{\partial \tau}(v, \eta) \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} &= \int_{\Omega_\epsilon} (\nabla u_\epsilon \cdot \nabla q_\epsilon) \nabla k \cdot \mathbf{v} \\
&\quad + \left[[k(\nabla u_\epsilon \cdot \nabla q_\epsilon) + u_\epsilon q_\epsilon] \mathbb{I} - k(\nabla u_\epsilon \otimes \nabla q_\epsilon) - k(\nabla q_\epsilon \otimes \nabla u_\epsilon) \right] \cdot \nabla \mathbf{v} d\Omega_\epsilon.
\end{aligned}$$

Considera-se agora o terceiro termo da Eq. (1.12). A descrição material de $b = b(\mathbf{x}_\tau)$ será chamada $b_m = b(\mathbf{x} + \tau \mathbf{v})$ para simplificar a notação e usando a identidade (1.39) tem-se

$$\begin{aligned}
\left. \frac{\partial l_\tau}{\partial \tau}(\eta) \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} &= \left. \frac{\partial}{\partial \tau} \int_{\Omega_\tau} b \eta d\Omega_\tau \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} \\
&= \left. \frac{\partial}{\partial \tau} \int_{\Omega_\epsilon} b_m \eta \det \mathbb{F}_\tau d\Omega_\epsilon \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} \\
&= \int_{\Omega_\epsilon} \left. \frac{\partial}{\partial \tau} (b_m \eta \det \mathbb{F}_\tau) \right|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} d\Omega_\epsilon \\
&= \int_{\Omega_\epsilon} \left(\frac{\partial b_m}{\partial \tau} \cdot \mathbf{v} \eta \det \mathbb{F}_\tau + b_m \eta \frac{\partial \det \mathbb{F}_\tau}{\partial \tau} \right) d\Omega_\epsilon \Big|_{\substack{v=u_\tau \\ \eta=q_\tau}} \Big|_{\tau=0}.
\end{aligned}$$

Logo, utilizando as identidades (1.46) e (1.43), tem-se

$$= \int_{\Omega_\epsilon} (\nabla b \cdot \mathbf{v} q_\epsilon + b q_\epsilon (\mathbb{I} \cdot \nabla \mathbf{v})) d\Omega_\epsilon.$$

Juntando novamente os termos calculados obtém-se a expressão final

$$\begin{aligned} \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \Big|_{\tau=0} &= \int_{\Omega_\epsilon} \left(\left(J(u_\epsilon, \nabla u_\epsilon) + k(\nabla u_\epsilon \cdot \nabla q_\epsilon) + u_\epsilon q_\epsilon - bq_\epsilon \right) \mathbb{I} - \left(\nabla u_\epsilon \otimes \frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \right) \right. \\ &\quad \left. - k \left[(\nabla u_\epsilon \otimes \nabla q_\epsilon) + (\nabla q_\epsilon \otimes \nabla u_\epsilon) \right] \right) \cdot \nabla \mathbf{v} + \left((\nabla u_\epsilon \cdot \nabla q_\epsilon) \nabla k - q_\epsilon \nabla b \right) \cdot \mathbf{v} \, d\Omega_\epsilon, \end{aligned} \quad (1.19)$$

o que conclui o passo 1 do cálculo da derivada topológica. O passo 2 é identificar o tensor generalizado momento-energia de Eshelby Σ_ϵ , que no caso é dado por

$$\begin{aligned} \Sigma_\epsilon &= \left(J(u_\epsilon, \nabla u_\epsilon) + k(\nabla u_\epsilon \cdot \nabla q_\epsilon) + u_\epsilon q_\epsilon - bq_\epsilon \right) \mathbb{I} - \left(\nabla u_\epsilon \otimes \frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \right) \\ &\quad - k \left[(\nabla q_\epsilon \otimes \nabla u_\epsilon) + (\nabla u_\epsilon \otimes \nabla q_\epsilon) \right]. \end{aligned}$$

Então, a sensibilidade à mudança de forma pode ser escrita como

$$\frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \Big|_{\tau=0} = \int_{\Omega_\epsilon} \Sigma_\epsilon \cdot \nabla \mathbf{v} + \left((\nabla u_\epsilon \cdot \nabla q_\epsilon) \nabla k - q_\epsilon \nabla b \right) \cdot \mathbf{v} \, d\Omega_\epsilon. \quad (1.20)$$

Considerando a identidade (1.51) do Apêndice 1.B e o Teorema da Divergência (Eq. (1.53)) pode-se operar no primeiro termo do segundo membro de (1.20) e obter

$$\begin{aligned} \int_{\Omega_\epsilon} \Sigma_\epsilon \cdot \nabla \mathbf{v} \, d\Omega_\epsilon &= - \int_{\Omega_\epsilon} \text{div} \Sigma_\epsilon \cdot \mathbf{v} \, d\Omega_\epsilon + \int_{\Omega_\epsilon} \text{div}(\Sigma_\epsilon^T \mathbf{v}) \, d\Omega_\epsilon \\ &= - \int_{\Omega_\epsilon} \text{div} \Sigma_\epsilon \cdot \mathbf{v} \, d\Omega_\epsilon + \int_{\partial\Omega_\epsilon} \Sigma_\epsilon \mathbf{n} \cdot \mathbf{v} \, d\partial\Omega_\epsilon. \end{aligned}$$

Logo, a sensibilidade é dada por

$$\frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \Big|_{\tau=0} = \int_{\Omega_\epsilon} \left(-\text{div} \Sigma_\epsilon + \left((\nabla u_\epsilon \cdot \nabla q_\epsilon) \nabla k - q_\epsilon \nabla b \right) \right) \cdot \mathbf{v} \, d\Omega_\epsilon + \int_{\partial\Omega_\epsilon} \Sigma_\epsilon \mathbf{n} \cdot \mathbf{v} \, d\partial\Omega_\epsilon. \quad (1.21)$$

A primeira integral da Eq. (1.21) é nula

$$\int_{\Omega_\epsilon} \left(-\text{div} \Sigma_\epsilon + \left((\nabla u_\epsilon \cdot \nabla q_\epsilon) \nabla k - q_\epsilon \nabla b \right) \right) \cdot \mathbf{v} \, d\Omega_\epsilon = 0. \quad (1.22)$$

De fato

$$\begin{aligned} \text{div}(\Sigma_\epsilon) &= \text{div} \left[\left(J(u_\epsilon, \nabla u_\epsilon) + k(\nabla u_\epsilon \cdot \nabla q_\epsilon) + u_\epsilon q_\epsilon - bq_\epsilon \right) \mathbb{I} \right. \\ &\quad \left. - \left(\nabla u_\epsilon \otimes \frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \right) - k \left((\nabla q_\epsilon \otimes \nabla u_\epsilon) + (\nabla u_\epsilon \otimes \nabla q_\epsilon) \right) \right] \\ &= \nabla [J(u_\epsilon, \nabla u_\epsilon)] + \nabla [k(\nabla u_\epsilon \cdot \nabla q_\epsilon)] + \nabla [u_\epsilon q_\epsilon] - \nabla [bq_\epsilon] \\ &\quad - \text{div} \left[\nabla u_\epsilon \otimes \frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \right] - \text{div} [k \nabla u_\epsilon \otimes \nabla q_\epsilon] - \text{div} [k \nabla q_\epsilon \otimes \nabla u_\epsilon]. \end{aligned} \quad (1.23)$$

Considerando as identidades (1.48), (1.49), (1.50) e (1.51) obtém-se

$$\begin{aligned}
 \operatorname{div}(\Sigma_\epsilon) &= \frac{\partial J}{\partial u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \nabla u_\epsilon + \underbrace{(\nabla \nabla u_\epsilon)^T \frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon)}_{(3)} \\
 &+ \underbrace{k[(\nabla \nabla u_\epsilon)^T \nabla q_\epsilon]}_{(1)} + \underbrace{(\nabla \nabla q_\epsilon)^T \nabla u_\epsilon}_{(2)} + (\nabla u_\epsilon \cdot \nabla q_\epsilon) \nabla k \\
 &+ q_\epsilon \nabla u_\epsilon + u_\epsilon \nabla q_\epsilon - q_\epsilon \nabla b - b \nabla q_\epsilon - \operatorname{div} \left[\frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \right] \nabla u_\epsilon \\
 &- \underbrace{(\nabla \nabla u_\epsilon) \frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon)}_{(3)} \\
 &- k[\nabla u_\epsilon \operatorname{div}(\nabla q_\epsilon) + \underbrace{(\nabla \nabla u_\epsilon) \nabla q_\epsilon}_{(1)}] - (\nabla u_\epsilon \otimes \nabla q_\epsilon) \nabla k \\
 &- k[\nabla q_\epsilon \operatorname{div}(\nabla u_\epsilon) + \underbrace{(\nabla \nabla q_\epsilon) \nabla u_\epsilon}_{(2)}] - (\nabla q_\epsilon \otimes \nabla u_\epsilon) \nabla k.
 \end{aligned}$$

As expressões indicadas com (1), (2) e (3) se cancelam mutuamente desde que u_ϵ e q_ϵ são campos escalares, ou seja, $\nabla \nabla u_\epsilon^T = \nabla \nabla u_\epsilon$ e $\nabla \nabla q_\epsilon^T = \nabla \nabla q_\epsilon$. Logo, tem-se que

$$\begin{aligned}
 \operatorname{div}(\Sigma_\epsilon) &= \frac{\partial J}{\partial u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \nabla u_\epsilon + (\nabla u_\epsilon \cdot \nabla q_\epsilon) \nabla k + q_\epsilon \nabla u_\epsilon + u_\epsilon \nabla q_\epsilon - q_\epsilon \nabla b - b \nabla q_\epsilon \\
 &- \operatorname{div} \left[\frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \right] \nabla u_\epsilon - k \nabla u_\epsilon \operatorname{div}(\nabla q_\epsilon) - (\nabla u_\epsilon \otimes \nabla q_\epsilon) \nabla k - k \nabla q_\epsilon \operatorname{div}(\nabla u_\epsilon) \\
 &- (\nabla q_\epsilon \otimes \nabla u_\epsilon) \nabla k.
 \end{aligned}$$

Rearrmando

$$\begin{aligned}
 \operatorname{div}(\Sigma_\epsilon) &= \left(-k \operatorname{div}(\nabla q_\epsilon) - (\nabla q_\epsilon \cdot \nabla k) + q_\epsilon + \frac{\partial J}{\partial u_\epsilon}(u_\epsilon, \nabla u_\epsilon) - \operatorname{div} \left[\frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \right] \right) \nabla u_\epsilon \\
 &- \left(k \operatorname{div}(\nabla u_\epsilon) + (\nabla u_\epsilon \cdot \nabla k) - u_\epsilon + b \right) \nabla q_\epsilon + \nabla k (\nabla u_\epsilon \cdot \nabla q_\epsilon) - q_\epsilon \nabla b \\
 &= \underbrace{\left[-\operatorname{div}(k \nabla q_\epsilon) + q_\epsilon + \frac{\partial J}{\partial u_\epsilon}(u_\epsilon, \nabla u_\epsilon) - \operatorname{div} \left[\frac{\partial J}{\partial \nabla u_\epsilon}(u_\epsilon, \nabla u_\epsilon) \right] \right]}_{=0, \text{ equação adjunta, Eq. (1.17)}} \nabla u_\epsilon \\
 &- \underbrace{(\operatorname{div}(k \nabla u_\epsilon) - u_\epsilon + b)}_{=0, \text{ equação de estado, Eq. (1.15)}} \nabla q_\epsilon + \nabla k (\nabla u_\epsilon \cdot \nabla q_\epsilon) - q_\epsilon \nabla b \\
 &= \nabla k (\nabla u_\epsilon \cdot \nabla q_\epsilon) - q_\epsilon \nabla b.
 \end{aligned} \tag{1.24}$$

A prova é concluída combinando (1.24) e (1.22).

Sendo assim, a sensibilidade (Eq. (1.21)) pode ser re-escrita como

$$\left. \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \right|_{\tau=0} = \int_{\partial\omega_\epsilon} \Sigma_\epsilon \mathbf{n} \cdot \mathbf{v} \, d\partial\Omega_\epsilon. \tag{1.25}$$

Para avançar ainda mais no cálculo da derivada, e relembrando a Eq. (1.6), a

expressão de sensibilidade (1.25), fica

$$\left. \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \right|_{\tau=0} = - \int_{\partial\omega_\epsilon} \boldsymbol{\Sigma}_\epsilon \mathbf{n} \cdot \mathbf{n} \, d\partial\Omega_\epsilon.$$

Consideremos agora uma função custo particular que, como se verá mais adiante, será útil no processamento de imagens

$$\mathcal{J}(\Omega_\epsilon) = \int_{\Omega_\epsilon} \nabla u_\epsilon \cdot \nabla u_\epsilon \, d\Omega_\epsilon,$$

logo

$$\frac{\partial J(\nabla u_\epsilon)}{\partial \nabla u_\epsilon} = 2\nabla u_\epsilon.$$

Tem-se então que o tensor generalizado de Eshelby para esta função custo esta dado por

$$\begin{aligned} \boldsymbol{\Sigma}_\epsilon &= \left(\nabla u_\epsilon \cdot \nabla u_\epsilon + k(\nabla u_\epsilon \cdot \nabla q_\epsilon) + u_\epsilon q_\epsilon - bq_\epsilon \right) \mathbb{I} \\ &- \left(\nabla u_\epsilon \otimes \nabla u_\epsilon - k(\nabla q_\epsilon \otimes \nabla u_\epsilon + \nabla u_\epsilon \otimes \nabla q_\epsilon) \right). \end{aligned}$$

Logo

$$\begin{aligned} \boldsymbol{\Sigma}_\epsilon \mathbf{n} \cdot \mathbf{n} &= \left((\nabla u_\epsilon \cdot \nabla u_\epsilon) + k(\nabla u_\epsilon \cdot \nabla q_\epsilon) + u_\epsilon q_\epsilon - bq_\epsilon \right) \underbrace{(\mathbf{n} \cdot \mathbf{n})}_{=1} \\ &- (\nabla u_\epsilon \otimes \nabla u_\epsilon) \mathbf{n} \cdot \mathbf{n} - k(\nabla q_\epsilon \otimes \nabla u_\epsilon + \nabla u_\epsilon \otimes \nabla q_\epsilon) \mathbf{n} \cdot \mathbf{n} \\ &= (\nabla u_\epsilon \cdot \nabla u_\epsilon) + k(\nabla u_\epsilon \cdot \nabla q_\epsilon) + u_\epsilon q_\epsilon - bq_\epsilon \\ &- \left(\frac{\partial u_\epsilon}{\partial \mathbf{n}} \right)^2 - 2k \left(\frac{\partial u_\epsilon}{\partial \mathbf{n}} \frac{\partial q_\epsilon}{\partial \mathbf{n}} \right) \end{aligned}$$

Considerando que em $\partial\omega_\epsilon$ é satisfeito que $\frac{\partial u_\epsilon}{\partial \mathbf{n}} = 0$ e $\frac{\partial q_\epsilon}{\partial \mathbf{n}} = 0$ e como $k = 0$ em ω_ϵ se verifica que $u_\epsilon = b$ em $\partial\omega_\epsilon$, logo tem-se que

$$\int_{\partial\omega_\epsilon} \boldsymbol{\Sigma}_\epsilon \mathbf{n} \cdot \mathbf{n} \, d\Omega_\epsilon = \int_{\partial\omega_\epsilon} \left(\nabla u_\epsilon \cdot \nabla u_\epsilon + k \nabla u_\epsilon \cdot \nabla q_\epsilon \right) d\Omega_\epsilon \quad (1.26)$$

Substituindo este resultado na Eq. (1.13) a derivada topológica para a função custo considerada e para um furo arbitrário ω esta caracterizada pela seguinte expressão

$$D_T(\widehat{\mathbf{x}}) = -\lim_{\epsilon \rightarrow 0} \frac{1}{f'(\epsilon)} \int_{\partial\omega_\epsilon} \left(\nabla u_\epsilon \cdot \nabla u_\epsilon + k \nabla u_\epsilon \cdot \nabla q_\epsilon \right). \quad (1.27)$$

Com isto o Passo 2 no cálculo da derivada topológica fica concluído. O Passo 3 (Análise Assintótica) consiste no estudo do comportamento das funções u_ϵ e q_ϵ para $\epsilon \rightarrow 0$. Pode-se mostrar (ver [33] e [168]) que a expansão assintótica da solução u_ϵ toma a forma

$$u_\epsilon(\mathbf{x}) = u(\mathbf{x}) + \epsilon^2 \nabla \varphi(\widehat{\mathbf{x}}, \mathbf{x}) \cdot \mathbb{P} \nabla u(\widehat{\mathbf{x}}) + \mathcal{O}(\epsilon^2), \quad (1.28)$$

onde $\varphi(\widehat{\mathbf{x}}, \mathbf{x})$ é a solução fundamental associada ao operador da equação de estado que,

para o caso em estudo, esta dada por

$$\varphi(\hat{\mathbf{x}}, \mathbf{x}) = -\frac{1}{2\pi} \log |\mathbf{x} - \hat{\mathbf{x}}| \quad \Rightarrow \quad \nabla \varphi(\hat{\mathbf{x}}, \mathbf{x}) = -\frac{1}{2\pi} \frac{\mathbf{x} - \hat{\mathbf{x}}}{|\mathbf{x} - \hat{\mathbf{x}}|^2} . \quad (1.29)$$

Por sua vez, \mathbb{P} é o tensor de polarização [168] que depende da forma do furo ω . Analisando primeiro o caso de um furo circular de raio unitário o tensor de polarização resulta

$$\mathbb{P} = -2\pi \mathbb{I} . \quad (1.30)$$

Substituindo estas expressões na expansão assintótica (Eq. (1.28)) tem-se

$$u_\epsilon(\mathbf{x}) = u(\mathbf{x}) + \frac{\epsilon^2}{|\mathbf{x} - \hat{\mathbf{x}}|^2} \nabla u(\hat{\mathbf{x}}) \cdot (\mathbf{x} - \hat{\mathbf{x}}) + \mathcal{O}(\epsilon^2) .$$

Expandindo $u(\mathbf{x})$ em série de Taylor na vizinhança de $\hat{\mathbf{x}}$ obtém-se

$$u_\epsilon(\mathbf{x}) = u(\hat{\mathbf{x}}) + \nabla u(\hat{\mathbf{x}}) \cdot (\mathbf{x} - \hat{\mathbf{x}}) + \epsilon^2 \nabla u(\hat{\mathbf{x}}) \cdot \frac{\mathbf{x} - \hat{\mathbf{x}}}{|\mathbf{x} - \hat{\mathbf{x}}|^2} + \mathcal{O}(\epsilon^2) .$$

A expressão anterior fornece a expansão assintótica da solução da equação de estado u_ϵ definida no domínio perturbado (Ω_ϵ) em função da solução da equação de estado u definida no domínio não perturbado (Ω). Empregando o resultado anterior o gradiente toma a forma

$$\nabla u_\epsilon(\mathbf{x}) = \nabla u(\hat{\mathbf{x}}) + \epsilon^2 \left(\frac{1}{|\mathbf{x} - \hat{\mathbf{x}}|^2} \mathbb{I} - \frac{2}{|\mathbf{x} - \hat{\mathbf{x}}|^4} (\mathbf{x} - \hat{\mathbf{x}}) \otimes (\mathbf{x} - \hat{\mathbf{x}}) \right) \nabla u(\hat{\mathbf{x}}) + \mathcal{O}(\epsilon^2) .$$

Para todo ponto $\mathbf{x} \in \partial\omega_\epsilon$ tem-se $\mathbf{x} - \hat{\mathbf{x}} = -\epsilon \mathbf{n}$, logo

$$\nabla u_\epsilon(\mathbf{x})|_{\partial\omega_\epsilon} = \nabla u(\hat{\mathbf{x}}) + (\mathbb{I} - 2\mathbf{n} \otimes \mathbf{n}) \nabla u(\hat{\mathbf{x}}) + \mathcal{O}(\epsilon^2) \quad (1.31)$$

$$= 2(\mathbb{I} - \mathbf{n} \otimes \mathbf{n}) \nabla u(\hat{\mathbf{x}}) + \mathcal{O}(\epsilon^2) . \quad (1.32)$$

Da mesma forma, a expansão assintótica da função adjunta q_ϵ é dada por

$$q_\epsilon(\mathbf{x}) = q(\hat{\mathbf{x}}) + \nabla q(\hat{\mathbf{x}}) \cdot (\mathbf{x} - \hat{\mathbf{x}}) + \epsilon^2 \nabla q(\hat{\mathbf{x}}) \cdot \frac{\mathbf{x} - \hat{\mathbf{x}}}{|\mathbf{x} - \hat{\mathbf{x}}|^2} + \mathcal{O}(\epsilon^2) ,$$

cujo gradiente resulta em

$$\nabla q_\epsilon(\mathbf{x}) = \nabla q(\hat{\mathbf{x}}) + \epsilon^2 \left(\frac{1}{|\mathbf{x} - \hat{\mathbf{x}}|^2} \mathbb{I} - \frac{2}{|\mathbf{x} - \hat{\mathbf{x}}|^4} (\mathbf{x} - \hat{\mathbf{x}}) \otimes (\mathbf{x} - \hat{\mathbf{x}}) \right) \nabla q(\hat{\mathbf{x}}) + \mathcal{O}(\epsilon^2) ,$$

que avaliado sobre $\partial\omega_\epsilon$ fica

$$\nabla q_\epsilon(\mathbf{x})|_{\partial\omega_\epsilon} = 2(\mathbb{I} - \mathbf{n} \otimes \mathbf{n}) \nabla q(\hat{\mathbf{x}}) + \mathcal{O}(\epsilon^2) . \quad (1.33)$$

Substituindo ambos resultados dados pelas Eqs. (1.32) e (1.33) na Eq. (1.27) tem-se

$$\begin{aligned} D_T(\widehat{\mathbf{x}}) &= -\lim_{\epsilon \rightarrow 0} \frac{1}{f'(\epsilon)} \int_{\partial\omega_\epsilon} 4 \left((\mathbb{I} - \mathbf{n} \otimes \mathbf{n}) \cdot \mathbb{A} \right) \\ &= -\lim_{\epsilon \rightarrow 0} \frac{4}{f'(\epsilon)} \int_{\partial\omega_\epsilon} \left(\text{tr}(\mathbb{A}) - (\mathbf{n} \otimes \mathbf{n}) \cdot \mathbb{A} \right), \end{aligned}$$

onde

$$\begin{aligned} \mathbb{A} &:= \nabla u(\widehat{\mathbf{x}}) \otimes \nabla u(\widehat{\mathbf{x}}) + k \nabla u(\widehat{\mathbf{x}}) \otimes_s \nabla q(\widehat{\mathbf{x}}), \\ \text{tr}(\mathbb{A}) &= \nabla u(\widehat{\mathbf{x}}) \cdot \nabla u(\widehat{\mathbf{x}}) + k \nabla u(\widehat{\mathbf{x}}) \cdot \nabla q(\widehat{\mathbf{x}}). \end{aligned}$$

Aplicando o teorema da divergência no segundo termo do integrando obtém-se

$$\begin{aligned} D_T(\widehat{\mathbf{x}}) &= -\lim_{\epsilon \rightarrow 0} \frac{4}{f'(\epsilon)} \left(\int_{\partial\omega_\epsilon} \text{tr}(\mathbb{A}) + \int_{\omega_\epsilon} \nabla \mathbf{n} \cdot \mathbb{A} \right) \\ &= -\lim_{\epsilon \rightarrow 0} \frac{4}{f'(\epsilon)} \left(\int_{\partial\omega_\epsilon} \text{tr}(\mathbb{A}) - \frac{1}{\epsilon} \int_{\omega_\epsilon} \mathbb{I} \cdot \mathbb{A} \right) \\ &= -\lim_{\epsilon \rightarrow 0} \frac{4}{f'(\epsilon)} \left(\int_{\partial\omega_\epsilon} \text{tr}(\mathbb{A}) - \frac{1}{\epsilon} \int_{\omega_\epsilon} \text{tr}(\mathbb{A}) \right) \\ &= -\lim_{\epsilon \rightarrow 0} \frac{4}{f'(\epsilon)} \left(\text{tr}(\mathbb{A}) 2\pi\epsilon - \frac{1}{\epsilon} \text{tr}(\mathbb{A}) \pi\epsilon^2 \right) \\ &= -2\text{tr}(\mathbb{A}) \lim_{\epsilon \rightarrow 0} \frac{1}{f'(\epsilon)} 2\pi\epsilon \end{aligned}$$

lembrando que a troca de sinal vem do fato de a normal $\mathbf{n} = -(\mathbf{x} - \widehat{\mathbf{x}})/\epsilon$ ser saliente a $\partial\omega_\epsilon$ (\mathbf{n} aponta para o centro da bola).

Escolhendo $f'(\epsilon) = 2\pi\epsilon$ ($f(\epsilon) = \pi\epsilon^2$), a derivada topológica para a função custo escolhida e para o caso de um furo circular resulta

$$\begin{aligned} D_T(\widehat{\mathbf{x}}) &= -2\text{tr}(\mathbb{A}) \\ &= -2 \left(\nabla u(\widehat{\mathbf{x}}) \cdot \nabla u(\widehat{\mathbf{x}}) + k \nabla u(\widehat{\mathbf{x}}) \cdot \nabla q(\widehat{\mathbf{x}}) \right). \end{aligned} \quad (1.34)$$

O resultado anterior vem ao encontro do já mencionado anteriormente: a derivada topológica é uma função escalar que depende somente das soluções das equações de estado e adjunta definidas no domínio **não perturbado** Ω e, por tanto, conhecidas estas funções a derivada topológica é obtida através de um simples cálculo.

Para um furo elíptico e denotando por \mathbf{t} e \mathbf{m} os vetores unitários respectivamente associados aos eixos maior e menor desta elipse, o tensor de polarização toma a forma seguinte

$$\mathbb{P} = -|\omega| \left(\frac{r_1 + r_2}{r_1} \mathbf{t} \otimes \mathbf{t} + \frac{r_1 + r_2}{r_2} \mathbf{m} \otimes \mathbf{m} \right),$$

onde r_1 e r_2 são, respectivamente, os semi-eixos maior e menor da elipse e $|\omega|$ a área do

furo elíptico. Lembrando que $|\omega| = \pi r_1 r_2$ o tensor de polarização pode ser reescrito como

$$\begin{aligned}\mathbb{P} &= -\pi r_1 r_2 \left(\frac{r_1 + r_2}{r_1} \mathbf{t} \otimes \mathbf{t} + \frac{r_1 + r_2}{r_2} \mathbf{m} \otimes \mathbf{m} \right) \\ &= -\pi r_1 \left(\frac{r_2(r_1 + r_2)}{r_1} \mathbf{t} \otimes \mathbf{t} + (r_1 + r_2) \mathbf{m} \otimes \mathbf{m} \right).\end{aligned}$$

Tomando $r_2 \rightarrow 0$ furo elíptico converge para um talho ortogonal à direção \mathbf{m} e, adotando $r_1 = 1$, seu correspondente tensor de polarização converge para

$$\mathbb{P} = -\pi(\mathbf{m} \otimes \mathbf{m}).$$

Desta expressão obtém-se

$$\mathbb{P} \nabla u(\hat{\mathbf{x}}) = -\pi(\nabla u(\hat{\mathbf{x}}) \cdot \mathbf{m}) \mathbf{m} = -\pi \frac{\partial u(\hat{\mathbf{x}})}{\partial \mathbf{m}} \mathbf{m}.$$

Em outras palavras, somente a derivada na direção \mathbf{m} de u no ponto $\hat{\mathbf{x}}$ é significativa na expansão assintótica de u_ϵ . Assim, seguindo os mesmos passos que no caso de furo circular e considerando a mesma função custo, a derivada topológica para um furo em forma de talho infinitesimal ortogonal à direção \mathbf{m} , resulta em

$$\begin{aligned}D_T(\hat{\mathbf{x}}) &= -\left((\nabla u(\hat{\mathbf{x}}) \cdot \mathbf{m})^2 + k(\nabla u(\hat{\mathbf{x}}) \cdot \mathbf{m})(\nabla q(\hat{\mathbf{x}}) \cdot \mathbf{m}) \right) \\ &= -\left(\nabla u \otimes \nabla u + k(\nabla u \otimes_s \nabla q) \right) \mathbf{m} \cdot \mathbf{m} \\ &= \mathbb{M} \mathbf{m} \cdot \mathbf{m} = D_T(\hat{\mathbf{x}}, \mathbf{m}),\end{aligned}\tag{1.35}$$

onde \mathbb{M} é uma matriz (tensor) simétrica dada por

$$\mathbb{M} = -\left(\nabla u \otimes \nabla u + k(\nabla u \otimes_s \nabla q) \right).\tag{1.36}$$

Comparando esta derivada topológica com a correspondente a um furo circular (Eq. (1.34)) observa-se que no caso de uma perturbação caracterizada por um talho ortogonal à direção \mathbf{m} a derivada topológica resulta **também** função da orientação.

Logo, para um dado ponto $\hat{\mathbf{x}}$, $D_T(\hat{\mathbf{x}}, \mathbf{m})$ alcança seu mínimo valor quando \mathbf{m} coincide com o auto-vetor associado ao menor auto-valor κ_{min} da matriz \mathbb{M} . Este valor mínimo será adotado como a derivada topológica associada a criação de um talho no ponto $\hat{\mathbf{x}}$. Na orientação “ótima” do talho, a sua normal coincide com o auto-vetor correspondente a κ_{min} .

Como será mostrado no Capítulo 3, a derivada topológica calculada desta maneira desempenha um papel fundamental no desenvolvimento de algoritmos para a restauração de imagens deterioradas pela presença de diferentes tipos de ruído.

Apêndice Capítulo 1

1.A Propriedades de $a(\cdot, \cdot)$

Sejam u, u_1, u_2, v, v_1 e v_2 funções arbitrárias em \mathcal{U} então $a : \mathcal{U} \times \mathcal{U} \mapsto \mathbb{R}$, é

- bilinear, se

$$\begin{aligned} a(\alpha u, v) &= a(u, \alpha v) = \alpha a(u, v), \\ a(u_1 + u_2, v) &= a(u_1, v) + a(u_2, v) \quad \text{e} \\ a(u, v_1 + v_2) &= a(u, v_1) + a(u, v_2), \end{aligned}$$

- simétrica, se

$$a(u, v) = a(v, u),$$

- contínua, se

$$|a(u, v)| \leq C \|u\| \|v\|,$$

- coerciva, se

$$a(u, u) \geq C \|u\|^2.$$

Em particular, para a classe de problemas considerados, \mathcal{U} e \mathcal{V} são um espaços satisfazendo

$$\mathcal{U} \subset H^1(\Omega)$$

$$\mathcal{V} \subset H^1(\Omega)$$

sendo $H^1(\Omega)$ o espaço de Hilbert de ordem um, i.e., funções integráveis com primeira derivada integrável.

Para forma bilineares $a(\cdot, \cdot)$ deste tipo, as hipóteses do corolário Lax-Milgram são atendidas. Este corolário, também conhecido com Teorema de Lax-Milgram, é enunciado a seguir:

Corolário 1. *Lax-Milgram* *Seja $a(u, \eta)$ uma forma bilinear, contínua e coerciva. Logo, para todo $\eta \in \mathcal{V}(\Omega)$ existe um único $u \in \mathcal{U}$ tal que*

$$a(u, \eta) = l(\eta) \quad \forall \eta \in \mathcal{V}(\Omega).$$

1.B Algumas identidades úteis

Derivadas espaciais e materiais O índice m indica descrição material (sendo ∇_m o gradiente na descrição material).

$$\mathbf{x}_\tau = \mathbf{x} + \tau \mathbf{v}(\mathbf{x}) \quad (1.37)$$

$$d\mathbf{x}_\tau = \frac{\partial \mathbf{x}_\tau}{\partial \mathbf{x}} = \mathbb{F} d\mathbf{x} \quad (1.38)$$

$$d\Omega_\tau = \det \mathbb{F}_\tau d\Omega_\epsilon \quad (1.39)$$

$$(\nabla u)_m = \mathbb{F}_\tau^{-T} \nabla_m u \quad (1.40)$$

$$\begin{aligned} \mathbb{F}_\tau &= \nabla \chi_\tau = \mathbb{I} + \tau \nabla \mathbf{v}(\mathbf{x}) \quad \text{logo} \\ &\Rightarrow \mathbb{F}_\tau|_{\tau=0} = \mathbb{I} \end{aligned} \quad (1.41)$$

$$\begin{aligned} \mathbb{F}_\tau^{-1} &= \mathbb{I} - \tau \nabla \mathbf{v}(\mathbf{x}) + \mathcal{O}(\tau) \quad \text{logo} \\ &\Rightarrow \mathbb{F}_\tau^{-1}|_{\tau=0} = \mathbb{I} \end{aligned} \quad (1.42)$$

$$\det \mathbb{F}_\tau|_{\tau=0} = 1 \quad (1.43)$$

$$\frac{\partial \mathbb{F}_\tau}{\partial \tau} = \nabla \mathbf{v} \quad (1.44)$$

$$\left. \frac{\partial \mathbb{F}_\tau^{-1}}{\partial \tau} \right|_{\tau=0} = -\nabla \mathbf{v} \quad (1.45)$$

$$\frac{\partial \det \mathbb{F}_\tau}{\partial \tau} = \mathbb{I} \cdot \nabla \mathbf{v} \det \mathbb{F}_\tau \quad (1.46)$$

Derivada de campos espaciais Sejam u e v campos escalares, \mathbf{u} e \mathbf{v} campos vetoriais e \mathbb{T} campo tensorial suaves, logo

$$\text{div}(v\mathbf{u}) = v \text{div} \mathbf{u} + (\nabla v) \cdot \mathbf{u} \quad (1.47)$$

$$\nabla(v\mathbf{u}) = v \nabla \mathbf{u} + \mathbf{u} \otimes \nabla v \quad (1.48)$$

$$\text{div}(\mathbf{v} \otimes \mathbf{u}) = \mathbf{v} \text{div} \mathbf{u} + (\nabla \mathbf{v}) \cdot \mathbf{u} \quad (1.49)$$

$$\nabla(\mathbf{v} \cdot \mathbf{u}) = (\nabla \mathbf{v})^T \mathbf{u} + (\nabla \mathbf{u})^T \mathbf{v} \quad (1.50)$$

$$\text{div}(\mathbb{T}^T v) = v \text{div}(\mathbb{T}) + \mathbb{T} \nabla v \quad (1.51)$$

$$\text{div}(\mathbb{T}^T \mathbf{v}) = \text{div}(\mathbb{T}) \cdot \mathbf{v} + \mathbb{T} \cdot \nabla \mathbf{v} \quad (1.52)$$

Teorema da Divergência Sejam Ω uma região regular limitada, e seja \mathbf{v} um campo vetorial suave, logo

$$\int_{\partial\Omega} \mathbf{v} \cdot \mathbf{n} \, d\partial\Omega = \int_{\Omega} \text{div} \mathbf{v} \, d\Omega \quad (1.53)$$

sendo \mathbf{n} o vetor normal a $d\partial\Omega$.

Capítulo 2

Restauração de Imagens

Para uma análise visual das imagens médicas, a clareza de detalhes e visibilidade dos objetos são fatores importantes, mas para fins de processamento avançado de imagens, um alto “*signal-to-noise*” ratio (SNR) é necessário, já que outras etapas neste processo (como segmentação e classificação) são muito sensíveis ao ruído.

Ao longo dos anos, diferentes técnicas têm sido estudadas para melhorar o SNR de uma imagem degradada. Estes métodos podem ser classificados como:

- Métodos que afetam o tempo de aquisição e/ou tamanho dos pixels:
 - adquirir repetidamente a mesma imagem e calcular o valor médio para cada pixel nas diferentes aquisições (este método é ineficiente e aumenta o tempo necessário para adquirir uma imagem, expondo o paciente de maneira perigosa),
 - escanear com elementos maiores (produz uma perda de resolução).
- Métodos que não afetam o tempo de aquisição e/ou tamanho dos pixels:
 - melhorar o hardware de aquisição (depende fortemente da tecnologia disponível),
 - pós-processar as imagens depois destas serem adquiridas (técnicas de processamento de imagens).

Os parâmetros de aquisição podem ser otimizados com o objetivo de melhorar o contraste e o SNR da imagem, mas este tipo de métodos geralmente resultam em um incremento significativo no tempo de aquisição (aumentando o tempo de exposição do paciente). As técnicas de pós-processamento têm a vantagem de não afetar o processo de aquisição [75, 97]. O simples cálculo de médias sobre pixels vizinhos (chamado também de difusão isotrópica) aumenta o SNR consideravelmente, mas tem o efeito adverso de degradar a qualidade das bordas (tais como linhas e pontos) que existem na imagem deixando-a mais “difusa” (efeito de *blurring*). Este efeito pode ser reduzido utilizando filtros não-lineares. O filtro de mediana possui a característica de manter as bordas, mas seus detalhes são perdidos, deteriorando a resolução da imagem. Uma das mais populares técnicas de remoção de ruído introduzidas nos últimos anos é a difusão anisotrópica, inicialmente proposta por Perona e Malik [164].

Na próxima seção será estudado o modelo usualmente utilizado para representar a degradação de uma imagem. Posteriormente, é apresentado o estado da arte na área de restauração de imagens por meio do uso do Cálculo Variacional e EDP’s (Equações Diferenciais Parciais).

2.1 Degradação de Imagens - Modelo Matemático

A restauração de imagens degradadas por algum fenômeno é uma das mais antigas preocupações do processamento de imagens e ainda hoje é um passo de pré-processamento importante para muitas aplicações. Durante a aquisição, transmissão e armazenamento das imagens, estas podem ser deterioradas. Em geral, este tipo de deterioração pode ser provocada por dois tipos de fenômenos. O primeiro deles é determinístico e está relacionado à um mau ajuste nas lentes ou dispositivos de aquisição, movimentos do objeto ou do dispositivo no momento de aquisição, entre outros. O segundo é de caráter aleatório e corresponde ao ruído proveniente de qualquer tipo de transmissão de informação.

O objetivo da restauração de imagens é justamente remover ou diminuir este tipo de degradação. Nesta seção, são estudados alguns métodos que permitem diminuir estes efeitos. Lamentavelmente, em geral é difícil identificar o tipo de ruído que afeta uma imagem real. Assim, ao não se dispor de um modelo de degradação exato alguma suposição deve ser feita. Geralmente o seguinte modelo é adotado: sejam $\bar{u} : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ a imagem original descrevendo uma cena real e u_0 a imagem observada desta cena (a imagem degradada), então

$$u_0 = R\bar{u} + \rho \quad (2.1)$$

sendo ρ um ruído aditivo aleatório e R é um operador linear¹ que representa o *blur* (geralmente uma convolução). Logo, o problema é: dado u_0 , reconstruir \bar{u} satisfazendo (2.1). Este é um problema mal posto por não garantir solução única e ainda por apresentar instabilidades na solução quando o ruído supera certos valores limites. Em geral se diz que um problema é bem posto segundo Hadamard [84] se a solução existe, é única e é estável. Quando alguma destas condições não é satisfeita o problema é dito mal posto (ver [16] para mais detalhes).

Para mostrar o desempenho dos algoritmos mencionados ao longo deste capítulo, serão utilizadas as imagens apresentadas na Figura 2.1, usualmente empregadas na literatura para testar os algoritmos. Nestas imagens foi aplicado um ruído uniforme de intensidade 20 (8% da diferença entre a intensidade máxima e mínima da imagem)². Nas diferentes implementações foi utilizado o software Matlab [190].

2.2 O Uso de EDP's na Restauração de Imagens

A seguir são estudadas duas alternativas para tratar problemas de restauração de imagens. Na primeira delas a restauração é tratada como um problema de minimização de um funcional (função custo), mais precisamente como um problema inverso. Logo, conhecido o modelo de degradação (dado pela Eq. (2.1)), objetiva-se achar uma aproximação u (imagem restaurada) da imagem \bar{u} (imagem original) a partir do dado u_0 (imagem

¹Seja α uma constante arbitrária, então R é linear se $R(\alpha u + v) = \alpha Ru + Rv$.

²O modelo utilizado para deteriorar as imagens é descrito com maior detalhe no Capítulo 4.



Figura 2.1: Imagens de Lena e do barco utilizadas nos exemplos (originais e com ruído uniforme).

degradada). Uma alternativa é estudar o problema do ponto de vista de mínimos quadrados, minimizando a norma L^2 do erro. No entanto, este é um problema mal posto que requer uma análise mais aprofundada (Seção 2.3).

Uma outra alternativa é considerar a restauração de imagens como um problema evolutivo. Nesta linha, tem sido introduzida a noção de espaço de escala (*scale space*). Os primeiros pesquisadores a propor este conceito (que será denominado de análise multi-escala) foram Koendernik [101] e Witkin [202]. Eles foram os responsáveis por formular rigorosamente a noção de análise multi-escala, que neste contexto tem o sentido de representação da imagem simultaneamente em múltiplas escalas. Neste caso, as estruturas das escalas mais grosseiras (*coarser scales*) devem constituir simplificações de estruturas em escalas mais finas.

Seguindo o trabalho inicial de Alvarez *et al.* [3] pode-se encontrar uma conexão entre a análise multi-escala e EDP's de maneira rigorosa. Neste trabalho, partindo de um

conjunto de axiomas (baseados nas propriedades que se deseja que a imagem possua) os autores mostram que a imagem resultante do processamento é a solução de uma EDP.

2.3 O Método da Energia

Como já foi mencionado, o problema de restauração pode ser interpretado como um problema inverso. Em particular, assumindo também que o modelo apresentado na Eq. (2.1) é válido, objetiva-se então recuperar \bar{u} a partir de u_0 . Este não é um problema fácil, dado que não se dispõe de muita informação acerca de ρ , além de algumas características estatísticas como a sua média e variância.

Supondo então que ρ é um ruído aleatório de média nula, e segundo o princípio do máximo (*maximum likelihood principle*), pode ser achada uma aproximação u de \bar{u} resolvendo o problema de mínimos quadrados dado por

$$\inf_{u \in \mathcal{U}} \int_{\Omega} |u_0 - Ru|^2 d\Omega = \inf_{u \in \mathcal{U}} J(u), \quad (2.2)$$

sendo Ω o domínio da imagem e $J(\cdot) : L^2(\Omega) \mapsto \mathbb{R}$ o funcional (função custo) que mede a qualidade da imagem restaurada u , onde $L^2(\Omega)$ é o espaço de funções quadraticamente integráveis. A solução u deste problema também satisfaz que $\delta J(u; \eta)$ (primeira variação de $J(u)$ na direção $\eta \in L^2$) é nula, i.e.,

$$\delta J(u; \eta) = 0 \quad \forall \eta \in \mathcal{V} \quad (2.3)$$

Logo

$$\begin{aligned} \delta J(u; \eta) &= \frac{d}{d\tau} J(u + \tau\eta)|_{\tau=0} \\ &= \int_{\Omega} \frac{d}{d\tau} (u_0 - R(u + \tau\eta))^2|_{\tau=0} d\Omega \\ &= \int_{\Omega} -2(u_0 - R(u + \tau\eta)) \cdot R\eta|_{\tau=0} d\Omega \\ &= \int_{\Omega} -2(u_0 - Ru) \cdot R\eta d\Omega \\ &= \int_{\Omega} -2R^*(u_0 - Ru) \cdot \eta d\Omega = 0. \end{aligned}$$

Como isto é válido para todo $\eta \in \mathcal{V}$, tem-se que

$$R^*Ru = R^*u_0$$

sendo R^* o operador adjunto de R . Como R^*R pode não ter inversa (ou caso tenha, seus autovalores podem ser próximos de zero o que a faz instável), para achar a solução da Eq. (2.2) deve ser utilizada alguma técnica de regularização.

2.3.1 Regularização do Problema

A maneira clássica de resolver problemas deste tipo é introduzir um termo de regularização [103], i.e.

$$E(u) = \frac{1}{2} \int_{\Omega} |u_0 - Ru|^2 d\Omega + \lambda \int_{\Omega} \phi(|\nabla u|) d\Omega. \quad (2.4)$$

Diferentes alternativas são obtidas para diferentes escolhas da função $\phi(\cdot)$. Por exemplo, utilizando $\phi(x) = x^2$ obtém-se o termo de regularização de Tikhonov-Arsenin, ou seja:

$$\mathcal{F}(u) = \frac{1}{2} \int_{\Omega} |u_0 - Ru|^2 d\Omega + \lambda \int_{\Omega} |\nabla u|^2 d\Omega. \quad (2.5)$$

Observa-se que o primeiro termo está associado à fidelidade aos dados e o segundo representa uma suavização. Este último termo irá penalizar os gradientes, ou seja, $\mathcal{F}(u)$ assume valores maiores em locais que apresentam valores elevados do gradiente. Para que o funcional $\mathcal{F}(u)$ esteja bem definido, u deve ser estar em $H^1(\Omega)$ [32], ou seja $u \in H^1(\Omega)$, onde $H^1(\Omega)$ é o espaço de Hilbert de primeira ordem ($u \in L^2(\Omega)$ e $\nabla u \in L^2(\Omega)$).

Novamente, a condição necessária de mínimo deste funcional é dada pelo valor u para o qual a primeira variação de $\mathcal{F}(u)$ é nula, logo

$$\begin{aligned} \delta\mathcal{F}(u; \eta) &= \frac{d}{d\tau} \left[\frac{1}{2} \int_{\Omega} |u_0 - R(u + \tau\eta)|^2 d\Omega + \lambda \int_{\Omega} |\nabla(u + \tau\eta)|^2 d\Omega \right]_{\tau=0} \\ &= \left[\int_{\Omega} -(u_0 - R(u + \tau\eta)) \cdot R\eta d\Omega + \lambda \int_{\Omega} 2\nabla(u + \tau\eta) \cdot \nabla\eta d\Omega \right]_{\tau=0} \\ &= \int_{\Omega} (R^*Ru - R^*u_0)\eta d\Omega + 2\lambda \int_{\Omega} \nabla u \cdot \nabla\eta d\Omega = 0. \quad \forall \eta \in \mathcal{V} \end{aligned}$$

Obtém-se então a equação variacional do problema: Achar $u \in \mathcal{U}$ tal que

$$\int_{\Omega} R^*Ru\eta d\Omega + 2\lambda \int_{\Omega} \nabla u \cdot \nabla\eta d\Omega = \int_{\Omega} R^*u_0\eta d\Omega \quad \forall \eta \in \mathcal{V}. \quad (2.6)$$

Para chegar nas equações de Euler-Lagrange, deve-se integrar por partes esta expressão. Utilizando o Teorema da Divergência [82], tem-se que

$$\int_{\Omega} \nabla u \cdot \nabla\eta d\Omega = - \int_{\Omega} \eta \Delta u d\Omega + \int_{\partial\Omega} \frac{\partial u}{\partial \mathbf{n}} \eta d\partial\Omega$$

sendo \mathbf{n} a normal a $\partial\Omega$, fronteira de Ω . Levando em consideração o anterior chega-se na forma integral da equação de Euler-Lagrange

$$\int_{\Omega} (R^*Ru - R^*u_0 - 2\lambda \Delta u)\eta d\Omega + \int_{\partial\Omega} \frac{\partial u}{\partial \mathbf{n}} \eta d\partial\Omega = 0 \quad \forall \eta \in \mathcal{V}. \quad (2.7)$$

Observando que esta condição é válida para todo $\eta \in \mathcal{V}$, obtém-se finalmente a forma forte

do problema, ou seja

$$\begin{cases} R^*Ru - 2\lambda\Delta u = R^*u_0, & \text{em } \Omega \\ \frac{\partial u}{\partial \mathbf{n}} = 0, & \text{sobre } \partial\Omega \end{cases}, \quad (2.8)$$

Que pode ser interpretado como a equação de calor isotrópica estacionária. Ao tratar uma imagem utilizando esta equação, o ruído presente é reduzido, mas também as bordas presentes são severamente degradadas. O parâmetro λ controla a intensidade da difusão, eliminando maior quantidade de ruído e bordas para valores mais elevados. Esta degradação ocorre pelo fato da equação ser isotrópica, o que irá difundir a imagem igualmente em todas as direções e em todos os pontos, não considerando as estruturas presentes.

Os resultados numéricos apresentados na Fig. (2.2) foram obtidos utilizando o Método dos Elementos Finitos (ver Apêndice (A.1)). Para isto foi aproximada a equação variacional (2.6) redefinindo-a no espaço de aproximação caracterizado por elementos finitos quadriláteros bilineares. Os nós da malha de elementos finitos foram centrados nos pixels da imagem de maneira que os valores de cada nó na solução correspondesse com a intensidade do pixel processado. A imagem de entrada u_0 é utilizada para construir o termo do lado direito da equação (Eq. (2.6)). Uma vez obtido o sistema de equações algébricas correspondente, este foi resolvido iterativamente utilizando o algoritmo de gradientes conjugados.

Para poder analisar em detalhe as características do termo de regularização propõe-se estudar a versão mais geral do funcional (apresentado na Eq. (2.4)). Calcula-se então a primeira variação de $E(u)$

$$\begin{aligned} \delta E(u; \eta) &= \frac{d}{d\tau} \left[\frac{1}{2} \int_{\Omega} |u_0 - R(u + \tau\eta)|^2 d\Omega + \lambda \int_{\Omega} \phi(|\nabla(u + \tau\eta)|) d\Omega \right]_{\tau=0} \\ &= \left[\int_{\Omega} -(u_0 - R(u + \tau\eta)) \cdot R\eta d\Omega + \lambda \int_{\Omega} \phi'(|\nabla(u + \tau\eta)|) \frac{\nabla u}{|\nabla u|} \cdot \nabla \eta d\Omega \right]_{\tau=0} \\ &= \int_{\Omega} -R^*(u_0 - Ru)\eta d\Omega + \lambda \int_{\Omega} \phi'(|\nabla u|) \frac{\nabla u}{|\nabla u|} \cdot \nabla \eta d\Omega \end{aligned}$$

para achar a condição de optimalidade do funcional

$$\int_{\Omega} R^*(Ru - u_0)\eta d\Omega + \lambda \int_{\Omega} \phi'(|\nabla u|) \frac{\nabla u}{|\nabla u|} \cdot \nabla \eta d\Omega = 0 \quad \forall \eta \in \mathcal{V}. \quad (2.9)$$

Operando de maneira semelhante ao caso isotrópico linear, obtém-se a forma integral da equação de Euler-Lagrange

$$\begin{aligned} \int_{\Omega} \left(R^*Ru - R^*u_0 - \lambda \operatorname{div} \left(\phi'(|\nabla u|) \frac{\nabla u}{|\nabla u|} \right) \right) \eta d\Omega \\ + \lambda \int_{\partial\Omega} \frac{\phi'(|\nabla u|)}{|\nabla u|} \frac{\partial u}{\partial \mathbf{n}} \eta d\Omega = 0 \quad \forall \eta \in \mathcal{V}. \end{aligned} \quad (2.10)$$

Logo, assumindo que E tem um mínimo em u e como Eq. (2.9) é válida para todo v ,



Figura 2.2: Restauração com difusão, imagem com ruído (SNR = 26.92dB) e imagem processada com $\lambda = 1$ (SNR = 25.32dB), $\lambda = 5$ (SNR = 21.57dB) e $\lambda = 25$ (SNR = 18.53dB).

pode-se escrever o problema (2.9) na sua forma forte

$$\begin{cases} R^*Ru - \lambda \operatorname{div} \left(\phi'(|\nabla u|) \frac{\nabla u}{|\nabla u|} \right) = R^*u_0 & \text{em } \Omega \\ \frac{\partial u}{\partial \mathbf{n}} = 0, & \text{em } \partial\Omega. \end{cases}.$$

Para entender melhor as propriedades deste problema, o termo $\operatorname{div} \left(\phi'(|\nabla u|) \frac{\nabla u}{|\nabla u|} \right)$

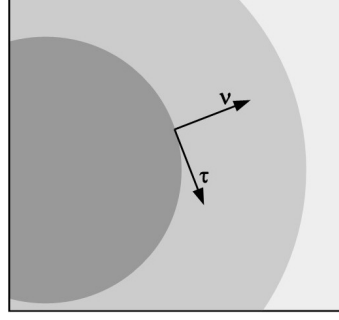


Figura 2.3: Sistema de referência local nas direções normal (ν) e tangente (τ) às isolinhas da imagem.

é analisado. Utilizando as identidades apresentadas na Seção 1.B, tem-se que

$$\begin{aligned}
 \operatorname{div} \left(\phi'(|\nabla u|) \frac{\nabla u}{|\nabla u|} \right) &= \frac{\phi'(|\nabla u|)}{|\nabla u|} \Delta u + \nabla \left(\frac{\phi'(|\nabla u|)}{|\nabla u|} \right) \cdot \nabla u \\
 &= \frac{\phi'(|\nabla u|)}{|\nabla u|} \Delta u + \left(\frac{\phi''(|\nabla u|)}{|\nabla u|^2} - \frac{\phi'(|\nabla u|)}{|\nabla u|^3} \right) (\nabla \nabla u) \nabla u \cdot \nabla u \\
 &= \frac{\phi'(|\nabla u|)}{|\nabla u|} \left(\frac{1}{|\nabla u|^2} (\Delta u |\nabla u|^2 - (\nabla \nabla u) \nabla u \cdot \nabla u) \right) \\
 &+ \phi''(|\nabla u|) \left(\frac{1}{|\nabla u|^2} ((\nabla \nabla u) \nabla u \cdot \nabla u) \right), \tag{2.11}
 \end{aligned}$$

sendo $\nabla \nabla u$ o gradiente de segunda ordem do campo escalar u (dado por um tensor, também de segunda ordem). Procura-se, então, uma função $\phi(s)$ que permita obter uma solução u quase constante por partes e com bordas nítidas. A Eq. (2.11) pode ser interpretada como uma decomposição do termo da divergência usando as estruturas presentes na imagem, neste caso, a direção normal e tangente às isolinhas (linhas nas quais a imagem tem intensidade constante, Fig. (2.3)). Define-se, então, um sistema de referência local, caracterizado em cada ponto \mathbf{x} onde $|\nabla u(\mathbf{x})| \neq 0$ por dois vetores: $\boldsymbol{\nu} = \frac{\nabla u}{|\nabla u|}$ e $|\boldsymbol{\tau}| = 1$ sendo que $\boldsymbol{\nu} \cdot \boldsymbol{\tau} = 0$. Logo, tem-se que

$$\begin{aligned}
 \frac{1}{|\nabla u|^2} (\Delta u |\nabla u|^2 - (\nabla \nabla u) \nabla u \cdot \nabla u) &= \Delta u - (\nabla \nabla u) \boldsymbol{\nu} \cdot \boldsymbol{\nu} = u_{\tau\tau} \\
 \frac{1}{|\nabla u|^2} ((\nabla \nabla u) \nabla u \cdot \nabla u) &= (\nabla \nabla u) \boldsymbol{\nu} \cdot \boldsymbol{\nu} = u_{\nu\nu}
 \end{aligned}$$

Obtém-se que então

$$R^* R u - \lambda \left(\frac{\phi'(|\nabla u|)}{|\nabla u|} u_{\tau\tau} + \phi''(|\nabla u|) u_{\nu\nu} \right) = R^* u_0, \tag{2.12}$$

onde $u_{\nu\nu}$ e $u_{\tau\tau}$ são as derivadas segundas na direção normal e tangente respectivamente.

Tem-se então que a função ϕ deve ser escolhida de maneira a satisfazer algumas propriedades:

- **Para gradientes pequenos.** Onde a variação da função u é pequena, a difusão deve ser forte em todas as direções (suavizado isotrópico) de maneira que elimine

a maior quantidade de ruído possível. Assumindo que ϕ é regular, as condições a serem satisfeitas para permitir suavizado isotrópico serão

$$\phi'(0) = 0; \lim_{s \rightarrow 0^+} \frac{\phi'(s)}{s} = \lim_{s \rightarrow 0^+} \phi''(s) = \phi''(0) > 0,$$

neste caso (2.12) fica

$$R^*Ru + \lambda\phi''(0)\Delta u = R^*u_0. \quad (2.13)$$

- **Para gradientes elevados.** Na vizinhança das bordas, a imagem apresenta fortes gradientes, os quais devem ser preservados. Logo, adota-se a atitude de difundir ao longo das isolinhas (ou seja na direção τ), para isto, na presença de gradientes elevados o coeficiente na direção normal deve ser eliminado. Tem-se então que

$$\lim_{s \rightarrow \infty^+} \phi''(s) = 0 \quad \text{e} \quad \lim_{s \rightarrow \infty^+} \frac{\phi'(s)}{s} = \beta > 0. \quad (2.14)$$

É fácil verificar que as condições (2.13) e (2.14) são incompatíveis (ϕ' cresce linearmente no ∞ , logo ϕ'' deveria ser pelo menos constante no ∞ , o que contradiz a primeira parte de (2.14)), logo deve-se procurar um equilíbrio entre as duas. Por exemplo, pode-se exigir que as taxas de convergência delas sejam diferentes quando $s \rightarrow \infty^+$

$$\begin{aligned} \lim_{s \rightarrow \infty^+} \phi''(s) &= \lim_{s \rightarrow \infty^+} \frac{\phi'(s)}{s} = 0, \text{ e também} \\ \lim_{s \rightarrow \infty^+} \frac{\phi''(s)}{\frac{\phi'(s)}{s}} &= 0. \end{aligned} \quad (2.15)$$

Um exemplo é a função chamada de *Hipersurface Minimal Function*, dada por

$$\phi(s) = \sqrt{1 + s^2} \quad (2.16)$$

Questões referentes a existência e unicidade da solução do problema de mínimo

$$\inf_{u \in \mathcal{U}} \{E(u)\}, \quad (2.17)$$

não são aprofundadas neste texto, o leitor interessado pode-se referir a [16](pgs.66-69).

Na Fig. (2.4) são apresentados alguns resultados numéricos obtidos ao tratar a imagem de Lena poluída com ruído Gaussiano com esta técnica. Como no caso descrito anteriormente, para obter a solução numérica do problema (2.9) foi utilizado Método dos Elementos Finitos. Neste caso, também adotam-se elementos quadriláteros bilineares e a discretização das equações foi feita de forma equivalente ao caso isotrópico, mas considerando o coeficiente de difusividade não linear (sendo a função $\phi(s)$ dada por $\phi(s) = \sqrt{1 + s^2}$). Para resolver a não linearidade foi utilizado o algoritmo de Minimização Semi-Quadrática descrito em [103]. Ao se estudar o SNR das imagens processadas observa-se que existe uma diminuição considerável no ruído (SNR = 26.91 na imagem degradada, e 31.44, 29.16,



Figura 2.4: Remoção de ruído (SNR inicial = 26.92dB) com difusão não linear, imagem processada com $\lambda = 1$ (SNR = 31.44dB), $\lambda = 5$ (SNR = 29.16dB) e $\lambda = 25$ (SNR = 24.51dB). Na segunda coluna foi plotado o valor de $\phi(s)$ para cada caso.

24.51 nas imagens restauradas para os diferentes valores de $\lambda = 1, 5$ e 25 respectivamente).

Esta alternativa consegue remover o ruído, e ao mesmo tempo manter as bordas da imagem. Entretanto, como pode ser observado na terceira fila da Fig. 2.4, o resultado depende fortemente do parâmetro λ . A utilização de um valor de λ muito alto irá degradar as bordas da imagem, e a utilização de um valor muito pequeno implica na remoção de pouco ruído (este ponto será estudado em mais detalhe no Capítulo 4).

2.4 Métodos Evolutivos

Uma alternativa diferente à apresentada na seção anterior, onde se propõe resolver o problema de restauração como um problema de mínimo associado a equações de Euler-Lagrange estacionárias, é trabalhar com equações evolutivas. Esta classe de modelos pode ser formalmente escrita da seguinte maneira

$$\begin{cases} u_{,t}(\mathbf{x}, t) + f(\mathbf{x}, u(\mathbf{x}, t), \nabla u(\mathbf{x}, t), \nabla \nabla u(\mathbf{x}, t)) = 0 & \text{em } \Omega \times (0, T) \\ \frac{\partial u}{\partial \mathbf{n}} = 0 & \text{em } \partial\Omega \times (0, T) \\ u(\mathbf{x}, 0) = u_0(\mathbf{x}) \end{cases}, \quad (2.18)$$

para $u(\mathbf{x}, T)$ a imagem reconstruída a partir dos dados iniciais $u_0(\mathbf{x})$, ∇u o gradiente de u e $\nabla \nabla u$ o gradiente de segunda ordem. Pode-se ressaltar que a principal diferença em relação aos modelos apresentados até agora é a presença do parâmetro t , que caracteriza o tempo. A idéia é, a partir da imagem dada $u_0(\mathbf{x})$, construir uma família de imagens $\{u(\mathbf{x}, t)\}_{t>0}$, representando diferentes versões (ou escalas) de $u_0(\mathbf{x})$ [2, 3, 101, 202]. Na medida que t avança, espera-se que $u(\mathbf{x}, t)$ seja uma imagem mais simplificada, ou seja, estruturas existentes em $u(\mathbf{x}, t_2)$ são simplificações de estruturas presentes em $u(\mathbf{x}, t_1)$ para $t_1 < t_2$. Desta maneira, novas estruturas não podem ser criadas, por este motivo o parâmetro t é chamado de “variável de escala”.

Determinar $f(., ., ., .)$ é um passo importante no processo já que existem objetivos contraditórios:

- eliminar ruído de u (introduzindo difusão na imagem),
- preservar características da imagem como bordas, ângulos e outras singularidades.

Seguindo a classificação tradicional de EDP's, pode-se identificar o tipo de operação resultante sobre a imagem:

- * **Forward Parabolic EDP:** geralmente utilizadas para remover ruído de uma imagem.
- * **Backward Parabolic EDP:** geralmente utilizadas para remover ruído e realçar bordas de uma imagem.
- * **Hiperbolic EDP:** utilizadas para realçar bordas, quando uma imagem apresenta perda da definição.

Os casos apresentados a seguir correspondem ao primeiro grupo. Além do mais, objetiva-se suavizar uma imagem que apresenta ruído, para o qual será utilizada a equação de calor. Utilizando, portanto, uma generalização do métodos proposta por Perona e Malik [164], o ruído é reduzido preservando as bordas de imagem. Posteriormente, outros métodos amplamente utilizados para restauração de imagens baseados nestas idéias são apresentados.

2.4.1 A Equação de Calor

Provavelmente a equação em derivadas parciais mais estudada em processamento de imagens seja a equação linear de calor [3, 202]

$$\begin{cases} u_t(\mathbf{x}, t) - \Delta u(\mathbf{x}, t) = 0 & \text{para } t > 0, \mathbf{x} \in \mathbb{R}^2 \\ u(\mathbf{x}, 0) = u_0(\mathbf{x}) \end{cases}. \quad (2.19)$$

Primeiramente considera-se $u_0(\mathbf{x})$ definida dentro da região $[0, 1] \times [0, 1]$ do plano $x - y$. Por simetria, pode-se estender o domínio de u a $\Omega = [-1, 1] \times [-1, 1]$, e supondo que esta é periódica, u pode ser estendida a \mathbb{R}^2 . A motivação para levar esta hipótese em consideração vem do fato que resolver esta equação para o dado inicial $u_0(\mathbf{x})$, é análogo a aplicar um filtro Gaussiano linear na imagem. Em particular, seja $u_0 \in L^1(\Omega)$, então a solução explícita de (2.19) é dada por

$$u(\mathbf{x}, t) = \int_{\mathbb{R}^2} G_{\sqrt{2t}}(\mathbf{x} - \mathbf{y}) u(\mathbf{y}) d\mathbf{y} = (G_{\sqrt{2t}} \star u_0)(\mathbf{x}), \quad (2.20)$$

sendo $G_\sigma(\mathbf{x})$ o *kernel* Gaussiano 2D, dado por

$$G_\sigma(\mathbf{x}) = \frac{e^{-\frac{|\mathbf{x}|^2}{2\sigma^2}}}{2\pi\sigma^2}. \quad (2.21)$$

Esta operação corresponde à aplicação de um filtro passa-baixa na imagem. Desta equação pode-se obter a relação entre o tempo t e o parâmetro σ que determina a amplitude do *kernel* Gaussiano. As propriedades de $u(\mathbf{x}, t)$ podem ser resumidas na seguinte proposição:

Proposição 1. *Sejam $u_0 \in L^1(\Omega)$ e $u(\mathbf{x}, t)$ dada por (2.20). Logo, para todo $t > 0$ e $\mathbf{x} \in \mathbb{R}^2$, $u(\mathbf{x}, t)$ satisfaz a equação de calor com condição inicial $u_0(\mathbf{x})$, ou seja*

$$\begin{aligned} u_t(\mathbf{x}, t) - \Delta u(\mathbf{x}, t) &= 0 \quad e \quad \lim_{t \rightarrow 0} \int_{\Omega} |u(\mathbf{x}, t) - u_0(\mathbf{x})| d\Omega = 0 \\ u(\mathbf{x}, 0) &\in L^1(\Omega) \quad e \quad u \in C^\infty((0, T) \times \mathbb{R}^2) \quad \forall T > 0. \end{aligned} \quad (2.22)$$

Além disso, se t_1 é qualquer real positivo, existe uma constante $C(t_1)$ tal que para $t \in [t_1, \infty]$

$$\sup_{\mathbf{x} \in \mathbb{R}^2} |u(\mathbf{x}, t)| \leq C(t_1) |u_0|_{L^1(\Omega)}, \quad (2.23)$$



Figura 2.5: Restauração com equação de calor isotrópica, imagem do barco com ruído (mesmo ruído utilizado nos casos anteriores, $\text{SNR} = 26.92\text{dB}$) e processada depois de 5 ($\text{SNR} = 24.38\text{dB}$), 25 ($\text{SNR} = 20.18\text{dB}$) e 125 ($\text{SNR} = 17.72\text{dB}$) iterações.

se $u_0 \in L^\infty(\Omega)$ tem-se também o princípio do máximo

$$\inf_{x \in \mathbb{R}^2} u_0(x) \leq u(x, t) \leq \sup_{x \in \mathbb{R}^2} u_0(x). \quad (2.24)$$

É interessante ressaltar que a unicidade de u depende da sua periodicidade em \mathbb{R}^2 . Neste caso a equação foi resolvida utilizando um esquema de diferenças. A imagem foi discretizada por pixel em um malha de quadrados regular. As derivadas no espaço (tanto em x quanto em y) foram substituídas por diferenças finitas (ver Apêndice A.1), tomando como intervalo característico (Δx e Δy) o tamanho do pixel, considerado também de tamanho unitário. Para calcular as derivadas temporais foi utilizado um algoritmo explícito. Para os resultados apresentados nesta seção, foi escolhido o parâmetro $\sigma = 1$ e

$\Delta t = 1/4$ (ver Apêndice A.1.2).

No caso, não importando a regularidade dos dados iniciais, as bordas são perdidas. De fato, a equação de calor tem velocidade de propagação infinita, o que não é uma propriedade desejável, já que as bordas e outras características de interesse da imagem são severamente danificadas. Assim, embora a equação de calor tenha sido amplamente utilizada em processamento de imagens, observam-se algumas desvantagens, dentre outras, a imagem resultante é muito suavizada (Figura 2.5).

2.4.2 Difusão Não-linear (Perona e Malik)

Para evitar o problema de perda de bordas, diferentes alternativas podem ser encontradas na literatura. Uma delas, é proposta por Perona e Malik [164], dada por

$$\begin{cases} u_{,t} = \text{div}(c(|\nabla u|^2)\nabla u) & \text{em } \Omega \times [0, T] \\ \frac{\partial u}{\partial \mathbf{n}} = 0 & \text{em } \partial\Omega \times [0, T] \\ u(\mathbf{x}, 0) = u_0(\mathbf{x}) & \text{em } \Omega, \end{cases} \quad (2.25)$$

para $c(s) : [0, \infty) \rightarrow [0, \infty)$. Neste caso foi escolhida f na Eq. (2.18) como sendo

$$f(\mathbf{x}, u(\mathbf{x}, t), \nabla u(\mathbf{x}, t), \nabla \nabla u(\mathbf{x}, t)) = \text{div}(c(|\nabla u|)\nabla u). \quad (2.26)$$

É fácil verificar que, se $c \equiv 1$, a equação de calor isotrópica é recuperada. A principal função de c é deter a difusão perto das bordas presentes na imagem, ou seja, se $c(0) = 1$ e $\lim_{s \rightarrow \infty} c(s) = 0$, têm-se que:

- na presença de altos gradientes (em geral aparecem nas bordas da imagem), ocorrerá pouca difusão.
- na presença de gradientes pequenos (regiões homogêneas da imagem), a difusão será forte.

Seguindo os mesmos passos que no caso estacionário, é possível reescrever o operador divergência na seguinte forma, tem-se

$$\text{div}(c(|\nabla u|^2)|\nabla u|) = \left(2|\nabla u|^2 c'(|\nabla u|^2) + c(|\nabla u|^2)\right) u_{\nu\nu} + c(|\nabla u|^2) u_{\tau\tau}.$$

Denominando $b(s) = c(s) + 2sc'(s)$, obtém-se

$$u_{,t} = c(|\nabla u|^2) u_{\tau\tau} + b(|\nabla u|^2) u_{\nu\nu}. \quad (2.27)$$

Esta equação pode ser interpretada como a soma de difusões nas direções ν e τ às isolinhas da imagem (Fig. 2.3), sendo c e b dois coeficientes de peso.

Para preservar bordas (altos gradientes), será de interesse que b seja sempre menor que c , ou seja $\lim_{s \rightarrow \infty} \frac{b(s)}{c(s)} = 0$. Usando a definição de $b(s)$, tem-se

$$\lim_{s \rightarrow \infty} \frac{b(s)}{c(s)} = \lim_{s \rightarrow \infty} \left[\frac{c(s)}{c(s)} + \frac{2sc'(s)}{c(s)} \right] = 0 \Rightarrow \lim_{s \rightarrow \infty} \frac{sc'(s)}{c(s)} = -\frac{1}{2}.$$

Pensando em funções crescentes em potências³, tem-se que este limite implica que $c(s) \approx \frac{1}{\sqrt{s}}$ quando $s \rightarrow \infty$. Para analisar a parabolicidade⁴ de (2.25), reescreve-se (2.27) da seguinte maneira

$$u_{,t} = A_{11}(|\nabla u|^2)u_{xx} + 2A_{12}(|\nabla u|^2)u_{xy} + A_{22}(|\nabla u|^2)u_{yy}, \quad (2.28)$$

sendo

$$\begin{aligned} A_{11}(|\nabla u|^2) &= 2u_x^2 c'(|\nabla u|^2) + c(|\nabla u|^2) \\ A_{12}(|\nabla u|^2) &= 2u_x u_y c'(|\nabla u|^2) \\ A_{22}(|\nabla u|^2) &= 2u_y^2 c'(|\nabla u|^2) + c(|\nabla u|^2) \end{aligned}$$

Logo, para que (2.28) seja parabólica, a matriz

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{12} & A_{22} \end{bmatrix},$$

deve ser definida positiva, i.e., $\det(A) > 0$, logo

$$\begin{aligned} \det A &= A_{11}A_{22} - A_{12}^2 \\ &= \left(2u_x^2 c'(|\nabla u|^2) + c(|\nabla u|^2)\right) \left(2u_y^2 c'(|\nabla u|^2) + c(|\nabla u|^2)\right) - \left(2u_x u_y c'(|\nabla u|^2)\right)^2 \\ &= 4u_x^2 u_y^2 c'(|\nabla u|^2) + 2u_x^2 c'(|\nabla u|^2)c(|\nabla u|^2) \\ &\quad + \left(2u_x^2 c'(|\nabla u|^2)c(|\nabla u|^2)\right) - 4u_x^2 u_y^2 c'(|\nabla u|^2) + c(|\nabla u|^2)^2 \\ &= 2u_x^2 c'(|\nabla u|^2)c(|\nabla u|^2) + \left(2u_x^2 c'(|\nabla u|^2)c(|\nabla u|^2)\right) + c(|\nabla u|^2)^2 \\ &= c'(|\nabla u|^2) \left(2u_x^2 c'(|\nabla u|^2) + (2u_x^2 c'(|\nabla u|^2)) + c(|\nabla u|^2)\right) \\ &= \underbrace{c'(|\nabla u|^2)}_{>0} \underbrace{\left(2|\nabla u|^2 c'(|\nabla u|^2) + c(|\nabla u|^2)\right)}_{b(|\nabla u|^2)}. \end{aligned} \quad (2.29)$$

Assim, a seguinte condição deve ser satisfeita

$$b(s) > 0.$$

Tem-se então que, para que a Eq. (2.25) seja útil para eliminar ruído mantendo as bordas da imagem, $c(s)$ deve satisfazer

$$\begin{cases} c : [0, \infty) \rightarrow [0, \infty) \text{ decrescente,} \\ c(0) = 1 ; c(s) \approx \frac{1}{\sqrt{s}} \text{ para } s \rightarrow \infty, \\ b(s) = c(s) + sc'(s) > 0. \end{cases}$$

³ $c(s) = s^\alpha$ para $\alpha \in \mathbb{R}$.

⁴Encontrando uma condição que torne a equação (2.25) parabólica, pode-se garantir a existência e unicidade da solução.

Pode-se apreciar na Figura 2.6⁵ que as bordas são preservadas, mas alguns detalhes são perdidos. No caso foi utilizada a função $c(s)$ proposta originalmente por Perona e Malik, ou seja

$$c(s) = \frac{1}{1 + \frac{s^2}{\sigma^2}}. \quad (2.30)$$

Ao analisar o SNR destas imagens observa-se que existe uma diminuição considerável no ruído da imagem depois da restauração (sendo na imagem degradada $\text{SNR} = 26.87$ e nas imagens processadas 28.06dB, 31.20dB e 24.44dB para 5, 25 e 125 iterações respectivamente).

2.4.3 Suavizado Seletivo (Alvarez, Morel e Lions)

No trabalho de Alvarez, Morel e Lions [4] é apresentado um esquema baseado na idéia chamado *Suavizado Seletivo*. Neste trabalho, os autores propõem a seguinte equação

$$\begin{cases} u_t = g(|G \star \nabla u|)((1 - h(|\nabla u|))\Delta u + \\ \quad h(|\nabla u|)|\nabla u| \operatorname{div}(\frac{\nabla u}{|\nabla u|})) \\ u(\mathbf{x}, 0) = u_0(\mathbf{x}) \end{cases} \quad (2.31)$$

sendo G um *kernel* de suavizado Gaussiano, $|G \star \nabla u|$ uma estimativa local do gradiente sem ruído e $g(x)$ uma função real decrescente que tende a zero em ∞ . Os autores utilizaram

$$\begin{aligned} g(x) &= \exp\left(-\left(\frac{2x}{g_\delta}\right)^2\right), \quad \text{e} \\ h(x) &\begin{cases} 0 & \text{se } x < h_\delta \\ 1 & \text{caso contrário.} \end{cases} \end{aligned} \quad (2.32)$$

sendo g_δ e h_δ constantes a serem escolhidas pelo usuário. Neste caso, o termo de difusão pode ser reescrito da seguinte forma

$$g(|G \star \nabla u|)u_{\tau\tau} + g(|G \star \nabla u|)(1 - h(|\nabla u|))u_{\nu\nu}, \quad (2.33)$$

obtendo-se então uma expressão decomposta na direção normal e tangente às isolinhas da imagem. Esta alternativa apresenta bons resultados como pode ser visto na Figura 2.7. O SNR confirma novamente que existe uma diminuição (embora não muito significativa) no nível de ruído destas imagens ($\text{SNR} = 26.87$ na imagem poluída, e $\text{SNR} = 29.98\text{dB}$, 28.24dB e 26.36dB para 5, 25 e 125 iterações, respectivamente).

Pode-se observar que as bordas são preservadas por esta técnica, mesmo assim este método apresenta dificuldade para remover o ruído ao longo delas. Além disso, um critério heurístico de parada deve ser introduzido para evitar a degradação completa da imagem.

⁵Como no caso anterior, a equação foi resolvida utilizando um esquema de diferenças, tanto no espaço como no tempo.

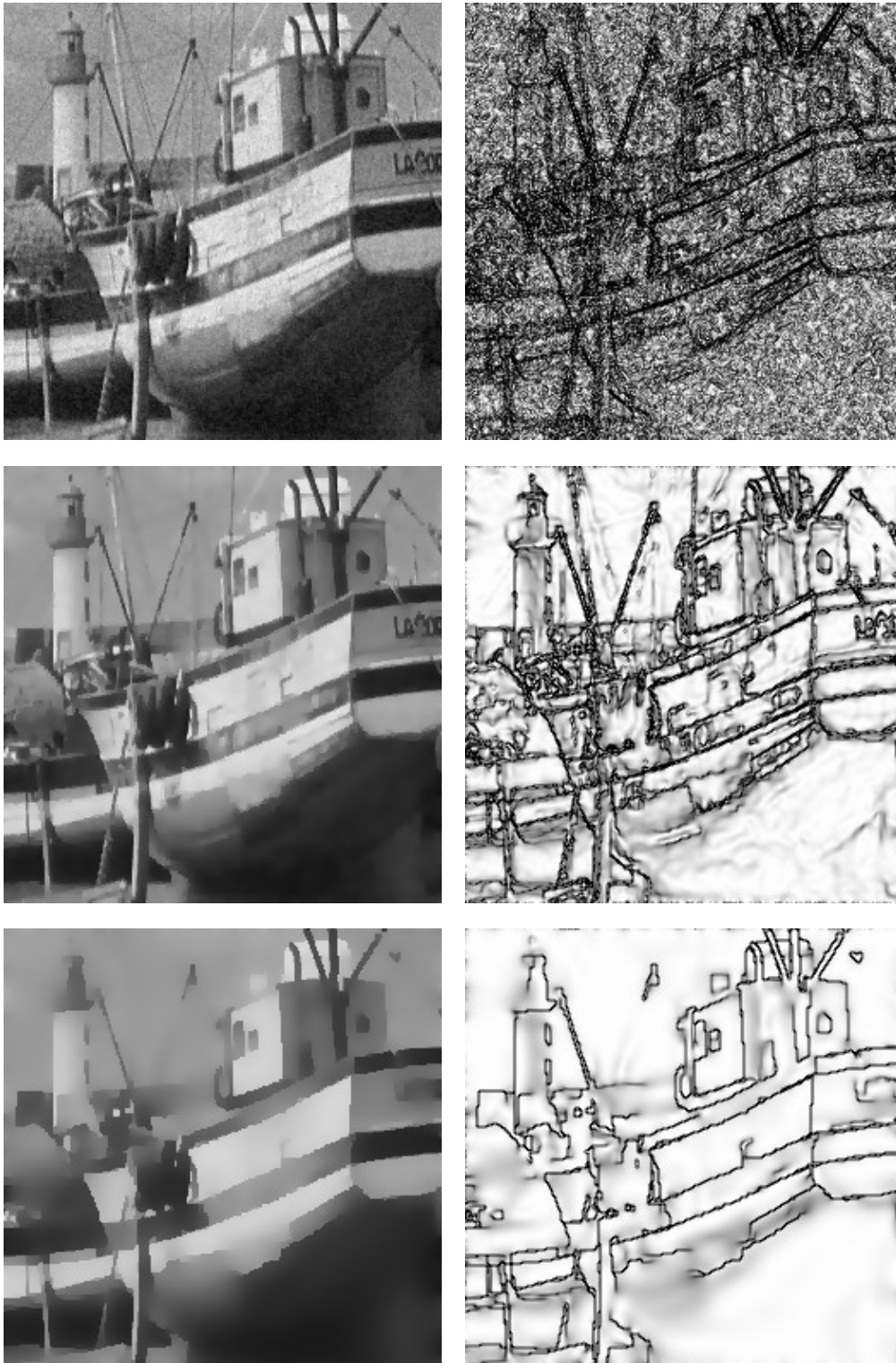


Figura 2.6: Restauração com equação de calor não linear, imagem do barco (mesmo ruído utilizado nos casos anteriores, $\text{SNR} = 26.88\text{dB}$) processada depois de 5 ($\text{SNR} = 28.06\text{dB}$), 25 ($\text{SNR} = 31.20\text{dB}$) e 125 ($\text{SNR} = 24.44\text{dB}$) iterações ($\sigma = 7$). Na segunda coluna foi plotado o valor da função c em cada ponto do domínio para cada imagem processada.



Figura 2.7: Restauração com o algoritmo de suavizado seletivo, imagem do barco com ruído (SNR = 26.87dB) e processada depois de 5 (SNR = 29.98), 25 (SNR = 28.24) e 125 (SNR = 26.36) iterações.

2.4.4 Difusão Anisotrópica Robusta (Black *et al.*)

Black *et al.* [27] propõem várias alternativas para a função $c(s)$, uma delas é a norma do erro de Tukey. Esta função tem a vantagem de impedir completamente a difusão para gradientes superiores a um determinado valor.

No trabalho citado, as seguintes 3 alternativas para a função $c(s)$ (coeficiente de difusão isotrópica) são estudadas (Fig. 2.8):

- Função de Huber [92]:

$$c(s) = \begin{cases} 1/\sigma & |s| \leq \sigma \\ 1/|s| & \text{caso contrário.} \end{cases} \quad (2.34)$$

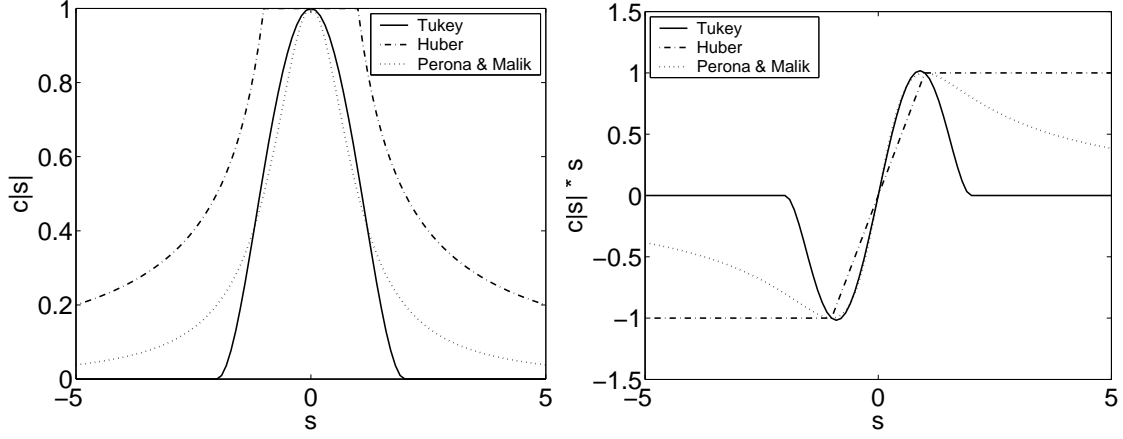


Figura 2.8: Diferentes alternativas para a função $c(s)$. À direita é apresentado o gráfico da função $c(|s|)$ e à esquerda o correspondente a $c(|s|)s$.

- Perona e Malik [164]:

$$c(s) = \frac{2}{\sigma(1 + \frac{s^2}{\sigma^2})}. \quad (2.35)$$

- Norma do erro robusta de Tukey [27]:

$$c(s) = \begin{cases} \frac{1}{0.5626\sigma} \left(1 - \frac{s^2}{4\sigma^2}\right)^2 & |s| < 2\sigma \\ 0 & \text{caso contrário.} \end{cases} \quad (2.36)$$

Mesmo sendo $c(s)$ um coeficiente escalar, a equação diferencial pode ser discretizada como anisotrópica (ver Apêndice A.1.1). A discretização utilizada (e.g., diferenças finitas) nas equações apresentadas é descrita na Seção A.1 (Eq. (A.15)).

Black *et al.* mostram as vantagens de utilizar uma função $c(s)$ que consegue “deter” a difusão através das bordas na imagem (e.g., a norma do erro de Tukey), já que para gradientes acima de um certo valor, a difusão é completamente detida.

Na Figura (2.8) é apresentado o gráfico das funções $c(s)$ e $c(s)s$ para as três alternativas (Huber, Perona e Malik, e Tukey). Estas funções foram escalonadas de maneira que as três comesçassem a “rejeitar” gradientes a partir de um mesmo valor σ e para todas terem o mesmo valor máximo. Nos testes realizados no Capítulo 4, onde são apresentadas algumas comparações entre os diferentes métodos, as funções de Perona e Malik, e Tukey foram utilizadas.

No trabalho de Black *et al.*, também é proposta uma metodologia para estimar o valor do parâmetro σ , calculando-o como

$$\sigma = 1.4826 \text{ mediana}(|\nabla u_0 - \text{mediana}(\nabla u_0)|). \quad (2.37)$$

Este cálculo é baseado na hipótese que ∇u tem distribuição de probabilidade normal. A mediana é o ponto médio de uma série de números; sendo a metade dos valores maiores que a mediana e a outra metade menor (notar que a mediana não necessariamente é

igual a média). A constante nesta fórmula (que é um fator de escala) vem do fato que o desvio mediano absoluto de uma distribuição normal com média nula e variância unitária é $0.6745 = 1/1.4826$ [27]. Esta simples técnica, que permite estimar o valor de σ , é utilizada nos casos apresentados no Capítulo 4. Nos resultados e figuras mostrados neste capítulo o valor de σ foi determinado de maneira arbitrária (sendo o valor escolhido $\sigma = 20$).

No trabalho de Black *et al.* os autores também analisam a introdução de processos de otimização de linhas na remoção de ruído. Esta alternativa introduz restrições de coerência espacial, que mostrou ter melhorado a continuidade das bordas restauradas. Esta questão não é abordada neste trabalho.

Alguns resultados para esta técnica são apresentados na Fig. 2.9. Ao comparar o SNR das imagens restauradas com o da imagem degradada, observa-se também uma diminuição no ruído da imagem restaurada (SNR imagem degradada = 26.87, SNR imagem restaurada = 31.33dB, 30.83dB e 26.34dB para 5, 25 e 125 iterações respectivamente).

2.5 Comentários

O método da energia consegue remover grande quantidade de ruído da imagem quando o valor de λ é bem escolhido. Por outro lado, se este parâmetro não for determinado de forma correta, a imagem pode ser severamente danificado. Em outras palavras, dependendo do nível de ruído na imagem, a escolha de um valor de λ muito pequeno irá remover pouco ruído da imagem, e um valor muito elevado, irá danificar as bordas e outros detalhes de interesse.

Uma propriedade que caracteriza os métodos de restauração de imagens baseados em difusão não-linear isotrópica é a dificuldade para remover ruído ao longo das bordas presentes na imagem. Esta propriedade altamente indesejada da difusão não linear e isotrópica é parcialmente reduzida quando um tensor de difusão não linear anisotrópico é introduzido [64]. Nesse caso, o tensor de difusividade $\mathbf{c}(|\nabla u|)$ é construído a partir dos autovetores e correspondentes autovalores do tensor $\mathbb{J} = \nabla u \otimes \nabla u$ também satisfazendo que $\mathbf{c}(s) \rightarrow 0$ quando $s \rightarrow \infty$. Com isto se pretende que $\mathbf{c}(\cdot)$ diminua a difusão através das bordas e um aumento desta na direção tangente, eliminando assim maior quantidade de ruído. No entanto, e como a difusão através das bordas não é completamente detida, critérios de parada heurísticos devem ser introduzidos na evolução no tempo para evitar a perda de detalhes da imagem.

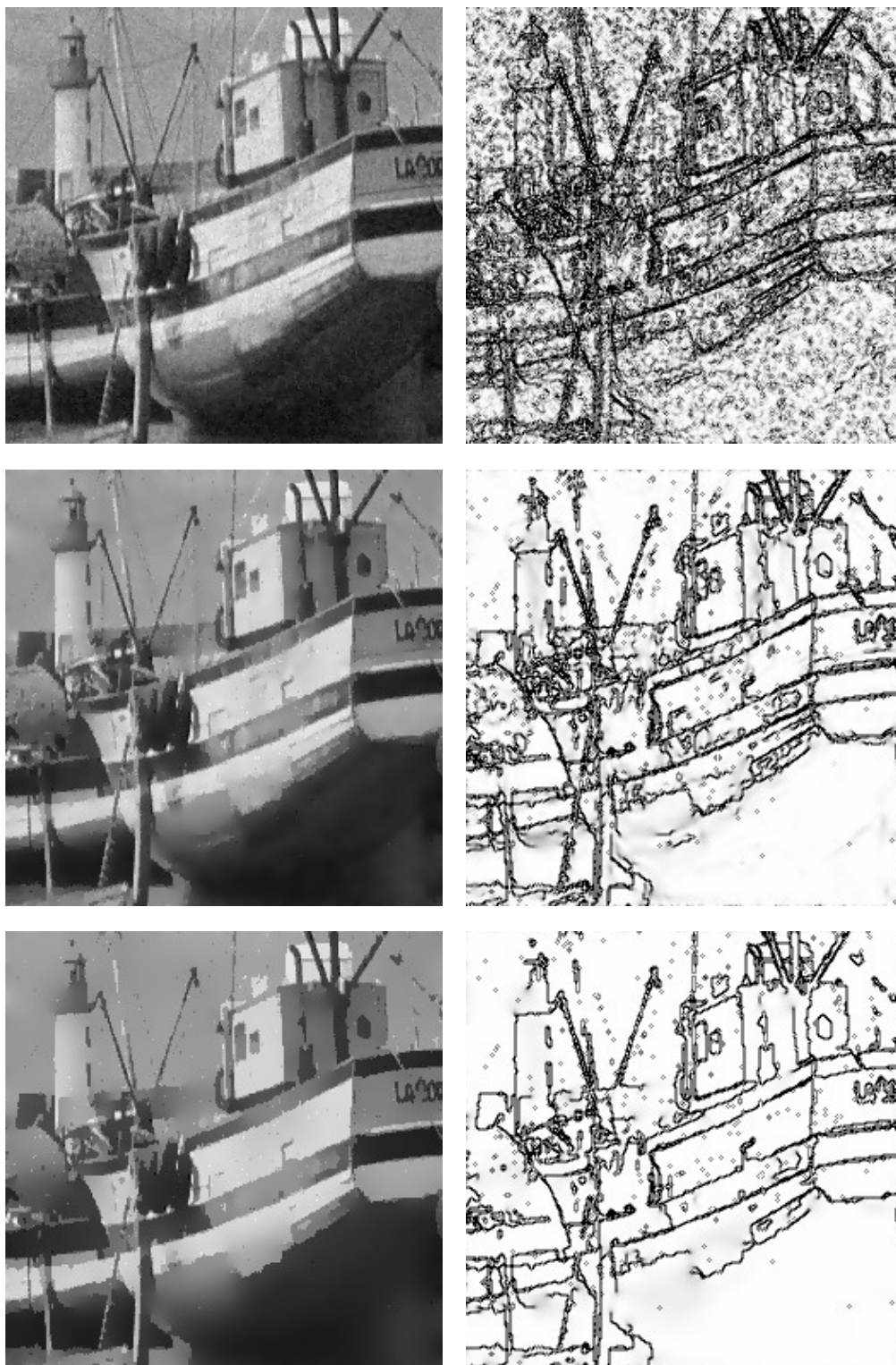


Figura 2.9: Restauração com equação de calor não linear, função $c(x)$ proposta por Black *et al.* (norma do erro de Tukey), imagem do barco (SNR inicial = 26.88dB) processada depois de 5 (SNR = 31.33), 25 (SNR = 30.83) e 125 (SNR = 26.34) iterações.

Apêndice Capítulo 2

2.A Medidas de erro em imagens

O *Signal-to-Noise Ratio* (SNR) e *Peak Signal-to-Noise Ratio* (PSNR) são estimativas que permitem quantificar a qualidade de uma imagem, por exemplo, depois do processo de restauração. A idéia é calcular um único número que consiga refletir a qualidade da imagem reconstruída. Imagens com medidas maiores são rotuladas como melhores. De fato, medidas tradicionais de SNR não consideram a percepção subjetiva do olho humano. Por este motivo, é importante ressaltar que valores maiores do SNR nem sempre indicam melhor qualidade.

Neste caso, são utilizadas como medidas quantitativas o SNR, o PSNR (Peak-SNR) e o Erro Médio Quadrático (MSE). Os mesmos são calculados para \bar{u} a imagem original e u a imagem reconstruída, da seguinte forma:

- Erro Meio Quadrático (MSE):

$$\text{MSE}(\bar{u}, u) = \frac{\sum_s (\bar{u}^s - u^s)^2}{(N \times M)} \quad (2.38)$$

- *Peak Signal-to-Noise Ratio* (PSNR):

$$\text{PSNR}(\bar{u}, u) = 10 * \log_{10} \left(\frac{255^2}{\text{MSE}(\bar{u}, u)} \right) \quad (2.39)$$

- *Signal-to-Noise Ratio* (SNR):

$$\text{SNR}(\bar{u}, u) = 10 * \log_{10} \left(\frac{(\max(\bar{u}^s) - \min(\bar{u}^s))^2}{\text{MSE}(\bar{u}, u)} \right) \quad (2.40)$$

O SNR e o PSNR são medidos em decibéis (dB). O dB é uma unidade de medida de força de um sinal, usualmente é a relação entre o sinal transmitido e uma certa fonte. A cada +3 dB o sinal aumenta 50% na sua força, logo, para uma perda de 6 dB, existe uma queda na força do sinal de 75%. No caso, é de interesse relacionar o sinal transmitido (a intensidade da imagem) com o ruído que ela apresenta (erro com relação a imagem original não degradada).

Como o erro é uma variável aleatória, pode-se calcular a sua média $\mu(e)$ e o desvio padrão $\sigma(e)$ de $e(s)$, estes sendo

$$e(s) = \bar{u}^s - u^s, \quad (2.41)$$

$$\mu(e) = \frac{\sum_s e(s)}{(N \times M)}, \quad (2.42)$$

$$\sigma(e) = \sqrt{\frac{\sum_s [\mu(e) - e(s)]^2}{(N * M)}}. \quad (2.43)$$

Capítulo 3

Derivada Topológica em Restauração de Imagens

Neste capítulo será mostrado como a derivada topológica pode ser empregada para resolver o problema de restaurar uma imagem degradada pela presença de ruído. Em particular, quando o Método de Energia (Capítulo 2, Seção 2.3) foi apresentado, a equação de Euler-Lagrange associada ao mínimo do funcional dado pela Eq. (2.4) correspondia a uma equação estacionária de calor onde o coeficiente de difusão estava dado por (Eq. (2.9))

$$c(|\nabla u|) = \frac{\phi'(|\nabla u|)}{|\nabla u|}.$$

Então, o problema consistia em determinar a função ϕ que permita diminuir o ruído preservando bordas. Uma propriedade que caracteriza os métodos de restauração de imagens baseados em difusão não-linear isotrópica é a dificuldade para remover ruído ao longo das bordas presentes na imagem. Esta propriedade altamente indesejada da difusão não linear, porém isotrópica, foi parcialmente reduzida quando um tensor de difusão não linear anisotrópico foi introduzido [64]. Nesse caso, um tensor de difusividade $\mathbf{c}(|\nabla u|)$ foi construído a partir dos autovetores e correspondentes autovalores do tensor $\mathbb{J} = \nabla u \otimes \nabla u$. Contudo, e como a difusão através das bordas não é completamente detida, critérios de parada heurísticos devem ser introduzidos para evitar a perda de detalhes da imagem.

Como será mostrado nesta seção, este critério não leva em consideração a correta sensibilidade do funcional quando perturbações são introduzidas. De fato, impedir a difusão na direção ortogonal às iso-linhas da intensidade u da imagem corresponde a introduzir um talho na direção das mesmas. Assim, o emprego da derivada topológica na recuperação de imagens surge naturalmente já que é o único conceito que permite analisar variações abruptas nas propriedades materiais por exemplo, fornecendo automaticamente o tensor de difusão que, no que segue, será designado por \mathbb{K} .

Para formalizar o acima descrito e lembrando o Capítulo 1, considere a imagem caracterizada por sua intensidade $u_0 \in L^2(\Omega)$ definida no domínio limitado e aberto $\Omega \in \mathbb{R}^2$ (a extensão a \mathbb{R}^3 é imediata). Para cada imagem restaurada, caracterizada pela intensidade $u \in H^1(\Omega)$, pode-se associar a função custo

$$\mathcal{J}(u) = \int_{\Omega} \nabla u \cdot \nabla u \, d\Omega,$$

que mede a “qualidade” da restauração dada pela solução do seguinte problema variacional: Determinar $u \in H^1(\Omega)$ tal que

$$\int_{\Omega} k \nabla u \cdot \nabla \eta \, d\Omega + \int_{\Omega} (u - u_0) \eta \, d\Omega = 0 \quad \forall \eta \in H^1(\Omega).$$

Como apresentado no capítulo anterior, existem diversos métodos para remover

ruído e realçar bordas em uma imagem degradada por algum fenômeno. Estes podem ser distinguidos em dois tipos: baseados na solução de um problema estacionário e baseados na solução de um problema evolutivo. Os dois tipos de métodos baseiam-se em aplicar difusão não-linear/anisotrópica em uma imagem. Em ambos casos, o coeficiente ou tensor de difusão é calculado como uma função do gradiente naquela região da imagem. Assim, este coeficiente assume valores pequenos quando o gradiente é elevado (i.e., nas bordas da imagem) de maneira a deter a difusão, e valores maiores quando o gradiente é pequeno (nas regiões homogêneas da imagem) o que promove uma difusão maior. Ambos os métodos possuem dois parâmetros. O método estacionário tem um parâmetro que determina quais gradientes serão considerados bordas e quais não, e um segundo parâmetro que caracteriza a intensidade da difusão aplicada. No caso do método evolutivo, o primeiro deles é de alguma maneira semelhante ao utilizado pelo método estacionário que determina quais valores do gradiente serão considerados bordas, e o número de iterações. Para os dois métodos, o parâmetro que determina quais gradientes serão considerados bordas pode ser estimado. Não acontece o mesmo com os outros parâmetros que devem ser estimados para cada imagem dependendo do seu tipo ou nível de ruído. Nos dois casos, a escolha destes parâmetros irá determinar a qualidade do resultado.

Por outro lado, o conceito de derivada topológica (apresentado no Capítulo 1), permite quantificar a sensibilidade de uma medida de performance ou função custo quando o domínio de definição do problema é perturbado. Assim, se uma função custo for associada ao ruído presente na imagem, será possível empregar sua derivada topológica para desenvolver algoritmos apropriados de restauração.

Neste capítulo, o problema de restauração de uma imagem poluída com ruído é estudado dentro desta perspectiva. Neste contexto, duas equações de estado são estudadas: a primeira baseada em um problema estacionário e a segunda em um problema evolutivo. Como mencionado no Capítulo 1, a característica notável desta derivada é a de permitir calcular a sensibilidade de uma função custo com respeito a um parâmetro que pode variar de maneira não suave. Em particular foi visto que a derivada topológica permite determinar o lugar e a orientação dos talhos que deveriam ser introduzidos no domínio para reduzir o valor da função custo. Do ponto de vista da equação de estado, os talhos impedem a difusão na direção ortogonal a eles, sendo possível difundir apenas na direção tangente. Em outras palavras, a derivada topológica fornece um procedimento para calcular o campo tensorial anisotrópico de difusão/condutividade que deve ser empregado para eliminar a maior quantidade de ruído preservando as bordas. Finalmente a imagem restaurada é calculada como a solução da equação de estado, mas empregando o tensor de difusão obtido com a derivada topológica.

Na Seção 3.1, esta técnica para a equação de estado estacionária é estudada e também são apresentadas algumas variantes que conseguem melhorar os resultados numéricos com ela obtidos.

Baseada em uma equação de estado de tipo evolutivo, na Seção 3.2 é apresentada a formulação discreta desse problema dando lugar a novo algoritmo de restauração robusto e computacionalmente eficiente. Finalmente, são apresentados resultados obtidos com estes

métodos para imagens artificialmente poluídas com ruído.

3.1 Abordagem Contínua

Para um ϵ pequeno, seja $\Omega_\epsilon = \Omega \setminus \gamma_\epsilon$ o domínio perturbado pela inserção de um talho (fissura) $\gamma_\epsilon = \hat{\mathbf{x}} + \epsilon\gamma$, onde $\hat{\mathbf{x}} \in \Omega$ e $\gamma(\mathbf{m})$ uma fissura reta, sendo \mathbf{m} a direção normal de γ (Fig. 3.1). Seja então u_ϵ a solução do mesmo problema variacional no domínio perturbado Ω_ϵ e $\mathcal{J}(u_\epsilon)$ a função custo associada a ela. Logo, tem-se a seguinte expressão assintótica topológica de $\mathcal{J}_\epsilon(u_\epsilon)$ quando $\epsilon \rightarrow 0$, i.e.,

$$\mathcal{J}_\epsilon(u_\epsilon) = \mathcal{J}(u) + f(\epsilon)D_T(\hat{\mathbf{x}}, \mathbf{m}) + \mathcal{O}(f(\epsilon)),$$

onde, como foi mostrado no Capítulo 1 (Eq. (1.35))

$$D_T(\hat{\mathbf{x}}, \mathbf{m}) = \mathbb{M} \mathbf{m} \cdot \mathbf{m}$$

sendo \mathbb{M} um tensor simétrico dado por

$$\mathbb{M} = -\left(\nabla u \otimes \nabla u + k(\nabla u \otimes_s \nabla q)\right). \quad (3.1)$$

e q a solução da equação adjunta

$$\int_{\Omega} (k \nabla q \cdot \nabla \eta + q \eta) d\Omega = - \int_{\Omega} \nabla u \cdot \nabla \eta d\Omega \quad \forall \eta \in \mathcal{V}.$$

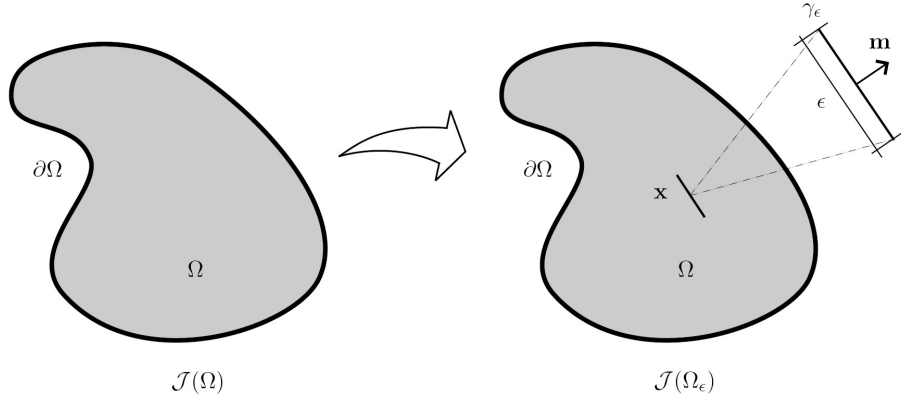


Figura 3.1: Conceito da Derivada Topológica para fissura.

Como mencionado, para um $\hat{\mathbf{x}}$ qualquer, $D_T(\hat{\mathbf{x}}, \mathbf{m})$ alcança o valor mínimo quando \mathbf{m} é o autovetor associado ao menor autovalor κ_{min} de \mathbb{M} . Então, considera-se como direção ótima da orientação da fissura $\gamma_\epsilon(\mathbf{m})$ o autovetor correspondente ao autovalor κ_{min} . Por sua vez, este valor mínimo será adotado como a derivada topológica associada a criação de um talho no ponto $\hat{\mathbf{x}}$.

Utilizando a informação dada pela D_T , Belaid *et al.* [22] propõem um método de remoção de ruído que preserva as bordas da imagem. Entretanto, estes autores cometem

um erro ao calcular a expansão assintótica não incorporando o coeficiente k na Eq. (3.1). Assim, os resultados apresentados nesse trabalho somente são corretos para $k = 1$. Este algoritmo consiste em calcular a derivada topológica e simular pequenas fissuras nos locais que a derivada é menor que um determinado valor $D_{T\text{Lim}}$. Os autores apresentam dois algoritmos baseados nesta idéia: um isotrópico e outro anisotrópico.

Para resolver este problema de maneira numérica, interpreta-se a introdução de um coeficiente de difusividade pequeno (ou um tensor de condutividade que atua somente em uma direção) como uma fissura. Nos algoritmos propostos, o tensor de difusividade $\mathbb{K}(\mathbf{x})$ para os casos isotrópico e anisotrópico é calculado como uma função da D_T , isto é $\mathbb{K} = \mathbb{K}(\mathbf{x}, D_T)$:

- Difusão isotrópica baseada na D_T (D_T -Iso):
 - $\mathbb{K}(\mathbf{x}) = k_\epsilon \mathbb{I}$ se $D_T(\mathbf{x}) \leq D_{T\text{Lim}}$;
 - $\mathbb{K}(\mathbf{x}) = k_0 \mathbb{I}$ caso contrário.
- Difusão anisotrópica baseada na D_T (D_T -Aniso):
 - $\mathbb{K}(\mathbf{x}) = k_\epsilon(\mathbf{n} \otimes \mathbf{n}) + k_0(\mathbf{t} \otimes \mathbf{t})$ se $D_T(\mathbf{x}) \leq D_{T\text{Lim}}$;
 - $\mathbb{K}(\mathbf{x}) = k_0 \mathbb{I}$ caso contrário.

para $k_\epsilon \ll 1$ e k_0 um número real positivo.

Com este tensor de difusão assim definido a imagem restaurada é obtida resolvendo o seguinte problema variacional: Determinar $u^* \in H^1(\Omega)$ tal que

$$\int_{\Omega} \mathbb{K} \nabla u^* \cdot \nabla \eta \, d\Omega + \int_{\Omega} (u^* - u_0) \eta \, d\Omega = 0 \quad \forall \eta \in H^1(\Omega).$$

Como a solução u^* do problema variacional dado pela Eq. (1.14) não pode, em geral, ser conhecida explicitamente é necessário achar uma solução aproximada. Para este fim, é utilizado o Método dos Elementos Finitos [93]. Logo, utilizando o elemento finito mais simples dado por um quadrilátero bilinear (para o caso de uma imagem bidimensional) ou por um paralelepípedo trilinear (para o caso tridimensional) com os pontos nodais coincidentes com os centros dos elementos da imagem, soluções aproximadas u^h de u , q^h de q e u^{*h} de u^* podem ser facilmente obtidas para qualquer imagem $u_0 \in L^2(\Omega)$ (ver Seção A.2). Utilizando esta solução, a aproximação por elementos finitos \mathbb{M}^h do tensor \mathbb{M} toma a forma

$$\mathbb{M}^h = - \left(\nabla u^h \otimes \nabla u^h + k(\nabla u^h \otimes_s \nabla q^h) \right). \quad (3.2)$$

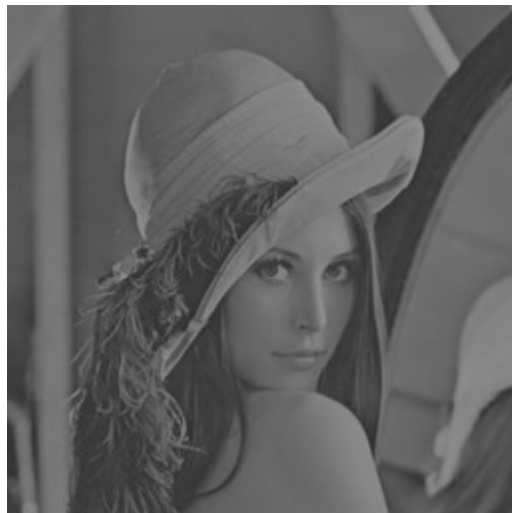
onde u^h e q^h são as interpolações de elementos finitos no ponto $\hat{\mathbf{x}}$ das funções u e q respectivamente.

Assim, para achar a imagem restaurada devem-se resolver 3 problemas de valor de contorno (Algoritmo 1) correspondentes aos campos escalares u^h do problema isotrópico, q^h do problema adjunto e u^{*h} solução do problema com o tensor de difusividade $\mathbb{K}(\mathbf{x})$ calculado utilizando D_T -Iso ou D_T -Aniso.

Como mencionado, este algoritmo utiliza um parâmetro ($D_{T\text{Lim}}$) para escolher os elementos da imagem que irão ter o seu coeficiente de difusividade modificado. Logo,



(a) Imagem de Lena original.



(b) Imagem de Lena com baixo contraste.



(c) Derivada topológica para as imagens anteriores.

Figura 3.2: Imagem de Lena com diferentes contrastes. A segunda fila corresponde à derivada topológica para cada imagem (desenhada utilizando a mesma escala em ambos casos).

dependendo dos valores da derivada topológica da imagem tratada, o valor deste coeficiente será modificado em diferente quantidade de pontos (e.g., duas imagens semelhantes com diferente contraste irão produzir uma derivada topológica com diferente escala de valores, já que a mesma depende de ∇u , ver Fig. 3.2). É fácil verificar que o parâmetro D_{TLim} deve ter um valor no intervalo $[D_{TMIN}, 0)$ (sendo D_{TMIN} o menor valor da D_T). Por outro lado, D_{TMIN} e a distribuição de valores da D_T neste intervalo pode variar consideravelmente para diferentes imagens, o que requer reajustar este parâmetro para diferentes imagens dificultando assim a sua estimativa. Por este motivo, D_{TLim} varia consideravelmente de uma imagem para outra, tornando-se um parâmetro de difícil ajuste.

Algoritmo 1 Restauração de imagens baseada na derivada topológica contínua

Entrada: Imagem degradada $u_0 \in L^2(\Omega)$, os parâmetros D_{TLim} e k_0 .

Saída: A imagem restaurada $u^* \in H^1(\Omega)$.

calcular u e q , soluções da equação de estado e adjunta respectivamente,
 calcular a matriz 2×2 \mathbb{M} e o seu menor autovalor κ_{min} em cada ponto do domínio,
 achar \mathbb{K} usando D_T -Iso ou D_T -Aniso,
 calcular u^* , a imagem restaurada, usando $\mathbb{K}(\mathbf{x})$ obtido anteriormente.

Uma alternativa diferente, apresentada em [111], é modificar o valor do coeficiente de difusividade utilizando um algoritmo de ponto fixo. Isto consiste em ordenar de forma crescente os valores da derivada topológica e escolher um percentual α dos mais negativos para inserir as “fissuras” nos locais correspondentes da imagem. Desta maneira, α controla o percentual de elementos da imagem que irão ter fissuras. Define-se então o conjunto \mathcal{M}_α como

$$\mathcal{M}_\alpha := \{D_T(s) : D_T(s) < 0 \text{ e } D_T(s) \text{ esta nos } \alpha\% \text{ valores mais negativos da } D_T\}. \quad (3.3)$$

Para s entre um e número de pixels na imagem. Esta alternativa permite um controle maior no algoritmo (Algoritmo 2).

Algoritmo 2 Restauração de imagens baseada na derivada topológica contínua II

Entrada: Imagem degradada $u_0 \in L^2(\Omega)$, os parâmetros α e k_0 .

Saída: A imagem restaurada $u^* \in H^1(\Omega)$.

calcular u e q , soluções da equação de estado e adjunta respectivamente,
 calcular a matriz 2×2 \mathbb{M} e o seu menor autovalor κ_{min} em cada ponto do domínio,
 achar \mathbb{K} usando D_T -Iso ou D_T -Aniso e Eq. (3.3),
 calcular u^* , a imagem restaurada, usando $\mathbb{K}(\mathbf{x})$ obtido anteriormente.

Pode-se observar na Figura 3.3 o resultado obtido com o Algoritmo 2 para D_T -Iso utilizando o tensor \mathbb{K} para a imagem de Lena poluída com ruído Gaussiano branco (a imagem processada é a apresentada na Fig. 2.1, utilizada nos métodos descritos no Capítulo 2). Em todos os testes realizados foi utilizado $k_0 = 2$. Visualmente, observa-se que o ruído é removido nos três casos. Quantitativamente, os 3 resultados (sejam, $\alpha = 0.10$, 0.20 e 0.30 respectivamente) apresentam melhoras consideráveis no SNR (indo de 26.92 na imagem degradada para 29.57, 30.36 e 30.88 nas imagens processadas respectivamente). Ao analisar os detalhes destas imagens, observa-se que, assim como os outros métodos não-lineares isotrópicos, apresenta dificuldade para remover ruído nas bordas da imagem. Na Fig. 3.4 são apresentados os resultados correspondentes a D_T -Aniso para calcular \mathbb{K} e a mesma imagem de Lena. Neste caso foram utilizados os mesmos valores de α ($\alpha = 0.10$, 0.20 e 0.30). Novamente o SNR é aumentado (indo de 26.92 na imagem degradada para 28.47, 28.85 e 29.05 nas imagens processadas, respectivamente). Desta vez, mesmo não alcançando o SNR do método isotrópico, este consegue remover ruído ao longo das bordas de uma maneira muito mais eficiente. Por último, na Fig. 3.5 é apresentada em



Figura 3.3: Restauração com algoritmo de derivada topológica contínua D_T -Iso da imagem de Lena (imagem poluída artificialmente utilizada no Capítulo 2) com $k_0 = 2$, fila superior: $\alpha = 0.10, 0.20$, fila inferior: $\alpha = 0.30$ e valor da derivada topológica para cada pixel.

detalhe a derivada topológica, os vetores normais (em preto) e tangentes (em branco), as fissuras e a imagem restaurada.

Neste momento, surge naturalmente uma questão: é possível usar a informação fornecida pela derivada topológica para construir um algoritmo de restauração de imagens completamente discreto sem perda de qualidade com relação ao algoritmo contínuo?. A resposta é afirmativa e será discutida na seção seguinte.

3.2 Abordagem Discreta

Como apresentado anteriormente, a derivada topológica tem-se mostrado uma ferramenta eficiente no processo de detecção de singularidades em uma imagem ao se considerar o problema estacionário. A seguir, é apresentado um novo método de restauração (originalmente introduzido em [113] e revisado em [111]) de imagens também baseado na derivada



Figura 3.4: Restauração com algoritmo de derivada topológica contínua D_T -Aniso da imagem de Lena (imagem poluída artificialmente utilizada no Capítulo 2) com $k_0 = 2$, fila superior: $\alpha = 0.10, 0.20$, fila inferior: $\alpha = 0.30$ e valor da derivada topológica para cada pixel).

topológica mas utilizando uma equação evolutiva como problema auxiliar. Novamente, fissuras são introduzidas na imagem para impedir a difusão em certos pontos e direções. Assim, a informação fornecida pela derivada é utilizada para determinar estes locais. Uma característica notável deste método é que, a condição de parada aparece de forma natural, eliminando a necessidade de introduzir heurísticas.

Como trata-se de uma abordagem puramente discreta a imagem é caracterizada por uma matriz de pixels, com uma intensidade associada a cada um deles, a partir de um esquema de diferenças mais precisamente, considera-se uma imagem bidimensional u dada por um conjunto de $M \times N$ pixels s . Em cada pixel s a intensidade da imagem u será denominada u^s . Logo, uma imagem qualquer pertence ao espaço \mathcal{U}^d

$$\mathcal{U}^d := \{u; u^s = \text{constante em } \omega^s, s = 1 \cdots M \times N\} , \quad (3.4)$$

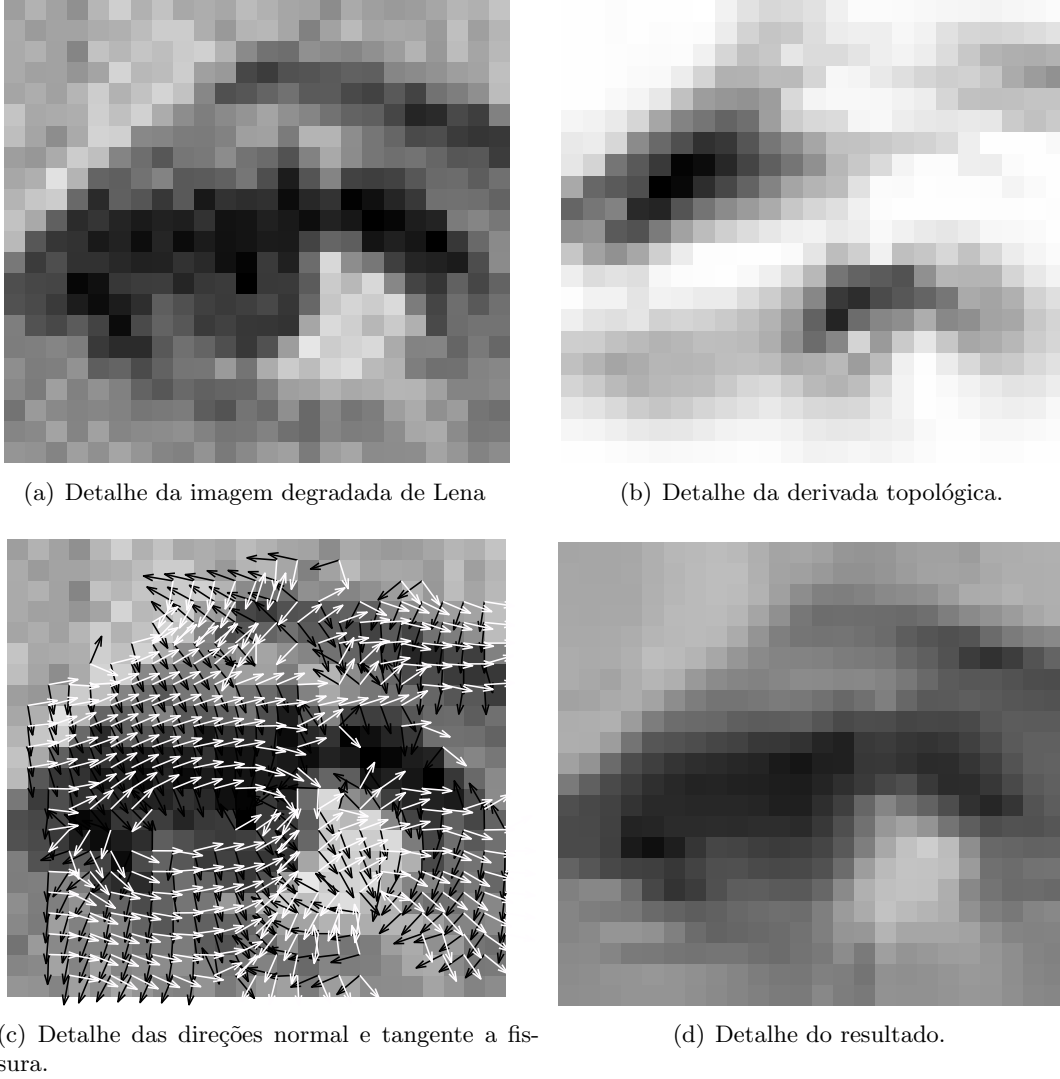


Figura 3.5: Detalhe na imagem de Lena da derivada topológica, das direções normal (preto) e tangente (branco) das fissuras inseridas e do resultado.

com $\bar{\Omega} = \cup_s \bar{\omega}^s$, sendo ω^s o domínio de s . O conjunto n^s de vizinhos do pixel s foi definido como os 4 pixels¹ (ver Fig. 3.6(a)) compartilhando um lado com s .

O funcional adotado para este esquema discreto é o seguinte

$$\mathcal{J}^d(u_t^s) = \sum_s \sum_{p \in n^s} k^{s,p} \hat{\Delta} u_t^{s,p} \cdot \hat{\Delta} u_t^{s,p},$$

o que pode ser interpretado como uma aproximação discreta da norma da energia do campo u . Nesta expressão o termo $k^{s,p}$ é o coeficiente de difusão do pixel s com o vizinho p , $n^s = \{n, s, e, w\}$ são os vizinhos do pixel s e $\hat{\Delta} u_t^{s,p}$ está definido como

$$\hat{\Delta} u_t^{s,p} = u_t^p - u_t^s. \quad (3.5)$$

¹i.e. *north*, *south*, *east* e *west* do pixel s .

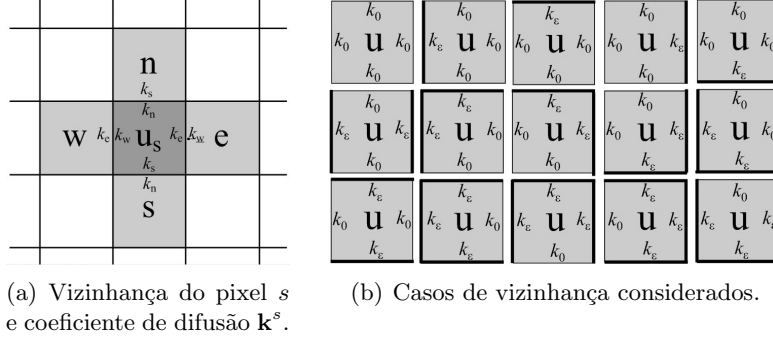


Figura 3.6: Vizinhança considerada.

Neste caso, u_t^s é explicitamente computada usando a seguinte expressão

$$u_t^s(\mathbf{k}^s) = u_{t-1}^s + \Delta t \sum_{p \in n^s} k^{s,p} \hat{\Delta} u_{t-1}^{s,p} \quad (3.6)$$

onde o índice $t \geq 1$ representa o número de iteração, sendo u_0^s a intensidade da imagem original no pixel s , $\mathbf{k}^s = \{k^{s,o}, k^{s,l}, k^{s,n}, k^{s,s}\}$ caracteriza o conjunto de coeficientes associados ao pixel s , Δt é o tamanho do passo no tempo artificial. Além do mais, e com a notação $u_t^s(\mathbf{k}^s)$ objetiva-se destacar que u_t^s é uma função explícita dos coeficientes de difusão associados ao pixel s . Finalmente, a Eq. (3.6) pode ser vista como uma aproximação de diferenças finitas da Eq. (2.25) (ver Apêndice A.1).

3.2.1 Cálculo da Derivada Topológica

No algoritmo contínuo, a introdução de um coeficiente de difusão k_ε nulo (ou muito pequeno $k_\varepsilon = \varepsilon$) é interpretado como a criação de um furo ou talho. No caso discreto acontece algo semelhante. O fato de $k^{s,p} = k_\varepsilon$ para um pixel s com o vizinho p é interpretado como a introdução de um talho na aresta que eles compartilham (não ocorre difusão ao longo desta aresta). Logo, objetiva-se encontrar a configuração para \mathbf{k}^s que mais preserve os detalhes removendo ruído ao calcular u_t^s . Como neste caso u_t^s é uma função explícita do conjunto \mathbf{k}^s , pode-se calcular a *variação total exata* do funcional custo para uma dada perturbação em $k^{s,p}$. Além do mais, denota-se \mathbf{k}_ε^s a configuração perturbada dos coeficientes de difusividade do pixel s . Logo, o valor da função custo quando esta perturbação é introduzida é dado por

$$\mathcal{J}^d(u_t^s(\mathbf{k}_\varepsilon^s)) = \mathcal{J}^d(u_t^s(\mathbf{k}^s)) + D_T(s, \mathbf{k}_\varepsilon^s), \quad (3.7)$$

onde $D_T(s, \mathbf{k}_\varepsilon^s)$ representa a *variação total* do funcional devido a uma perturbação nos coeficientes de difusividade no pixel s caracterizada pelo conjunto \mathbf{k}_ε^s . Como no caso contínuo, a introdução de uma perturbação no pixel s onde D_T é negativa irá produzir uma diminuição no valor da função custo \mathcal{J}^d . Logo, usando esta informação podem-se selecionar os melhores pixels para se introduzir perturbações.

Para isto, assume-se $k^{s,p} \in \{k_\epsilon, k_0\}$, logo o conjunto de todas as possíveis configurações para \mathbf{k}^s é definido como

$$\mathcal{C}(s) := \{\mathbf{k}^s = (k^{s,w}, k^{s,e}, k^{s,n}, k^{s,s}); k^{s,p} \in \{k_\epsilon, k_0\}, p = \{w, e, n, s\}\}.$$

Observa-se que existem 16 combinações possíveis para \mathbf{k}^s (as possibilidades são $k^{s,p} = k_\epsilon$ ou $k^{s,p} = k_0$, para cada vizinho, logo $2^4 = 16$ casos são possíveis). O caso $k^{s,w} = k^{s,e} = k^{s,n} = k^{s,s} = k_\epsilon$ não é levado em consideração por não produzir uma variação na função custo (o que não iria modificar a imagem original). As 15 combinações restantes são

- não aplicar difusão com um vizinho,
- não aplicar difusão com dois vizinhos que compartilham um vértice,
- não aplicar difusão com 3 vizinhos,
- aplicar difusão na direção x,
- aplicar difusão na direção y,
- aplicar difusão em todas as direções (difusão isotrópica).

O último caso corresponde a difusão isotrópica e é definido como $\mathbf{k}_{iso}^s = \{k_0, k_0, k_0, k_0\}$.

Para calcular o valor da D_T para um determinado pixel, é necessário introduzir uma perturbação. Isto é feito ao trocar, para um pixel determinado s , o conjunto \mathbf{k}^s por $\mathbf{k}_\epsilon^s \in \mathcal{C}_\sigma$. Logo, a função custo $\mathcal{J}_\epsilon^d(\epsilon u_t^s)$ assume o valor

$$\mathcal{J}_\epsilon^d(\epsilon u_t^s) = \mathcal{J}^d(u_t^s) - \sum_{p \in n^s} k_\epsilon^{s,p} \hat{\Delta} u_t^{s,p} \cdot \hat{\Delta} u_t^{s,p} + \sum_{p \in n^s} k_\epsilon^{s,p} \hat{\Delta} \epsilon u_t^{s,p} \cdot \hat{\Delta} \epsilon u_t^{s,p}, \quad (3.8)$$

para $u_t^s = u_t^s(\mathbf{k}^s)$ e $\epsilon u_t^s = u_t^s(\mathbf{k}_\epsilon^s)$ calculado usando a Eq. (3.6) e $\hat{\Delta} \epsilon u_t^{s,p} = u_t^p - \epsilon u_t^s$, respectivamente.

Para as Eqs. (3.7) e (3.8) a variação total da função custo \mathcal{J}^d devido a uma perturbação \mathbf{k}_ϵ^s pode ser escrita como

$$D_T(s, \mathbf{k}_\epsilon^s) = \sum_{p \in n^s} k_\epsilon^{s,p} \hat{\Delta} \epsilon u_t^{s,p} \cdot \hat{\Delta} \epsilon u_t^{s,p} - \sum_{p \in n^s} k_\epsilon^{s,p} \hat{\Delta} u_t^{s,p} \cdot \hat{\Delta} u_t^{s,p}. \quad (3.9)$$

Como na alternativa contínua, será considerada a perturbação \mathbf{k}_ϵ^s que minimize o valor da $D_T(s, \mathbf{k}_\epsilon^s)$. Usando esta informação é proposto o seguinte algoritmo discreto de restauração de imagens baseado na derivada topológica (Algoritmo 3).

O conjunto \mathcal{M}_α é definido como em (3.3). Como no caso contínuo, no algoritmo anteriormente apresentado o parâmetro α permite controlar os valores da derivada topológica que irão produzir mudanças em \mathbf{k}^s . Em todos os casos foi utilizado $\Delta t = \frac{1}{4}$ e $k_0 = 1$, os valores máximos que garantem a condição Courant-Friedrichs-Levy (estabilidade, ver Apêndice A.1.2) do método de resolução iterativo. Na Fig. 3.7 são apresentados alguns

Algoritmo 3 Restauração de imagens baseada em uma versão discreta da derivada topológica

Entrada: Uma imagem 2D $u_0 \in \mathcal{U}$, o coeficiente de difusividade k_0 e o parâmetro α .

Saída: A imagem restaurada $\bar{u}^s \in \mathcal{U}$.

```
fazer t=1; Parar = FALSE;  $\mathbf{k}^s = \mathbf{k}_{iso}^s, s = 1..M \times N$ 
while Parar = FALSE do
  for cada pixel  $s$  do
    for cada  $\mathbf{k}_\epsilon^s \in \mathcal{C}_\sigma$  do
      calcular  $D_T(s, \mathbf{k}_\epsilon^s)$  seguindo a Eq. (3.9)
    end for
  end for
  for cada pixel  $s \in \mathcal{M}_\alpha$  do
    fazer  $D_T(s) = \min_{\epsilon^*} \{D_T(s, \mathbf{k}_\epsilon^s), \mathbf{k}_\epsilon^s \in \mathcal{C}_\sigma\}$ 
    fazer  $\mathbf{k}^s = \mathbf{k}_{\epsilon^*}^s$  o coeficiente de difusividade associado a  $D_T(s)$ 
  end for
  calcular  $u_t^s(\mathbf{k}^s)$  usando Eq. (3.6).
  if  $|\mathcal{J}_\epsilon^d(u_t^s) - \mathcal{J}_\epsilon^d(u_{t-1}^s)| > tol$  then
     $t = t + 1$ 
  else
     $\bar{u}^s = u_t^s, s = 1, \dots, M \times N$ , Parar = TRUE
  end if
end while
```

resultados obtidos com esta técnica. As três imagens apresentadas (correspondentes a resultados para valores de $\alpha = 0.05, 0.15$ e 0.25 respectivamente) apresentam uma melhora considerável no SNR (indo de 26.91 na imagem degradada para 28.49, 29.61 e 29.53 nas imagens processadas respectivamente). Nas Figuras 3.9 e 3.8 são apresentados os detalhes das bordas que foram introduzidas na imagem em diferentes iterações. Pode-se observar que depois de um certo número de iterações, a imagem tem bordas em praticamente todas as arestas. Assim, a variação da função custo é praticamente nula entre duas iterações consecutivas, o que irá deter o algoritmo. A quantidade de bordas que são adicionadas na imagem em cada iteração é controlada pelo parâmetro α .

3.3 Comentários

Como apresentado, os métodos de restauração baseados na derivada topológica utilizam informação da sensibilidade de uma função custo associada a um determinado problema à mudança na topologia do domínio no qual encontra-se definido. Esta informação é utilizada para achar a topologia ótima do domínio que, na presença de difusão, elimina maior quantidade de ruído preservando as bordas da imagem.

No caso contínuo, observa-se que esta técnica elimina boa quantidade de ruído mas apresenta dificuldade para remover ruído em regiões de gradientes elevados. No caso do algoritmo discreto, observa-se que o ruído é removido em toda a imagem inclusive nas bordas. Além disso, esse algoritmo consegue melhorar consideravelmente a nitidez das bordas realçando as características presentes na imagem e impedindo a perda de informação.



Figura 3.7: Restauração com o algoritmo de derivada topológica discreta da imagem da Lena (imagem poluída com o mesmo ruído que no Capítulo 2). Imagens correspondem a valores de $k_0 = 1$ e $\alpha = 0.05, 0.15$ e 0.25 respectivamente.

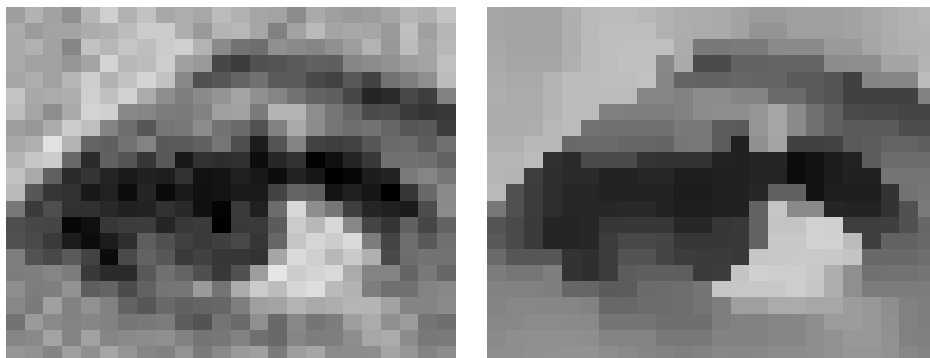


Figura 3.8: Detalhe na imagem de Lena antes e depois do processamento ($k_0 = 1$ e $\alpha = 0.20$).

Respeito do desempenho computacional, pode-se mencionar que o algoritmo discreto efetivamente melhora a performance do contínuo. Para imagens acima de 32×32 o

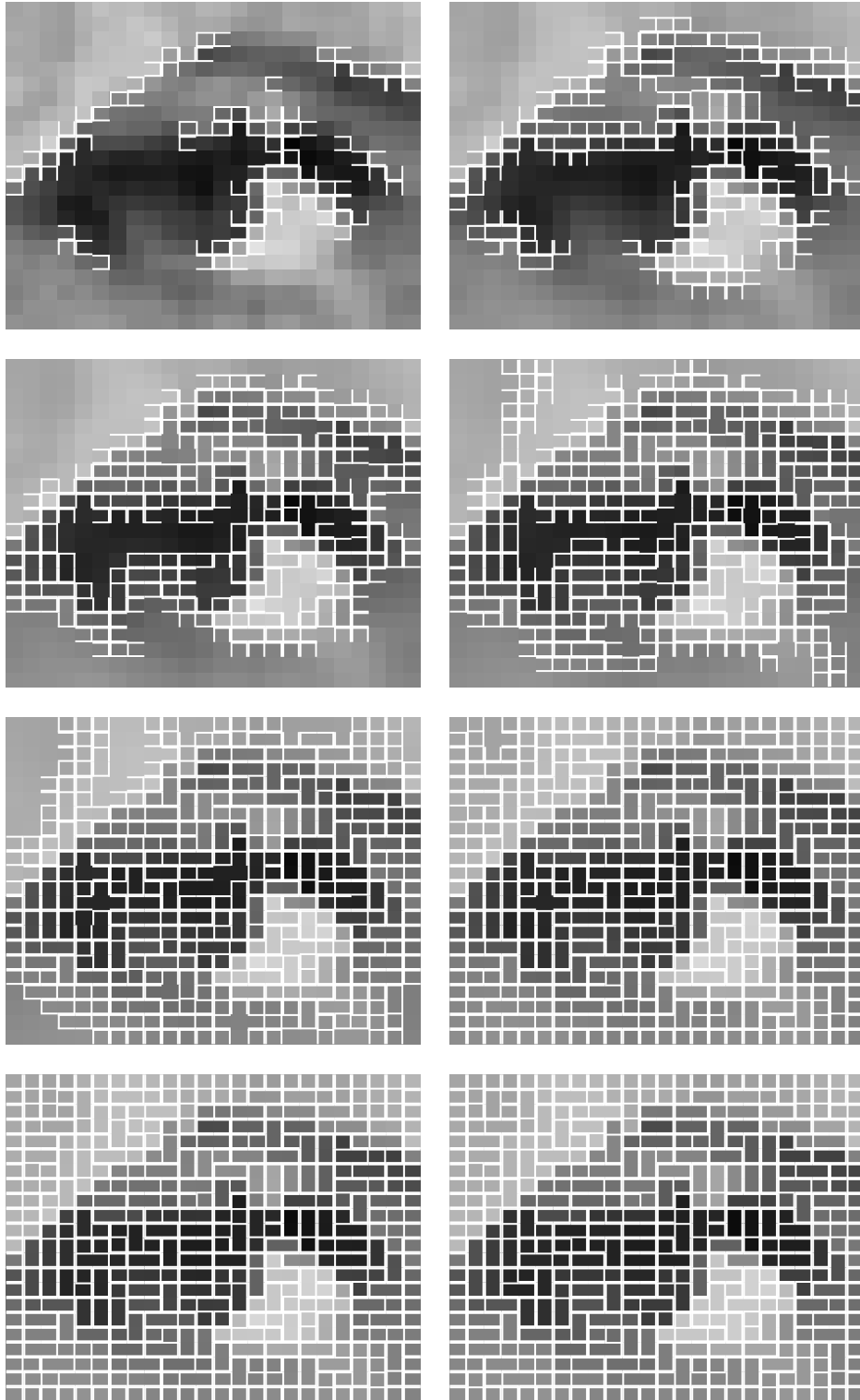


Figura 3.9: Detalhe (na imagem de Lena) da configuração das bordas introduzidas na imagem durante o processamento ($k_0 = 1$ e $\alpha = 0.20$).

algoritmo discreto tem menor custo computacional como pode ser observado na Figura 3.10. Os testes de performance foram feitos em um computador Pentium 4 de 3 GHz com 512 Mb de memória e os algoritmos implementados em Matlab 6.5 [190].

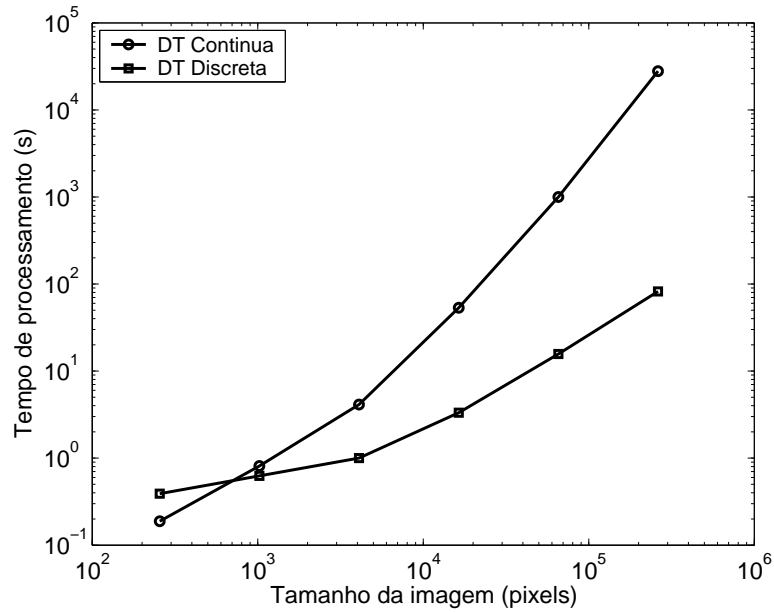


Figura 3.10: Tamanho da imagem (pixels) vs. tempo de processamento (s).

Uma outra vantagem da abordagem discreta, é que mesmo sendo um algoritmo iterativo, ele tem um critério de parada próprio e a quantidade de iterações não é um parâmetro requerido pelo métodos ou que afete a sua qualidade.

Capítulo 4

Análises e Comparações das Técnicas Propostas e Existentes

Este capítulo tem por objetivo comparar o desempenho dos métodos clássicos discutidos no Capítulo 2 e dos métodos baseados na derivada topológica apresentados no Capítulo 3. Estes métodos são utilizados para restaurar imagens artificialmente degradadas com ruído de diferentes tipos e intensidades. Em particular, sejam \bar{u} a imagem real original (não poluída) e u_0 a imagem observada (degradada), então os seguintes modelos de ruído foram utilizados

- Ruído uniforme: com média nula e probabilidade uniforme. Assim, um ruído r (em níveis de intensidade, i.e., $r \in [0, 255]$) é dado por

$$u_0^s = \bar{u}^s + 2 * r * (rand - 0.5), \quad (4.1)$$

sendo $rand$ um número aleatório entre zero e um, com distribuição uniforme. Logo, o ruído encontra-se no intervalo $[-r, r]$.

- Ruído Gaussiano branco: com média μ (sempre nula) e desvio padrão $\bar{\sigma}$ (em níveis de intensidade). Este ruído de caráter aditivo é denotado por $\rho(\mu, \bar{\sigma})$ é uma variável aleatória com distribuição de probabilidade Gaussiana. A imagem é poluída usando o modelo

$$u_0^s = \bar{u}^s + \rho(\mu, \bar{\sigma}). \quad (4.2)$$

- Ruído sal e pimenta: esse ruído consiste em mudar o valor de alguns pixels na imagem para branco ou preto. No caso, p pixels (sendo p um percentual) são alterados para 0 ou 255 com igual probabilidade.
- Ruído multiplicativo speckle: a imagem é poluída com ruído multiplicativo, distribuído de maneira uniforme com média $\mu = 0$ e desvio padrão $\bar{\sigma}$ (em níveis de intensidade), denotado por $n(\mu, \bar{\sigma})$, dado pelo modelo

$$u_0^s = \bar{u}^s + n(\mu, \bar{\sigma}) * \bar{u}^s. \quad (4.3)$$

Na Fig. 4.1 e 4.2 são apresentadas as imagens utilizadas (originais e poluídas com os tipos de ruídos descritos anteriormente), de tamanho 256×256 pixels e 256 níveis de intensidade.



(a) Imagens originais (superfície da lua, imagem de Lena e imagem do barco)



(b) Imagens poluídas com ruído uniforme (SNR aproximadamente 26.80dB)



(c) Imagens poluídas com ruído Gaussiano (SNR aproximadamente 18.80dB)

Figura 4.1: Imagens artificialmente poluídas utilizadas nos testes. As filas (de cima para baixo) correspondem a imagem original, ruído uniforme ($r = 20$) e ruído Gaussiano branco ($\bar{\sigma} = 30$) (imagens obtidas de [148]).

4.1 Teste 1 - Remoção de ruído uniforme

Nesta seção, os métodos descritos nos Capítulos 2 e 3 foram utilizados para restaurar imagens poluídas com ruído uniforme. Como mencionado anteriormente, estes métodos podem ser classificados em dois grupos, evolutivos e estacionários, ou seja

- Métodos evolutivos:
 - Perona & Malik (Seção 2.4.2, com $c(x)$ dado por Eq. (2.35)),
 - Black. *et al.* (Seção 2.4.2, com $c(x)$ dado por Eq. (2.36)),



(a) Imagens poluídas com ruído tipo sal e pimenta (SNR aproximadamente 13.70dB)



(b) Imagens poluídas com ruído tipo speckle (SNR aproximadamente 18.50dB)

Figura 4.2: (Cont.) Imagens artificialmente poluídas utilizadas nos testes. Correspondentes (de cima para baixo) a ruído sal e pimenta ($p = 15$) e ruído speckle ($\bar{\sigma} = 60$).

- Suavizado seletivo (Alvarez, Morel e Lions, Seção 2.4.3) e
- R_{D_T} Discreto.
- Métodos estacionários:
 - Minimização Semi-Quadrática (Kornprobst & Aubert, descrito na Seção 2.3),
 - R_{D_T} Contínua (Belaïd *et al.*).

As imagens utilizadas nesta primeira parte foram degradadas com ruído uniforme ($r = 20$), sendo o SNR correspondente a aproximadamente 27 dB.

O desempenho destes métodos pode ser ajustado utilizando parâmetros específicos a cada um deles. Todos os métodos clássicos dependem de um parâmetro σ com significado equivalente. Este parâmetro serve para determinar os valores do gradiente que indicam a presença de bordas. Isto é, quando o gradiente é superior a σ existem na vizinhança pixels de outras regiões (*outliers*). Este parâmetro é utilizado para controlar a difusão nas bordas de maneira a preservar as características de interesse. Para os métodos clássicos, σ foi ajustado utilizando a técnica proposta por Black *et al.*[27], (Eq. (2.37)). Isto permite estimar o valor de σ em função do gradiente da imagem sendo tratada. No caso dos métodos evolutivos, existe outro parâmetro importante e difícil de ser ajustado: o número de iterações.

Método	Parâmetro	Valor
Perona & Malik	σ iterações	Eq. (2.37) 10 - 20 - 30
Black. <i>et al.</i>	σ iterações	Eq. (2.37) 10 - 20 - 30
Suavizado seletivo (A-M-L)	σ iterações	Eq. (2.37) 10 - 20 - 30
Minimização Semi-Quadrática (K-A)	σ λ	Eq. (2.37) 10
R_{D_T} Discreto	α	0.18
R_{D_T} Contínua	α	0.80

Tabela 4.1: Valores utilizados dos parâmetros nos diferentes métodos.

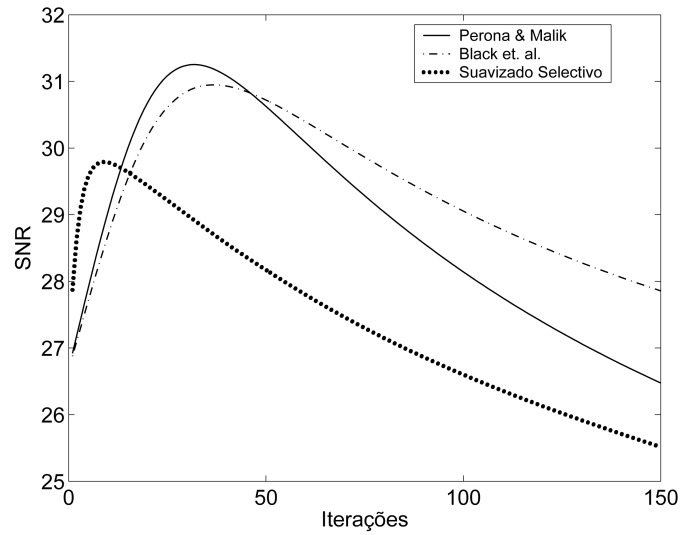
Para estimar a quantidade de iterações necessárias para obter resultados de boa qualidade pelos métodos evolutivos, o SNR foi estudado para 150 iterações. Dependendo do método, o máximo SNR é alcançado entre a décima e a trigésima iteração aproximadamente (Fig. 4.3(a)). Pode-se observar que, se o método não for detido a tempo, este pode produzir uma degradação significativa na imagem. Assim, no caso dos métodos evolutivos, as imagens degradadas foram processadas 10, 20 e 30 iterações em todos os casos. Os resultados apresentados em todos os casos corresponde ao que apresentou o melhor SNR dentre estes três resultados.

Para o método de minimização semi-quadrática, a mesma análise foi feita com o parâmetro λ . Como apresentado na Fig. 4.3(b), o melhor valor para este parâmetro é alcançado em $\lambda \approx 10$.

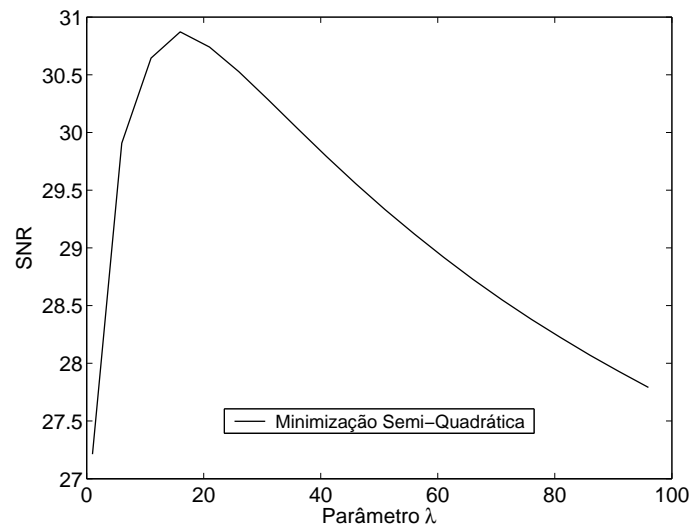
A escolha de parâmetros nas técnicas D_T é bem diferente. Tanto o algoritmo R_{D_T} -Discreto como o R_{D_T} -Contínuo dependem do parâmetro α (um valor real entre 0 e 1), que determina a quantidade de “fraturas” a serem introduzidas na imagem. Este parâmetro determina a quantidade (em %) de pixels nos quais serão introduzidas fissuras para deter (ou diminuir) a difusão. Por este motivo não é possível estabelecer uma relação entre os métodos R_{D_T} e os métodos clássicos, o que requer o estudo de α de maneira separada. O parâmetro α foi analisado como nos casos anteriores, selecionando assim o valor que produz o melhor SNR (no caso de R_{D_T} -Discreto $\alpha = 0.18$ e para R_{D_T} -Contínua $\alpha = 0.80$, Fig. 4.3(c)). No entanto, estes métodos apresentam uma ampla faixa de valores para os quais os resultados têm uma qualidade boa, o que indica uma pequena dependência em relação a esse parâmetro. Os valores dos diferentes parâmetros são resumidos na Tabela 4.1. Uma vez determinado este valor, o mesmo foi mantido fixo em todos os testes.

Esta análise de parâmetros foi feita utilizando a imagem do barco em todos os casos. Como será apresentado mais adiante, testes semelhantes foram realizados para diferentes tipos e intensidades de ruído. Estas análises foram feitas para observar o comportamento dos métodos e não para estimar valores ótimos dos parâmetros. Por este motivo, os resultados utilizados em **todas** as comparações correspondem aos parâmetros mencionados na Tabela 4.1, já que geralmente em situações reais achar valores “ótimos” não é possível.

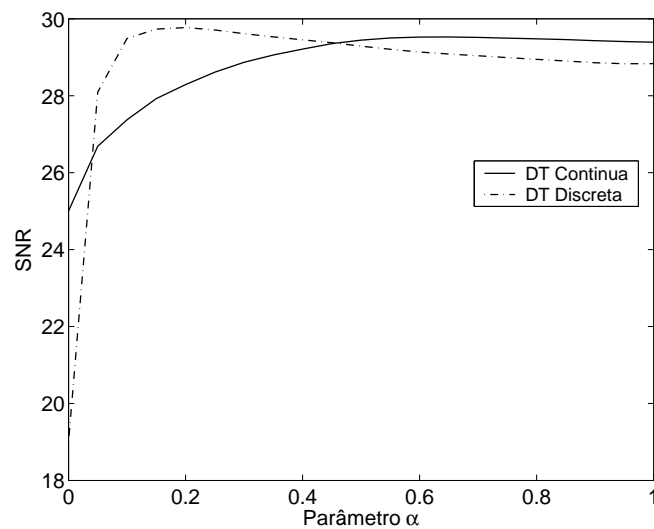
Na Tabela 4.2 são apresentados o SNR e estatísticas do ruído correspondentes aos diferentes métodos para $r = 20$. Observa-se que os métodos R_{D_T} não apresentam melhoras



(a) SNR vs. número de iteração - Métodos evolutivos.



(b) SNR vs. parâmetro λ - Minimização Semi-Quadrática.



(c) SNR vs. parâmetro α - Métodos D_T .

Figura 4.3: Análise de parâmetros (imagem do barco). Ruído uniforme $r = 20$, SNR inicial de aproximadamente 27 dB.

	PSNR (dB)	SNR (dB)	$\mu(e)$	$\bar{\sigma}(e)$
Imagem poluída (lua)	26.86	26.66	0.4776	11.560
Evolutivo (Perona e Malik) (its = 30)	31.69	31.48	0.9776	6.5657
Evolutivo (Black <i>et al.</i>) (its = 30)	29.82	29.62	0.9799	8.1697
Evolutivo (Suavizado seletivo) (its = 20)	30.68	30.48	0.9705	7.3898
Minimização Semi-Quadrática	31.07	30.87	0.9772	7.0586
D_T Discreto	30.50	30.29	1.0058	7.5497
D_T Contínua	30.84	30.63	0.9767	7.2539
Imagem poluída (Lena)	26.92	26.88	0.5213	11.487
Evolutivo (Perona e Malik) (its = 30)	31.67	31.63	1.0217	6.5772
Evolutivo (Black <i>et al.</i>) (its = 30)	31.20	31.17	1.0218	6.9467
Evolutivo (Suavizado seletivo) (its = 10)	30.58	30.55	1.0565	7.4646
Minimização Semi-Quadrática	31.08	31.04	1.0182	7.0517
D_T Discreto	29.60	29.57	1.0765	8.3744
D_T Contínua	28.96	28.93	1.0195	9.0272
Imagem poluída (barco)	26.88	26.67	0.6005	11.535
Evolutivo (Perona e Malik) (its = 30)	31.36	31.15	1.1013	6.8077
Evolutivo (Black <i>et al.</i>) (its = 30)	30.98	30.77	1.1003	7.1221
Evolutivo (Suavizado seletivo) (its = 10)	29.98	29.77	1.0106	8.0220
Minimização Semi-Quadrática	30.75	30.54	1.0968	7.3185
D_T Discreto	29.93	29.73	1.1463	8.0455
D_T Contínua	29.63	29.42	1.0944	8.3446

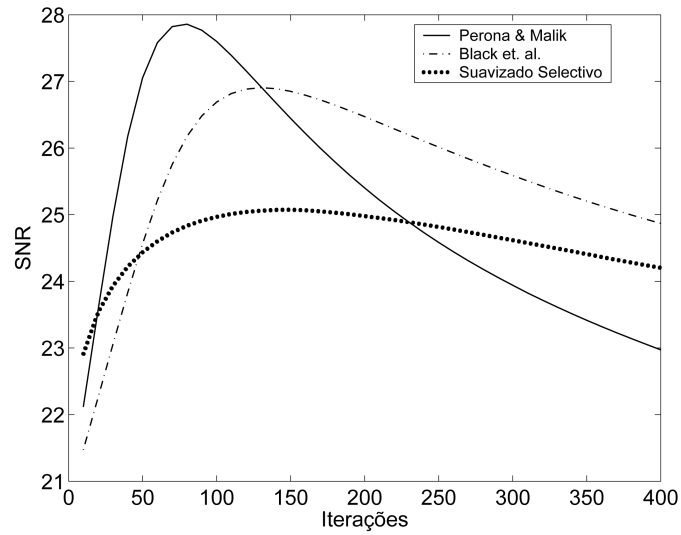
Tabela 4.2: Comparação para ruído uniforme. $r = 20$, SNR inicial de aproximadamente 27 dB.

em relação aos métodos existentes. Em particular, dependendo da imagem processada a qualidade do resultado varia sutilmente. Neste caso, os resultados obtidos com o R_{D_T} -Discreto são os que mais se aproximam dos obtidos com os métodos clássicos.

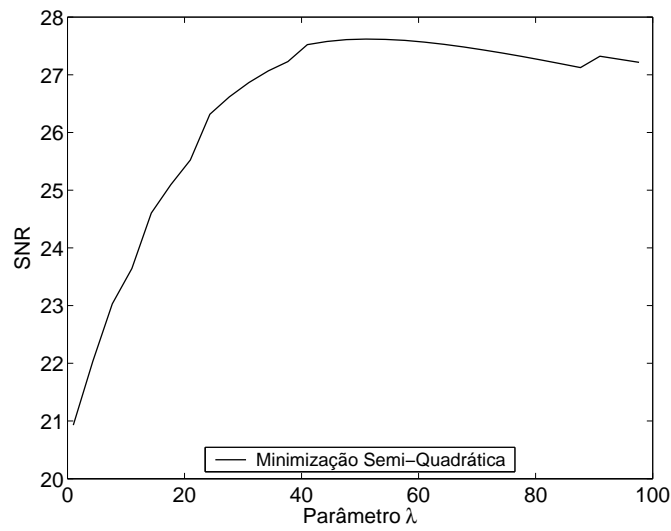
Em uma segunda análise, é considerado um ruído uniforme de maior intensidade ($r = 40$). O SNR das imagens degradadas é de aproximadamente 21 dB. As imagens poluídas com ruído uniforme foram processadas utilizando os mesmos métodos e com os mesmos parâmetros que no caso anterior.

Pode-se observar neste caso que os métodos D_T se mostram comparáveis com os métodos clássicos. No caso dos algoritmos estacionários, em particular R_{D_T} - Contínuo e Minimização Semi-Quadrática, observa-se que o primeiro se comporta melhor, conseguindo remover maior quantidade de ruído. Já no caso do R_{D_T} - Discreto, nota-se que a qualidade dos resultados depende da imagem e do ruído nela presente, conseguindo valores de SNR semelhantes aos obtidos pelos outros métodos, mas com a vantagem de não precisar o número de iterações. Esta diferença é importante, já que iterar muito degrada a imagem e iterar pouco impede que suficiente quantidade de ruído seja removida.

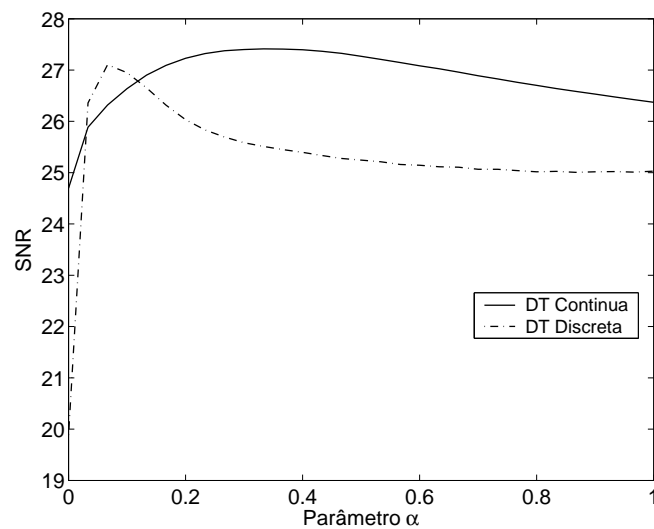
Ao repetir a análise de sensibilidade a parâmetros feita anteriormente para um ruído de maior intensidade, observa-se que nos métodos clássicos os valores (quantidade de iterações e λ) ótimos mudam consideravelmente (Fig. 4.4(a) e 4.4(b)). Esta característica dificulta a tarefa de estimar esses parâmetros (já que dependem da intensidade do ruído).



(a) SNR vs. número de iteração - Métodos evolutivos.

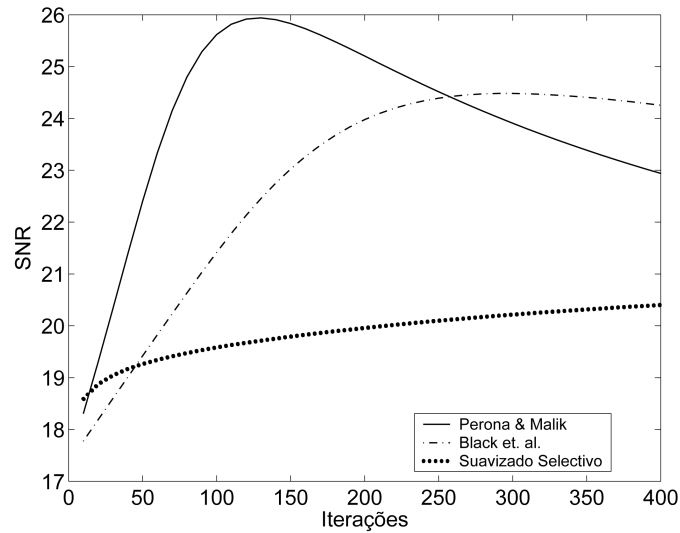


(b) SNR vs. parâmetro λ - Minimização Semi-Quadrática.

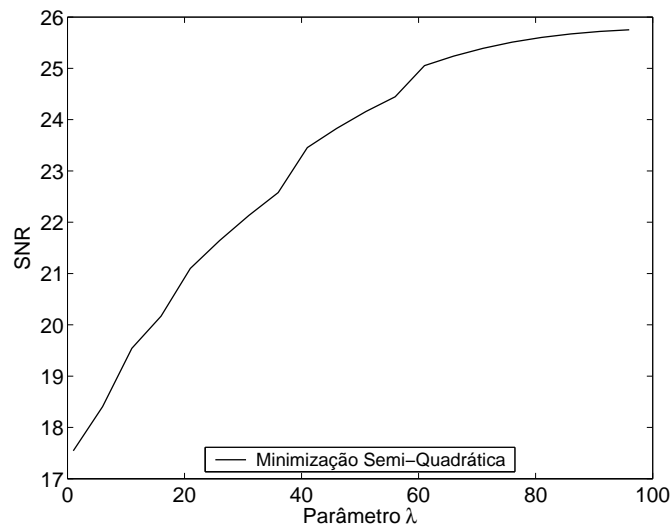


(c) SNR vs. parâmetro α - Métodos D_T .

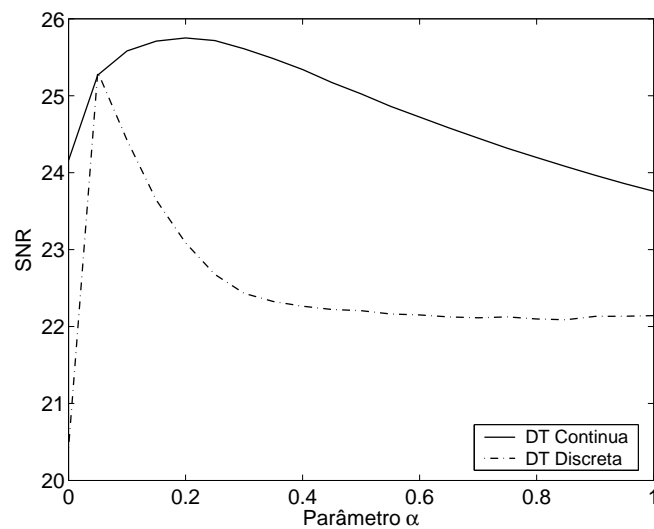
Figura 4.4: Análise de parâmetros (imagem do barco). Ruído uniforme $r = 40$, SNR inicial de aproximadamente 21 dB.



(a) SNR vs. número de iteração - Métodos evolutivos.



(b) SNR vs. parâmetro λ - Minimização Semi-Quadrática.



(c) SNR vs. parâmetro α - Métodos D_T .

Figura 4.5: Análise de parâmetros (imagem do barco). Ruído uniforme $r = 60$, SNR inicial de aproximadamente 17.50 dB.

	PSNR (dB)	SNR (dB)	$\mu(e)$	$\bar{\sigma}(e)$
Imagem poluída (lua)	20.88	20.67	0.552	23.039
Evolutivo (Perona e Malik) (its = 30)	25.00	24.79	1.0501	14.301
Evolutivo (Black <i>et al.</i>) (its = 30)	22.10	21.89	1.0516	19.993
Evolutivo (Suavizado seletivo) (its = 30)	24.57	24.37	1.0836	15.022
Minimização Semi-Quadrática	24.16	23.95	1.0512	15.765
D_T Discreto	26.62	26.41	1.0431	11.852
D_T Contínua	27.23	27.03	1.0485	11.039
Imagem poluída (Lena)	20.96	20.92	0.3065	22.833
Evolutivo (Perona e Malik) (its = 30)	25.21	25.17	0.8061	13.976
Evolutivo (Black <i>et al.</i>) (its = 30)	23.13	23.10	0.8059	17.759
Evolutivo (Suavizado seletivo) (its = 30)	24.39	24.35	0.8712	15.360
Minimização Semi-Quadrática	23.78	23.75	0.8074	16.475
D_T Discreto	26.33	26.30	0.8278	12.270
D_T Contínua	26.46	26.43	0.8122	12.092
Imagem poluída (barco)	20.92	20.71	0.3859	22.941
Evolutivo (Perona e Malik) (its = 30)	25.17	24.96	0.8867	14.038
Evolutivo (Black <i>et al.</i>) (its = 30)	23.25	23.04	0.8848	17.515
Evolutivo (Suavizado seletivo) (its = 30)	24.16	23.95	0.9665	15.765
Minimização Semi-Quadrática	23.66	23.46	0.8864	16.698
D_T Discreto	26.39	26.19	0.8867	12.184
D_T Contínua	26.89	26.68	0.8820	11.505

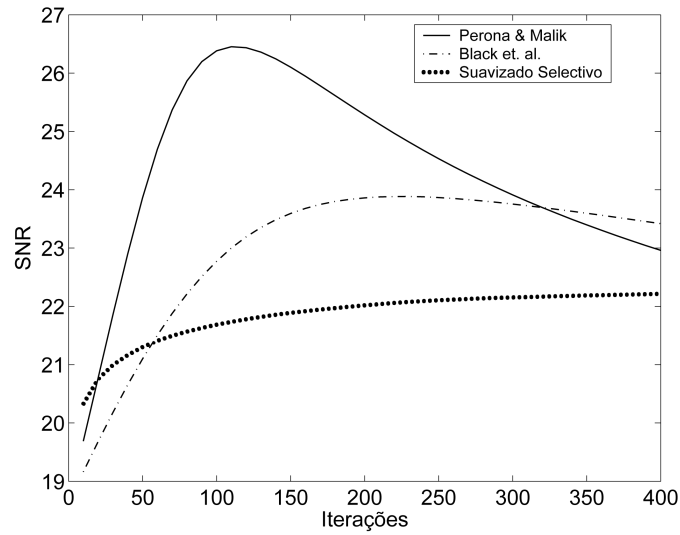
Tabela 4.3: Comparação para ruído uniforme. $r = 40$, SNR inicial de aproximadamente 21 dB.

O último teste realizado para ruído uniforme corresponde a $r = 60$, o que produz um SNR de aproximadamente 17.50 dB. Novamente, os parâmetros utilizados são os mesmos que nas situações anteriores. Desta vez, mesmo quando todos os métodos apresentam melhoras nas imagens após o processamento, os resultados dos métodos D_T se mostram superiores. Na Figura 4.5 são apresentadas as curvas correspondentes a análise de parâmetros feita para este nível de ruído. Observa-se que, na medida que o ruído aumenta, os parâmetros ótimos dos métodos clássicos variam consideravelmente, enquanto nos métodos D_T a situação é diferente. No caso de R_{D_T} -Discreto, os valores ótimos do parâmetro α oscilam entre 0.10 e 0.20, e o R_{D_T} -Contínua apresenta valores comparáveis para α abaixo de 0.4.

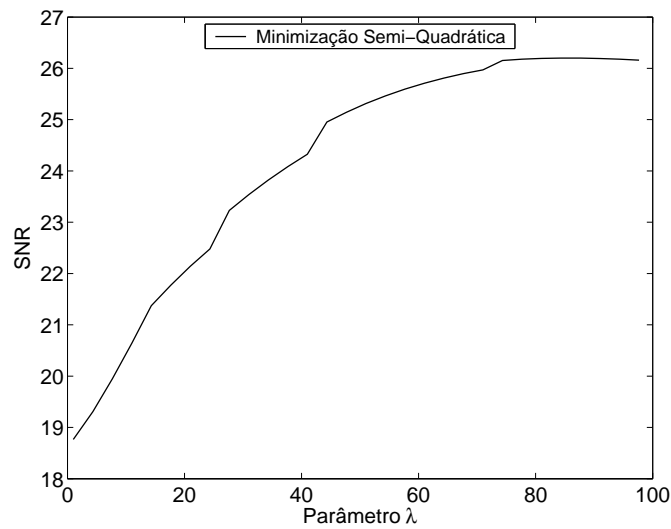
4.2 Teste 2 - Remoção de ruído Gaussiano

Nesta seção, o desempenho dos métodos é estudado para remover ruído Gaussiano branco. As imagens apresentadas na Fig. 4.1 foram poluídas com $\bar{\sigma} = 30$, o que produz um SNR ≈ 18.70 dB. Os parâmetros de todos os métodos foram mantidos iguais que nos casos anteriores (Tabela 4.1). Na Tabela 4.5 são apresentados os resultados para os diferentes métodos.

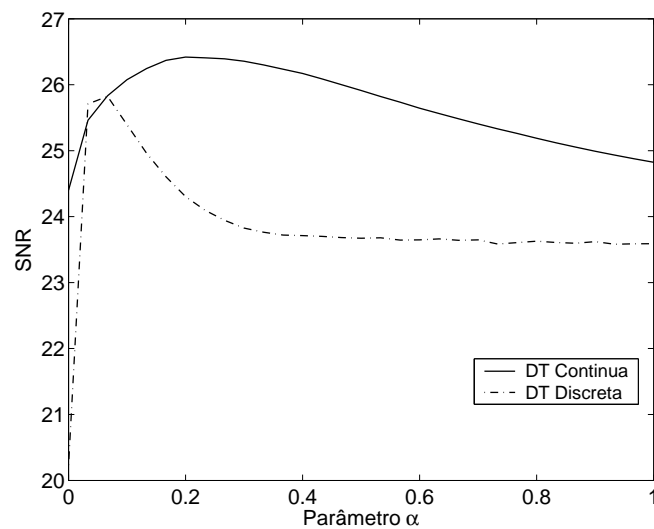
Observa-se para este tipo de ruído que os métodos baseados na derivada topológica apresentam resultados de melhor qualidade. Cabe ressaltar que nenhum parâmetro teve



(a) SNR vs. número de iteração - Métodos evolutivos.



(b) SNR vs. parâmetro λ - Minimização Semi-Quadrática.



(c) SNR vs. parâmetro α - Métodos D_T .

Figura 4.6: Análise de parâmetros (imagem do barco). Ruído Gaussiano $\bar{\sigma} = 30$, SNR inicial de aproximadamente 18.80 dB.

	PSNR (dB)	SNR (dB)	$\mu(e)$	$\bar{\sigma}(e)$
Imagem poluída (lua)	17.35	17.14	0.3788	34.606
Evolutivo (Perona e Malik) (its = 30)	20.08	19.87	0.8785	25.263
Evolutivo (Black <i>et al.</i>) (its = 30)	18.08	17.87	0.8798	31.792
Evolutivo (Suavizado seletivo) (its = 30)	19.02	18.82	0.8638	28.522
Minimização Semi-Quadrática	19.41	19.20	0.8776	27.281
D_T Discreto	23.74	23.54	0.8095	16.552
D_T Contínua	24.36	24.16	0.8746	15.406
Imagem poluída (Lena)	17.56	17.53	0.1088	33.755
Evolutivo (Perona e Malik) (its = 30)	20.48	20.45	0.6086	24.116
Evolutivo (Black <i>et al.</i>) (its = 30)	18.66	18.63	0.6081	29.741
Evolutivo (Suavizado seletivo) (its = 30)	19.29	19.26	0.8081	27.657
Minimização Semi-Quadrática	19.64	19.60	0.6065	26.576
D_T Discreto	23.55	23.52	0.5830	16.927
D_T Contínua	24.09	24.06	0.6084	15.906
Imagem poluída (barco)	17.57	17.37	0.0523	33.719
Evolutivo (Perona e Malik) (its = 30)	20.54	20.33	0.5546	23.964
Evolutivo (Black <i>et al.</i>) (its = 30)	18.81	18.60	0.5537	29.255
Evolutivo (Suavizado seletivo) (its = 30)	19.29	19.09	0.5114	27.659
Minimização Semi-Quadrática	19.61	19.41	0.5511	26.659
D_T Discreto	23.47	23.27	0.5493	17.083
D_T Contínua	24.40	24.19	0.5476	15.359

Tabela 4.4: Comparação para ruído uniforme. $r = 60$, SNR inicial de aproximadamente 17.50 dB.

que ser reajustado para obter estes resultados. Neste caso, os métodos clássicos apresentam qualidade inferior, provavelmente devido a que os parâmetros utilizados já não são ótimos. Uma das vantagens dos métodos D_T de restauração consiste em que o ajuste dos parâmetros é mais fácil devido, principalmente, a que o parâmetro mais importante é limitado ($\alpha \in [0, 1]$) e a qualidade do resultado não é muito afetada por mudanças neste valor. Estas características tornam os métodos D_T mais robustos.

Na Fig. 4.6 são apresentadas as curvas correspondentes à análise de parâmetros para ruído Gaussiano. Observa-se que para este tipo de ruído, o valor ótimo do parâmetro α encontra-se bem próximo do valor ótimo para ruído uniforme.

Este tipo de ruído é comum em imagens de tomografia computadorizada. Na Seção 4.5 estes métodos são utilizados para remover ruído de imagens médicas que apresentam ruído Gaussiano branco [76, 203].

4.3 Teste 3 - Remoção de ruído sal e pimenta

Nesta seção o desempenho dos métodos é estudado na remoção de ruído tipo sal e pimenta. Os parâmetros de todos os métodos foram mantidos iguais que nos testes anteriores (Tabela 4.1).

Na Tabela 4.6 e Figura 4.8 são apresentados os resultados para as imagens processadas utilizando os diferentes métodos. Para este tipo de ruído observa-se que os métodos

	PSNR (dB)	SNR (dB)	$\mu(e)$	$\bar{\sigma}(e)$
Imagem poluída (lua)	18.60	18.40	-0.1596	29.948
Evolutivo (Perona e Malik) (its = 30)	21.58	21.37	0.3408	21.259
Evolutivo (Black <i>et al.</i>) (its = 30)	19.55	19.34	0.3414	26.856
Evolutivo (Suavizado seletivo) (its = 30)	21.10	20.89	0.3133	22.464
Minimização Semi-Quadrática	20.49	20.28	0.3413	24.094
D_T Discreto	24.88	24.67	0.2852	14.538
D_T Contínua	25.46	25.25	0.3479	13.602
Imagem poluída (Lena)	18.81	18.77	-0.4546	29.244
Evolutivo (Perona e Malik) (its = 30)	21.96	21.93	0.0451	20.341
Evolutivo (Black <i>et al.</i>) (its = 30)	20.23	20.19	0.0467	24.842
Evolutivo (Suavizado seletivo) (its = 30)	21.26	21.22	0.2458	22.060
Minimização Semi-Quadrática	20.72	20.69	0.0474	23.459
D_T Discreto	24.65	24.62	0.0960	14.921
D_T Contínua	25.07	25.04	0.0609	14.217
Imagem poluída (barco)	18.84	18.64	-0.2764	29.128
Evolutivo (Perona e Malik) (its = 30)	22.06	21.86	0.2234	20.110
Evolutivo (Black <i>et al.</i>) (its = 30)	20.39	20.19	0.2231	24.372
Evolutivo (Suavizado seletivo) (its = 30)	21.24	21.03	0.3578	22.114
Minimização Semi-Quadrática	20.72	20.51	0.2178	23.468
D_T Discreto	24.67	24.47	0.1652	14.889
D_T Contínua	25.40	25.19	0.2108	13.698

Tabela 4.5: Comparação para ruído Gaussiano. $\bar{\sigma} = 30$, SNR inicial de aproximadamente 18.80 dB.

D_T conseguem remover uma quantidade maior de ruído que os métodos clássicos. Na Fig. 4.9 é feita a análise de parâmetros para este tipo de ruído. Enquanto os métodos clássicos apresentam mudanças consideráveis nos valores ótimos dos parâmetros, os métodos baseados na derivada topológica apresentam valores semelhantes aos casos anteriores.

4.4 Teste 4 - Remoção de ruído speckle

Nesta seção é analisada a capacidade dos métodos apresentados para remover ruído speckle, o qual é de caráter multiplicativo. Este tipo de ruído é comum em imagens de ultrassom [49, 104, 131] e de *Synthetic Aperture Radar (SAR)*[128, 151], as quais costumam apresentar este tipo de degradação dependente do sinal. Os parâmetros de todos os métodos foram mantidos iguais aos dos casos anteriores (Tabela 4.1).

Na Tabela 4.7 e Figura 4.10 são apresentados os resultados para as imagens processadas utilizando os diferentes métodos para imagens poluídas com ruído tipo speckle com $\bar{\sigma} = 60$. Para ruído do tipo multiplicativo, observa-se que os dois métodos D_T conseguem remover uma quantidade maior de ruído que os outros. Observa-se também que os resultados são consideravelmente melhores que os obtidos com métodos clássicos.

A remoção deste tipo de ruído é um problema particularmente importante em imagens médicas de Ultrassom. Na Seção 4.5 os métodos propostos são utilizados para restaurar imagens reais.

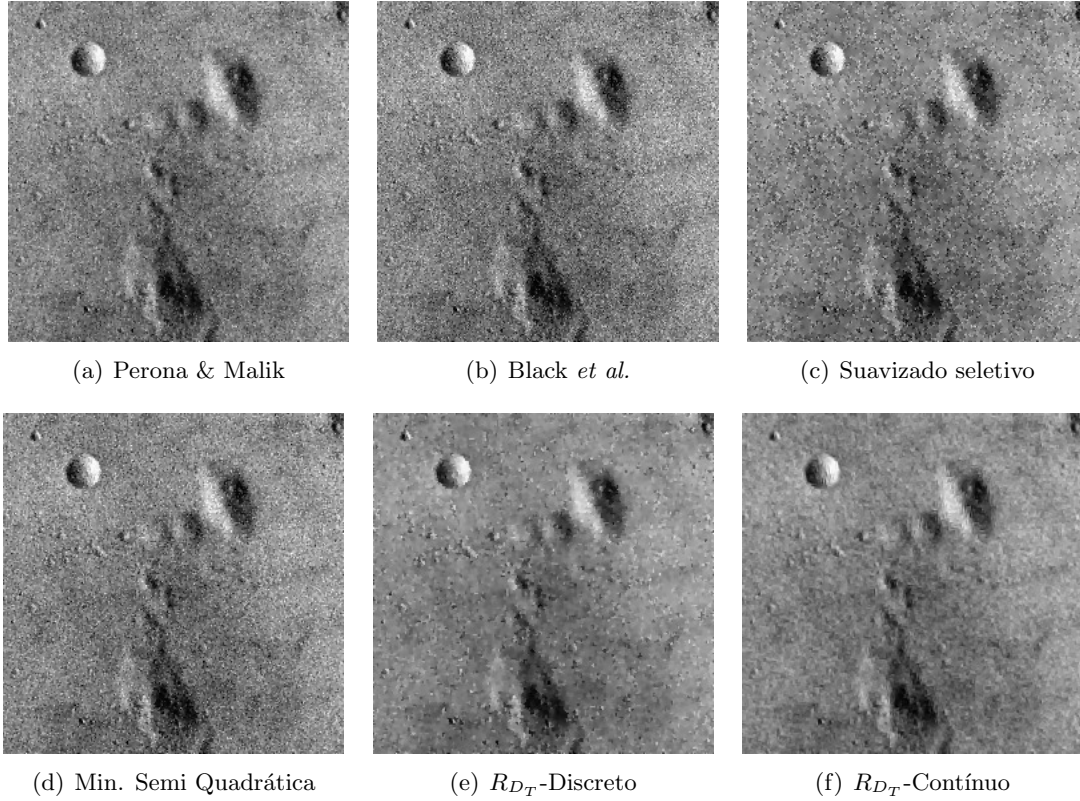


Figura 4.7: Resultados para a imagem da lua com cada método. Ruído Gaussiano $\bar{\sigma} = 30$, SNR inicial de aproximadamente 18.80 dB.

4.5 Teste 5 - Processamento de imagens médicas

As Figuras 4.12, 4.13 e 4.14 correspondem a várias imagens de angiografia com diferentes níveis de ruído Gaussiano [76] (primeira fila). Na Fig. 4.12, 4.13 e 4.14 também são apresentados os resultados correspondentes a estas imagens processadas com os algoritmos baseados na derivada topológica (i.e., R_{DT} -Contínuo e R_{DT} -Discreto).

Na Fig. 4.15 são apresentadas 3 imagens de Ultrassom. Estas imagens apresentam ruído tipo speckle [105]. A duas primeiras imagens (Figuras 4.15(a) e 4.15(d)) correspondem à artéria carótida de um paciente de sexo feminino. A primeira delas ao momento da sístole e a segunda à diástole. A terceira imagem (Fig. 4.15(g)) corresponde a um estudo de 4 câmeras do coração. Estas imagens apresentam um nível de ruído intenso o que dificulta outros processamento como segmentação. Os parâmetros utilizados para restaurar estas imagens foram $\Delta t = 1$, $k_0 = 1$ e $\alpha = 0.18$ para R_{DT} -Discreto, $k_0 = 2$ e $\alpha = 0.8$ para R_{DT} -Contínuo. Nas segunda e terceira colunas da Fig. 4.15 são apresentados os resultados correspondentes a estas imagens processadas com ambos algoritmos D_T , R_{DT} -Contínuo e R_{DT} -Discreto, respectivamente.

4.6 Comentários

Nesta primeira parte foi abordado o problema de restauração de imagens no contexto de Cálculo Variacional e EDP's. Em particular, foram apresentados alguns dos

	PSNR (dB)	SNR (dB)	$\mu(e)$	$\bar{\sigma}(e)$
Imagem poluída (lua)	14.12	13.91	0.0092	50.209
Evolutivo (Perona e Malik) (its = 30)	14.90	14.69	0.5075	45.884
Evolutivo (Black <i>et al.</i>) (its = 10)	14.14	13.94	0.5101	50.036
Evolutivo (Suavizado seletivo) (its = 30)	17.05	16.85	0.3954	35.790
Minimização Semi-Quadrática	14.79	14.58	0.5060	46.477
D_T Discreto	21.76	21.55	0.5565	20.818
D_T Contínua	21.50	21.29	0.5077	21.450
Imagem poluída (Lena)	13.48	13.44	-1.4972	54.022
Evolutivo (Perona e Malik) (its = 30)	14.23	14.20	-0.9967	49.533
Evolutivo (Black <i>et al.</i>) (its = 30)	13.56	13.52	-0.9916	53.521
Evolutivo (Suavizado seletivo) (its = 30)	16.20	16.16	-0.4243	39.507
Minimização Semi-Quadrática	14.14	14.11	-0.9933	50.038
D_T Discreto	20.64	20.61	-0.0652	23.688
D_T Contínua	20.33	20.30	-0.9854	24.516
Imagem poluída (barco)	13.44	13.23	-1.1405	54.254
Evolutivo (Perona e Malik) (its = 30)	14.19	13.98	-0.6417	49.793
Evolutivo (Black <i>et al.</i>) (its = 30)	13.54	13.33	-0.6363	53.653
Evolutivo (Suavizado seletivo) (its = 30)	16.29	16.09	0.1472	39.066
Minimização Semi-Quadrática	14.07	13.87	-0.6435	50.440
D_T Discreto	20.48	20.27	0.2695	24.123
D_T Contínua	20.34	20.13	-0.6414	24.511

Tabela 4.6: Comparação para ruído sal e pimenta. $p = 15$, SNR inicial de aproximadamente 13.50 dB.

mais utilizados métodos de restauração existentes na literatura. Este métodos foram computacionalmente implementados, sendo apresentados também resultados numéricos na restauração de imagens degradadas em diferentes situações.

Em seguida, e como uma contribuição na área, é utilizada a derivada topológica para tratar o problema de restauração. Esta ferramenta, originariamente concebida para tratar de problemas de otimização de topologia e forma e problemas inversos em mecânica, tem se mostrado muito eficiente ao ser aplicada no processamento de imagens. Neste caso, o método de restauração proposto originalmente por Belaid *et al.* [22] é revisado, sendo apresentado o cálculo detalhado da derivada topológica por meio do *Topological-Shape Sensitivity Method*. Este método, baseado na equação estacionária de condução de calor, tem se mostrado eficiente na remoção de ruído. Posteriormente, e utilizando uma abordagem discreta, um novo método de restauração baseado na derivada topológica e na equação de calor transiente é proposto. Esta abordagem discreta, mais simples na sua implementação e de menor custo computacional, apresentou bons resultados na remoção de ruído.

Finalmente, os métodos propostos são comparados com os existentes na literatura para diferentes intensidades e tipos de ruído. Em geral, para níveis pequenos de ruído os métodos D_T não se mostram superiores aos métodos existentes. No entanto, na medida que os ruído aumenta, os métodos D_T continuam apresentando resultados bons sem necessitar reajustar seus parâmetros enquanto a performance dos outros métodos é severamente

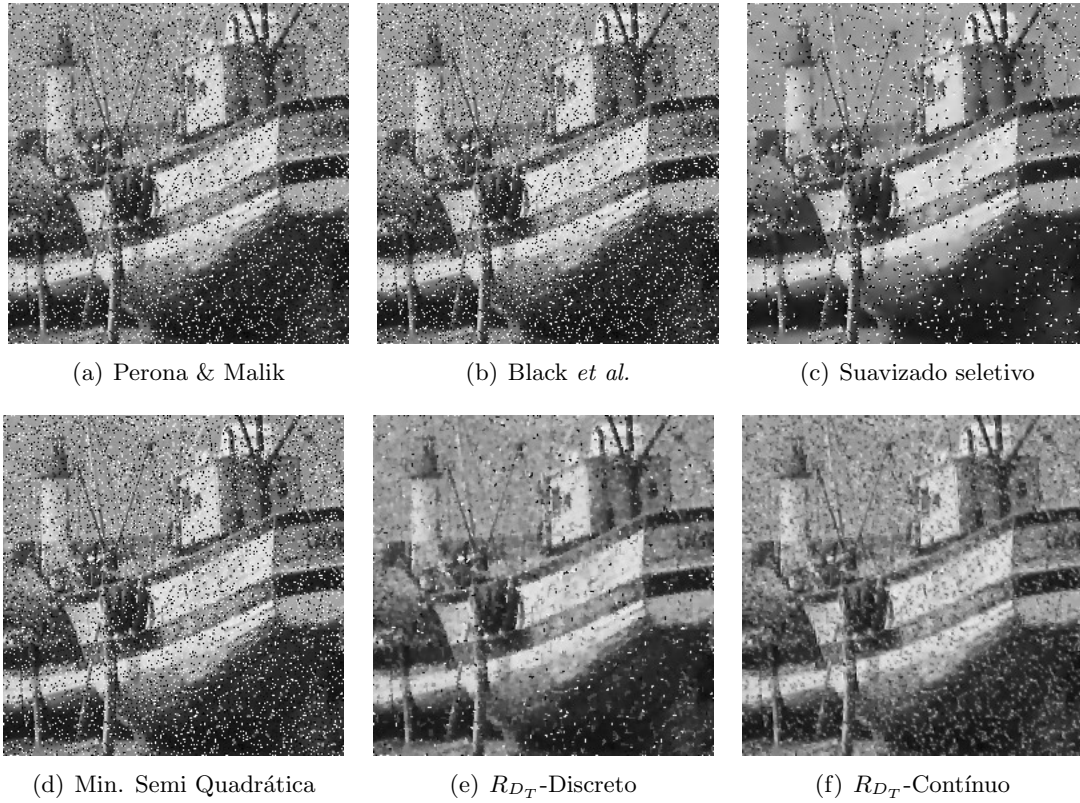
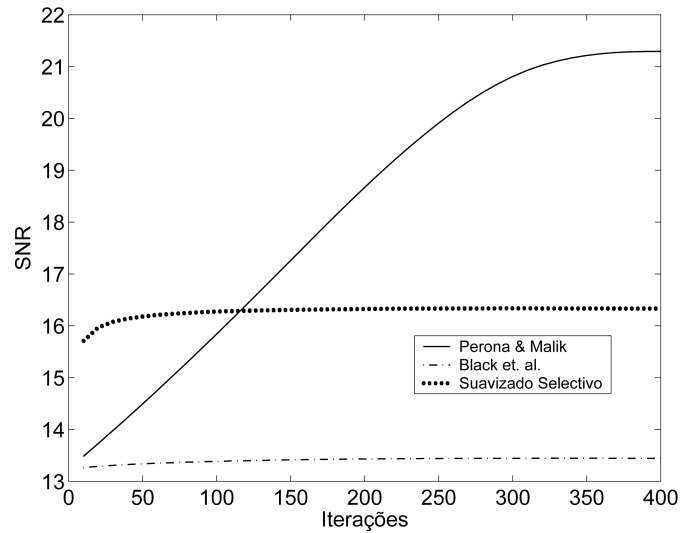


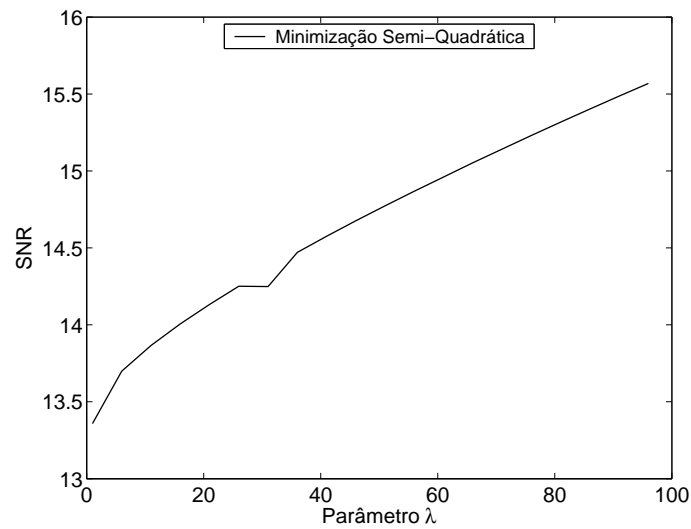
Figura 4.8: Resultados para a imagem de barco com cada método. Ruído sal e pimenta $p = 15$.

afetada na mesma situação, característica que facilita o seu uso.

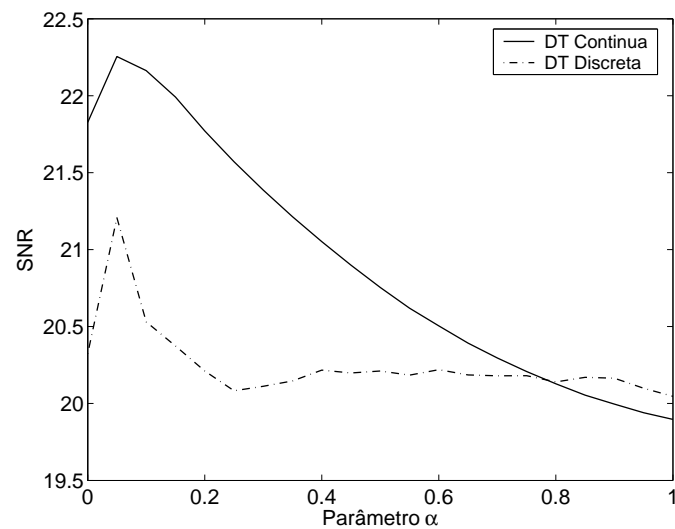
A pouca dependência nos parâmetros, evidencia a robustez dos métodos D_T , não requerendo informações adicionais (tipo de imagem, intensidade ou tipo de ruído, etc.) para proporcionar bons resultados no processamento.



(a) SNR vs. número de iteração - Métodos evolutivos.



(b) SNR vs. parâmetro λ - Minimização Semi-Quadrática.



(c) SNR vs. parâmetro α - Métodos D_T .

Figura 4.9: Análise de parâmetros (imagem do barco). Ruído sal e pimenta $p = 15$, SNR inicial de aproximadamente 13.50 dB.

	PSNR (dB)	SNR (dB)	$\mu(e)$	$\bar{\sigma}(e)$
Imagem poluída (lua)	18.42	18.21	0.2045	30.590
Evolutivo (Perona e Malik) (its = 30)	21.36	21.15	0.7029	21.796
Evolutivo (Black <i>et al.</i>) (its = 30)	19.28	19.07	0.7063	27.708
Evolutivo (Suavizado seletivo) (its = 30)	20.42	20.21	0.9881	24.279
Minimização Semi-Quadrática	20.69	20.49	0.7052	23.530
D_T Discreto	24.38	24.17	0.7228	15.388
D_T Contínua	25.28	25.07	0.7123	13.872
Imagem poluída (Lena)	18.67	18.64	0.5552	29.716
Evolutivo (Perona e Malik) (its = 30)	21.45	21.42	1.0557	21.549
Evolutivo (Black <i>et al.</i>) (its = 30)	19.88	19.84	1.0548	25.847
Evolutivo (Suavizado seletivo) (its = 30)	20.35	20.32	0.9789	24.472
Minimização Semi-Quadrática	20.46	20.43	1.0594	24.158
D_T Discreto	23.91	23.88	1.1942	16.205
D_T Contínua	24.75	24.71	1.0550	14.725
Imagem poluída (barco)	18.64	18.44	0.7254	29.799
Evolutivo (Perona e Malik) (its = 30)	21.44	21.24	1.2250	21.561
Evolutivo (Black <i>et al.</i>) (its = 30)	20.06	19.86	1.2268	25.288
Evolutivo (Suavizado seletivo) (its = 30)	20.25	20.04	1.0884	24.754
Minimização Semi-Quadrática	20.33	20.13	1.2318	24.511
D_T Discreto	23.82	23.61	1.2764	16.376
D_T Contínua	24.95	24.74	1.2253	14.374

Tabela 4.7: Comparação para ruído speckle. $\bar{\sigma} = 60$, SNR inicial de aproximadamente 18.50 dB.

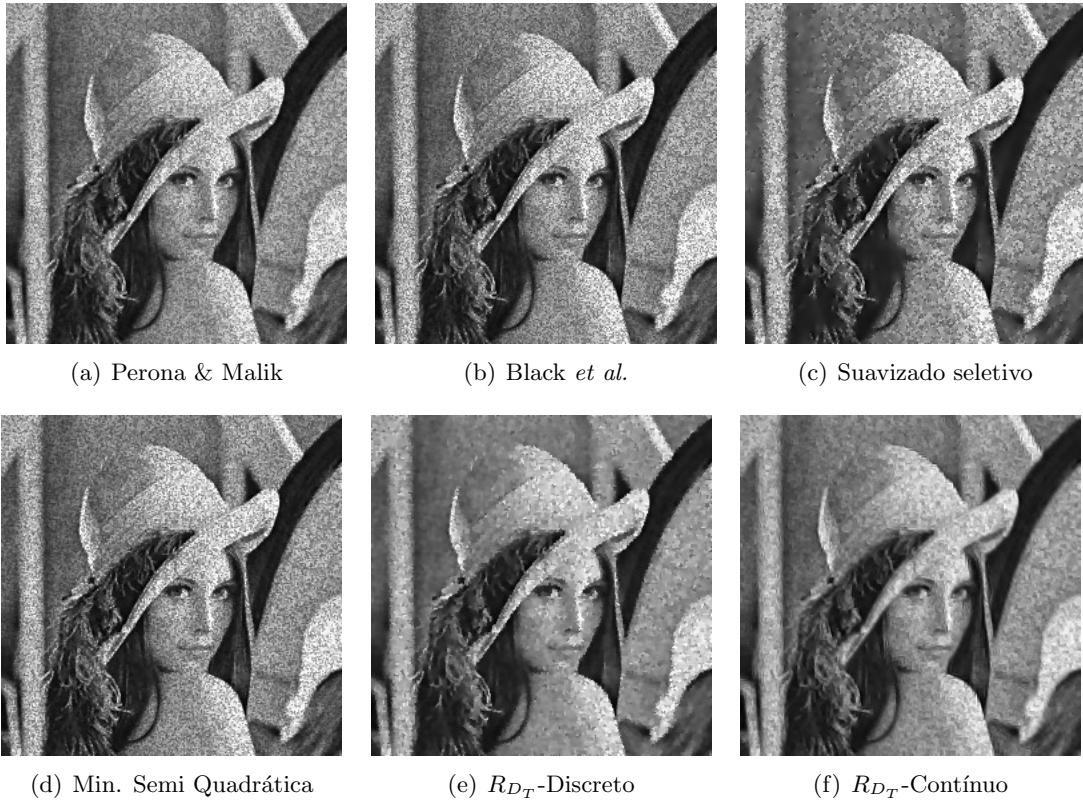
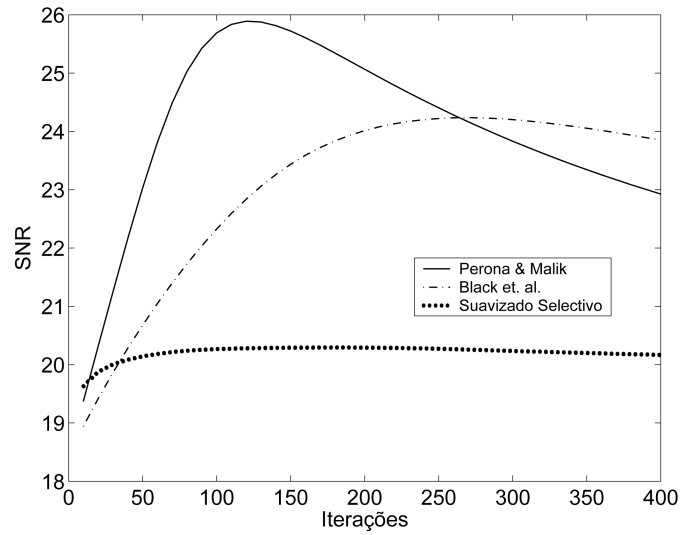
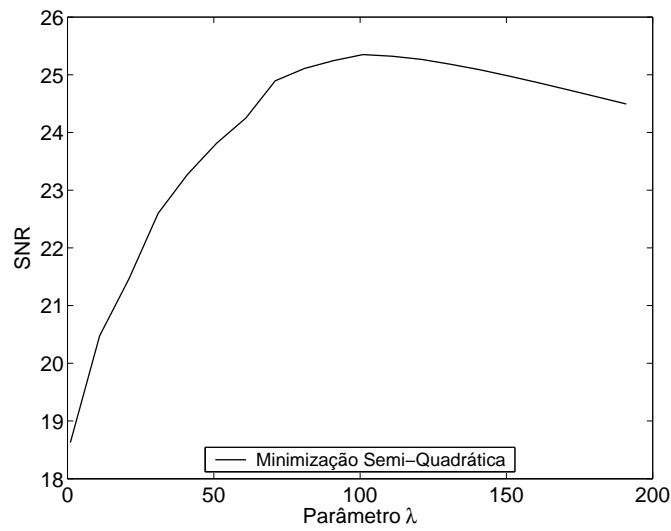


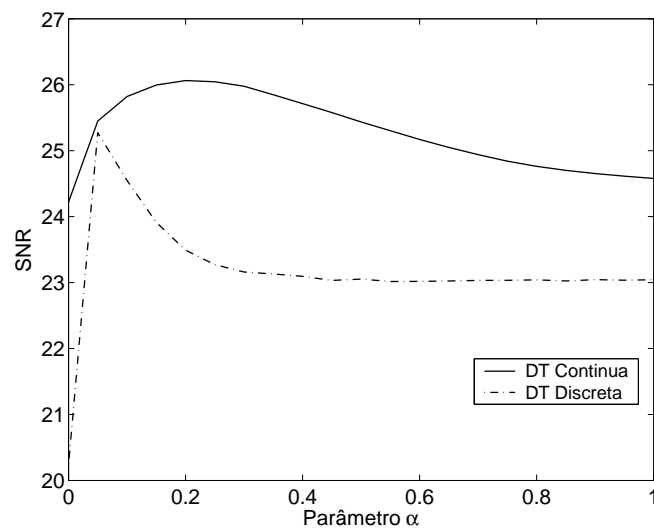
Figura 4.10: Resultados para a imagem de Lena com cada método. Ruído speckle $\bar{\sigma} = 60$.



(a) SNR vs. número de iteração - Métodos evolutivos.



(b) SNR vs. parâmetro λ - Minimização Semi-Quadrática.



(c) SNR vs. parâmetro α - Métodos D_T .

Figura 4.11: Análise de parâmetros (imagem do barco). Ruído speckle $\bar{\sigma} = 60$, SNR inicial de aproximadamente 13.50 dB.

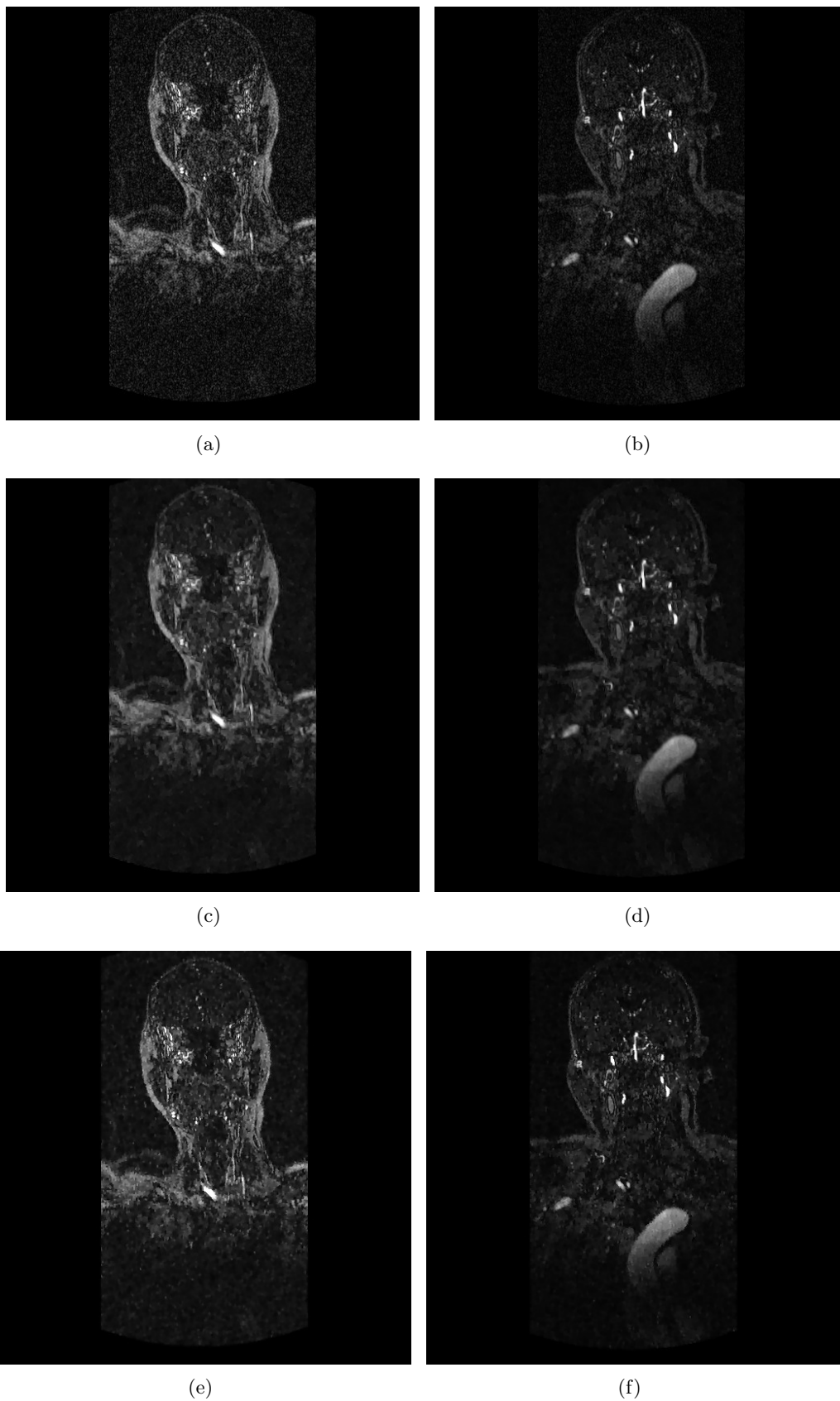


Figura 4.12: Imagens de Angiografia (região cervical) - Exemplos de ruído Gaussiano (primeira fila). Imagens processadas com R_{DT} -Contínuo (segunda fila, $k_0 = 2$ e $\alpha = 0.8$) e R_{DT} -Discreto (terceira fila, $k_0 = 1$ e $\alpha = 0.18$).

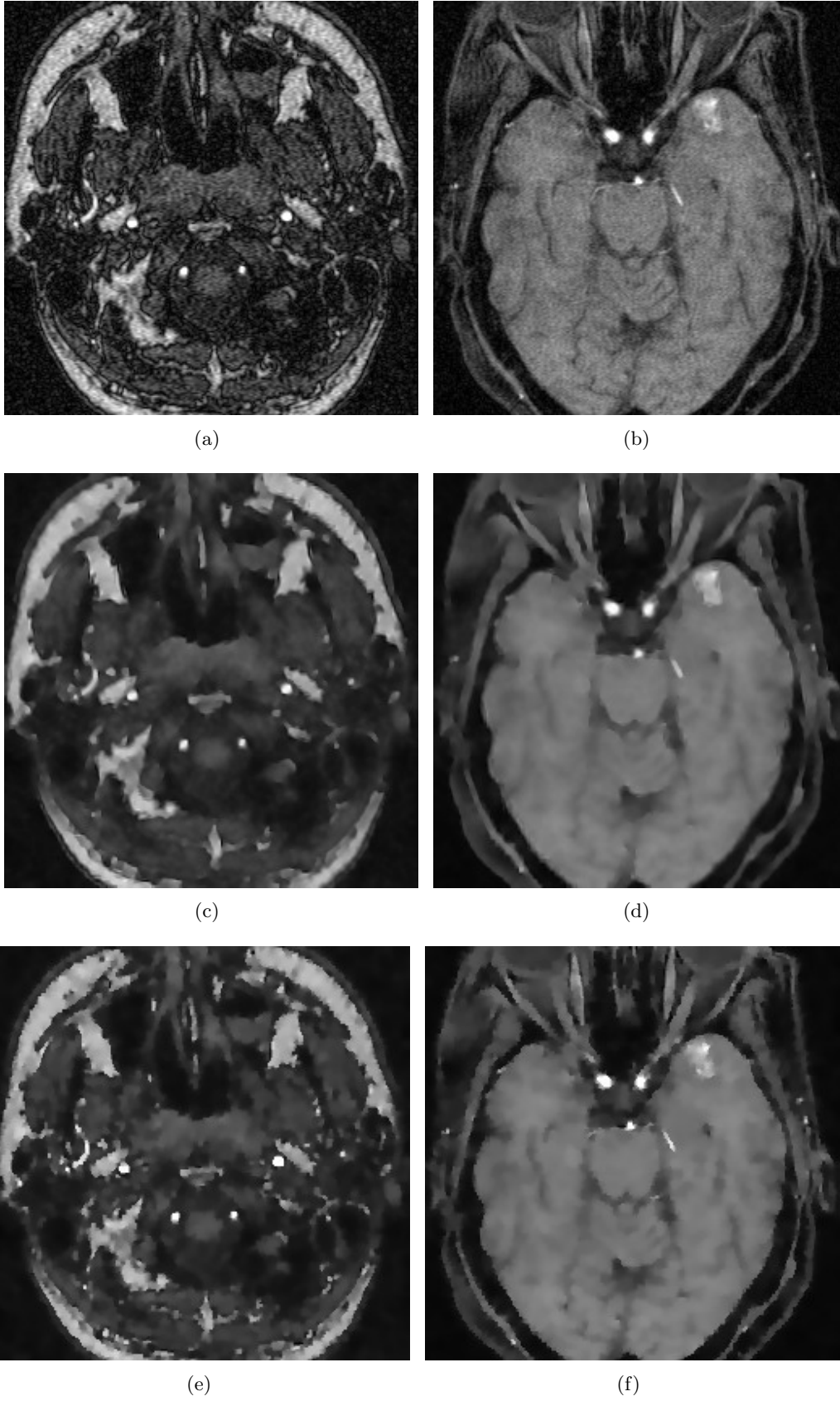


Figura 4.13: (Cont.) Diferentes imagens de Angiografia (artérias do cérebro) - Exemplos de ruído Gaussiano (primeira fila). Imagens processadas com R_{DT} -Contínuo (segunda fila, $k_0 = 2$ e $\alpha = 0.8$) e R_{DT} -Discreto (terceira fila, $k_0 = 1$ e $\alpha = 0.18$).

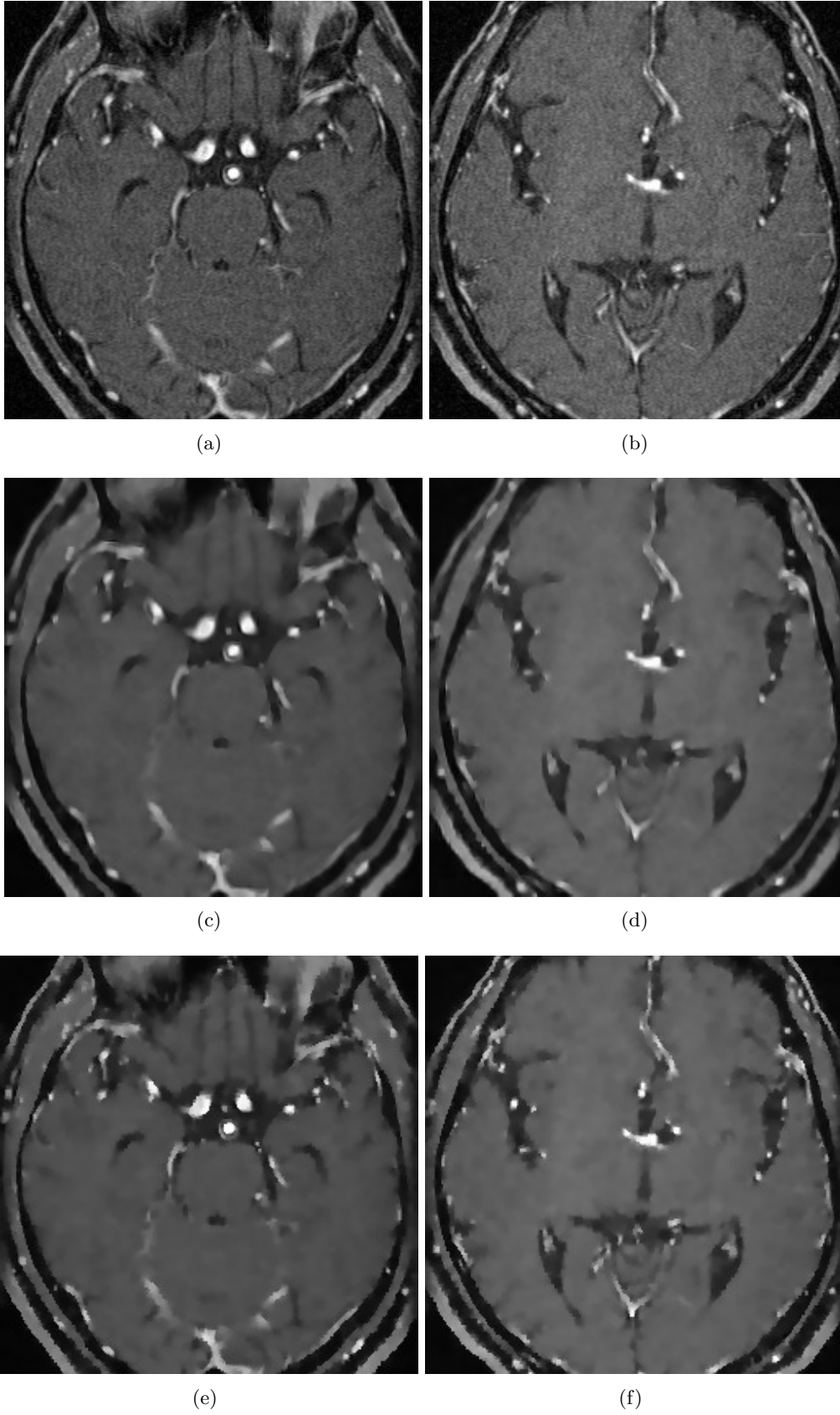


Figura 4.14: (Cont.) Diferentes imagens de Angiografia (artérias do cérebro) - Exemplos de ruído Gaussiano (primeira fila). Imagens processadas com R_{DT} -Contínuo (segunda coluna, $k_0 = 2$ e $\alpha = 0.8$) e R_{DT} -Discreto (terceira coluna, $k_0 = 1$ e $\alpha = 0.18$).

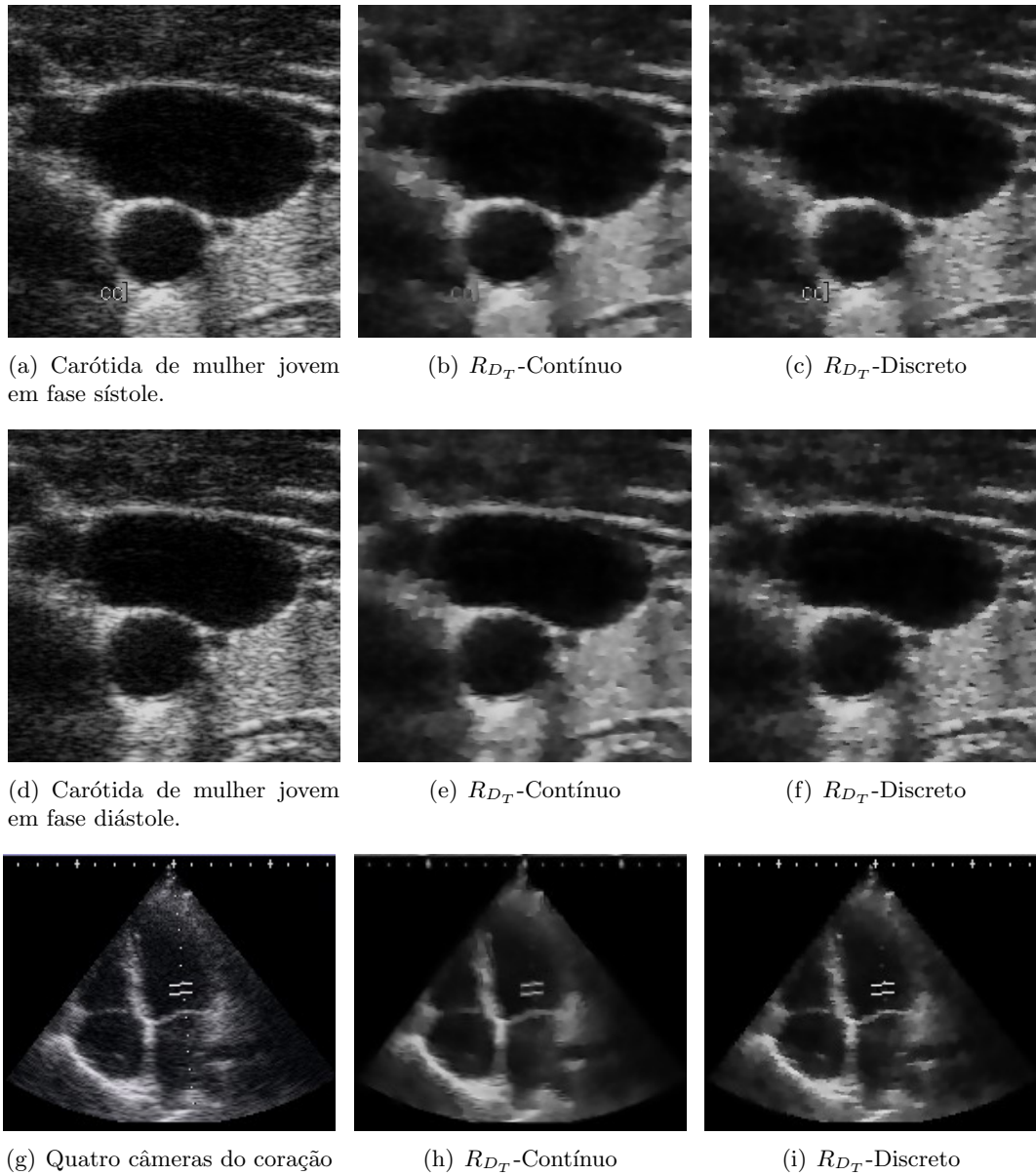


Figura 4.15: Diferentes imagens de ultrasom - Exemplos de ruído speckle. Imagens restauradas com R_{DT} -Contínuo (coluna central, $k_0 = 2$ e $\alpha = 0.8$) e R_{DT} -Discreto (coluna da direita, $k_0 = 1$ e $\alpha = 0.18$).

Parte II

Segmentação de Imagens

Introdução

Na segunda parte desta tese é abordado o problema de segmentação de imagens com um especial interesse em imagens médicas, mais uma vez, utilizando a derivada topológica. Sendo assim, no Capítulo 5 é estudado o funcional de Mumford-Shah, o qual foi o ponto de partida no estudo deste problema no contexto do Cálculo Variacional e EDP's. Além disso, este funcional tem servido como guia na definição de outros apresentados neste trabalho. No Capítulo 6 é apresentada uma nova técnica de segmentação de imagens baseada no conceito de derivada topológica. Os resultados obtidos com este método tem se mostrado promissores, em particular na presença de ruído. Posteriormente, baseado neste método um segundo algoritmo é proposto. Mais simples na sua implementação utilizando uma versão discreta dessa mesma idéia, é proposto um método mais simples e de menor custo computacional, que também se mostra promissor e robusto na presença de ruído. Por último, no Capítulo 7 os métodos propostos são comparados quantitativamente para vários níveis de ruído com outros freqüentemente utilizados. Também são estudados os resultados de ambos métodos propostos para diferentes parâmetros e sua influência nos resultados da segmentação.

Os resultados apresentados nesta parte representam adaptações de trabalhos publicados em anais de congressos e revistas internacionais, quais sejam:

- I. Larrabide, R. A. Feijóo, A. A. Novotny, E. Taroco and M. Masmoudi. An Image Segmentation Method Based on a Discrete Version of the Topological Derivative. www.wcsmo6.org. *WCSMO6 - 6th World Congress on Structural and Multidisciplinary Optimization*, Rio de Janeiro, Brazil, 30th May-03rd June 2005.
- I. Larrabide, R. A. Feijóo, E. Taroco, A. Novotny. Configurational Derivative As a Tool for Image Segmentation. *Proceedings of the ECCM 2006*. LNEC - Lisbon, Portugal 5th-9th June 2006.
- I. Larrabide, R. A. Feijóo, A. A. Novotny and E. A. Taroco. Topological Derivative: A Tool for Image Processing. *Computers & Structures - An International Journal*. Editors: K. J. Bathe & B. H. V. Topping. Guest Editor for the Special Issue: C. A. Mota Soares, M. Bendsoe, K. K. Choi and J. Herskovits. Accepted for publication - December 2006.
- I. Larrabide, A. A. Novotny, M. Masmoudi, R. A. Feijóo and E. A. Taroco. Topological Derivative as a Tool for Image Processing - Part I: Image Segmentation. *IEEE Transactions in Image Processing*. Submitted - December 2006.

Capítulo 5

Segmentação de Imagens

A segmentação de imagens, processo pelo qual pretende-se separar diferentes objetos do fundo ou *background* de uma imagem digital, ainda é um dos mais desafiantes problemas de processamento de imagens. Diferentes aplicações da segmentação de imagens podem ser mencionadas em várias áreas de pesquisa. Por exemplo, na análise de imagens de satélite, segmentação é utilizada para identificar cidades, rodovias, cultivos, lagos, etc. Em acompanhamento de movimento é de interesse identificar um mesmo objeto em uma seqüência de imagens [55, 78, 160, 204]. A introdução na medicina de técnicas vindas de outras áreas como Modelagem Computacional dentre outras, requer o uso de informações específicas dos pacientes como por exemplo imagens médicas. Estas imagens podem ser adquiridas de diferentes fontes como Tomografia Computadorizada (CT), Ressonância Magnética (MRI), *Single Photon Emission Computed Tomography* (SPECT), *Positron Emission Tomography* (PET) e Ultrassom (US). Uma das formas mais comuns de incorporar dados reais nas técnicas de modelagem é através de imagens reais. Para isto, primeiramente é necessário interpretar estas imagens e identificar as estruturas e objetos nelas presentes. Este processo é chamado de segmentação e é uma das mais antigas preocupações do processamento de imagens.

As técnicas de aquisição de imagens médicas fornecem aos especialistas informação útil (anatômica e funcional) do interior do corpo humano. Conseqüentemente, a demanda por ferramentas de segmentação que permitam facilitar operações como diagnóstico, tratamento, planejamento cirúrgico e monitoramento dentre outras, tem crescido consideravelmente desde a introdução deste tipo de tecnologia. Em particular, a aplicação da segmentação de imagens médicas que têm motivado este trabalho é na área de hemodinâmica, onde a segmentação de vasos sanguíneos para posterior reconstrução é amplamente utilizada para simular computacionalmente o fluxo sanguíneo [41, 63, 108, 193, 200].

Neste capítulo será brevemente analisado o estado da arte em segmentação de imagens no contexto de Cálculo Variacional e EDP's. Será apresentado o modelo introduzido por Mumford-Shah no ano 1989 [133], o qual coloca o processo de segmentação como um problema de mínimo. Também é discutida na Seção 5.2 a existência de solução deste problema, assim como às condições de optimalidade.

5.1 Estado da arte em Segmentação de Imagens via EDP's

As técnicas de aquisição de imagens médicas mencionadas fornecem imagens detalhadas da anatomia do corpo humano. Os conjuntos de dados 3D associados a estas imagens dão informação para análise não disponível em imagens 2D e ao mesmo tempo desafiam os profissionais habituados a trabalhar e interpretar imagens em duas dimensões.

Mesmo que as imagens sejam 3D, elas ainda são amplamente utilizadas como imagens 2D individuais. Informação quantitativa como tamanho e forma dos órgãos pode ser extraída destas imagens para realizar atividades como diagnóstico ou planejamento cirúrgico. Para conseguir realizar estas atividades, o primeiro passo é identificar os diferentes tecidos e estruturas anatômicas envolvidas. Por este motivo, o processo de segmentação deve ser preciso e repetível para ser clinicamente útil.

Uma primeira definição de segmentação poderia ser: *a segmentação é o processo que divide uma imagem nos objetos que a compõem*, mas esta definição é muito vaga. Para fornecer uma definição mais precisa deve-se considerar o objetivo ou aplicação específica na qual esta técnica vai ser utilizada. Por exemplo, na segmentação de imagens médicas, o objetivo é identificar os diferentes órgãos, regiões e estruturas anatômicas nos dados adquiridos via CT, MRI, Ultrassom, etc.

Podem ser identificados dois enfoques diferentes para o problema de segmentação. No primeiro deles o objetivo é achar as regiões ou conjuntos de pixels correspondentes a um objeto em uma imagem, identificando-os por regiões homogêneas. Neste enfoque, objetiva-se identificar os limites e regiões ocupadas por todos os objetos da imagem. Uma outra alternativa é identificar objetos isoladamente, por exemplo, separar uma pessoa do fundo em uma fotografia ou identificar um dente vários em uma placa dentária. Neste caso, procura-se uma curva que limite o objeto de interesse. Logo, o tipo de técnica utilizada dependerá da aplicação considerada.

Uma questão fundamental comum a todos os métodos de segmentação é a detecção de bordas. De fato, esse problema tem sido motivo de preocupação desde o início do processamento de imagens. As técnicas clássicas de segmentação de imagens baseiam-se em características locais das derivadas das bordas, como por exemplo as derivadas primeiras ou segundas [169, 173, 181]. No caso de derivadas primeiras pode-se citar o filtro proposto por Canny [37], que ainda é muito utilizado. Já no caso de derivadas segundas pode ser mencionada a proposta de Marr e Hildreth [127] baseada nos cruzamentos de zero do Laplaciano do Gaussiano (chamado LoG). Estas alternativas combinam derivadas de diferentes ordens, mas apresentam a desvantagem, de serem locais. Além do mais, estas técnicas levam em conta duas características básicas dos *pixels*: descontinuidades e similaridades. Muitas das técnicas clássicas (e.g., *multiple thresholding*, *region growing*, *morphologic filtering* entre outras [75, 97]) têm sido utilizadas para tratar este problema, fornecendo diferentes resultados [91, 187]. No entanto, tais técnicas são pouco confiáveis quando a região segmentada está rodeada por outras com intensidade similar (estruturas de pouco contraste).

Métodos mais sofisticados, como *level sets*, utilizam equações de evolução para acompanhar as superfícies em movimento. Estes métodos são baseados no cálculo de soluções de sistemas de equações hiperbólicas lineares/não-lineares ou de evolução para descrever o movimento. A aproximação inicial da solução (semente) evolui até achar os limites da região de interesse. Estes métodos requerem interação do usuário para introduzir uma ou mais sementes que serão o ponto de partida do algoritmo [124, 125, 172, 179]. Embora segundo os autores esta técnica se mostra útil na segmentação de imagens médicas, o

cálculo da solução das equações evolutivas envolvidas requer de um elevado esforço computacional. Uma outra técnica muito utilizada para segmentar imagens médicas é *active contours* [30, 121, 205, 206] (também chamada de *snakes*) freqüentemente empregada em diferentes tipos de imagens médicas, em particular, para segmentação de imagens do cérebro. Novamente, o uso de equações evolutivas implica em um custo computacional elevado.

Utilizando *markov random fields*, em [88] e [208] são descritas técnicas de segmentação completamente automáticas, especialmente desenhadas para segmentar imagens MRI do cérebro. Estas técnicas capturam três características espaciais das imagens de MRI: distribuição não paramétrica das intensidades dos tecidos, correlação dos tecidos e inomogeneidades no sinal. Uma vez que estes campos são calculados (usando modelos probabilísticos apropriados), um algoritmo de otimização (*Iterated Conditional Modes*, *Simulated Annealing*, Maximização da esperança, etc.) é utilizado para recalcula-los até a convergência. Novamente, a limitação desta técnica é o excessivo custo computacional.

No método *Bootstrap* de segmentação [21, 100], a idéia é utilizar um outro método de segmentação (por exemplo, *multiple thresholding*) para produzir uma primeira estimação da imagem segmentada. Assim, calculando estimativas como média e variabilidade das intensidades nas diferentes regiões e a partir da primeira estimação da imagem segmentada, novas imagens são geradas e segmentadas sucessivamente, repetindo este processo até chegar em um ponto estacionário. Outro método de segmentação muito utilizado é o *K-Means* [77, 159]. *K-means* é um algoritmo de agrupamento muito conhecido que reúne pixels em uma imagem em K grupos ao minimizar a variância total inter-grupo. O *Fuzzy C-means* (FCM) [48, 24], é um método de agrupamento não supervisionado, que tem sido amplamente utilizado em diferentes problemas de computação gráfica e processamento de imagens. Neste caso, os dados são classificados em conjuntos definido protótipos dos grupos (ou classes iniciais), uma norma associada e minimizando uma função custo.

O enfoque apresentado neste capítulo, aborda o problema de segmentação do ponto de vista do Cálculo Variacional e EDP's.

5.2 Segmentação como um Problema de Minimização

No ano 1989 foi introduzido o funcional de Mumford-Shah [133], o qual coloca o problema de segmentação como um problema de mínimo. A maior complicação que este funcional apresenta é que envolve variáveis de dois tipos diferentes: a função de intensidade u (um campo escalar definido no domínio da imagem $\Omega \in \mathbb{R}^N$, $N = 2, 3$) e o conjunto de descontinuidades K (conjunto de curvas de dimensão $N - 1$).

A seguir este funcional é formalmente apresentado e posteriormente são discutidas questões referentes a existência de um mínimo para este funcional. Também serão deduzidas as condições de optimalidade que o mínimo deste funcional deve satisfazer.

5.2.1 Funcional de Mumford-Shah

Sejam Ω uma região limitada aberta de \mathbb{R}^N ($N = 2, 3$), e $v_0(\mathbf{x})$, $\mathbf{x} \in \Omega$ a imagem a ser segmentada (onde $v_0(\mathbf{x})$ representa a intensidade da imagem em cada ponto). Sem perda de generalidade, pode-se assumir que $0 \leq v_0(\mathbf{x}) \leq 1$ q.t.p. $\mathbf{x} \in \Omega$. Procura-se um par (u, K) , onde $K \subset \Omega$ é o conjunto de descontinuidades que minimiza

$$\mathcal{F}(u, K) = \int_{\Omega-K} (v_0 - u)^2 d\Omega + \alpha \int_{\Omega-K} |\nabla u|^2 d\Omega + \beta \int_K d\sigma, \quad (5.1)$$

sendo $\int_K d\sigma$ a medida de K , $\alpha, \beta \geq 0$ e $u \in H^1(\Omega - K)^1$.

Conjetura 1. *Assume-se que existe um mínimo de \mathcal{F} tal que as bordas (conjunto de descontinuidades K) são a união de um conjunto finito de curvas embebidas em Ω [132].*

5.2.2 Existência da Solução

Neste funcional há uma restrição muito forte proveniente do fato de K ser um conjunto de curvas $C^1(\Omega)$. Para poder contornar esta dificuldade, define-se a longitude de K como sendo a sua medida de Hausdorff $(N - 1)$ -dimensional $\mathcal{H}^{N-1}(K)$ (ver Apêndice 5.A ou [16]). Logo, o funcional fica definido como

$$\mathcal{F}(u, K) = \int_{\Omega-K} (v_0 - u)^2 d\Omega + \alpha \int_{\Omega-K} |\nabla u|^2 d\Omega + \beta \mathcal{H}^{N-1}(K). \quad (5.2)$$

Eliminando soluções triviais, alguns casos especiais deste funcional podem ser analisados. Por exemplo, considerando $v_0 \neq u$ e $K \neq \emptyset$, o funcional de Mumford-Shah reduzido é dado por²

$$E(u, K) = \int_{\Omega-K} (v_0 - u)^2 d\Omega + \beta \mathcal{H}^{N-1}(K). \quad (5.3)$$

Este funcional é de particular interesse porque posteriormente será utilizado um funcional similar a este.

Para poder aplicar o método direto do cálculo das variações e achar as condições de optimalidade do funcional é necessário encontrar uma topologia que assegure ao mesmo tempo semi-continuidade inferior (*Lower-semicontinuity*) de \mathcal{F} e também que a seqüência minimizante seja compacta (para maiores detalhes, ver [16] e referências ali citas). Por esta razão, aparece a necessidade de encontrar outra formulação alternativa para $\mathcal{F}(u, K)$. Esta nova formulação envolve o espaço $BV(\Omega)$ de funções de variação limitada em Ω , seja

$$BV(\Omega) = \left\{ u \in L^1(\Omega); \int_{\Omega} |Du| < \infty \right\}.$$

sendo D o operador gradiente. Pode-se então definir o seguinte funcional

$$G(u) = \int_{\Omega} (v_0 - u)^2 d\Omega + \alpha \int_{\Omega} |\nabla u|^2 d\Omega + \beta \mathcal{H}^{N-1}(S_u). \quad (5.4)$$

¹ Sendo $\Omega - K$ equivalente a $\Omega \setminus K$.

² Tomando $\alpha = 0$, ou seja, desprezando a influência do termo do gradiente, $u = v_0$ e $K = \emptyset$ são solução ($\mathcal{F}(v_0, \emptyset) = 0$). Logo, a solução coincide com a imagem sendo segmentada, com o conjunto de descontinuidades vazio, o que não é uma segmentação!

A idéia consiste em associar o conjunto de bordas K ao conjunto de salto S_u de u (o conjunto de pontos nos quais u não é contínua), eliminando assim a variável K . A condição de semi-continuidade inferior não pode ser obtida para conjuntos, mas há resultados mostrando que esta condição pode ser obtida para funções. Este espaço ainda apresenta algumas funções não constantes patológicas que são contínuas e têm derivada igual a zero q.t.p. (podem-se citar como exemplo as funções de Cantor-Vitali [6]). Por este motivo, necessita-se utilizar um espaço mais restrito. Neste caso é utilizado o espaço $SBV(\Omega)$ de funções especiais de variação limitada (para mais detalhes, o leitor pode dirigir-se a [16] pgs. 135-137), que elimina estas funções patológicas. Têm-se então dois problemas:

$$(P_1) \inf_{u,K} \{ \mathcal{F}(u, K), u \in H^1(\Omega - K) \cap L^\infty(\Omega), K \subset \Omega, K \text{ fechado}, \mathcal{H}^{N-1}(K) < \infty \}$$

$$(P_2) \inf_u \{ G(u), u \in SBV(\Omega) \cap L^\infty(\Omega) \}$$

Uma relação entre estes dois problemas foi achada por Ambrosio [5], quem apresenta o seguinte resultado:

Teorema 2. *Sejam $K \subset \Omega$ um conjunto fechado tal que $\mathcal{H}^{N-1}(K) < \infty$ e $u \in H^1(\Omega - K) \cap L^\infty(\Omega)$. Logo $u \in SBV(\Omega)$ e $S_u \subset K$.*

Deste teorema segue que $\inf P_2 < \inf P_1$, dado que K e S_u diferem só em um conjunto de medida nula como foi provado por Di Giorgi *et al.* [45]. A prova de que (P_2) tem solução é uma consequência do seguinte teorema [16]:

Teorema 3. *Seja $u_n \in SBV(\Omega)$ uma seqüência de funções tal que existe uma constante C com $|u_n(x)| \leq C < \infty$ q.t.p. $x \in \Omega$ e $\int_\Omega |\nabla u_n|^2 d\Omega + \mathcal{H}^{N-1}(S_{u_n}) \leq C$. Logo existe uma subsequência u_{n_k} convergindo q.t.p. x para a função $u \in SBV(\Omega)$. Também, ∇u_{n_k} converge de forma fraca em $L^2(\Omega)^N$ a ∇u , e $\lim \mathcal{H}^{N-1}(S_{u_{n_k}}) \geq \mathcal{H}^{N-1}(S_u)$.*

Obtém-se uma solução para (P_2) ao aplicar o Teorema 3 em uma seqüência minimizante qualquer de (P_2) , observando-se de ante mão que a análise de seqüências minimizantes pode ser restrita satisfazendo $|u_n|_{L^\infty(\Omega)} \leq |u_0|_{L^\infty(\Omega)}$. Segue-se então que as diferentes variantes do funcional de Mumford-Shah são equivalentes. Por simplicidade, na próxima seção será estudada a forma original do funcional na Eq. (5.1), devido as dificuldades que aparecem ao trabalhar com a medida $\mathcal{H}^{N-1}(K)$. Usando os resultados discutidos anteriormente, serão deduzidas as condições de optimalidade que garantem a existência do mínimo. Procura-se então um par (u, K) com regularidade suficiente (de maneira que todas as operações estejam bem definidas) e que satisfaça

$$\mathcal{F}(u, K) \leq \mathcal{F}(\hat{u}, \hat{K})$$

para todo \hat{u} e \hat{K} nos correspondentes espaços. Considera-se $K = \bigcup_i \Gamma_i$, sendo Γ_i as fronteiras das componentes Ω_i conectadas de Ω , e $\Omega_i \subset \Omega$. Considera-se também por simplicidade, uma imagem com um único objeto, sendo Ω_i a região ocupada pelo objeto, e Ω_e o fundo (*background*) da imagem. Logo, existe uma única curva Γ que limita o objeto e o fundo, então:

Teorema 4. *Seja (u, Γ) o par que minimiza $\mathcal{F}(u, \Gamma)$, ambos suficientemente regulares, logo*

$$\begin{aligned} -\alpha \Delta u &= v_0 - u \quad \text{em } \Omega, \\ \frac{\partial u}{\partial \mathbf{n}} &= 0 \quad \text{em } \Gamma_i \cup \Gamma_e \cup \partial\Omega, \\ \varphi_i(u) - \varphi_e(u) + \beta \text{ curv}\Gamma &= 0 \quad \text{em } \Gamma. \end{aligned}$$

Sendo $\varphi_i(u)|_{\Omega_i}$ e $\varphi_e(u)|_{\Omega_e}$, e $\varphi(u) = (v_0 - u)^2 + \alpha |\nabla u|^2$.

Prova 2. *Para provar este resultado é calculada a variação total do funcional $\mathcal{F}(u, \Gamma)$. Primeiramente é estudada a variação de $\mathcal{F}(u, \Gamma)$ em relação a u na direção η , denotada $\mathcal{F}(u, \Gamma; \eta, 0)$. A condição de optimalidade em u é dada por $\mathcal{F}(u, \Gamma; \eta, 0) = 0$ para todo η suficientemente regular. Tem-se então*

$$\mathcal{F}(u, \Gamma) = \int_{\Omega-\Gamma} (v_0 - u)^2 d\Omega + \alpha \int_{\Omega-\Gamma} |\nabla u|^2 d\Omega + \beta \int_{\Gamma} d\sigma.$$

Calcula-se a primeira variação de \mathcal{F} em u

$$\begin{aligned} \mathcal{F}(u, \Gamma; \eta, 0) &= \frac{d}{dt} \mathcal{F}(u + t\eta, \Gamma)|_{t=0} \\ &= \frac{d}{dt} \left[\int_{\Omega-\Gamma} (v_0 - (u + t\eta))^2 d\Omega + \alpha \int_{\Omega-\Gamma} \nabla(u + t\eta) \cdot \nabla(u + t\eta) d\Omega + \beta \int_{\Gamma} d\sigma \right]_{t=0} \\ &= \left[\int_{\Omega-\Gamma} -2(v_0 - u - t\eta)\eta d\Omega + \alpha \int_{\Omega-\Gamma} 2(\nabla(u + t\eta) \cdot \nabla\eta) d\Omega \right]_{t=0} \\ &= \int_{\Omega-\Gamma} -2(v_0 - u)\eta d\Omega + \alpha \int_{\Omega-\Gamma} 2(\nabla u \cdot \nabla\eta) d\Omega. \end{aligned}$$

Usando o Teorema da Divergência, tem-se

$$\int_{\Omega-\Gamma} -2((v_0 - u) + \alpha \Delta u)\eta d\Omega + 2\alpha \int_{\Gamma_i \cup \Gamma_e \cup \partial\Omega} \eta \nabla u \cdot \mathbf{n} d\sigma = 0 \quad \forall \eta \text{ suficientemente regular.}$$

Como esta condição é satisfeita para todo η , necessariamente

$$\begin{cases} (v_0 - u) + \alpha \Delta u = 0 & \text{em } \Omega, \text{ e} \\ \frac{\partial u}{\partial \mathbf{n}} = 0 & \text{em } \Gamma_i \cup \Gamma_e \cup \partial\Omega. \end{cases}$$

o que conclui a primeira parte do teorema.

Na segunda parte desta prova, será estudada a variação de $\mathcal{F}(u, \Gamma)$ com respeito a Γ . Para isto, serão utilizados conceitos de Análise de Sensibilidade a Mudança de Forma. Primeiramente estuda-se a sensibilidade dos dois primeiros termos de $\mathcal{F}(u, \Gamma)$, deixando o termo $\int_{\Gamma} d\sigma$ para ser analisado posteriormente. Seja então $\mathbf{x}_\tau = \mathbf{x} + \tau \mathbf{v}(\mathbf{x})$, onde \mathbf{v} é o campo de velocidade que caracteriza a mudança de forma, Γ_τ é a interface entre Ω_{e_τ} e Ω_{i_τ} , \mathbf{n}_e e \mathbf{n}_i são as normais respectivas a cada parte do domínio e $\Omega_\tau = \Omega_{e_\tau} \cup \Omega_{i_\tau} \cup \Gamma_\tau$. Objetiva-se então estudar a variação de Γ_τ sob o efeito de $\mathbf{v}(\mathbf{x})$. Considera-se também a função $\varphi(u)$, dada por

$$\varphi(u) = (v_0 - u)^2 + \alpha \nabla u \cdot \nabla u.$$

Logo, utilizando o Teorema de Transporte de Reynolds [82] a derivada total com respeito de τ em $\tau = 0$ é dada por

$$\begin{aligned} \left. \frac{d}{d\tau} \int_{\Omega_\tau} \varphi \, d\Omega_\tau \right|_{\tau=0} &= \left[\frac{d}{d\tau} \int_{\Omega_{e\tau}} \varphi_e \, d\Omega + \frac{d}{d\tau} \int_{\Omega_{i\tau}} \varphi_i \, d\Omega \right]_{\tau=0} \\ &= \int_{\Omega_e} \varphi'_e \, d\Omega + \int_{\Omega_i} \varphi'_i \, d\Omega + \int_{\partial\Omega} \varphi_e \underbrace{(\mathbf{v} \cdot \mathbf{n})}_{\mathbf{v}|_{\partial\Omega}=0} \, d\partial\Omega \\ &\quad + \int_{\Gamma} \varphi_e (\mathbf{v} \cdot \mathbf{n}_e) \, d\sigma + \int_{\Gamma} \varphi_i (\mathbf{v} \cdot \mathbf{n}_i) \, d\sigma. \end{aligned} \quad (5.5)$$

Dado que

$$\varphi' = 2(- (v_0 - u)u' + \alpha \nabla u \cdot \nabla u'),$$

pode-se re-escrever (5.5) como

$$\begin{aligned} \left. \frac{d}{d\tau} \int_{\Omega_\tau} \varphi \, d\Omega_\tau \right|_{\tau=0} &= \int_{\Omega_e} 2(- (v_0 - u_e)u'_e + \alpha \nabla u_e \cdot \nabla u'_e) \, d\Omega \\ &\quad + \int_{\Omega_i} 2(- (v_0 - u_i)u'_i + \alpha \nabla u_i \cdot \nabla u'_i) \, d\Omega \\ &\quad + \int_{\Gamma} \varphi_i (\mathbf{v} \cdot \mathbf{n}_i) - \varphi_e (\mathbf{v} \cdot \mathbf{n}_i) \, d\sigma, \end{aligned}$$

já que $\mathbf{n}_i = -\mathbf{n}_e$. Utilizando o Teorema da Divergência, tem-se

$$\begin{aligned} \left. \frac{d}{d\tau} \int_{\Omega_\tau} \varphi \, d\Omega_\tau \right|_{\tau=0} &= -2 \int_{\Omega_e} ((v_0 - u_e) + \alpha \Delta u_e) \cdot u'_e \, d\Omega - 2 \int_{\Omega_i} ((v_0 - u_i) + \alpha \Delta u_i) \cdot u'_i \, d\Omega \\ &\quad + \int_{\Gamma} u'_e \nabla u_e \cdot \mathbf{n}_i \, d\sigma + \int_{\Gamma} u'_i \nabla u_i \cdot \mathbf{n}_i \, d\sigma + \int_{\Gamma} (\varphi_e - \varphi_i) (\mathbf{v} \cdot \mathbf{n}_i) \, d\sigma. \end{aligned}$$

Da primeira parte da demonstração tem-se que $(v_0 - u_e) + \alpha \Delta u_e = 0$ e $(v_0 - u_i) + \alpha \Delta u_i = 0$, e também que $\nabla u_e \cdot \mathbf{n}_i = \frac{\partial u_e}{\partial \mathbf{n}} = 0$ e $\nabla u_i \cdot \mathbf{n}_i = \frac{\partial u_i}{\partial \mathbf{n}} = 0$ em Γ . Logo

$$\left. \frac{d}{d\tau} \int_{\Omega_\tau} \varphi \, d\Omega_\tau \right|_{\tau=0} = \int_{\Gamma} (\varphi_e - \varphi_i) (\mathbf{v} \cdot \mathbf{n}_i) \, d\sigma.$$

Considera-se agora a derivada $\left. \frac{d}{d\tau} \int_{\Gamma_\tau} d\sigma \right|_{\tau=0}$. Sem perda de generalidade e usando conceitos de Mecânica do Contínuo, pode-se considerar $d\sigma = \|d\mathbf{x}_\tau\|$, a norma de um elemento diferencial de x na configuração espacial (descrição Euleriana, ver [82]). Logo, utilizando as identidades descritas na Seção 1.B a derivada material com respeito a τ do

campo espacial esta dada por

$$\begin{aligned}
 \frac{d}{d\tau} \int_{\Gamma_\tau} \|d\mathbf{x}_\tau\| \Big|_{\tau=0} &= \int_{\Gamma} \frac{d}{d\tau} \|\mathbb{F}_\tau^{-T} d\mathbf{x}\| \Big|_{\tau=0} \\
 &= \int_{\Gamma} \frac{d}{d\tau} (\mathbb{F}_\tau^{-T} d\mathbf{x} \cdot \mathbb{F}_\tau^{-T} d\mathbf{x})^{\frac{1}{2}} \Big|_{\tau=0} \\
 &= \frac{1}{2} \int_{\Gamma} \frac{d}{d\tau} (\mathbb{F}_\tau^{-1} \mathbb{F}_\tau^{-T}) d\mathbf{x} \cdot \frac{d\mathbf{x}}{\|\mathbb{F}_\tau^{-T} d\mathbf{x}\|} \Big|_{\tau=0} \\
 &= \frac{1}{2} \int_{\Gamma} \frac{1}{\|d\mathbf{x}\|} ((\nabla \mathbf{v})^T + (\nabla \mathbf{v})) \cdot d\mathbf{x} \\
 &= \int_{\Gamma} \frac{(\nabla_s \mathbf{v}) d\mathbf{x} \cdot d\mathbf{x}}{\|d\mathbf{x}\|}.
 \end{aligned}$$

Sendo $\nabla_s \mathbf{v}$ a parte simétrica do gradiente da velocidade. Considerando $\mathbf{v} = \hat{v} \mathbf{n}$ para \mathbf{n} a normal de Γ , tem-se para $\hat{v} = cte$

$$\begin{aligned}
 \int_{\Gamma} \frac{\nabla_s(\hat{v} \mathbf{n}) d\mathbf{x} \cdot d\mathbf{x}}{\|d\mathbf{x}\|} &= \int_{\Gamma} \frac{\overbrace{(\nabla_s \hat{v} \otimes \mathbf{n} + \hat{v} \nabla_s \mathbf{n})}^{=0} d\mathbf{x} \cdot d\mathbf{x}}{\|d\mathbf{x}\|} \\
 &= \int_{\Gamma} \hat{v} \nabla_s \mathbf{n} d\mathbf{x} \cdot \frac{d\mathbf{x}}{\|d\mathbf{x}\|}.
 \end{aligned}$$

Tendo em vista que em Γ $d\mathbf{x} = \|d\mathbf{x}\| \mathbf{t}$, \mathbf{t} o vetor tangente de Γ , na expressão anterior obtém-se

$$\int_{\Gamma} \hat{v} \nabla_s \mathbf{n} d\mathbf{x} \cdot \frac{d\mathbf{x}}{\|d\mathbf{x}\|} = \int_{\Gamma} \hat{v} (\nabla_s \mathbf{n}) \mathbf{t} \cdot \mathbf{t} \|d\mathbf{x}\|,$$

logo

$$\frac{d}{dt} \int_{\Gamma} d\sigma \Big|_{\tau=0} = \int_{\Gamma} \hat{v} (\nabla_s \mathbf{n}) \mathbf{t} \cdot \mathbf{t} d\sigma.$$

Por outro lado tem-se que [23] (Cap.8 - pgs. 290-298)

$$\frac{\partial \mathbf{n}}{\partial \mathbf{t}} = \text{curv}(K) \mathbf{t} + \text{tor}(K) \boldsymbol{\nu},$$

sendo $\text{tor}(K)$ a torção de K e $\boldsymbol{\nu}$ a bi-normal de K . Logo

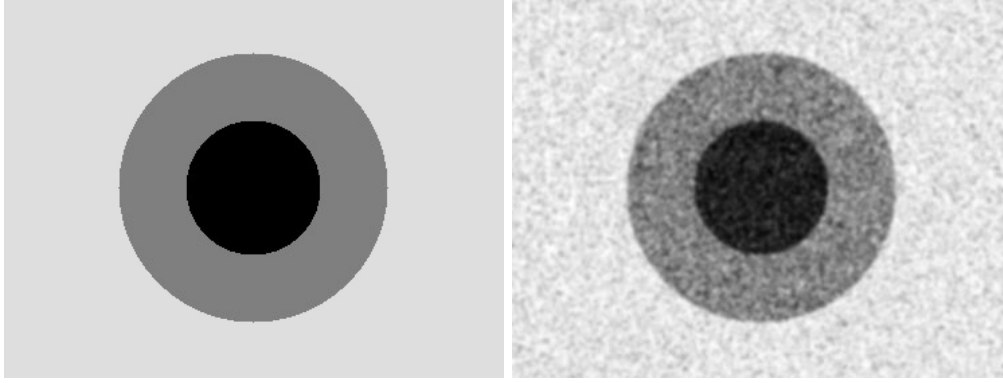
$$\frac{\partial \mathbf{n}}{\partial \mathbf{t}} = (\nabla_s \mathbf{n}) \mathbf{t} = \text{curv}(K) \mathbf{t} + \text{tor}(K) \boldsymbol{\nu},$$

mas como a curva K é plana, a sua torção é nula. Logo

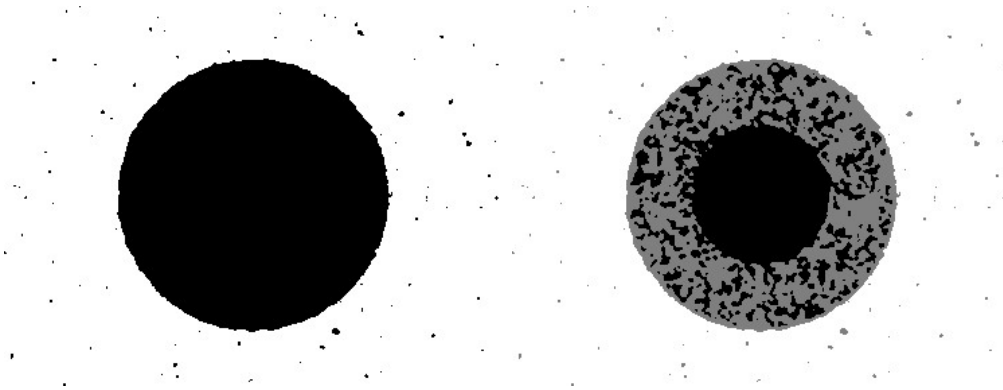
$$(\nabla_s \mathbf{n}) \mathbf{t} = \text{curv}(K) \mathbf{t} \Rightarrow (\nabla_s \mathbf{n}) \mathbf{t} \cdot \mathbf{t} = \text{curv}(K) \mathbf{t} \cdot \mathbf{t} = \text{curv}(K),$$

o que conclui a prova do teorema.

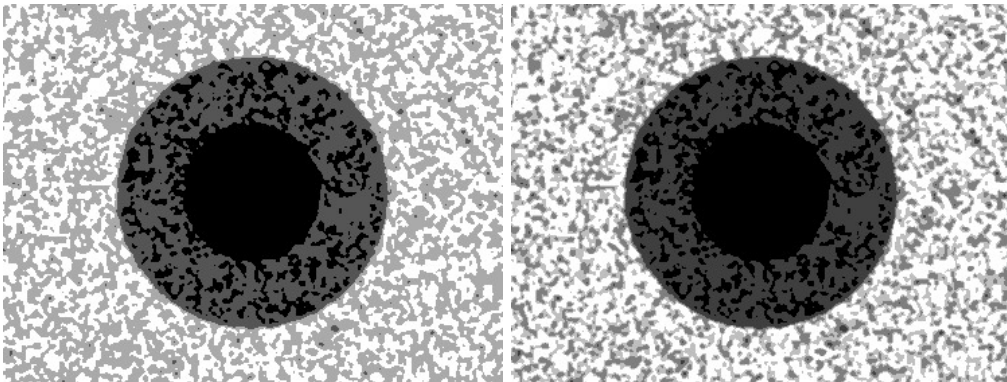
A unicidade da solução do funcional de Mumford-Shah ainda é um problema em aberto. Para mais detalhes neste assunto o leitor pode dirigir-se a [16], pgs. 133-146. Nesta referência, também são discutidas diferentes alternativas para aproximar o mínimo deste funcional. As mais utilizadas são discretizar o funcional utilizando um esquema de



(a) Imagens original e poluída com ruído Gaussiano branco



(b) Imagens segmentadas para 2 e 3 classes



(c) Imagens segmentadas para 4 e 5 classes

Figura 5.1: Resultado da segmentação por união de regiões usando o funcional de Mumford e Shah reduzido como função custo. No caso são apresentados (de esquerda a direita e de cima para baixo) a imagem original e a mesma imagem poluída com ruído Gaussiano branco (com variância $\sigma^2 = 0.1$), na segunda fila o resultado para 2 e 3 classes, e na terceira fila o resultado para 4 e 5 classes.

diferenças ou elementos finitos [31].

Na Figura 5.1 são apresentados resultados experimentais ao utilizar o funcional de Mumford-Shah modificado, ou seja o modelo constante por partes. Neste caso somente dois termos do funcional são considerados, ou seja

$$\mathcal{F}(u, K) = \theta \int_{\Omega-K} (v_0 - u)^2 d\Omega + (1 - \theta) \int_K d\sigma,$$

que é o funcional reduzido de Mumford-Shah.

Neste caso a estratégia utilizada para segmentar a imagem usando o funcional de Mumford-Shah é relativamente simples. A mesma pertence ao conjunto das chamadas “união de regiões” (*region-merging*). A mesma consiste nos seguintes passos:

- Seja u_0 uma segmentação $\Omega^0 = \bigcup_{i=1}^M (\Omega_i^0 \cup K^0)$ inicial de uma imagem v_0 dada. O mínimo correspondente, u^* de $\mathcal{F}(u, K)$, é uma função constante por partes, onde as constantes são os valores médios de v_0 sobre cada região Ω_i^0 .
- Unindo as diferentes regiões iterativamente sempre que a união diminua o funcional \mathcal{F}^3 , o parâmetro θ permite controlar o peso de cada termo na equação.

Como os resultados obtidos com este algoritmo simples não apresentaram boa qualidade respeito dos obtidos com outros métodos, os mesmos não foram utilizados nas comparações apresentadas no Capítulo 7.

³Pelo fato de se unir regiões de diferentes intensidades (substituindo as intensidades correspondentes por valores médios dentro de uma mesma região), necessariamente o valor da função custo sempre cresce. Neste caso as uniões são feitas da maneira que produzam o menor aumento da função custo em cada iteração.

Apêndice Capítulo 5

5.A Medida de Hausdorff

Sejam $k \in [0, +\infty]$ e $A \subset \mathbb{R}^N$. A medida k -dimensional de Hausdorff de A é dada por

$$\mathcal{H}^k(A) = \lim_{\delta \rightarrow 0} \mathcal{H}_\delta^k(A), \quad (5.6)$$

onde para $0 < \delta \leq \infty$, $\mathcal{H}_\delta^k(A)$ é definido como

$$\mathcal{H}_\delta^k(A) = \frac{w_k}{2^k} \left\{ \sum_{i \in I} |\text{diam}(A_i)|^k, \text{diam}(A_i) \leq \delta, A \subset \cup_{i \in I} A_i \right\} \quad (5.7)$$

para coberturas $(A_i)_{i \in I}$ finitas ou contáveis; $\text{diam}(A_i)$ denota o diâmetro do conjunto A_i , e w_k é um fator de normalização igual a $\pi^{k/2} \Gamma(1 + k/2)$ onde $\Gamma(t) = \int_0^\infty s^{t-1} e^{-s} ds$ é a função gamma (w_k coincide com a medida de Lebesgue da bola unitária de \mathbb{R}^k se $k \geq 1$ é um inteiro). Define-se também a dimensão de Hausdorff de A como

$$\mathcal{H} - \dim(A) = \inf \left\{ k \geq 0; \mathcal{H}^k(A) = 0 \right\}. \quad (5.8)$$

Um exemplo de medida de Hausdorff infinita é o caso dos fractais.

Capítulo 6

Derivada Topológica em Segmentação de Imagens

No Capítulo 5 foi estudado o funcional de Mumford-Shah. Este funcional representa a primeira abordagem em segmentação de imagens como um problema de minimização do ponto de vista do Cálculo Variacional e EDP's, sendo amplamente estudado na literatura desde a sua introdução.

A seguir é proposta uma técnica de segmentação de imagens inovadora baseada em uma nova forma de enxergar este problema de minimização, utilizando para isto a expansão assintótica topológica (apresentada no Capítulo 1). Nesta abordagem, é calculada a sensibilidade a pequenas perturbações de uma medida de performance ou função custo associada a uma segmentação específica da imagem tratada. A função custo utilizada foi motivada pelo funcional de Mumford-Shah mas com algumas modificações. Neste caso, é calculada a derivada topológica para uma perturbação no termo fonte do problema. Esta abordagem será designada *contínua* já que nenhuma aproximação é realizada nas funções envolvidas. Em particular, a derivada topológica é utilizada como um critério de descida para minimizar a função custo, obtendo-se assim um novo algoritmo de segmentação de imagens. Este novo algoritmo baseia-se na solução de uma equação variacional na qual o termo de fonte é perturbado sucessivamente em um processo iterativo de maneira que o valor da função custo sempre diminua.

Posteriormente, um segundo método de segmentação é proposto baseado em uma simplificação da alternativa contínua. Em poucas palavras, um caso particular do funcional contínuo é discretizado e um novo termo é introduzido. Como o funcional é discreto, é possível calcular a sua variação total frente às perturbações na imagem segmentada. Esta variação total pode ser identificada como a derivada topológica da função custo discreta. Esta informação é utilizada para propor um novo algoritmo de segmentação de simples implementação, robusto e rápido. Finalmente, como estes métodos dependem da escolha das classes, é proposto também um algoritmo de otimização dos valores das classes.

6.1 Abordagem Contínua

Em geral, uma imagem pode ser caracterizada por uma matriz bidimensional de *pixels* ou tridimensional de *voxels*. A seguir este elemento básico da imagem (pixel/voxel) será chamado de *elemento da imagem*, tendo a cada um deles uma intensidade associada. Logo, a imagem original pode ser descrita por uma função real v_0 que é constante em cada elemento da imagem

$$v_0 \in \mathcal{V} = \{w \in L^2(\Omega) : w \text{ constante a nível de elemento da imagem}\} \quad (6.1)$$

onde Ω é um domínio aberto limitado em $\mathbb{R}^n, n = 2, 3$. Considera-se também o conjunto de classes \mathcal{C} :

$$\mathcal{C} = \{c_i \in \mathbb{R} : i = 1, \dots, Nc\}, \quad (6.2)$$

onde Nc é o número de classes predefinidas nas quais a imagem original v_0 será segmentada e c_i representa a intensidade que caracteriza a $i^{esima} - classe$.

Conseqüentemente, o problema de segmentação de imagens pode ser colocado como: Dada uma imagem $v_0 \in \mathcal{V}$ encontre a imagem segmentada $u^* \in \mathcal{U}$ que minimize o funcional $\mathcal{J} : \mathcal{U} \mapsto \mathbb{R}$ associado ao custo de uma imagem segmentada específica e sendo \mathcal{U} definido como

$$\mathcal{U} = \{u \in \mathcal{V} : u(\mathbf{x}) \in \mathcal{C}, \forall \mathbf{x} \in \Omega\}. \quad (6.3)$$

Tendo como motivação o funcional de Mumford-Shah e trabalhos anteriores em processamento de imagens [18, 22, 110, 113, 114], a seguinte função custo \mathcal{J} associada a imagem segmentada $u \in \mathcal{U}$ é adotada

$$\mathcal{J}(u) = \frac{1}{2} \int_{\Omega} \mathbb{K} \nabla \varphi \cdot \nabla \varphi \, d\Omega + \frac{1}{2} \int_{\Omega} (\varphi - (v_0 - u))^2 \, d\Omega, \quad (6.4)$$

onde φ é a solução do seguinte problema variacional: Encontre $\varphi \in H^1(\Omega)$, tal que

$$a(\varphi, \eta) = l(\eta) \quad \forall \eta \in H^1(\Omega), \quad (6.5)$$

com a forma bilinear $a(\cdot, \cdot) : H^1(\Omega) \times H^1(\Omega) \rightarrow \mathbb{R}$ e a forma linear $l(\cdot) : L^2(\Omega) \rightarrow \mathbb{R}$ definidas como

$$a(\varphi, \eta) := \int_{\Omega} \mathbb{K} \nabla \varphi \cdot \nabla \eta \, d\Omega + \int_{\Omega} \varphi \eta \, d\Omega \quad \text{e} \quad l(\eta) := \beta \int_{\Omega} (v_0 - u) \eta \, d\Omega. \quad (6.6)$$

O parâmetro β deve ser escolhido experimentalmente e o campo tensorial de difusividade de segunda ordem \mathbb{K} é constante a nível de cada elemento da imagem. Além disso, estas formas satisfazem

$$|a(\varphi, \eta)| \leq M \|\varphi\|_{H^1(\Omega)} \|\eta\|_{H^1(\Omega)}, \quad (6.7)$$

$$a(\varphi, \varphi) \geq m \|\varphi\|_{H^1(\Omega)}^2, \quad (6.8)$$

$$|l(\eta)| \leq \|v_0 - u\|_{L^2(\Omega)} \|\eta\|_{H^1(\Omega)}. \quad (6.9)$$

assegurando, pelo teorema de Lax-Milgram, a existência e unicidade da solução φ do problema variacional dado pela Eq. (6.5).

6.1.1 Problema Perturbado

Associado a φ é definida a função φ_ϵ solução do problema variacional perturbado. A perturbação é caracterizada por trocar a imagem segmentada u por uma nova u_T que é idêntica a u em todos os pontos do domínio Ω exceto em uma pequena região ω_ϵ centrada no ponto $\hat{\mathbf{x}} \in \Omega$, sendo $\Omega_\epsilon = \Omega \setminus \overline{\omega_\epsilon}$ e $\partial\Omega_\epsilon = \partial\Omega \cup \partial\omega_\epsilon$. Em ω_ϵ , u_T assume um dos valores

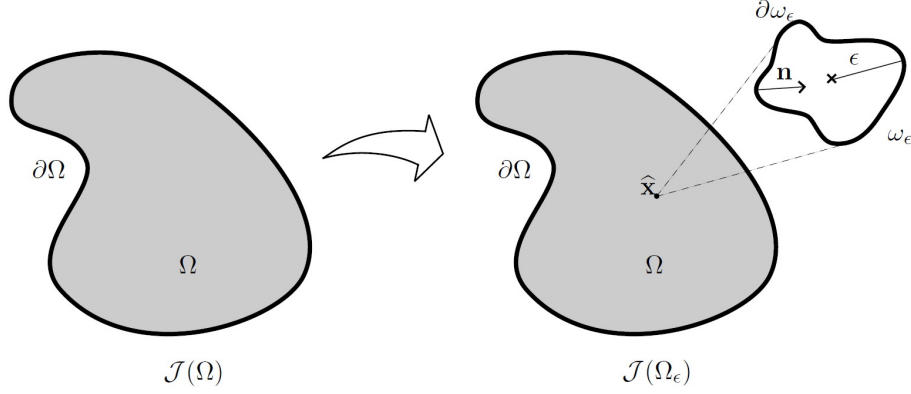


Figura 6.1: Conceito da derivada topológica.

$c_i \in \mathcal{C}$. Formalmente, $u_T(\mathbf{x}) = u(\mathbf{x}) \forall \mathbf{x} \in \Omega_\epsilon$ e $u_T(\mathbf{x}) = c_i, c_i \in \mathcal{C} \forall \mathbf{x} \in \omega_\epsilon$. Assim, a função custo perturbada se torna

$$\mathcal{J}_\epsilon(u_T) = \frac{1}{2} \int_{\Omega} \mathbb{K} \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon d\Omega + \frac{1}{2} \int_{\Omega} (\varphi_\epsilon - (v_0 - u_T))^2 d\Omega, \quad (6.10)$$

onde o campo φ_ϵ é a solução do problema variacional perturbado: Achar $\varphi_\epsilon \in H^1(\Omega)$ tal que

$$a_\epsilon(\varphi_\epsilon, \eta) = l_\epsilon(\eta) \quad \forall \eta \in H^1(\Omega), \quad (6.11)$$

com $l_\epsilon(\cdot) : L^2(\Omega) \rightarrow \mathbb{R}$, definido como

$$\begin{aligned} l_\epsilon(\eta) &= \beta \int_{\Omega} (v_0 - u_T) \eta d\Omega \\ &= \beta \int_{\Omega_\epsilon} (v_0 - u) \eta d\Omega + \beta \int_{\omega_\epsilon} (v_0 - c_i) \eta d\Omega, \quad c_i \in \mathcal{C} \end{aligned} \quad (6.12)$$

satisfaz as mesmas propriedades dadas pelas Eqs. (6.7)-(6.9). Além disso, destas propriedades, a seguinte estimativa é verdadeira (ver Apêndice 6.A para maiores detalhes)

$$\| \varphi_\epsilon - \varphi \|_{H^1(\Omega)} \leq C |\omega_\epsilon|^{1/2} \quad (6.13)$$

onde C é uma constante independente de ϵ e $|\omega_\epsilon|$ é a medida (área) de ω_ϵ .

6.1.2 Cálculo da Derivada Topológica

Para calcular a derivada topológica deste problema, a Eq. (1.12) toma a forma

$$\left. \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \right|_{\tau=0} = \left[\frac{\partial \mathcal{J}_\tau(u_T)}{\partial \tau} + \frac{\partial a_\tau(v, \eta)}{\partial \tau} - \frac{\partial l_\tau(\eta)}{\partial \tau} \right]_{\substack{v=\varphi_\tau \\ \eta=q_\tau}} \bigg|_{\tau=0}.$$

onde φ_ϵ é a solução da *equação de estado* (Eq. (6.11)) e q_ϵ é a solução da *equação adjunta* dada por: Achar $q_\epsilon \in H^1(\Omega)$, tal que

$$a_\epsilon(q_\epsilon, \eta) = - \left\langle \frac{\partial \mathcal{J}_\epsilon(u_T)}{\partial \varphi_\epsilon}, \eta \right\rangle = \frac{1-\beta}{\beta} l_\epsilon(\eta) \quad \forall \eta \in H^1(\Omega), \quad (6.14)$$

logo

$$q_\epsilon = \frac{1-\beta}{\beta} \varphi_\epsilon. \quad (6.15)$$

Tem-se assim

- a contribuição da função custo $\mathcal{J}_\epsilon(u_T)$

$$\begin{aligned} \left. \frac{\partial \mathcal{J}_\tau(u_T)}{\partial \tau} \right|_{\substack{v=\varphi_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} &= \frac{1}{2} \int_{\Omega} (\varphi_\epsilon - (v_0 - u_T))^2 \mathbb{I} \cdot \nabla \mathbf{v} \, d\Omega + \frac{1}{2} \int_{\Omega} (\mathbb{K} \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon) \mathbb{I} \cdot \nabla \mathbf{v} \, d\Omega \\ &\quad - \int_{\Omega} (\mathbb{K} \nabla \varphi_\epsilon \otimes \nabla \varphi_\epsilon) \cdot \nabla \mathbf{v} \, d\Omega, \end{aligned} \quad (6.16)$$

- a contribuição da forma bilinear $a_\epsilon(\cdot, \cdot)$

$$\begin{aligned} \left. \frac{\partial a_\tau(v, \eta)}{\partial \tau} \right|_{\substack{v=\varphi_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} &= \int_{\Omega} (\mathbb{K} \nabla \varphi_\epsilon \cdot \nabla q_\epsilon) \mathbb{I} \cdot \nabla \mathbf{v} \, d\Omega + \int_{\Omega} (\varphi_\epsilon q_\epsilon) \mathbb{I} \cdot \nabla \mathbf{v} \, d\Omega \\ &\quad - \int_{\Omega} (\mathbb{K} \nabla \varphi_\epsilon \otimes \nabla q_\epsilon + \mathbb{K} \nabla q_\epsilon \otimes \nabla \varphi_\epsilon) \cdot \nabla \mathbf{v} \, d\Omega, \end{aligned} \quad (6.17)$$

- a contribuição do funcional linear $l_\epsilon(\cdot)$

$$\left. \frac{\partial l_\tau(\eta)}{\partial \tau} \right|_{\substack{v=\varphi_\tau \\ \eta=q_\tau}} \Big|_{\tau=0} = \beta \int_{\Omega} ((v_0 - u_T) q_\epsilon) \mathbb{I} \cdot \nabla \mathbf{v} \, d\Omega, \quad (6.18)$$

onde \mathbf{v} se expande de forma contínua em Ω , e como a perturbação (mudança de forma) sobre $\partial\Omega_\epsilon$ somente é permitida em $\partial\omega_\epsilon$, obtém-se

$$\mathbf{v}(\mathbf{x}) = 0 \quad \forall \mathbf{x} \in \partial\Omega. \quad (6.19)$$

Dos resultados mencionados, a derivada a mudança de forma da função custo pode ser escrita como

$$\left. \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \right|_{\tau=0} = \int_{\Omega} \boldsymbol{\Sigma}_\epsilon \cdot \nabla \mathbf{v} \, d\Omega, \quad (6.20)$$

onde $\boldsymbol{\Sigma}_\epsilon$ pode ser interpretada como uma generalização do tensor momento-energia de Eshelby [54] que, para o problema considerado, é dado por

$$\begin{aligned} \boldsymbol{\Sigma}_\epsilon &= \frac{1}{2} \left[(\varphi_\epsilon - (v_0 - u_T))^2 + \mathbb{K} \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon + 2(\mathbb{K} \nabla \varphi_\epsilon \cdot \nabla q_\epsilon + \varphi_\epsilon q_\epsilon) - 2\beta(v_0 - u_T) q_\epsilon \right] \mathbb{I} \\ &\quad - \mathbb{K} \left[\nabla \varphi_\epsilon \otimes \nabla \varphi_\epsilon + \nabla \varphi_\epsilon \otimes \nabla q_\epsilon + \nabla q_\epsilon \otimes \nabla \varphi_\epsilon \right]. \end{aligned} \quad (6.21)$$

Considerando a relação tensorial

$$\operatorname{div}\left(\boldsymbol{\Sigma}_\epsilon^T \mathbf{v}\right) = \boldsymbol{\Sigma}_\epsilon \cdot \nabla \mathbf{v} + \operatorname{div} \boldsymbol{\Sigma}_\epsilon \cdot \mathbf{v}, \quad (6.22)$$

e da restrição sobre a campo de velocidade de mudança de forma dado pela Eq. (6.19), a derivada a mudança de forma dada pela Eq. (6.20) pode ser reescrita como

$$\left. \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \right|_{\tau=0} = \int_{\partial\omega_\epsilon} (\boldsymbol{\Sigma}_\epsilon^e - \boldsymbol{\Sigma}_\epsilon^i) \mathbf{n} \cdot \mathbf{v} \, d\partial\omega_\epsilon - \int_{\Omega} \operatorname{div} \boldsymbol{\Sigma}_\epsilon \cdot \mathbf{v} \, d\Omega. \quad (6.23)$$

Na expressão anterior, \mathbf{n} é a normal unitária ao longo de $\partial\Omega_\epsilon$ (ver Fig. 6.1). Além do mais, $\boldsymbol{\Sigma}_\epsilon^e$ e $\boldsymbol{\Sigma}_\epsilon^i$ representam o tensor de Eshelby generalizado em $\partial\omega_\epsilon$ vindo de Ω_ϵ e de ω_ϵ , respectivamente. Também, é fácil verificar que o tensor de Eshelby generalizado tem divergência nula, i.e., $\operatorname{div} \boldsymbol{\Sigma}_\epsilon = \mathbf{0}$ (ver Apêndice 6.B). Conseqüentemente, a derivada a mudança de forma da função custo $\mathcal{J}_\epsilon(u_T)$ transforma-se em uma integral definida no contorno $\partial\omega_\epsilon$, i.e.,

$$\left. \frac{d\mathcal{J}(\Omega_\tau)}{d\tau} \right|_{\tau=0} = \int_{\partial\omega_\epsilon} (\boldsymbol{\Sigma}_\epsilon^e - \boldsymbol{\Sigma}_\epsilon^i) \mathbf{n} \cdot \mathbf{v} \, d\partial\omega_\epsilon. \quad (6.24)$$

Em outras palavras, a derivada a mudança de forma deste problema somente depende dos valores dos campos envolvidos no contorno $\partial\omega_\epsilon$. Logo, se o domínio $\partial\omega_\epsilon$ é submetido a uma perturbação dada pela seguinte expansão uniforme

$$\mathbf{v} = -\mathbf{n} \text{ em } \partial\omega_\epsilon \quad (6.25)$$

e da Eq. (1.5), a derivada topológica é dada apenas por uma integral na fronteira da bola $\partial\omega_\epsilon$, i.e.,

$$D_T(\hat{\mathbf{x}}) = -\lim_{\epsilon \rightarrow 0} \frac{1}{f'(\epsilon)} \int_{\partial\omega_\epsilon} (\boldsymbol{\Sigma}_\epsilon^e - \boldsymbol{\Sigma}_\epsilon^i) \mathbf{n} \cdot \mathbf{n}. \quad (6.26)$$

Como todos os campos φ_ϵ , $\nabla \varphi_\epsilon$, q_ϵ , ∇q_ϵ e \mathbb{K} são contínuos em $\partial\omega_\epsilon$, obtém-se

$$\begin{aligned} 2 (\boldsymbol{\Sigma}_\epsilon^e - \boldsymbol{\Sigma}_\epsilon^i) \mathbf{n} \cdot \mathbf{n} &= \left[(\varphi_\epsilon - (v_0 - u_T))^2 - 2 \beta (v_0 - u_T) q_\epsilon \right]^e \\ &\quad - \left[(\varphi_\epsilon - (v_0 - u_T))^2 - 2 \beta (v_0 - u_T) q_\epsilon \right]^i \\ &= \left[(\varphi_\epsilon - (v_0 - u))^2 + 2 \beta u q_\epsilon \right] \\ &\quad - \left[(\varphi_\epsilon - (v_0 - c_i))^2 + 2 \beta c_i q_\epsilon \right] \\ &= (u - c_i) \left[(\varphi_\epsilon - (v_0 - u)) + (\varphi_\epsilon - (v_0 - c_i)) + 2 \beta q_\epsilon \right] \end{aligned} \quad (6.27)$$

onde $c_i \in \mathcal{C}$, $i = 1, \dots, Nc$. Logo, a Eq. (6.26) pode se reescrita como

$$D_T(\hat{\mathbf{x}}) = \frac{1}{2} \lim_{\epsilon \rightarrow 0} \frac{1}{f'(\epsilon)} \int_{\partial\omega_\epsilon} (c_i - u) \left[(\varphi_\epsilon - (v_0 - u)) + (\varphi_\epsilon - (v_0 - c_i)) + 2\beta q_\epsilon \right]. \quad (6.28)$$

Agora, considerando a estimativa dada pela Eq. (6.13), $f(\epsilon)$ pode ser escolhida como sendo

$$f(\epsilon) = \pi \epsilon^2,$$

e devido a continuidade de φ_ϵ e q_ϵ , o teorema da localização pode ser utilizado na Eq. (6.28). Logo, a derivada topológica é dada por

$$\begin{aligned} D_T(\hat{\mathbf{x}}) &= \frac{1}{2}(c_i - u) \left[(\varphi(\hat{\mathbf{x}}) - (v_0 - u)) + (\varphi(\hat{\mathbf{x}}) - (v_0 - c_i)) \right. \\ &\quad \left. + 2(1 - \beta) \varphi(\hat{\mathbf{x}}) \right] \quad \forall \hat{\mathbf{x}} \in \Omega, \end{aligned} \quad (6.29)$$

lembrando que $q_\epsilon = \frac{1-\beta}{\beta} \varphi_\epsilon$.

Da Eq. (6.29) observa-se que a derivada topológica em qualquer ponto $\hat{\mathbf{x}} \in \Omega$ somente depende do resultado naquele ponto da função φ solução do problema variacional dado pela Eq. (6.5) definido no domínio **não perturbado** Ω , da imagem de entrada v_0 , da imagem segmentada u e da perturbação dada por uma das intensidades que caracterizam as Nc classes $c_i \in \mathcal{C}$ nas quais a imagem v_0 será segmentada. Além disso, da Eq. (1.1) e como $f(\epsilon)$ é positiva, introduzindo perturbações em qualquer ponto $\hat{\mathbf{x}}$ onde $D_T(\hat{\mathbf{x}})$ seja negativa o valor da função custo será diminuído. Logo, a derivada topológica pode ser vista como uma função indicadora dos pontos mais adequados onde as perturbações devem ser introduzidas.

Como a solução φ do problema variacional dado pela Eq. (6.5) não pode, em geral, ser conhecido explicitamente é necessário achar uma solução aproximada. Neste sentido, o Método dos Elementos Finitos [93] será adotado nos experimentos numéricos apresentados a seguir. Logo, utilizando o elemento finito mais simples dado por um quadrilátero bilinear (para o caso de uma imagem bidimensional) ou por um paralelepípedo trilinear (para o caso tridimensional) com os pontos nodais coincidentes com os centros dos elementos da imagem, uma solução aproximada φ^h de φ pode ser facilmente obtida para qualquer $v_0 \in \mathcal{V}$ e $u \in \mathcal{U}$ (ver Seção A.2). Utilizando esta solução, uma aproximação de elementos finitos da derivada topológica toma a forma

$$\begin{aligned} D_T^h(\hat{\mathbf{x}}) &= \frac{1}{2}(c_i - u^h) \left[(\varphi^h(\hat{\mathbf{x}}) - (v_0^h - u^h)) + (\varphi^h(\hat{\mathbf{x}}) - (v_0^h - c_i)) \right. \\ &\quad \left. + 2(1 - \beta) \varphi^h(\hat{\mathbf{x}}) \right] \quad \forall \hat{\mathbf{x}} \in \Omega, \end{aligned} \quad (6.30)$$

onde v_0^h e u^h são as interpolações de elementos finitos no ponto $\hat{\mathbf{x}}$ das funções v_0 e u respectivamente. Além disso, considerando que a derivada topológica depende de c_i , denota-se \hat{c}_i a classe c_i que minimiza $D_T^h(\hat{\mathbf{x}})$, que será denotada por sua vez como $\widehat{D}_T^h(\hat{\mathbf{x}})$.

Como mencionado anteriormente, para uma imagem $v_0 \in \mathcal{V}$ deve-se achar a imagem segmentada $u^* \in \mathcal{U}$ que minimiza a função custo \mathcal{J} ao se escolher sucessivamente a classe que produz o valor mais negativo da derivada topológica. Assim, propõe-se o seguinte algoritmo de segmentação de imagens (Algoritmo 4) baseado na derivada topológica e chamado de agora em diante S_{D_T} -Continuous.

Algoritmo 4 S_{D_T} -Continuous um método de segmentação de imagens baseado na derivada topológica

Entrada: Uma imagem de entrada $v_0 \in \mathcal{V}$, o conjunto \mathcal{C} , uma segmentação inicial qualquer $u \in \mathcal{U}$, o campo tensorial de difusividade \mathbb{K} e os parâmetros β e $\alpha \in (0, 1)$.

Saída: A imagem segmentada $u^* \in \mathcal{U}$.

```

while  $D_{TMIN}^h < 0$  do
    achar a solução do problema variacional (6.5) para obter  $\varphi^h$ 
    calcular  $\widehat{D}_T^h$  e  $\widehat{c}_i$  no centro de cada elemento da imagem  $s$  (pontos nodais dos elementos finitos)
    avaliar  $D_{TMIN} = \min_s \{\widehat{D}_T^h(s), \widehat{D}_T^h(s) < 0\}$ 
    em cada elemento da imagem  $s$  escolher  $u(s) = \widehat{c}_i$  se  $\widehat{D}_T^h(s) \leq \alpha D_{TMIN}$ 
end while
 $u^* = u$ 
    
```

Neste ponto é interessante ressaltar que o tensor de difusividade \mathbb{K} em geral pode ser adotado como um tensor isotrópico homogêneo ($\mathbb{K} = k_0 \mathbb{I} \quad \forall \mathbf{x} \in \Omega$). No entanto, quando a remoção de ruído é feita utilizando algum método anisotrópico não linear [4, 27, 64] ou um método de restauração baseado na derivada topológica (como os apresentados no Capítulo 3, [18, 113]), o tensor \mathbb{K} pode ser escolhido igual ao tensor de difusividade fornecido por estes métodos.

6.2 Abordagem Discreta

Nesta seção é apresentado um segundo método de segmentação baseado em uma simplificação da idéia anterior. Como será apresentado, neste novo algoritmo não é necessário calcular o campo φ para obter a derivada topológica. De fato, considerando $\beta = 0$ na Eq. (6.5) obtém-se a solução trivial $\varphi \equiv 0$ para qualquer imagem segmentada $u \in \mathcal{U}$. Neste caso o funcional $\mathcal{J}(u)$ definido em \mathcal{U} se reduz a

$$\mathcal{J}(u) = \int_{\Omega} (v_0 - u)^2 d\Omega. \quad (6.31)$$

Além disso, como v_0 e u são constantes a nível dos elementos da imagem, o funcional pode ser reescrito como

$$\mathcal{J}^d(u) = \sum_s (v_0^s - u^s)^2, \quad (6.32)$$

onde v_0^s e u^s significam o valor de v_0 e u no elemento w^s da imagem respectivamente e $\overline{\Omega} = \bigcup_s w^s$. Segundo esta abordagem, o problema de segmentação de imagens pode ser reduzido a: Dada a imagem original $v_0 \in \mathcal{V}$, achar $u^* \in \mathcal{U}$ tal que

$$u^* := \arg \min_{u \in \mathcal{U}} \mathcal{J}^d(u). \quad (6.33)$$

O problema de minimização anterior pode ser facilmente resolvido. De fato, somente é necessário achar para cada elemento s da imagem o índice $\bar{i} := \arg \min_i |v_0^s - c_i|; i =$

$1, \dots, N_c; c_i \in \mathcal{C}$. Conseqüentemente, a segmentação u^* é caracterizada por $u^{*s} = c_{\bar{i}}$. Em outras palavras, a nível de cada elemento da imagem s , a imagem segmentada é obtida escolhendo o valor da classe $c_{\bar{i}} \in \mathcal{C}$ que mais próximo se encontra do valor v_0^s da imagem de entrada v_0 .

Como nesta formulação não existe controle na medida do contorno correspondendo ao subdomínio $\Omega_i = \{x \in \Omega; u(x) = c_i\}$ associado à classe $c_i \in \mathcal{C}, i = 1, \dots, N_c$, segmentações cujas regiões Ω_i têm bordas muito irregulares são permitidas. Logo, para obter resultados com bordas mais suaves e das idéias por traz do funcional do Mumford-Shah [133], o seguinte funcional é proposto:

$$\mathcal{F}^d(u) = \theta \mathcal{J}^d(u) + (1 - \theta) \mathcal{B}^d(u), \quad \text{com } \theta \in (0, 1] \subset \mathbb{R}, \quad (6.34)$$

onde o segundo termo $\mathcal{B}^d(u)$ é associado à medida da interface entre diferentes regiões. Em particular, este funcional é definido como

$$\mathcal{B}^d(u) = \frac{1}{4n} \sum_s \sum_p \chi(u^s, u^p). \quad (6.35)$$

Neste caso, $n = 2$ (3) para imagens 2D (3D) e $\chi(u^s, u^p)$ é uma função característica do contorno que o elemento da imagem s compartilha com o vizinho p e é definida como tomando o valor um quando $u^s \neq u^p$ e zero caso contrário. O parâmetro θ controla a contribuição (como uma combinação convexa) de cada termo ($\mathcal{J}^d(u)$ e $\mathcal{B}^d(u)$) na função custo $\mathcal{F}^d(u)$.

Logo, da definição do funcional $\mathcal{F}^d(u)$ dado pela Eq. (6.34), o custo associado a uma imagem segmentada $u \in \mathcal{U}$ é facilmente calculado. Além disso, ao se perturbar o valor de u em um determinado elemento da imagem w^s mudando o valor da sua classe de u^s para algum $c_i \in \mathcal{C}$ obtém-se uma nova imagem segmentada *perturbada* u_T . Logo, $\mathcal{F}^d(u_T)$ é dada por

$$\mathcal{F}^d(u_T) = \theta \mathcal{J}^d(u_T) + (1 - \theta) \mathcal{B}^d(u_T), \quad (6.36)$$

onde $\mathcal{J}^d(u_T)$ e $\mathcal{B}^d(u_T)$ podem ser reescritas como

$$\begin{aligned} \mathcal{J}^d(u_T) &= \mathcal{J}^d(u) - (v_0^s - u^s)^2 + (v_0^s - c_i)^2, \\ \mathcal{B}^d(u_T) &= \mathcal{B}^d(u) - \frac{1}{4n} \sum_p [\chi(u^s, u^p) - \chi(c_i, u^p)] \end{aligned} \quad (6.37)$$

dado que u_T é igual a u em todos os pontos exceto em um elemento s onde toma o valor $c_i \in \mathcal{C}$.

Das expressões acima descritas, a variação total do funcional \mathcal{F}^d devido a uma perturbação no elemento s da imagem será chamada $D_T(s)$ e é dada por

$$D_T(s) = \mathcal{F}^d(u_T) - \mathcal{F}^d(u), \quad (6.38)$$

onde

$$D_T(s) = \theta [(v_0^s - u_T^s)^2 - (v_0^s - u^s)^2] + (1 - \theta) \frac{1}{4n} \sum_p [\chi(u_T^s, u^p) - \chi(u^s, u^p)], \quad (6.39)$$

para $u_T^s = c_i, c_i \in \mathcal{C}, i = 1, \dots, N_c$.

Segue a continuação a variação total dada pela Eq. (6.38) reescrita como

$$\mathcal{F}^d(u_T) = \mathcal{F}^d(u) + D_T(s). \quad (6.40)$$

Observa-se, portanto, que esta expressão, quando comparada com a Eq. (1.1), pode ser interpretada como uma versão discreta da “expansão topológica” para a abordagem discreta. Além disso, em cada elemento da imagem s pode ser selecionada a perturbação u_T^s que produz o mínimo valor da variação total. O valor mínimo da variação total $D_T(s)$ e a classe correspondente c_i são denotadas $\widehat{D}_T(s)$ e \widehat{c}_i respectivamente. Logo, para uma abordagem discreta a variação total assume o mesmo papel que a derivada topológica no caso contínuo. De fato, a variação total D_T pode ser considerada como uma função indicadora dos elementos da imagem nos quais as perturbações devem ser introduzidas para reduzir o valor da função custo.

Usando estes resultados, um algoritmo de segmentação discreto, de baixíssimo custo computacional, é proposto. Como mencionado anteriormente, de acordo com a expansão topológica (Eq. (6.40)), para uma imagem $v_0 \in \mathcal{V}$ deve-se encontrar uma imagem segmentada $u^* \in \mathcal{U}$ que minimize a função custo \mathcal{F}^d selecionando sucessivamente a classe que produz o valor mais negativo da variação total. Assim, propõe-se o seguinte algoritmo de segmentação de imagens (Algoritmo 5) chamado de agora em diante S_{D_T} -Discrete.

Algoritmo 5 S_{D_T} -Discrete um método de segmentação de imagens baseado na derivada topológica discreta

Entrada: Uma imagem de entrada $v_0 \in \mathcal{V}$, o conjunto de classes \mathcal{C} , uma segmentação inicial $u \in \mathcal{U}$ e os parâmetros θ e $\alpha \in (0, 1)$.

Saída: A imagem segmentada $u^* \in \mathcal{U}$.

normalizar a imagem v_0 e os valores das classes ao intervalo $[0; 1]$

while $D_{TMIN} < 0$ **do**

 calcular $\widehat{D}_T(s)$ e \widehat{c}_i em cada elemento s da imagem

 avaliar $D_{TMIN} = \min_s \{\widehat{D}_T(s); \widehat{D}_T(s) < 0\}$

 em cada elemento s da imagem escolher $u^s = \widehat{c}_i$ se $\widehat{D}_T(s) \leq \alpha D_{TMIN}$

end while

$u^* = u$

Neste ponto é interessante ressaltar que nos dois algoritmos de segmentação de imagens foi adotado o critério de parada $D_{TMIN} \geq 0$, no entanto outros podem ser considerados. Por exemplo, um critério associado ao comportamento da função custo, e.g., se em duas iterações consecutivas a função custo decresce menos que uma determinada tolerância, o algoritmo é detido. Também, nos dois algoritmos é utilizada a condição

$\overline{D_T}(s) \leq \alpha D_{TMIN}$ para determinar os elementos nos quais o valor da classe será modificado. Uma outra abordagem pode ser enunciada como: seja L_{D_T} a lista de todos os elementos s da imagem com derivada topológica negativa ordenados por este valor, ou seja

$$L_{D_T}(N) = \{s_1, s_2, \dots, s_N\}, \quad (6.41)$$

onde $D_T(s_1) \leq D_T(s_2) \leq \dots \leq D_T(s_N) < 0$ e N é o número de elementos na imagem. Logo, a nova estratégia consiste em modificar o valor dos elementos da imagem segmentada u^s somente para $s \in L_{D_T}(\alpha N)$, sendo os αN primeiros elementos da lista $L_{D_T}(N)$.

É importante observar que, embora $\alpha \in [0, 1]$, este parâmetro tem significado diferente em ambos casos. De fato, no primeiro caso αD_{TMIN} para $\alpha = 0$ significa perturbar todos os elementos da imagem com derivada topológica negativa. Já utilizando a alternativa descrita pela Equação (6.41), este será o caso para $\alpha = 1$, indicando que todos os elementos da imagem com derivada topológica negativa devem ser perturbados.

6.3 Otimização dos Valores das Classes

É fácil perceber que, tanto no caso contínuo como no discreto, o resultado depende dos valores utilizados para caracterizar as classes (sejam, $c_i \in \mathcal{C}$). Quando informação precisa destes valores está disponível, espera-se uma segmentação mais precisa. Por outro lado, quando esta informação não se encontra disponível, o resultado pode ser afetado por erros na estimação destes valores. Para resolver este problema, uma técnica muito simples (baseada em idéias apresentadas em [18]) é proposta para ajustar os valores das classes.

A idéia aqui é ajustar os valores das classes em cada iteração. Para isto, no final de cada iteração do algoritmo de segmentação, os valores das classes são reajustados. O algoritmo proposto é apresentado em Algoritmo 6.

Algoritmo 6 Otimização do valor das classes

Entrada: Uma imagem de entrada $v_0 \in \mathcal{V}$, o conjunto \mathcal{C} e a imagem segmentada na iteração i , $u \in \mathcal{U}$.

Saída: O novo conjunto de classes \mathcal{C}^* .

```

 $\mathcal{C}^* = [ ]$ 
for  $c \in \mathcal{C}$  do
  for  $i = -1$  a  $1$  do
    escolher  $\mathcal{C}^i = (\mathcal{C} - \{c\}) \cup \{c + i\}$ 
    criar  $u^i$  fazendo em  $u$  a substituição  $\mathcal{C} \rightarrow \mathcal{C}^i$ 
    calcular  $j^i = \mathcal{J}(u^i)$ 
  end for
   $c^{min} = c + i$  sendo  $i = \min_i(j^i)$ 
   $\mathcal{C}^* = \mathcal{C}^* \cup c^{min}$ 
end for

```

Esta técnica simples melhora consideravelmente os resultados da segmentação como será mostrado na Seção 7.3 e pode ser igualmente aplicada para S_{D_T} -Discrete e S_{D_T} -

Continuous. Os novos algoritmos que incorporam esta modificação serão chamados S_{DT} -Discrete- OP e S_{DT} -Continuous- OP respectivamente.

Apêndice Capítulo 6

6.A Análise Assintótica para uma Perturbação na Fonte

Nesta seção é feita uma análise assintótica para o problema apresentado na Seção 6.1 considerando $\mathbb{K} = k\mathbb{I}$.

6.A.1 Formulação do problema associado ao domínio original

Seja o problema direto associado ao domínio original, definido como

- Achar $\varphi \in H^1(\Omega)$, tal que

$$a(\varphi, \eta) = l(\eta) \quad \forall \eta \in H^1(\Omega) \quad (6.42)$$

onde

$$\begin{aligned} a(\varphi, \eta) &= \int_{\Omega} k \nabla \varphi \cdot \nabla \eta \, d\Omega + \int_{\Omega} \varphi \eta \, d\Omega \\ l(\eta) &= \beta \int_{\Omega} (v_0 - u) \eta \, d\Omega \\ &= \beta \int_{\Omega \setminus \bar{\omega}_{\epsilon}} (v_0 - u) \eta \, d\Omega + \beta \int_{\omega_{\epsilon}} (v_0 - u) \eta \, d\Omega \end{aligned}$$

- Seja também $q \in H^1(\Omega)$, a solução da equação adjunta associada ao domínio original dada por

$$q = \frac{1 - \beta}{\beta} \varphi \quad (6.43)$$

6.A.2 Formulação do problema associado ao domínio perturbado

Seja o problema direto, associado ao domínio perturbado, definido como

- Achar $\varphi_{\epsilon} \in H^1(\Omega)$, tal que

$$a_{\epsilon}(\varphi_{\epsilon}, \eta) = l_{\epsilon}(\eta) \quad \forall \eta \in H^1(\Omega) \quad (6.44)$$

onde

$$\begin{aligned} a_{\epsilon}(\varphi_{\epsilon}, \eta) &= \int_{\Omega} k \nabla \varphi_{\epsilon} \cdot \nabla \eta \, d\Omega + \int_{\Omega} \varphi_{\epsilon} \eta \, d\Omega \\ l_{\epsilon}(\eta) &= \beta \int_{\Omega \setminus \bar{\omega}_{\epsilon}} (v_0 - u) \eta \, d\Omega + \beta \int_{\omega_{\epsilon}} (v_0 - u_T) \eta \, d\Omega \end{aligned}$$

- Seja também $q_{\epsilon} \in H^1(\Omega)$ a solução da equação adjunta associada ao domínio original

$$q_{\epsilon} = \frac{1 - \beta}{\beta} \varphi_{\epsilon} \quad (6.45)$$

6.A.3 Análise Assintótica

Teorema 5. *Considera-se a solução das equações de estado φ_ϵ , φ e adjunta q_ϵ , q , onde cada par φ , q e φ_ϵ , q_ϵ estão associadas aos problemas original (Eqs. (6.42), (6.43)) e perturbado (Eqs. (6.44), (6.45)) respectivamente. Logo, a seguinte estimativa é verdadeira*

$$\|\varphi_\epsilon - \varphi\|_{H^1(\Omega)} \leq C_1 |\omega_\epsilon|^{1/2} \quad e \quad \|q_\epsilon - q\|_{H^1(\Omega)} \leq C_2 |\omega_\epsilon|^{1/2},$$

onde as constante C_1 e C_2 são independentes do parâmetro ϵ e $|\omega_\epsilon|$ é a medida de Lebesgue da bola ω_ϵ .

Prova 3. *Considerando a diferença entre as equações variacionais associadas aos problemas perturbado (Eq. (6.44)) e original (Eq. (6.42)) respectivamente, obtém-se*

$$\begin{aligned} a_\epsilon(\varphi_\epsilon - \varphi, \eta) &= \beta \int_{\omega_\epsilon} (v_0 - u_T) \eta \, d\Omega - \beta \int_{\omega_\epsilon} (v_0 - u) \eta \, d\Omega \\ &= \beta \int_{\omega_\epsilon} (u - u_T) \eta \, d\Omega \quad \forall \eta \in H^1(\Omega). \end{aligned}$$

Considerando também $\eta = \varphi_\epsilon - \varphi$ obtém-se

$$a_\epsilon(\varphi_\epsilon - \varphi, \varphi_\epsilon - \varphi) = \beta \int_{\omega_\epsilon} (u - u_T)(\varphi_\epsilon - \varphi) \, d\Omega,$$

logo, da coercividade de $a_\epsilon(\cdot, \cdot)$ tem-se

$$m \|\varphi_\epsilon - \varphi\|_{H^1(\Omega)}^2 \leq a_\epsilon(\varphi_\epsilon - \varphi, \varphi_\epsilon - \varphi) = \beta \int_{\omega_\epsilon} (u - u_T)(\varphi_\epsilon - \varphi) \, d\Omega.$$

Considerando a desigualdade de Cauchy-Schwarz obtém-se

$$\begin{aligned} \|\varphi_\epsilon - \varphi\|_{H^1(\Omega)}^2 &\leq \frac{\beta}{m} \|u - u_T\|_{L^2(\omega_\epsilon)} \|\varphi_\epsilon - \varphi\|_{L^2(\omega_\epsilon)} \\ &\leq c_1 \frac{\beta}{m} \|u - u_T\|_{L^2(\omega_\epsilon)} \|\varphi_\epsilon - \varphi\|_{L^2(\Omega)} \\ &\leq c_2 \frac{\beta}{m} \|u - u_T\|_{L^2(\omega_\epsilon)} \|\varphi_\epsilon - \varphi\|_{H^1(\Omega)} \\ &\leq c_3 \frac{\beta}{m} \max_{\mathbf{x} \in \omega_\epsilon} |u - u_T| |\omega_\epsilon|^{1/2} \|\varphi_\epsilon - \varphi\|_{H^1(\Omega)}, \end{aligned}$$

sendo c_1 , c_2 e c_3 constantes independentes de ϵ . Logo

$$\|\varphi_\epsilon - \varphi\|_{H^1(\Omega)} \leq C_1 |\omega_\epsilon|^{1/2},$$

onde

$$C_1 = c_3 \frac{\beta}{m} |u(\hat{\mathbf{x}}) - u_T(\hat{\mathbf{x}})|.$$

lembrando que u e u_T são constantes em ω_ϵ .

Além disso, das Eqs. (6.43) e (6.45) tem-se

$$\|q_\epsilon - q\|_{H^1(\Omega)} \leq \frac{(1-\beta)}{m} \max_{\mathbf{x} \in \omega_\epsilon} |u - u_T| |\omega_\epsilon|^{1/2} = C_2 |\omega_\epsilon|^{1/2}.$$

Finalmente

$$\|q_\epsilon - q\|_{H^1(\Omega)} \leq C_2 |\omega_\epsilon|^{1/2},$$

e seguindo um argumento idêntico ao caso anterior, tem-se

$$C_2 = c_3 \frac{(1-\beta)}{m} |u(\hat{\mathbf{x}}) - u_T(\hat{\mathbf{x}})|,$$

o que conclui a prova.

6.B Tensor Momento-Energia de Eshelby

Proposição 2. No contexto de derivada topológica aplicada em segmentação de imagens apresentada anteriormente, o seguinte é verdadeiro

$$\operatorname{div}(\boldsymbol{\Sigma}_\epsilon) = 0.$$

Prova 4. Seja φ_ϵ a solução da equação de estado do problema perturbado e q_ϵ a correspondente da equação adjunta, logo

Equação de estado:

$$\begin{cases} -\operatorname{div}(k \nabla \varphi_\epsilon) + \varphi_\epsilon = \beta(v_0 - u) & \mathbf{x} \in \Omega \setminus \bar{\omega}_\epsilon \\ -\operatorname{div}(k \nabla \varphi_\epsilon) + \varphi_\epsilon = \beta(v_0 - u_T) & \mathbf{x} \in \omega_\epsilon \\ \frac{\partial \varphi_\epsilon}{\partial n} = 0 & \mathbf{x} \in \partial\Omega \end{cases}.$$

Equação adjunta:

$$q_\epsilon = \frac{1-\beta}{\beta} \varphi_\epsilon. \quad (6.46)$$

No problema sob consideração, o tensor de momento-energia de Eshelby é dado por

$$\begin{aligned} \boldsymbol{\Sigma}_\epsilon^e &= \frac{1}{2} \left(k \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon + (\varphi_\epsilon - (v_0 - u))^2 + 2(k \nabla \varphi_\epsilon \cdot \nabla q_\epsilon + \varphi_\epsilon q_\epsilon) - 2\beta(v_0 - u)q_\epsilon \right) \mathbb{I} \\ &- k(\nabla \varphi_\epsilon \otimes \nabla \varphi_\epsilon + \nabla \varphi_\epsilon \otimes \nabla q_\epsilon + \nabla q_\epsilon \otimes \nabla \varphi_\epsilon), \quad \forall \mathbf{x} \in \Omega \setminus \bar{\omega}_\epsilon \\ \boldsymbol{\Sigma}_\epsilon^i &= \frac{1}{2} \left(k \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon + (\varphi_\epsilon - (v_0 - u_T))^2 + 2(k \nabla \varphi_\epsilon \cdot \nabla q_\epsilon + \varphi_\epsilon q_\epsilon) - 2\beta(v_0 - u_T)q_\epsilon \right) \mathbb{I} \\ &- k(\nabla \varphi_\epsilon \otimes \nabla \varphi_\epsilon + \nabla \varphi_\epsilon \otimes \nabla q_\epsilon + \nabla q_\epsilon \otimes \nabla \varphi_\epsilon), \quad \forall \mathbf{x} \in \omega_\epsilon \end{aligned}$$

Considera-se primeiramente $\boldsymbol{\Sigma}_\epsilon^e$. Da Eq. (6.46) obtém-se

$$\begin{aligned} \boldsymbol{\Sigma}_\epsilon^e &= \frac{1}{2} \left(k \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon + (\varphi_\epsilon - (v_0 - u))^2 + 2 \frac{1-\beta}{\beta} (k \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon + \varphi_\epsilon \varphi_\epsilon) \right. \\ &- \left. 2(1-\beta)(v_0 - u)\varphi_\epsilon \right) \mathbb{I} - k \left(\nabla \varphi_\epsilon \otimes \nabla \varphi_\epsilon + 2 \frac{1-\beta}{\beta} \nabla \varphi_\epsilon \otimes \nabla \varphi_\epsilon \right). \end{aligned}$$

Da linearidade do operador $\text{div}(\cdot)$, pode-se calcular $\text{div}(\Sigma_\epsilon^e)$ termo a termo. Lembrando que

$$\text{div}(\varphi \mathbb{I}) = \varphi \text{div}(\mathbb{I}) + \mathbb{I} \nabla \varphi \quad e \quad \nabla(\mathbf{v} \cdot \mathbf{u}) = (\nabla \mathbf{v})^T \mathbf{u} + (\nabla \mathbf{u})^T \mathbf{v},$$

obtem-se

$$\begin{aligned} \text{div} \left(\frac{1}{2} (k \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon) \mathbb{I} \right) &= \frac{1}{2} \left(k(\nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon) \underbrace{\text{div}(\mathbb{I})}_{=0} \right) + \frac{1}{2} \nabla (k(\nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon)) \\ &= \frac{1}{2} (k(\nabla \nabla \varphi_\epsilon)^T \nabla \varphi_\epsilon + k(\nabla \nabla \varphi_\epsilon)^T \nabla \varphi_\epsilon) \\ &= k(\nabla \nabla \varphi_\epsilon)^T \nabla \varphi_\epsilon. \end{aligned}$$

Para o segundo termo, e considerando que as funções v e u são constantes a nível do elemento de imagem ω^s , tem-se que

$$\text{div} \left(\frac{1}{2} (\varphi_\epsilon - (v_0 - u))^2 \mathbb{I} \right) = (\varphi_\epsilon - (v_0 - u)) \nabla \varphi_\epsilon.$$

O terceiro e quarto termo de $\text{div}(\cdot)$ podem ser reescritos como

$$\text{div} \left(\frac{1-\beta}{\beta} (k \nabla \varphi_\epsilon \cdot \nabla \varphi_\epsilon + \varphi_\epsilon \varphi_\epsilon) \mathbb{I} \right) = 2 \frac{1-\beta}{\beta} (k(\nabla \nabla \varphi_\epsilon)^T \nabla \varphi_\epsilon + \varphi_\epsilon \nabla \varphi_\epsilon),$$

e

$$\text{div}(- (1-\beta)(v_0 - u) \varphi_\epsilon \mathbb{I}) = - (1-\beta)(v_0 - u) \nabla \varphi_\epsilon.$$

respectivamente. Operando de maneira semelhante nos termos restantes obtém-se

$$\text{div}(-k \nabla \varphi_\epsilon \otimes \nabla \varphi_\epsilon) = -\text{div}(k \nabla \varphi_\epsilon) \nabla \varphi_\epsilon - k(\nabla \nabla \varphi_\epsilon) \nabla \varphi_\epsilon \quad (6.47)$$

$$\text{div} \left(-2 \frac{1-\beta}{\beta} k \nabla \varphi_\epsilon \otimes \nabla \varphi_\epsilon \right) = -2 \frac{1-\beta}{\beta} \text{div}(k \nabla \varphi_\epsilon) \nabla \varphi_\epsilon - 2 \frac{1-\beta}{\beta} k(\nabla \nabla \varphi_\epsilon) \nabla \varphi_\epsilon.$$

Juntando todos os termos obtém-se

$$\begin{aligned} \text{div}(\Sigma_\epsilon^e) &= k(\nabla \nabla \varphi_\epsilon)^T \nabla \varphi_\epsilon + (\varphi_\epsilon - (v_0 - u)) \nabla \varphi_\epsilon + 2 \frac{1-\beta}{\beta} (k(\nabla \nabla \varphi_\epsilon)^T \nabla \varphi_\epsilon) \\ &+ 2 \frac{1-\beta}{\beta} \varphi_\epsilon \nabla \varphi_\epsilon - (1-\beta)(v_0 - u) \nabla \varphi_\epsilon - \text{div}(k \nabla \varphi_\epsilon) \nabla \varphi_\epsilon - k(\nabla \nabla \varphi_\epsilon) \nabla \varphi_\epsilon \\ &- 2 \frac{1-\beta}{\beta} \text{div}(k \nabla \varphi_\epsilon) \nabla \varphi_\epsilon - 2 \frac{1-\beta}{\beta} k(\nabla \nabla \varphi_\epsilon) \nabla \varphi_\epsilon. \end{aligned}$$

Como φ_ϵ é um campo escalar, o seguinte é verdadeiro

$$\nabla \nabla \varphi_\epsilon = (\nabla \nabla \varphi_\epsilon)^T. \quad (6.48)$$

Logo, da Eq. (6.48) pode se escrever

$$\begin{aligned}
 \operatorname{div}(\Sigma_\epsilon^e) &= \underbrace{k(\nabla\nabla\varphi_\epsilon)\nabla\varphi_\epsilon}_{(1)} + (\varphi_\epsilon - (v_0 - u))\nabla\varphi_\epsilon + \underbrace{2\frac{1-\beta}{\beta}(k(\nabla\nabla\varphi_\epsilon)\nabla\varphi_\epsilon)}_{(2)} \\
 &+ 2\frac{1-\beta}{\beta}\varphi_\epsilon\nabla\varphi_\epsilon - (1-\beta)(v_0 - u)\nabla\varphi_\epsilon - \operatorname{div}(k\nabla\varphi_\epsilon)\nabla\varphi_\epsilon - \underbrace{k(\nabla\nabla\varphi_\epsilon)\nabla\varphi_\epsilon}_{(1)} \\
 &- 2\frac{1-\beta}{\beta}\operatorname{div}(k\nabla\varphi_\epsilon)\nabla\varphi_\epsilon - \underbrace{2\frac{1-\beta}{\beta}(k(\nabla\nabla\varphi_\epsilon)\nabla\varphi_\epsilon)}_{(2)}. \tag{6.49}
 \end{aligned}$$

Como os termos indicados com (1) e (2) cancelam mutuamente, pode-se reescrever a Eq. (6.49) da seguinte forma

$$\begin{aligned}
 \operatorname{div}(\Sigma_\epsilon^e) &= (\varphi_\epsilon - (v_0 - u))\nabla\varphi_\epsilon \\
 &+ 2\frac{1-\beta}{\beta}\varphi_\epsilon\nabla\varphi_\epsilon - (1-\beta)(v_0 - u)\nabla\varphi_\epsilon - \operatorname{div}(k\nabla\varphi_\epsilon)\nabla\varphi_\epsilon \\
 &- 2\frac{1-\beta}{\beta}\operatorname{div}(k\nabla\varphi_\epsilon)\nabla\varphi_\epsilon.
 \end{aligned}$$

Somando e subtraindo $\beta(v_0 - u)$ obtém-se

$$\begin{aligned}
 \operatorname{div}(\Sigma_\epsilon^e) &= \underbrace{(\varphi_\epsilon - \beta(v_0 - u) - \operatorname{div}(k\nabla\varphi_\epsilon))}_{=0 \text{ equação de estado}} \nabla\varphi_\epsilon \\
 &+ \frac{\beta}{1-\beta} \underbrace{(\varphi_\epsilon - \beta(v_0 - u) - \operatorname{div}(k\nabla\varphi_\epsilon))}_{=0 \text{ equação de estado}} \nabla\varphi_\epsilon \\
 &+ \frac{\beta}{1-\beta} \underbrace{(\varphi_\epsilon - \beta(v_0 - u) - \operatorname{div}(k\nabla\varphi_\epsilon))}_{=0 \text{ equação de estado}} \nabla\varphi_\epsilon.
 \end{aligned}$$

Logo

$$\operatorname{div}(\Sigma_\epsilon^e) = 0.$$

A prova para $\operatorname{div}(\Sigma_\epsilon^i) = 0$ é equivalente, o que conclui a prova.

Capítulo 7

Análises e Comparações das Técnicas Propostas e Existentes

Neste capítulo são apresentados os resultados obtidos com as técnicas descritas no capítulo anterior. Também, os métodos propostos são quantitativamente comparados com outros já existentes. Esta comparação é feita utilizando dados sintéticos poluídos artificialmente com ruído. Posteriormente são apresentados alguns resultados para imagens médicas reais, mostrando a potencialidade destes métodos.

Na Seção 7.1 são feitas comparações entre as técnicas propostas e outras amplamente utilizadas, onde 4 diferentes índices de qualidade de segmentação são utilizados para quantificar os resultados. Como é sabido, no processo de remoção de ruído, o suavizado isotrópico elimina elementos da imagem não distinguindo entre ruído e detalhes da imagem. Assim, pretende-se realizar a menor quantidade de iterações de suavizado possível da imagem para não eliminar elementos de interesse, mas também o suficiente de maneira que elementos indesejados (como ruído) não perturbem o resultado da segmentação. Na Seção 7.2 será estudada a sensibilidade da segmentação com relação ao número de iterações de suavizado realizadas na imagem de entrada.

Como mencionado, o resultado dos métodos D_T depende do conjunto \mathcal{C} selecionado. Logo, a sensibilidade da segmentação com respeito a seleção dos valores das classes $c_i \in \mathcal{C}$ é analisado na Seção 7.3. Posteriormente, a influência dos parâmetros dos métodos S_{D_T} -Continuous e S_{D_T} -Discrete é estudada na Seção 7.4.

Como foi mencionado no início deste trabalho, é de principal interesse a aplicação destas técnicas na segmentação de imagens médicas, em particular para identificação de vasos sanguíneos. Assim, a performance destes métodos nesta aplicação particular é apresentada na Seção 7.5.

7.1 Teste 1 - Validação e comparação com outros métodos

A imagem sintética apresentada na Fig. 7.1 foi utilizada nos diferentes testes realizados. Esta é uma imagem em tonalidades de cinza de 8 *bits-per-pixel* (bpp) e 372×282 pixels de tamanho composta de dois círculos concêntricos, um menor (50 pixels de raio e intensidade 50) e um maior (100 pixels de raio e intensidade 100). A partir desta, diferentes imagens de teste (ou casos de teste - TC) foram geradas adicionando diferentes níveis de Ruído Gaussiano Branco (WGN). Este ruído é obtido como mencionado na Seção 2.1. Para eliminar a influência aleatória do ruído nos resultados (índices), para cada variância σ^2 oito imagens foram geradas. Os índices correspondentes as diferentes imagens foram

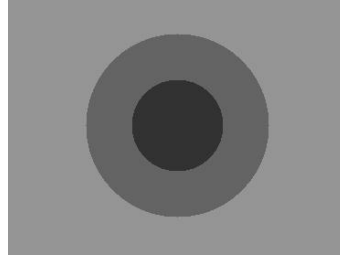


Figura 7.1: Imagem sintética.

Nível de ruído($\bar{\sigma}^2$)	0.01	0.02	0.03	0.04	0.05
Nome	TCn01	TCn02	TCn03	TCn04	TCn05
SNR(dB)	23.41	20.79	19.22	18.16	17.25
Nível de ruído($\bar{\sigma}^2$)	0.06	0.07	0.08	0.09	0.10
Nome	TCn06	TCn07	TCn08	TCn09	TCn10
SNR(dB)	16.54	15.92	15.64	15.25	14.93

Tabela 7.1: Nome dos diferentes casos baseado no nível de ruído na imagem

promediados eliminando assim esta influência. Além disso, estas imagens (TC) foram denotadas como TCn01 a TCn10 para $\bar{\sigma}^2$ variando desde 0.01 a 0.1 respectivamente (Tabela 7.1).

Os métodos de segmentação propostos foram comparados com outros usualmente utilizados em imagens médicas, quais sejam: *Bootstrap* [21, 100], *K-Means* [77, 159], *Fuzzy C-means* [48, 24] e Crescimento de Regiões (*Region Growing*). No caso de *Region Growing*, sete sementes foram selecionadas na imagem: quatro sementes no *background* da imagem (próximas as arestas) e três no círculo interior. O crescimento foi detido quando a diferença absoluta entre a média das intensidades das sementes nessa região \bar{u} e do pixel u^p sendo considerado superava um determinado limite $\delta > 0$ (foi utilizado $\delta = 20$ em todos os casos), i.e.,

$$\text{quando } |\bar{u} - u^p| > \delta, u^p \text{ não é incorporado a região.} \quad (7.1)$$

A qualidade dos resultados foi quantificada utilizando as seguintes métricas [1, 210]:

- **Índice Tanimoto:** Este índice é calculado como

$$I(A_1, A_2) = \frac{n(A_1 \cap A_2)}{n(A_1 \cup A_2)} \quad (7.2)$$

i.e., o quociente entre a quantidade de pixels na intersecção da região A_1 na imagem original e a região (correspondente) A_2 na imagem segmentada, e a quantidade de pixels na união de ambas regiões. O valor ótimo deste índice é 1.

- **Índice de *Overlap* (superposição):** É definido como [210]:

$$O(A_1, A_2) = 2 \cdot \frac{n(A_1 \cap A_2)}{n(A_1) + n(A_2)} \quad (7.3)$$

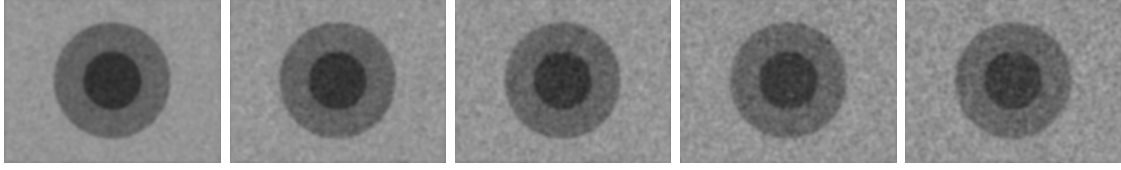


Figura 7.2: Imagens de entrada correspondentes a TCn02, TCn04, TCn06, TCn08 e TCn10 respectivamente.

i.e., o quociente entre a quantidade de pixels na intersecção da região A_1 na imagem original e a região (correspondente) A_2 na imagem segmentada, e a soma entre a quantidade de pixels em ambas as regiões. O valor ótimo deste índice é 1.

- **Desvio do centro de massa:** Este índice é dado pela distância (em pixels) entre os centros de massa da região original e a região correspondente na imagem segmentada. O valor ótimo deste índice é 0.
- **Distância entre as bordas:** Este índice é dado por

$$D(C_1, C_2) = \frac{\sum_{i=1}^{np1} d(\mathbf{x}_1^i, C_2) + \sum_{i=1}^{np2} d(\mathbf{x}_2^i, C_1)}{(np1 + np2)} \quad (7.4)$$

onde $d(\mathbf{x}, C)$ significa a distância (em pixels) do ponto \mathbf{x} à curva C ; C_1 e C_2 são as bordas das regiões original e segmentada respectivamente; $np1$ e $np2$ são o número de pontos que caracterizam estas bordas. Finalmente \mathbf{x}_1^i e \mathbf{x}_2^i representam pontos quaisquer nas bordas C_1 e C_2 respectivamente. O valor ótimo deste índice é 0.

Depois de degradar artificialmente as imagens com ruído Gaussiano branco, um filtro de suavizado isotrópico foi aplicado (convolução com um *kernel* Gaussiano 5x5). Como será apresentado na próxima seção, alguns destes métodos não foram capazes de segmentar a imagem para todos os níveis de ruído. Em geral, quando o nível de ruído aumenta, um maior número de iterações de suavizado foi necessário para assegurar que os diferentes métodos sejam capazes de fornecer um resultado razoável. Este aspecto será abordado em maior detalhe na Seção 7.2. Neste primeiro estudo, seis iterações de suavizado foram aplicadas nas diferentes imagens TC antes da segmentação. Na Fig. 7.2 são apresentadas algumas das imagens TC utilizadas e na Fig. 7.3 os resultados para os diferentes métodos.

Na Fig. 7.3 pode-se observar que, para altos níveis de ruído, *Bootstrap* e *Fuzzy C-means* têm problemas para identificar as diferentes regiões. Também, *Bootstrap*, *Fuzzy C-means*, *K-means*, *Region Growing* e *S_{DT} -Discrete* identificam componentes do ruído como sendo parte das regiões. Por outro lado *S_{DT} -Continuous* retorna resultados satisfatórios inclusive para ruído intenso.

O comportamento dos diferentes índices com respeito ao nível de ruído (caracterizado pela variância associada $\bar{\sigma}^2$) é apresentado na Fig. 7.4. Em todos os casos, *Bootstrap* e *Fuzzy C-means* têm problemas para baixos níveis de ruído (n04 e menos). Para os índices

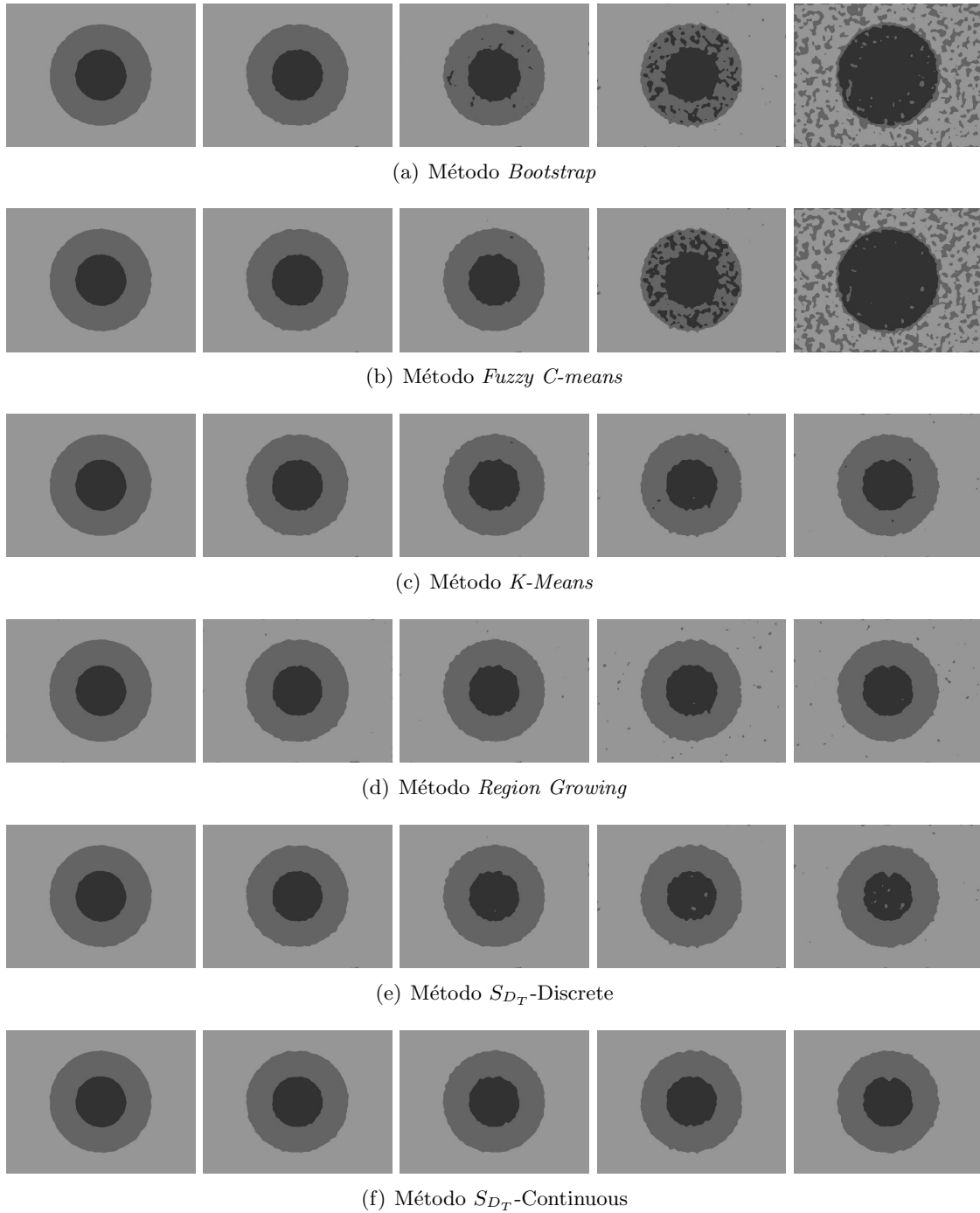


Figura 7.3: Resultados para *Bootstrap*, *Fuzzy C-means*, *K-Mean*, *Region Growing*, S_{DT} -Discrete e S_{DT} -Continuous. As colunas correspondem aos casos TCn02, TCn04, TCn06, TCn08 e TCn10 respectivamente.

de *Overlap* e Tanimoto (Fig.7.4(a) e Fig.7.4(b)) os melhores resultados são obtidos com *K-means*, *Region Growing*, S_{DT} -Discrete e S_{DT} -Continuous. Além disso, S_{DT} -Continuous apresenta melhores resultados que os outros inclusive para altos níveis de ruído. Por outro lado, o comportamento dos métodos S_{DT} -Discrete e *K-Means* são semelhantes.

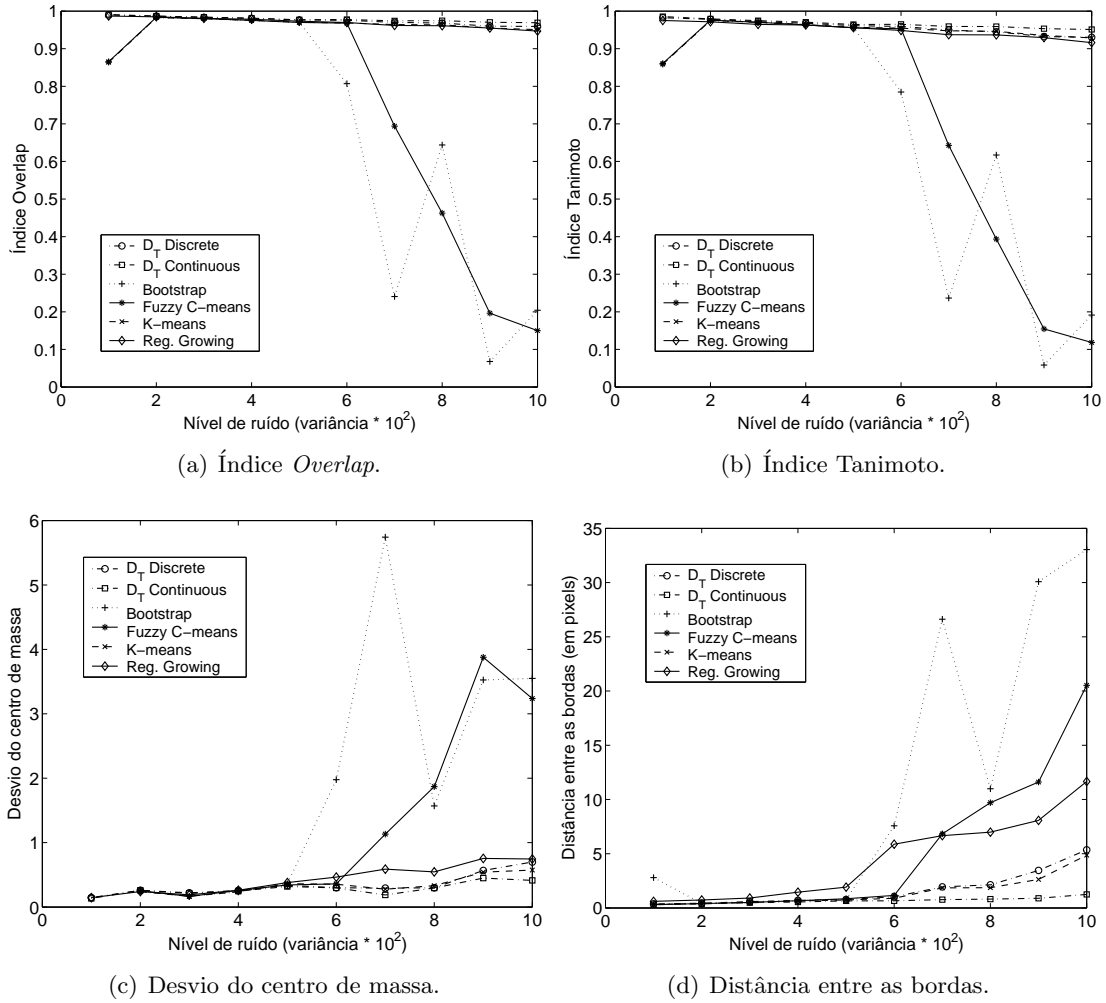


Figura 7.4: Índices vs. variância do ruído $\bar{\sigma}^2$.

Para a Desvio do centro de massa (Fig. 7.4(c)), *K-means*, *Region Growing*, *S_{D_T}-Discrete* e *S_{D_T}-Continuous* apresentam resultados compráveis. Todos estes métodos não apresentam um desvio maior que um pixel. Neste caso, *K-means* e *S_{D_T}-Discrete* são comparáveis, sendo *S_{D_T}-Continuous* o melhor. É importante ressaltar que a regularidade da estrutura sendo segmentada (dois círculo concêntricos), pode afetar os resultados (em particular para este índice)¹. Esta característica será analisada com maior detalhe mais adiante.

Para medir a distância entre bordas de regiões vizinhas, as bordas da região correspondendo à intensidade 100 foram identificados. Uma vez determinados os contornos desta região nas imagens original e segmentada, a distância entre as curvas foi calculada como descrito anteriormente. Estes resultados (Fig. 7.4(d)) evidenciam uma clara superioridade dos métodos *K-means*, *S_{D_T}-Discrete* e *S_{D_T}-Continuous*. Como será apresentado a seguir, os resultados são influenciados pelo número de iterações de suavizado aplicadas nos dados. Quanto maior o número de iterações aplicadas, maior a quantidade de ruído eliminado.

¹Como o objetivo é identificar uma região “geometricamente isotrópica”, o suavizado isotrópico não afeta a sua forma. Para estruturas mais complexas isto pode não acontecer.

Iterações de suavizado	2	4	6	8
Nome	TCn10-s02	TCn10-s04	TCn10-s06	TCn10-s08

Tabela 7.2: Nomes dos diferentes casos de teste baseados na quantidade de iterações de suavizado.

No entanto, as bordas também são perdidas e objetos pequenos são eliminados. Por este motivo é importante que o método seja capaz de segmentar a imagem com o menor número de iterações possível retornando um resultado de boa qualidade. Como as imagens utilizadas nesta seção são sintéticas e as regiões presentes são grandes quando comparadas ao tamanho da imagem (i.e., o tamanho dos círculos é da mesma ordem que a imagem), uma suavização isotrópica não afeta a sua forma. Contudo, quando imagens mais complicadas são processadas, esta situação pode mudar como será discutido nas seções seguintes.

7.2 Teste 2 - Robustez

No caso de imagens médicas, a capacidade do método para identificar tecidos ou órgãos que podem estar imersos em outros de intensidades semelhantes é muito importante. A principal desvantagem de suavizar muito uma imagem ruidosa é que detalhes podem ser perdidos e características importantes da imagem ser apagadas. Como mencionado anteriormente, dependendo do número de iterações de suavização aplicadas nos dados de entrada, os métodos podem ou não ser capazes de identificar (segmentar) as diferentes regiões da imagem. Nesta seção é estudado como o número de iterações de suavizado afeta o resultado da segmentação.

A seguir, será utilizada a imagem TCn10 (correspondente a ruído n10 mencionado na Seção 7.1). O ruído da imagem é parcialmente removido aplicando suavizado isotrópico (convolução com um *kernel* Gaussiano). Foram gerados quatro diferentes casos de teste correspondentes a 2, 4, 6 e 8 iterações respectivamente (Tabela 7.2). Os índices empregados anteriormente (i.e., *Overlap*, Tanimoto, Desvio do centro de massa e Distância entre as bordas) foram utilizados para avaliar a qualidade das segmentações obtidas com cada método. Os resultados são apresentados na Fig.7.5.

No caso dos índices *Overlap* e Tanimoto (Figs. 7.5(a) e 7.5(b)) observa-se que *K-means*, *Region Growing*, S_{D_T} -Continuous e S_{D_T} -Discrete se comportam de maneira similar acima de 6 iterações. No entanto, os métodos D_T e *Region Growing* se mostram melhores para menor número de iterações.

Para o Desvio do centro de massa (Fig. 7.5(c)), somente depois de 6 iterações *K-means* foi capaz de retornar resultados similares aos de *Region Growing*, S_{D_T} -Discrete e S_{D_T} -Continuous². Com poucas iterações, *Region Growing* é o que melhor consegue aproximar do S_{D_T} -Discrete, sendo que S_{D_T} -Continuous se mostra melhor que os outros em todos os casos.

²É importante destacar que a regularidade e simetria da imagem pode afetar estes resultados. Por este motivo, uma outra imagem foi utilizada como teste.

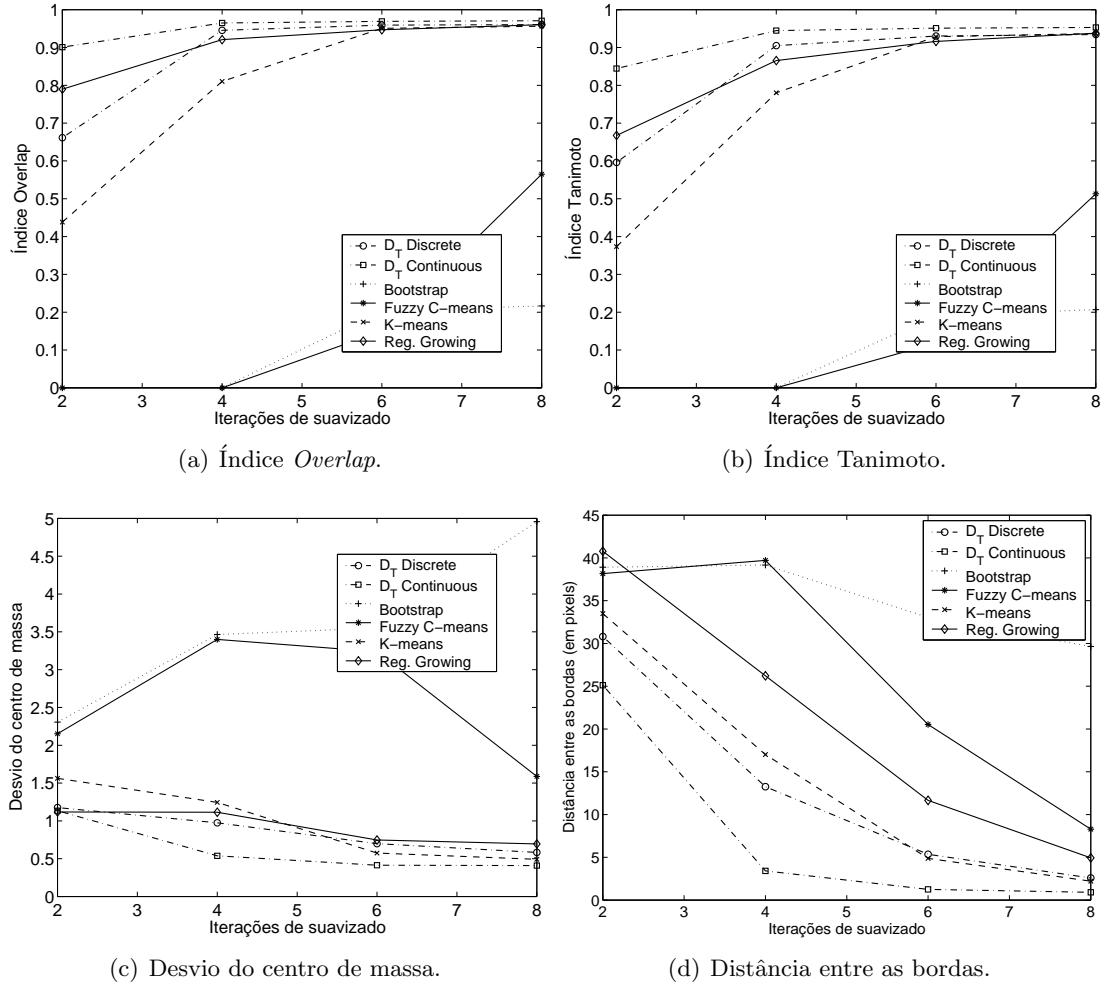


Figura 7.5: Índice vs. número de iterações de suavizado para imagens TCn10-s02 a TCn10-s08.

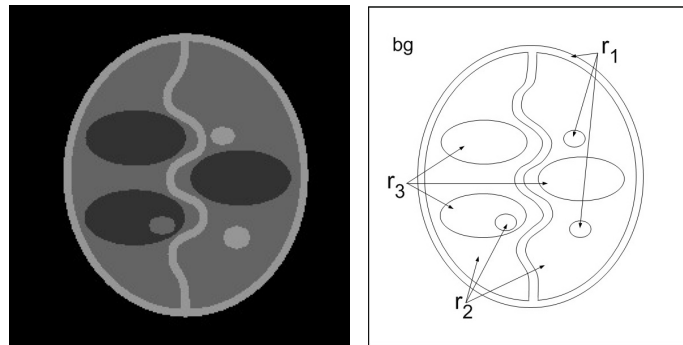


Figura 7.6: Imagem Médica Sintética - SMI.

Para a Distância entre as bordas (Fig. 7.5(d)) pode-se apreciar que S_{D_T} -Continuous se comporta muito melhor que os outros métodos em todos os casos, sendo K -means, S_{D_T} -Discrete e S_{D_T} -Continuous os que apresentaram resultados aceitáveis.

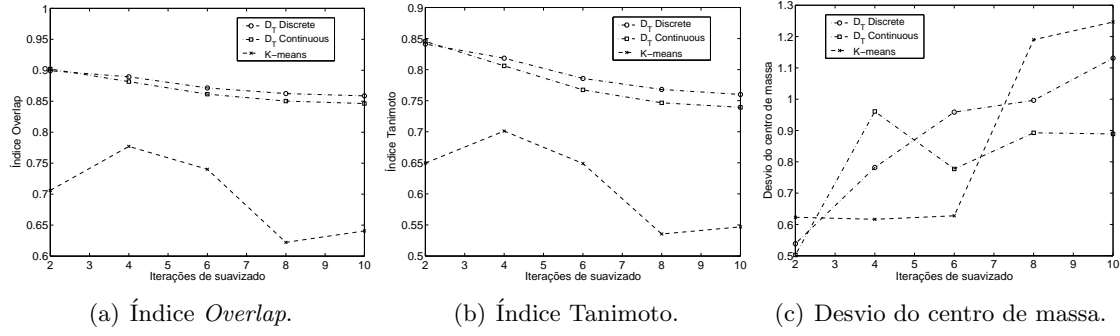


Figura 7.7: Resultados para *Overlap*, Tanimoto e Desvio do centro de massa na imagem SMI.

Como mencionado, a regularidade e simetria da imagem utilizada (Fig.7.1) pode ter uma influência nos resultados dos diferentes métodos, especialmente quando uma maior suavização é introduzida. Para analisar esta dependência, uma imagem diferente foi estudada. Esta imagem, chamada Imagem Médica Sintética - SMI (apresentada na Fig. 7.6) de 256×256 pixels de tamanho é composta por uma série de regiões de diferentes intensidades e formas. As intensidades são 0, 50, 100 e 150 correspondendo as regiões bg , r_3 , r_2 e r_1 respectivamente. A imagem foi poluída com ruído n05 para simular uma imagem médica real (Tabela 7.1). Esta imagem foi segmentada usando os dois métodos de segmentação baseados na D_T (S_{D_T} -Discrete e S_{D_T} -Continuous) e K -Means.

Os resultados obtidos são apresentados na Fig. 7.7, correspondendo à região de intensidade 50 (região r_3). Para os índices de *Overlap* e Tanimoto (Fig. 7.7(a) e 7.7(b)) observa-se que ambos os métodos D_T são superiores a K -Means. No caso de Desvio do centro de massa (Fig. 7.7(c)), para menos de três iterações o método S_{D_T} -Discrete se comporta sensivelmente melhor mas, em geral, a diferença entre os métodos não é significativa (já que os valores variam entre 0.5 e 1.2 pixels).

Para a Distância entre as bordas (Fig. 7.8(a) e Fig.7.8(b)), as bordas das regiões r_2 (isolinhas correspondentes a intensidades 75 e 125) e r_3 (isolinhas correspondentes a intensidades 25 e 75) foram calculadas. Observa-se que, para r_2 os métodos D_T se mostraram mais eficientes que K -Means mantendo este índice abaixo de 3 pixels e para r_3 abaixo de 1 pixel em todos os casos.

7.3 Teste 3 - Dependência nos valores das classes

É fácil perceber que o resultado da segmentação será influenciado pelos valores das classes \mathcal{C} . Por outro lado, os métodos utilizados nas comparações das seções anteriores não requerem a definição destas classes.

Esta diferença dos métodos propostos tem uma vantagem e uma desvantagem. A vantagem é que, dispondo de informação *a priori* que indique que uma região de interesse é caracterizada por uma dada intensidade, pode-se forçar o método a procurar por ela

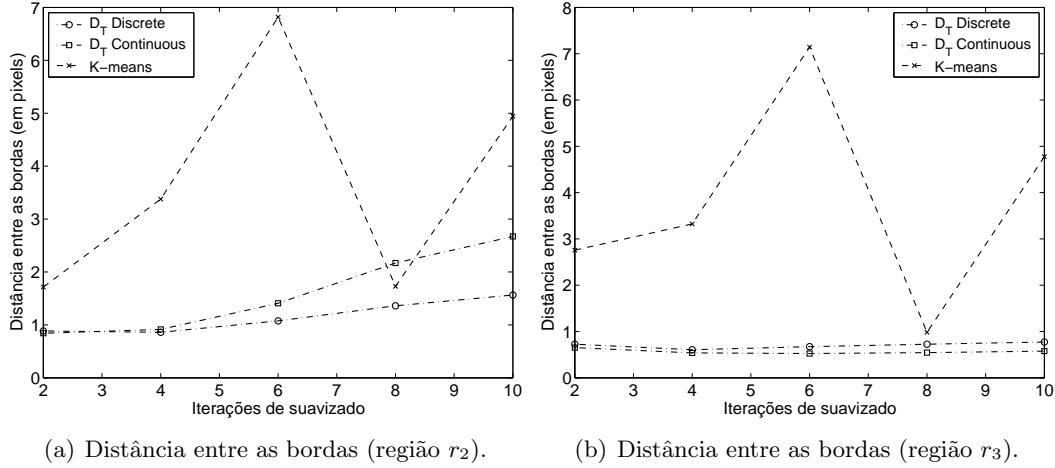


Figura 7.8: Resultados para Distância entre as bordas para a região na imagem SMI.

Percentual (p)	-10%	-9%	-8%	-7%	-6%	-5%	-4%	-3%	-2%	-1%	0%
Valor da classe c_2	74.50	77.05	79.59	82.15	84.70	87.25	89.80	92.34	94.90	97.45	100
Percentual (p)	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%	
Valor da classe c_2	102.5	105.1	107.6	110.2	112.75	115.30	117.85	120.40	122.95	125.50	

Tabela 7.3: Valores da classe c_2 dependendo do erro introduzido.

(ao escolher uma classe com aquele valor). Não obstante, quando esta informação não está disponível, uma inspeção por parte do usuário é necessária para determinar os valores destas classes. Neste caso, os valores das classes podem não estar corretamente definidos (i.e., conter erros). Logo, é importante avaliar como estes erros irão afetar a qualidade do resultado.

Nesta seção, a primeira imagem sintética foi segmentada (Fig 7.1), mas somente os métodos D_T foram testados. O objetivo desta análise é entender melhor como o valor das classes influencia o resultado da segmentação e qual método é mais sensível a erros na estimação das classes. Como mencionado anteriormente, a imagem original contém intensidades 50, 100 e 150 nas diferentes regiões. Somente a classe correspondente a intensidade 100 foi perturbada. A perturbação no valor da classe c_2 foi feita somando a este um erro $e_c(p)$, computado como

$$e_c(p) = p * \max(\mathcal{V}), \quad (7.5)$$

sendo p um valor entre 0 e 1, e $\max(\mathcal{V})$ é o valor máximo que a imagem $v_0 \in \mathcal{V}$ pode assumir. Neste caso, por se tratar de uma imagem de 8bpp, $\max(\mathcal{V}) = 255$. Na Tabela 7.3 são mostrados os valores de p utilizados nos diferentes testes. É interessante notar que para $p = 0.10$ o erro total é $\pm 25.5\%$ no valor adotado para c_2 .

A imagem TCn05, foi segmentada usando S_{D_T} -Discrete e S_{D_T} -Continuous. Foi considerado $\mathcal{C} = \{c_1, c_2, c_3\}$, para $c_1 = 50$ e $c_3 = 150$, e perturbando c_2 como mencionado na Tabela 7.3. Os resultados são apresentados na Fig.7.9.

No caso de índice de *Overlap* e índice Tanimoto (Figs. 7.9(a) e Fig. 7.9(b)), observa-

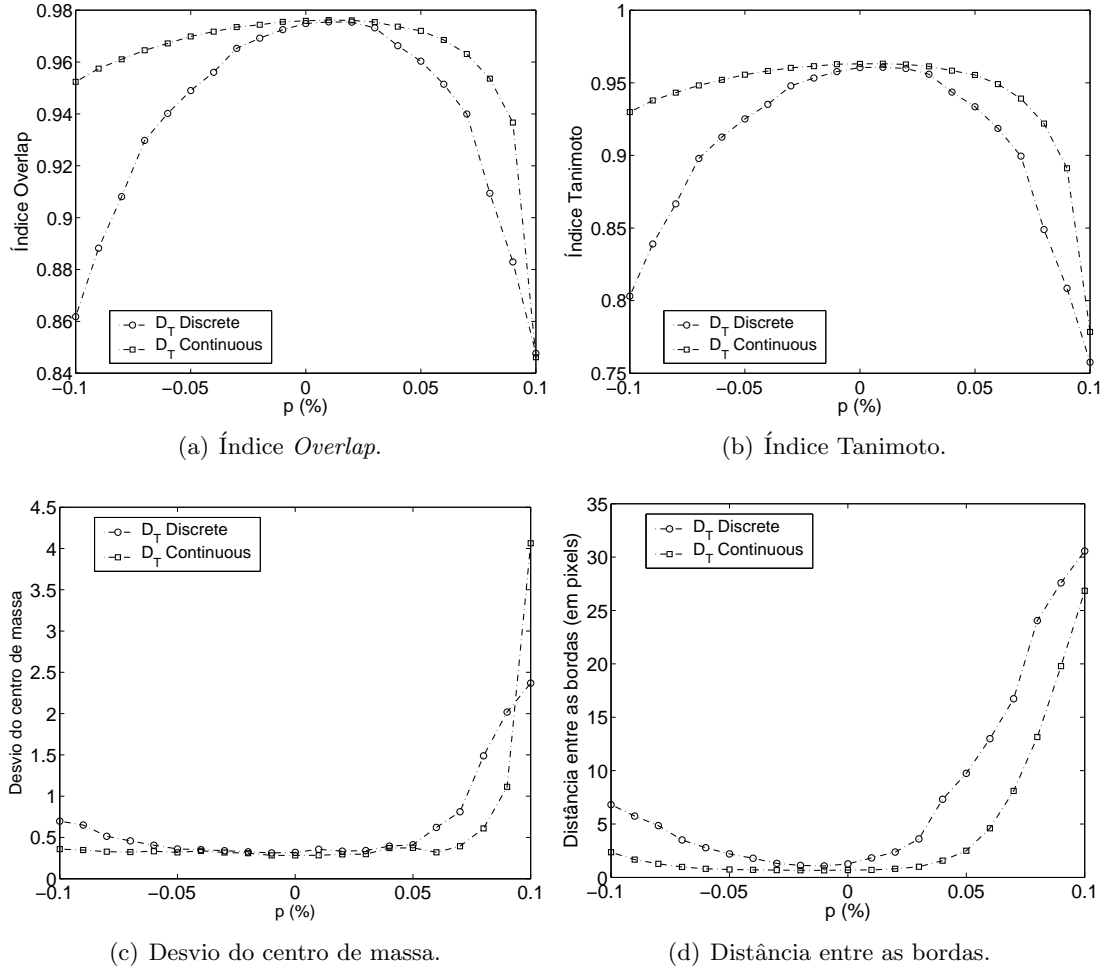


Figura 7.9: Índices como função de p , c_2 foi considerada como a classe perturbada e a imagem TCn05 foi segmentada. O valor da classe c_2 foi computado usando a Eq. (7.5): $c_2 = 100 \pm p \times \max(\mathcal{V}) = 100 \pm p \times 255$.

se que ambos os métodos obtiveram melhor resultado para o valor exato da classe. No entanto, a S_{D_T} -Continuous se mostra superior em todos os casos.

Para a Desvio do centro de massa (Fig. 7.9(c)), observa-se que S_{D_T} -Continuous é mais preciso em todos os casos. Em particular, para $p \in (-0.1; 0)$ o desvio do centro de massa foi quase constante em torno de $1/3$ de pixel.

A Figura 7.9(d) apresenta a Distância entre as bordas para a região de intensidade 100. Novamente, S_{D_T} -Continuous se mostra mais preciso para identificar os limites das regiões. Para $p \in (-0.1; 0)$ a borda é achada com um erro de aproximadamente 1 pixel para S_{D_T} -Continuous e 5 pixels para S_{D_T} -Discrete.

É importante destacar que as bordas são computadas para todas as regiões de intensidade $100 \pm p \times 255$ achadas pelo algoritmo e depois computada a distância à região correspondente (intensidade 100) na imagem original sem ruído. Por isto, cada pixel mal classificado irá influir no resultado final.

Em todos os testes realizados nesta seção o valor de cada classe foi definido com antecedência. Neste ponto, surge uma nova pergunta: é possível melhorar a qualidade da

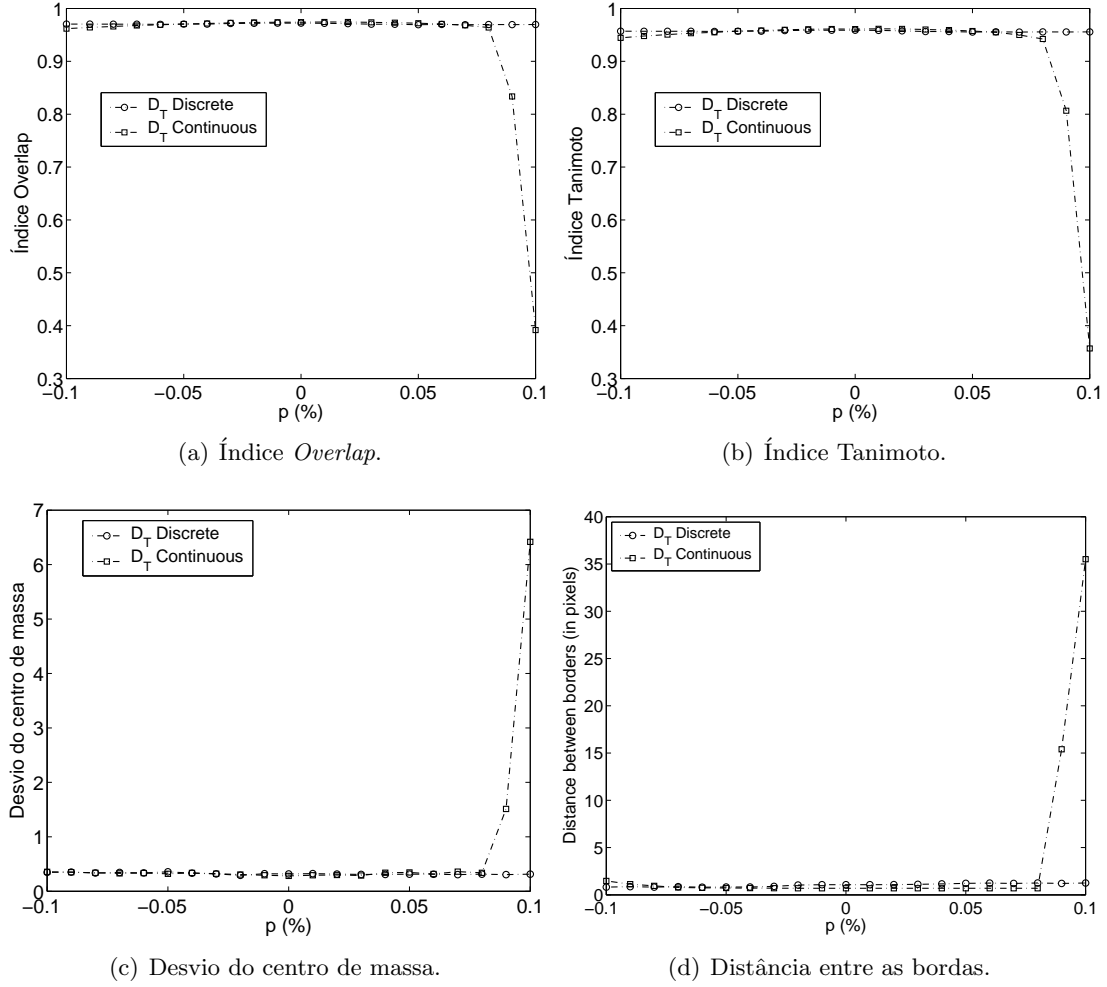


Figura 7.10: Índices como função de p sendo c_2 a classe perturbada para o caso TCn05. O valor da classe c_2 é otimizado utilizando o Algoritmo (6).

segmentação ajustando este valor por meio do processo de otimização descrito na Seção 6.3?. A resposta é afirmativa e os resultados obtidas com os métodos S_{D_T} -Continuous-OP e S_{D_T} -Discrete-OP são apresentados na Fig. 7.10. Esta técnica simples melhora consideravelmente a qualidade da segmentação fazendo com que a dependência dos resultados no valor das classes seja desprezível.

7.4 Teste 4 - Estimação de parâmetros

Além do conjunto de classes \mathcal{C} , os métodos têm outros parâmetros, sendo β e k (a análise está restrita ao caso do tensor $\mathbb{K} = k\mathbb{I}$, i.e., difusão isotrópica) para S_{D_T} -Continuous e θ para S_{D_T} -Discrete. No caso, β pode ser interpretado como uma variável de escala, relacionando o termo de força da equação de estado e φ a sua solução. Este parâmetro adota valores entre 0 e 1. Quando β é igual a 0, recupera-se o problema discreto que depende somente na distância entre a intensidade do pixel e a intensidade da classe. Quando $\beta = 1$, recupera-se o problema potencial, onde a equação de estado corresponde

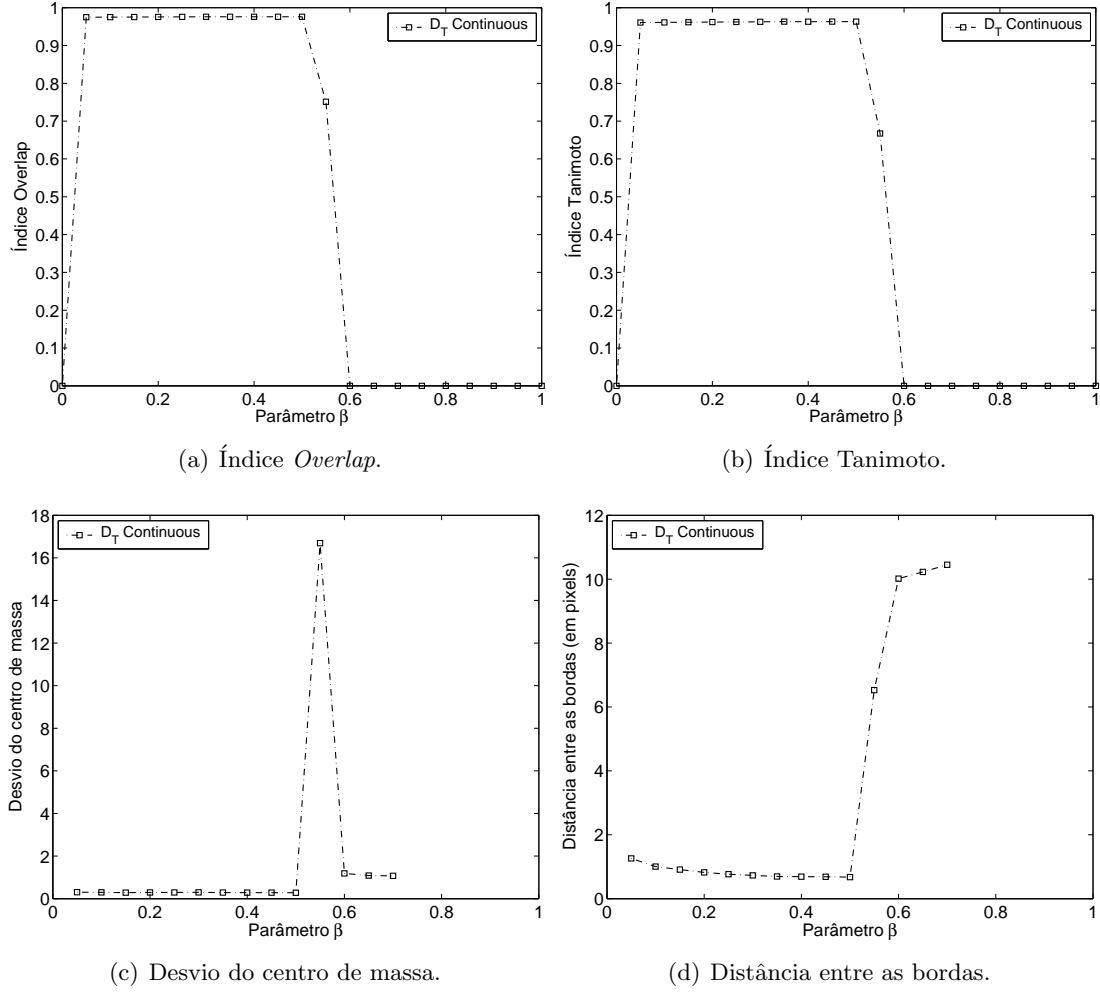


Figura 7.11: Curvas para S_{D_T} -Continuous para diferentes valores do parâmetro β . Os índices são computados por região correspondendo a intensidade 100.

a condição de mínimo da função custo. Por outro lado, k é o coeficiente que controla a “suavidade” de φ . No caso de S_{D_T} -Discrete, θ controla o peso de cada termo na função custo $\mathcal{F}^d(u^s)$ (Seção 6.2).

Outro parâmetro que ambos os métodos apresentam é α , usado para selecionar quantos pixels serão reclassificados em cada iteração do algoritmo. Esta variável pode ser sempre utilizada com o valor um, devendo ser modificado somente quando o algoritmo apresente oscilações. Se for o caso, este valor deve ser reduzido, diminuindo o tamanho do passo.

Como ambos os métodos apresentam diferentes parâmetros, eles foram analisados separadamente usando as imagens correspondentes a TCn05.

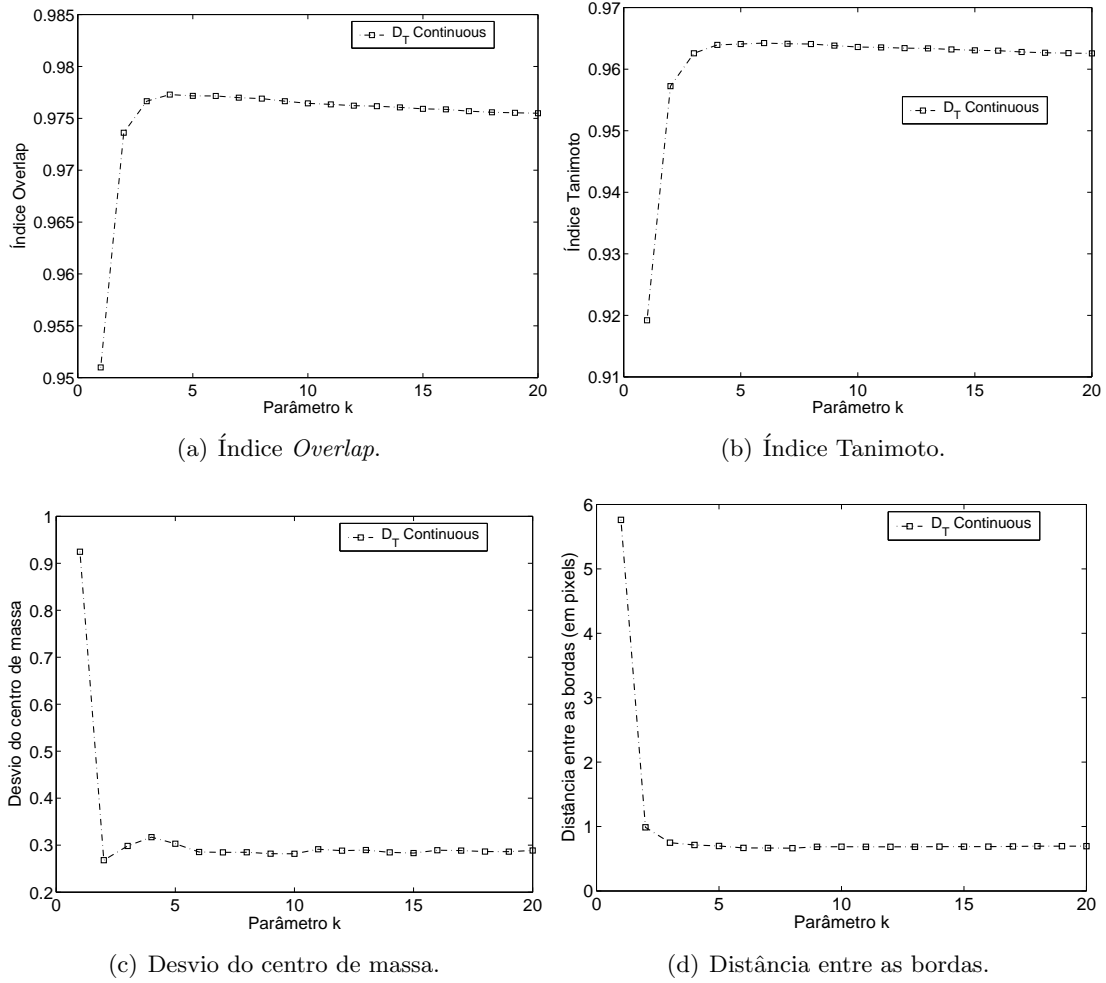


Figura 7.12: Curvas para S_{D_T} -Continuous para diferentes valores do parâmetro k . Os índices são computados por região correspondendo a intensidade 100.

7.4.1 S_{D_T} -Continuous

Os resultados para diferentes valores de β são apresentados na Figura 7.11. Neste caso β foi testado para valores no intervalo 0 a 1 em incrementos de 0.05. O melhor resultado para β varia entre 0.05 e 0.5. É interessante destacar que para valores de β acima de 0.5, a imagem não foi segmentada em absoluto (a condição inicial, $u_0 \equiv c_i$, não foi modificada para $\beta > 0.5$). Somente o desvio do centro de massa apresenta melhoras para $\beta > 0.6$. O motivo disto é que, nesta situação, o centro de massa corresponde ao centro da imagem original que se encontra muito próximo do centro de massa da região c_2 (região de intensidade 100).

A Fig. 7.12 mostra os resultados de S_{D_T} -Continuous para diferentes valores de k no intervalo de 1 a 20. Como apresentado na figura, valores de k maiores que 5 produzem melhores resultados.

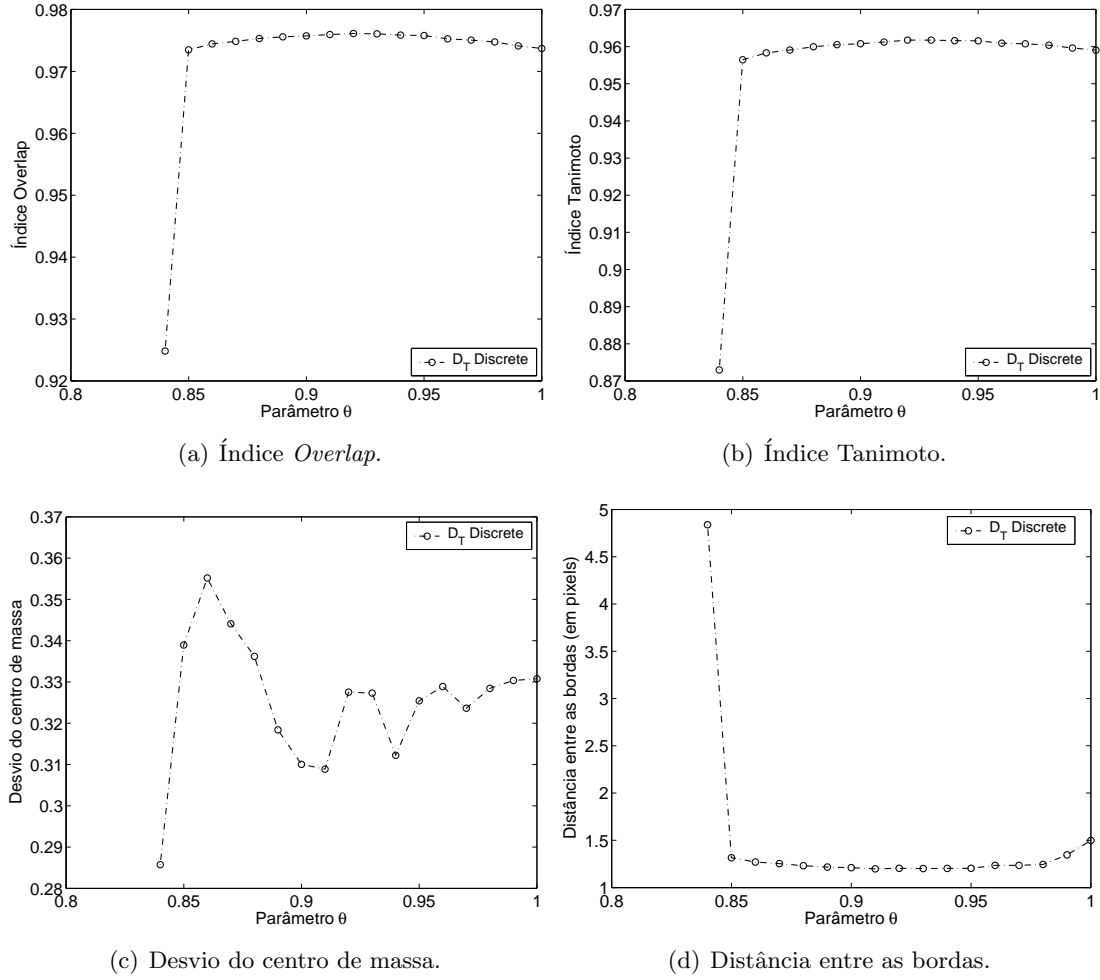


Figura 7.13: Curvas para S_{DT} -Discrete para diferentes valores do parâmetro θ .

7.4.2 S_{DT} -Discrete

O comportamento do método para diferentes valores do parâmetro θ é apresentado na Fig. 7.13. Para $\theta > 0.85$ os índices Tanimoto e *Overlap* não são muito influenciados por este parâmetro. No caso de Distância entre as bordas, o melhor resultado foi obtido para $\theta \in (0.87, 0.98)$. No caso do desvio do centro de massa (Fig. 7.13(c)), observa-se que os resultados se comportam de maneira oscilatória, porém as variações são muito pequenas (entre 0.29 e 0.36 pixels).

7.5 Teste 5 - Processamento de imagens médicas

Como mencionado, a motivação deste trabalho vem da necessidade de segmentar estruturas ou regiões a partir de imagens médicas para posterior reconstrução. O objetivo, neste caso particular, é recuperar a geometria dos vasos do sistema cardiovascular para serem utilizadas na simulação do fluxo sanguíneo em diferentes distritos. Esta técnica tem diversas aplicações no campo da medicina, em particular para diagnose e planejamento

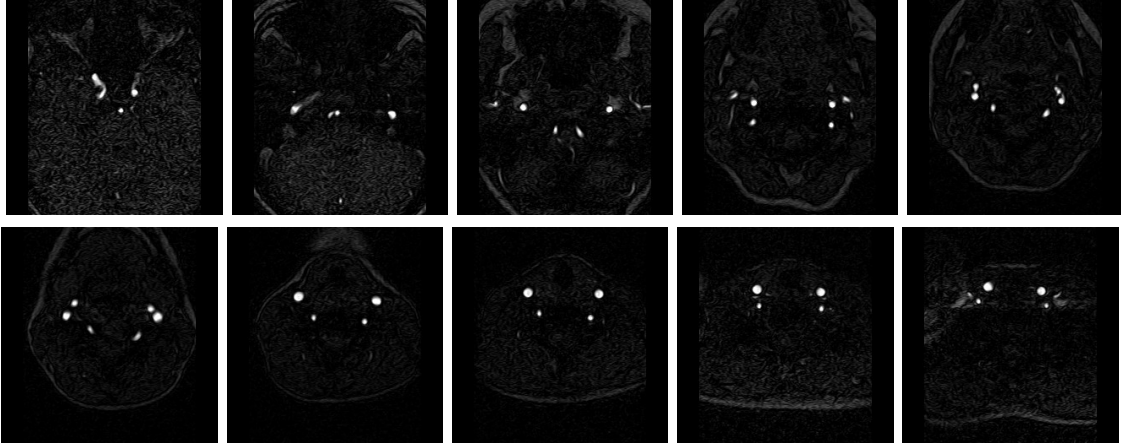


Figura 7.14: Imagem de CT (10 planos de 90).

cirúrgico [41, 63, 200, 193, 108]. Cabe a esta seção apresentar alguns resultados obtidos ao aplicar as técnicas anteriormente propostas em dados reais para segmentar vasos sanguíneos.

A imagem segmentada corresponde a uma CT composta de 90 planos (*slices*), sendo cada um deles de 256×256 pixels. Na Fig. 7.14 são apresentados 10 diferentes planos do volume total. Nestas imagens a artéria carótida interna e externa são claramente identificadas. Para segmentar estas artérias duas classes foram escolhidas. A primeira delas representando as artérias ($c_1 = 200$) e a segunda representando o *background* ($c_2 = 160$).

Na Fig. 7.15 são apresentadas as reconstruções 3D para a artéria carótida direita e esquerda segmentadas com os métodos S_{DT} -Discrete (Fig. 7.15(a)) e S_{DT} -Continuous (Fig. 7.15(b)) onde a segmentação foi processada como planos 2D independentes e posteriormente unidos em um único volume. A Fig. 7.15(c) apresenta a superfície S_{DT} -Discrete (branco) e a superfície para S_{DT} -Continuous (vermelho 50% de transparência) superpostas. Podem-se observar pequenas diferenças entre ambos os métodos, em particular na Artéria Carótida Externa (ECA). Em ambos os casos (esquerda e direita) o método S_{DT} -Continuous capturou mais detalhes da ECA.

Também, uma implementação 3D do método S_{DT} -Discrete (S_{DT} -Discrete 3D) foi utilizada para segmentar esta imagem como um único volume 3D. Diferentes comparações entre estes resultados são apresentadas nas Figs. (7.15(d)), (7.15(e)) e (7.15(f)).

Para comparar quantitativamente a diferença entre os métodos, a Fig. (7.16(a)) mostra o comportamento dos índices de *Overlap* e Tanimoto, onde o pior resultado foi obtido no plano 60. As diferenças entre ambas segmentações para dito plano são ressaltadas na Fig. (7.16(b)) onde foram colocados em cor amarela os pixels segmentados para S_{DT} -Continuous e em vermelho + amarelo os correspondentes a S_{DT} -Discrete.

7.6 Comentários

Nesta segunda parte foi abordado o problema de segmentação de imagens no contexto de Cálculo Variacional e EDP's. No Capítulo 5 foi estudado o funcional de Mumford-

Shah, o ponto de partida para o estudo da segmentação de imagens como um problema de minimização.

Posteriormente, o problema de segmentação é atacado utilizando a derivada topológica que representa mais uma contribuição deste trabalho na área de processamento de imagens. Assim, seguindo a mesma linha da primeira parte deste trabalho, dois novos algoritmos de segmentação são propostos. O desenvolvimento do cálculo da derivada topológica para uma função custo específica foi feito, chegando-se na expressão final da D_T . O primeiro algoritmo, utiliza a derivada topológica contínua como indicador para achar a classe em que cada pixel deve ser classificado de maneira que o valor da função custo seja mínimo. Este algoritmo se mostrou robusto na presença de ruído conseguindo segmentações de boa qualidade mesmo na presença de ruído intenso. No segundo algoritmo um caso particular da função custo é considerada conduzindo a uma versão discreta da derivada topológica. Assim, a expressão da derivada topológica é dada pela variação total da função custo simplificada. Este funcional considera a diferença absoluta entre a intensidade do pixel e a classe na qual foi classificado, e o comprimento das bordas das diferentes regiões, de maneira a evitar a criação de “ilhas” na imagem segmentada. Este segundo algoritmo, muito mais simples na sua implementação e de menor custo computacional (permitindo uma implementação 3D de baixíssimo custo), também tem se mostrado robusto na presença de ruído.

Respeito do desempenho computacional, o algoritmo discreto se mostrou muito mais eficiente que o contínuo. Em todos os casos o algoritmo discreto tem menor custo computacional como pode ser observado na Figura 7.17. Os testes de performance foram feitos em um computador Pentium 4 de 3 GHz com 512 Mb de memória e os algoritmos implementados em Matlab 6.5 [190].

Finalmente, os resultados obtidos com os métodos propostos foram avaliados quantitativamente e comparados com outros usualmente utilizados. Os métodos D_T apresentaram resultados de melhor qualidade, particularmente quando o ruído é de maior intensidade. Por último, estes métodos foram utilizados para segmentar imagens médicas obtendo-se resultados satisfatórios.

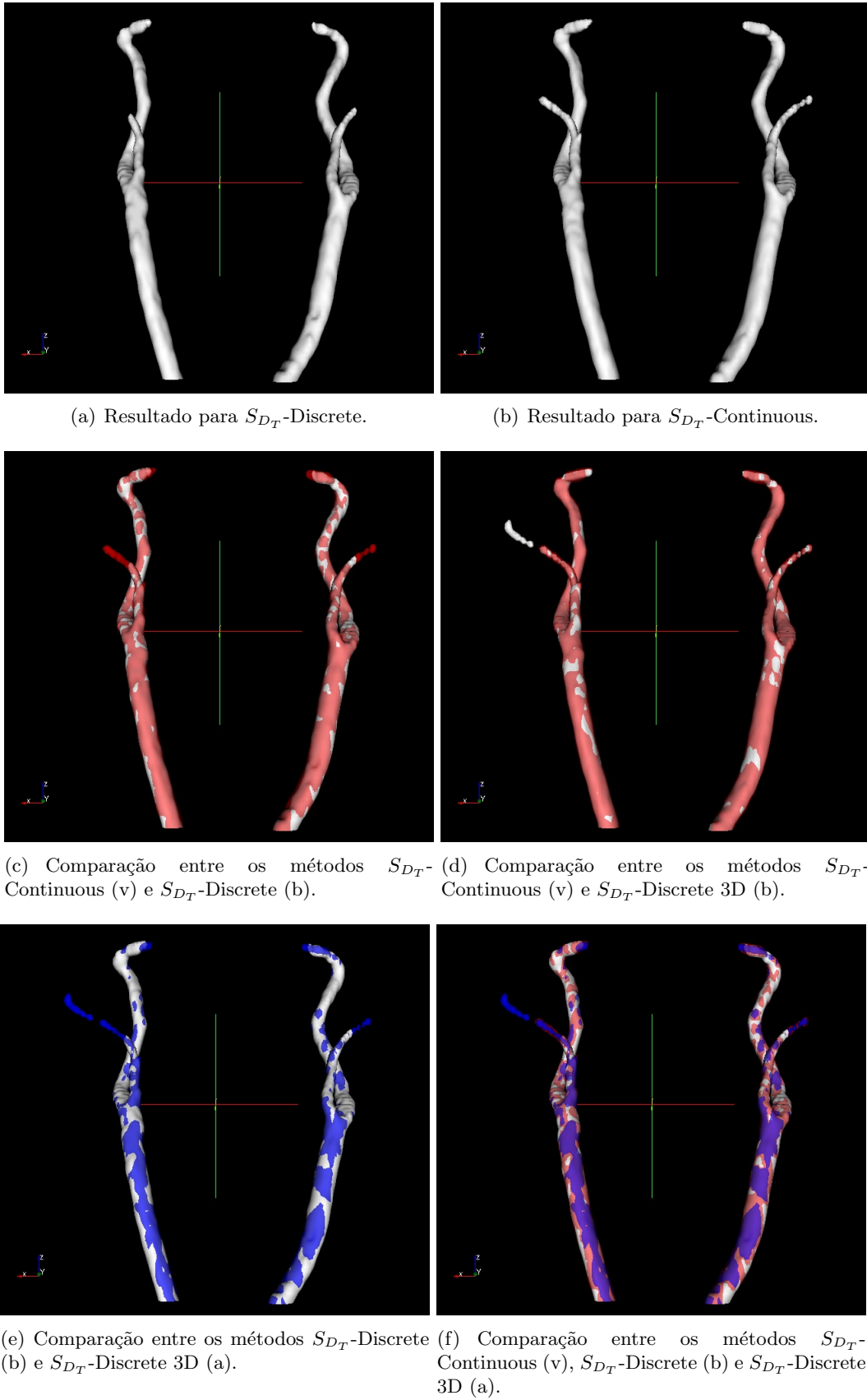


Figura 7.15: Reconstrução 3D para resultados correspondentes aos métodos S_{D_T} (v = vermelho, a = azul e b = branco).

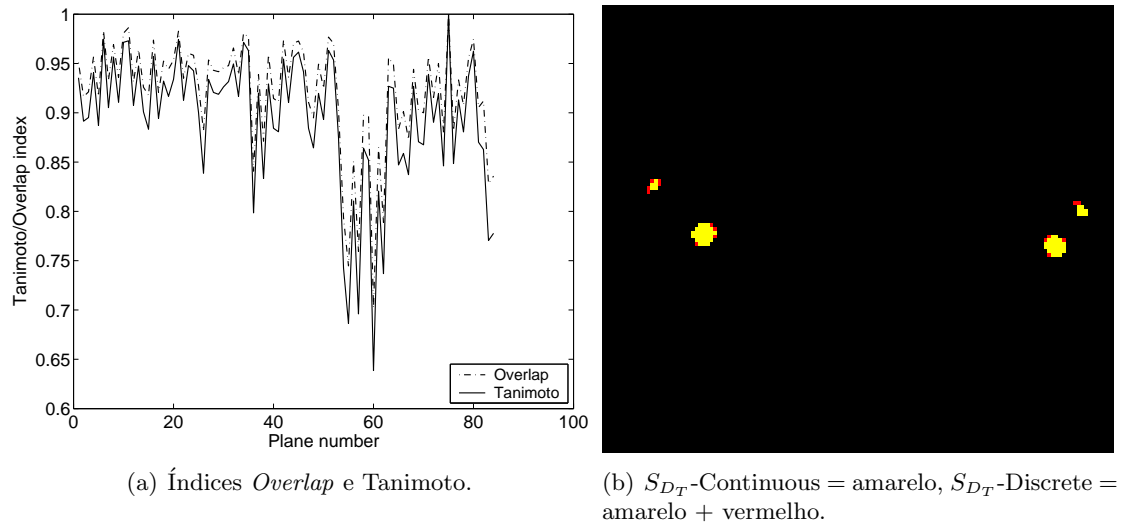


Figura 7.16: Comparação quantitativa entre os métodos S_{DT} -Continuous e S_{DT} -Discrete.

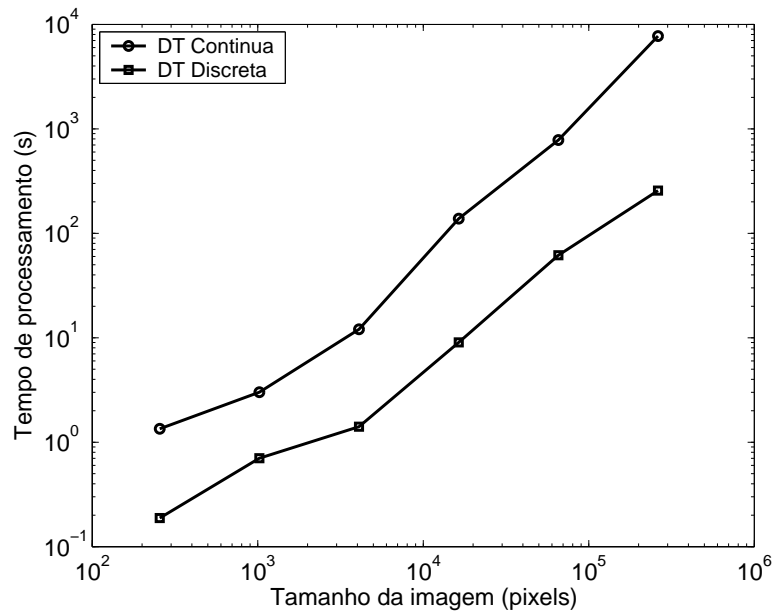


Figura 7.17: Tamanho da imagem (pixels) vs. tempo de processamento (s).

Parte III

Modelagem e Simulação Computacional do Sistema Cardiovascular Humano

Introdução

Nesta terceira e última parte, serão abordados diferentes aspectos do processo de modelagem e simulação computacional do Sistema Cardiovascular Humano (SCVH). Inicialmente, os modelos do SCVH utilizados na atualidade são apresentados no Capítulo 8. Neste caso, é feita a dedução do modelo unidimensional (Modelo 1D), uma simplificação das equações de Navier-Stokes, que permite representar as maiores artérias do corpo humano diminuindo assim o custo computacional do problema. São colocadas também as equações tridimensionais completas (Modelo 3D) utilizadas para obter informação detalhada do fluxo sanguíneo em pequenos distritos e, finalmente as condições de acoplamento necessárias para utilizar estes modelos conjuntamente. Para construir estes modelos, chamados multidimensionais, métodos e informações de diversas origens e natureza devem ser utilizados. Isto requer o emprego de ferramentas computacionais que permitam integrar estes modelos multidimensionais com os métodos necessários para a representação (como os propostos na Parte I e Parte II deste trabalho), simulação e visualização do sistema arterial, assim como operar grandes volumes de dados e resultados heterogêneos. Na atualidade isto representa um grande desafio já que não existem softwares especializados que proporcionem estas funcionalidades em um mesmo ambiente computacional. Com o intuito de contribuir na diminuição desta deficiência, neste trabalho foi desenvolvido o sistema HeMoLab - Laboratório de Modelagem em Hemodinâmica. Este sistema é descrito no Capítulo 9, onde é feita uma análise das suas necessidades assim como uma descrição dos diferentes módulos e componentes que integram esta ferramenta. Por último, no Capítulo 10, são apresentados alguns resultados obtidos com o HeMoLab. Um sistema computacional destas características em muito poderá contribuir para melhorar a compreensão do SCVH, auxiliando na prevenção, diagnose, terapia e reabilitação (planejamento cirúrgico) das mais diversas patologias ou disfunções cardiovasculares.

Finalmente, parte dos resultados apresentados nesta parte foram publicados nos seguintes trabalhos

- I. Larrabide, P. J. Blanco, S. A. Urquiza and R. A. Feijóo. Sensitivity of Blood Flow in Stenosed Carotid Bifurcation. *II International Conference on Computational Bioengineering*. H. Rodrigues *et al.* (Eds.). pp:663-674. Lisbon, Portugal, 14th-16th September, 2005.
- P. Blanco, I. Larrabide, S. Urquiza and R. Feijóo. Sensitivity of Blood Flow Patterns to the Constitutive Law of the Fluid. *Proceedings of the ECCM 2006*. LNEC - Lisbon, Portugal 5th-9th June 2006.
- I. Larrabide e R.A.Feijóo. HeMoLab - Laboratório de Modelagem em Hemodinâmica - Relatório de Pesquisa Nro. 13/2006 - Agosto 2006 - LNCC/MCT, Petrópolis - RJ - Brasil.
- I. Larrabide and R.A.Feijóo. HeMoLab - Hemodynamics Modeling Laboratory, An application for modelling the Human Cardiovascular System. Submitted to Com-

puter Graphics International 2007. May 30th - June 2nd 2007. Petrópolis - RJ - Brasil.

Capítulo 8

Modelagem Computacional Multidimensional do Sistema Cardiovascular Humano

Devido ao aumento na importância das doenças cardiovasculares nos últimos anos existe a cada dia um maior interesse por entender a origem e evolução das doenças do Sistema Cardiovascular Humano (SCVH), sendo o principal objetivo melhorar as técnicas de diagnóstico, prevenção e tratamento. Devido às limitações das experiências *in vivo*, os modelos computacionais cumprem um papel decisivo no progresso desta área de pesquisa. Sendo assim, a combinação de técnicas de modelagem computacional (como dinâmica dos fluidos, mecânica dos sólidos, modelagem multiescala, etc.) com imagens médicas tridimensionais é uma poderosa ferramenta que permite estudar o SCVH em detalhe. Atualmente é possível implementar modelos computacionais complexos e realistas que fornecem um melhor entendimento dos fenômenos presentes neste sistema, a um custo baixo e de maneira não invasiva. Logo, características do fluxo sanguíneo podem ser analisadas empregando dados específicos dos pacientes em diferentes situações e distritos vasculares.

Não obstante, na modelagem do Sistema Cardiovascular Humano é necessário vencer uma série de desafios. A reconstrução da geometria das artérias a partir de imagens médicas [41, 42], a solução de problemas não-lineares transientes que envolvem interação do sangue com as paredes arteriais deformáveis, são somente alguns destes desafios. Sendo assim, neste capítulo serão introduzidos os modelos físicos e matemáticos utilizados para estudar o fluxo sanguíneo no SCVH.

A representação deste sistema na sua totalidade com modelos tridimensionais é, no momento, impensável devido ao altíssimo custo computacional. Logo, para conseguir modelar o SCVH como um todo, são atualmente utilizados modelos multidimensionais [59, 60, 62, 107, 170, 192]. Estes modelos utilizam formulações diferentes nas distintas partes do sistema, dependendo do nível de detalhe que se deseje em cada uma delas:

- Modelos 3D [39, 44]: Estes modelos são utilizados para reproduzir pequenos distritos do SCVH. Nas regiões representadas com estes modelos dispõe-se de grande detalhe do fluxo sanguíneo.
- Modelos 1D [61, 95]: São utilizados para representar as maiores artérias do corpo humano. Estes modelos são baseados em uma simplificação das equações de Navier-Stokes 3D. Os mesmos são obtidos ao assumir uma série de hipóteses: 1) os vasos são retos, 2) o perfil de velocidade em qualquer ponto é conhecido e 3) uma relação constitutiva entre a pressão e a área do vaso.
- Modelo 0D [66, 150]: Utilizados na representação do comportamento do fluxo sanguíneo a partir de um ponto (e.g., para representar a resistência oferecida pelas arteríolas

Causas	% de mortes	Mortes por 100.000 por ano		
		Todos	Homens	Mulheres
Todas as causas	100.00	916.1	954.7	877.1
Doenças cardiovasculares	29.34	268.8	259.3	278.4
Doenças infecciosas e parasitárias	19.12	175.2	185.1	165.1
Isquemia do coração	12.64	115.8	121.4	110.1
Cancer	12.49	114.4	126.9	101.7
Doenças cerebrovasculares (derrame)	9.66	88.5	81.4	95.6
Infecções respiratórias	6.95	63.7	63.5	63.8
Doenças respiratórias	6.49	59.5	61.1	57.9
Acidentes	6.23	57.0	73.7	40.2
HIV/AIDS	4.87	44.6	46.2	43.0

Tabela 8.1: WHO - World Health Organization, 2004, [201].

e capilares). Estes modelos “condensam” em um ponto a resistência oferecida por uma parte do sistema não representada em detalhe.

- Acoplamento 1D-3D [63, 193]: Como os modelos multidimensionais introduzem contornos artificiais dentro do sistema arterial, é necessário suprir condições de acoplamento apropriadas que permitam a interação entre modelos com diferente dimensão. Desta maneira, é possível utilizar representações locais detalhadas para estudar os fenômenos associados à dinâmica do fluxo em distritos como bifurcações e outras singularidades.

Em síntese, os modelos 1D são empregados para representar a maior parte da árvore arterial e fornecer aos modelos 3D condições de contorno consistentes com o resto do sistema, utilizando terminais *Windkessel* (modelos 0D) para implementar as condições de contorno nos extremos das artérias. Desta forma, as condições de contorno dos modelos 3D são naturalmente ajustadas quando ocorre alguma mudança em qualquer parte do sistema, facilitando o estudo de diferentes situações.

Neste contexto a arteriosclerose (termo genérico empregado para indicar o espessamento e endurecimento das paredes arteriais) é responsável pela maioria das mortes nos países desenvolvidos e em desenvolvimento. Um tipo de arteriosclerose, a aterosclerose, é uma doença típica das grandes artérias, que está por trás das doenças nas artérias coronárias, carótídeas, no aneurisma de aorta e nas doenças vasculares cerebrais dentre outras. A aterosclerose é a causa principal de morte nos países desenvolvidos e em desenvolvimento para ambos os sexos e, em geral, após os 50 anos. Por sua vez, esta patologia se localiza em sítios onde prevalecem características hemodinâmicas alteradas.

Em concordância com isto, no ano 2004 por exemplo, no mundo todo 29.34% das mortes registradas foram por causa de doenças cardiovasculares (ver Tabela 8.1).

Estas regiões apresentam irregularidades na estrutura do fluxo como separação, refluxo e tensões oscilatórias. Em acordo com isto, diversos trabalhos na literatura apontam uma forte dependência dos fatores hemodinâmicos na geometria arterial [163], sendo esta uma área ideal para a aplicação de técnicas de modelagem computacional.

8.1 Modelos

São apresentadas nesta seção as equações que governam cada modelo, assim como as condições de contorno utilizadas para fechar o sistema de equações. O sistema de equações hiperbólicas utilizado para o modelo 1D é exposto. O sistema de equações correspondente ao modelo 3D é apresentado no contexto da formulação ALE (*Arbitrary Lagrangian-Eulerian*) devido ao movimento do domínio introduzido pelo deslocamento da parede arterial. As mudanças no domínio são modeladas utilizando um esquema simples de acoplamento fluido estrutura. As leis constitutivas empregadas também são colocadas, assim como alguns aspectos numéricos relacionados ao modelo constitutivo de Casson.

8.1.1 Modelo 1D

Assumindo hipóteses simplificadoras apropriadas sobre a geometria e características do fluxo sanguíneo, é possível obter um conjunto de equações unidimensionais que governam o fluxo do sangue nas artérias (este modelo é deduzido na Seção 8.A, considerando $x_3 = l$ a direção longitudinal da artéria). Como resultado destas simplificações obtém-se o seguinte sistema de equações diferenciais não-lineares

$$\begin{aligned} S_{,t} + (Sv)_{,l} &= 0 \\ (Sv)_{,t} + [v^2 S(1 + \bar{\delta})]_{,l} + \frac{S}{\rho} p_{,l} &= vN + \frac{\mu}{\rho} (Sv)_{,ll} \\ p &= p_0 + \frac{E\pi R_0 h_0}{S} \left(\sqrt{\frac{S}{S_0}} - 1 \right) + \frac{k\pi R_0 h_0}{S} \frac{1}{2\sqrt{S_0 S}} S_{,t}, \end{aligned} \quad (8.1)$$

sendo

$$\begin{aligned} N &= \frac{\mu}{\rho} \oint_{C(l,t)} \phi_{,\mathbf{m}} dl \\ \bar{\delta} &= \frac{1}{S} \int_S \phi^2 dA - 1, \end{aligned}$$

onde S representa a área da seção transversal da artéria, v a velocidade média do sangue e p a pressão média no ponto l . Foi denotado como ϕ o perfil de velocidade na seção transversal S , $C(l, t)$ a curva limitando S e \mathbf{m} a normal saindo de $C(l, t)$.

Para representar a resposta da parede arterial à pressão dentro do vaso, foi utilizada uma equação linear visco-elástica relacionando a pressão média p e a área da seção transversal, fechando assim o sistema. Na equação para p , R_0 é o raio da artéria, h_0 é a espessura da parede e S_0 é a área. Os três dados R_0 , h_0 e S_0 correspondem ao valor de referência da pressão p_0 . E é o módulo de elasticidade efetiva de Young e k é o coeficiente de fluidez associado a viscosidade parietal.

Note-se que, apesar de poder adotar diferentes modelos constitutivos para o fluido no modelo 3D, no modelo 1D o fluido é sempre considerado como Newtoniano. Isto se

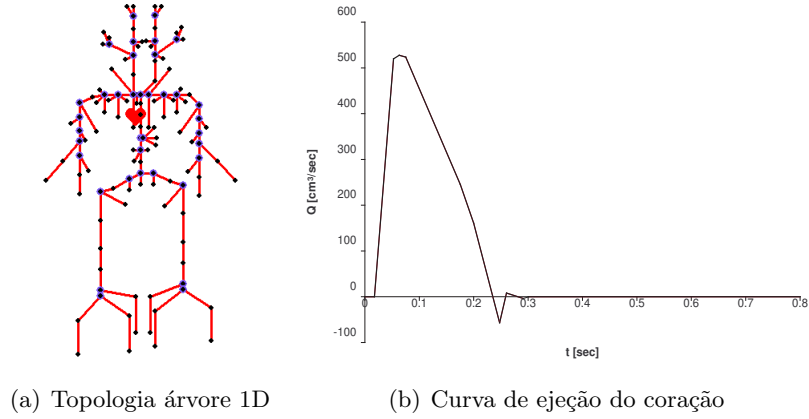


Figura 8.1: Árvore 1D.

deve ao fato de que variações na estrutura do fluxo devido a mudanças na lei constitutiva somente serão de interesse nas geometrias tridimensionais.

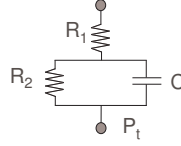
8.1.2 Geração de árvores

A solução do sistema de equações do modelo 1D é aproximada empregando o Método dos Elementos Finitos. Neste caso, foi utilizada a topologia proposta por Avolio [19], apresentada na Figura 8.1(a). No entanto, nada impede criar outras novas, o que será fundamental para gerar árvores personalizadas. Para isto é necessária uma ferramenta de edição que permita adicionar e eliminar elementos da árvore, modificar as suas propriedades e modificar a forma da curva do coração utilizada como entrada do sistema (Fig. 8.1(b)). Também, a partir desta informação deverá ser construída uma malha de elementos finitos. Estas necessidades serão estudadas em detalhe no Capítulo 9.

8.1.3 Modelo 0D

O papel das artérias e arteríolas é prover aos órgãos, tecidos e músculos uma quantidade suficiente de sangue. Estas regiões periféricas podem ser representadas, por exemplo, usando o modelo chamado *Windkessel* [66], uma analogia hidráulica-elétrica na qual as diferentes características do sistema arterial são representadas com componentes de um circuito elétrico (Tabela 8.2). Como este modelos condensam em um ponto o comportamento em uma região completa, são chamados Modelos 0D (Fig. 8.2).

Esta idéia é utilizada para modelar as condições de contorno nas extremidades do modelo 1D (representados por círculos na Fig. 8.1(a)), os quais levam em consideração a resistência oferecida pelos vasos periféricos [135, 150]. Estes elementos são representados utilizando uma resistência R_1 em série com um paralelo de outra resistência R_2 e um capacitor C . Os valores das resistências devem ser ajustados de maneira que a distribuição do fluxo nas diferentes regiões do corpo seja consistente com os valores observados por Nichols e O'Rourke [139], sempre verificando a relação $R_1 = 0.2R_T$ [66, 135] com $R_T = R_1 + R_2$. A compliância periférica total é tomada de [135]. Esta compliância é distribuída


Figura 8.2: Terminal *Windkessel*

Hidráulico	Elétrico
pressão p [mm Hg]	tensão elétrica U [V]
fluxo Q [ml / min]	corrente I [A]
resistência ao fluxo R [mm Hg \times min / ml]	resistor R [Ω]
compliância C [ml / mm Hg]	capacitância C [F]
inertância M [g / cm ⁴]	indutância L [H]
contratilidade G [ml / (min mmHg)]	condutância G [Ω^{-1}]
volume V [ml]	carga q [C]

Tabela 8.2: Analogia entre elementos de um circuito elétrico e características das artérias.

nos terminais Windkessel seguindo as linhas colocadas nesse trabalho. A equação utilizada em cada terminal é a seguinte:

$$Q_{,t} = \frac{1}{R_1 R_2 C} [R_2 C p_{,t} + p - (R_1 + R_2) Q].$$

Os parâmetros R_1 , R_2 e C devem ser ajustados para o caso particular que se deseje modelar.

8.1.4 Modelo 3D e Acoplamento

Sejam Ω a região 3D sob consideração, $\mathbf{x} = (x_1, x_2, x_3)$ um ponto arbitrário em Ω , $\mathbf{v} = \mathbf{v}(\mathbf{x}, t)$ a velocidade do fluxo sanguíneo e $\mathbf{u}(\mathbf{x}, t)$ o movimento do marco de referência consistente com a formulação ALE (*Arbitrary Lagrangian-Eulerian*). Logo, do PPV (Princípio das Potências Virtuais), tem-se

$$\begin{aligned} \int_{\Omega} \rho \mathbf{v}_{,t} \cdot \hat{\mathbf{v}} \, d\Omega &+ \int_{\Omega} \rho \nabla \mathbf{v} (\mathbf{v} - \mathbf{u}) \cdot \hat{\mathbf{v}} \, d\Omega - \int_{\Omega} p \, \text{div}(\hat{\mathbf{v}}) \, d\Omega + \int_{\Omega} \mu \nabla \mathbf{v} \cdot \nabla \hat{\mathbf{v}} \, d\Omega \\ &= \int_{\Omega} \mathbf{b} \cdot \hat{\mathbf{v}} \, d\Omega - \sum_{i=1}^n \int_{S_i} p_{1D} \mathbf{n} \cdot \hat{\mathbf{v}} \, dS \quad \forall \hat{\mathbf{v}} \in Var_v \end{aligned} \quad (8.2)$$

$$\int_{\Omega} \text{div}(\mathbf{v}) \hat{p} \, d\Omega = 0 \quad \forall \hat{p} \in Var_p, \quad (8.3)$$

sendo p_{1D} a tensão na superfície de acoplamento S_i vinda do modelo 1D. Logo o problema variacional pode ser colocado como:

Problema 1. *Problema Variacional - Modelo 3D para fluido Newtoniano Incompressível*

Dado \mathbf{b} , achar $\mathbf{v} \in Kin_v$ e $p \in Kin_p$ tal que

$$\begin{cases} (\mathbf{v}_t, \hat{\mathbf{v}}) + a(\mathbf{v}, \hat{\mathbf{v}}) + b(\mathbf{v}, \mathbf{u}, \hat{\mathbf{v}}) + g(\hat{\mathbf{v}}, p) = (\mathbf{b}, \hat{\mathbf{v}}) + \sum_{i=1}^n c(\bar{p}_i, \hat{\mathbf{v}}) \quad \forall \hat{\mathbf{v}} \in Var_v \\ g(\mathbf{v}, \hat{p}) = 0 \quad \forall \hat{p} \in Var_p \end{cases} \quad (8.4)$$

Para Kin_v , Kin_p , Var_v e Var_p definidos como

$$\begin{aligned} Kin_v &= \{\mathbf{v} \in \mathcal{V} ; \mathbf{v} = 0 \text{ em } \partial\Omega_{\bar{v}}\}, \quad Var_v = \{\hat{\mathbf{v}} \in \mathcal{V} ; \hat{\mathbf{v}} = 0 \text{ em } \partial\Omega_{\bar{v}}\} \\ Kin_p &= \{p \in L^2(\Omega)\}, \quad Var_p = \{\hat{p} \in L^2(\Omega)\}, \end{aligned}$$

Com condição inicial $\mathbf{v}|_{t=0} = \mathbf{v}_0$ tal que $\text{div}(\mathbf{v}_0) = 0$, e

$$\begin{aligned} a(\mathbf{v}, \hat{\mathbf{v}}) &= \mu(\nabla \mathbf{v}, \nabla \hat{\mathbf{v}}) \text{ uma forma bilinear em } \mathcal{V}; \\ b(\mathbf{v}, \mathbf{u}, \hat{\mathbf{v}}) &= ((\nabla \mathbf{v})(\mathbf{v} - \mathbf{u}), \hat{\mathbf{v}}) \text{ uma forma bilinear em } \mathcal{V}; \\ c(\bar{p}_i, \hat{\mathbf{v}}) &= - \int_{S_i} p_{1D} \mathbf{n} \cdot \hat{\mathbf{v}} dS \text{ uma forma bilinear em } \mathcal{V}; \\ g(\hat{\mathbf{v}}, \hat{p}) &= -(\text{div}(\hat{\mathbf{v}}), \hat{p}); \\ \bar{p}_i &= \text{pressão média na seção transversal de } \partial\Omega_{\bar{p}}; \\ (\cdot, \cdot) &= \text{produto interno usual em } \mathcal{U}. \end{aligned}$$

Vale a pena mencionar que se a fronteira sobre a qual a velocidade é imposta coincide com toda a fronteira do domínio, então para que a pressão seja única, o espaço $L^2(\Omega)$ deve ser substituído pelo espaço $L_0^2(\Omega)$ onde o índice 0 significa que as funções possuem média nula.

Para completar este sistema de equações é necessário introduzir condições de contorno apropriadas e relações constitutivas associando a pressão e o deslocamento da parede arterial. Neste caso, foi utilizado um modelo simples de anéis independentes consistente com o modelo 1D. O seguinte sistema de equações para os pontos sobre a superfície $\partial\Omega$ (representando a parede arterial) foi empregado

$$\begin{aligned} p &= p_0 + \frac{Eh}{R_0^2} \gamma + \frac{kh}{R_0^2} \gamma_t, \\ \Delta \mathbf{x} &= \gamma \mathbf{n} \text{ em } \partial\Omega, \\ \mathbf{v} &= \mathbf{x}_t, \end{aligned} \quad (8.5)$$

onde γ representa o deslocamento da parede arterial na direção \mathbf{n} (a normal saindo da parede).

Para acoplar os modelos 1D e 3D foi introduzido outro conjunto de equações na interface entre os dois modelos que estabelece as condições de acoplamento apropriadas garantindo que o problema esteja bem posto [192]. Para garantir a continuidade da massa a seguinte restrição é introduzida

$$\mathcal{Q}_i = - \int_{S_i} \mathbf{v} \cdot \mathbf{n} dS, \quad (8.6)$$

sendo \mathcal{Q}_i a taxa de fluxo correspondente ao modelo 1D na interface de acoplamento S_i .

Para garantir a continuidade da tensão na interface 1D-3D, tem-se

$$\int_{\partial\Omega_i} \mathbb{T}_{3D} \mathbf{n} \cdot \hat{\mathbf{v}} \, d\partial\Omega + \int_{S_i} p_{1D} \mathbf{n} \cdot \hat{\mathbf{v}} \, dS = 0, \quad i = 1, n \quad \forall \hat{\mathbf{v}} \in Var_v, \quad (8.7)$$

onde p_{1D} é a pressão média dada pelo modelo 1D na interface de acoplamento S_i , \mathbb{T}_{3D} é o tensor de tensões dado pelo modelo 3D na interface de acoplamento $\partial\Omega_i$ e \mathbf{n} é a normal saindo de $\partial\Omega_i$. Estas duas equações garantem continuidade da massa e tensão na interface. Resulta simples mostrar que, para números de Reynolds suficientemente grandes (como é o caso do fluxo sanguíneo), a Eq. (8.7) implica em continuidade da pressão.

8.1.5 Leis constitutivas

O comportamento do sangue pode ser modelado empregando equações constitutivas do tipo fluido Newtoniano (apropriadas para representar o comportamento do plasma) ou Não-Newtonianas. No caso Newtoniano o coeficiente de viscosidade dinâmica μ é constante, logo a relação é dada por

$$\text{Lei constitutiva Newtoniana} \quad \rightarrow \quad \mu = \mu_0. \quad (8.8)$$

Já no caso Não-Newtoniano, é utilizada uma relação mais complexa representada por

$$\text{Lei constitutiva de Casson} \quad \rightarrow \quad \mu(\hat{d}) = \left(\sqrt{\mu_0} + \sqrt{\frac{\tau_0}{2\hat{d}}} \right)^2, \quad (8.9)$$

onde $\hat{d} = \sqrt{\frac{1}{2} \mathbb{D}_{\mathbf{v}} : \mathbb{D}_{\mathbf{v}}}$ é um dos invariantes do tensor $\mathbb{D}_{\mathbf{v}}$ (i.e., a componente simétrica do gradiente de velocidade). No modelo constitutivo (8.9), μ_0 é o valor assintótico da viscosidade dinâmica quando $\hat{d} \rightarrow \infty$ e τ_0 é o limite para $\hat{d} \rightarrow 0$. Este modelo, conhecido como modelo de Casson, reproduz mudanças na viscosidade considerando variações na parte simétrica do tensor gradiente da velocidade. Logo, esperam-se variações no campo de velocidade quando as taxas de deformação são muito pequenas, já que a viscosidade nestas situações cresce consideravelmente tendendo a um movimento rígido no limite.

8.1.6 Reconstrução de Geometrias

Diferentes estudos apontam que a complexidade geométrica influencia fortemente as características hemodinâmicas e as ações mecânicas do fluxo sanguíneo nas paredes dos vasos [165, 163]. Por esta razão, a geometria vascular tem um papel muito importante no desenvolvimento de patologias como arteriosclerose. Por exemplo, a presença de zonas de re-circulação é fortemente influenciada pelo ângulo de bifurcação das artérias. Outro caso típico é dado pela seção distal¹ de uma anastomose. Diferentes ângulos de anastomose induzem zonas de re-circulação, provocando o êxito ou fracasso de um *bypass*.

¹i.e., ponto em que uma estrutura ou um órgão fica afastado de seu centro ou de sua origem.

Todas estas características destacam a importância da precisão no processo de obtenção de geometrias a partir de dados dos pacientes (e.g., processamento de imagens médicas e geração de malhas de elementos finitos). Estes passos são os seguintes:

- Obter os dados, geralmente em formatos padrão (e.g., DICOM) a partir de um estudo de imagens (como CT, MRI, Ultrasom, etc.) do paciente.
- Melhorar/restaurar a imagem e segmentar as regiões de interesse.
- Reconstruir a geometria em três dimensões.
- Gerar uma malha de Elementos Finitos para calcular as soluções dos modelos.

Este é de fato um processo complexo que requer a utilização de técnicas de diferentes áreas da ciência. A integração destas técnicas no processo de modelagem do SCVH será estudado no Capítulo 9.

Apêndice Capítulo 8

8.A Dedução do modelo unidimensional do Sistema Cardiovascular

É derivado (a seguir) um modelo unidimensional teórico que representa o fluxo sanguíneo em vasos de grande porte do SCVH, incluindo fluxos de saída devido a ramificações e/ou permeabilidade das paredes arteriais. Para chegar neste modelo, foi seguido o trabalho de Hughes & Lubliner ([95] e [200]). Os autores aplicam as hipóteses dos modelos no Teorema de Transporte de Reynolds para uma variável genérica ξ , chegando-se assim em uma expressão equivalente. Substituindo a variável genérica ξ por ρ , que representa a massa específica², obtém-se a equação de conservação da massa. Partindo da equação de Navier-Stokes, o mesmo procedimento é feito para achar a expressão da conservação de momento.

No presente trabalho, o problema é abordado de maneira um pouco diferente. Partindo do PPV é deduzida a equação de Navier-Stokes para o modelo de interesse. Como na situação particular sendo modelada, somente a velocidade axial (v_3) é de interesse, o sistema que originalmente tem três equações (respectivamente, v_1 , v_2 e v_3) é reduzido a uma única equação. Logo, usando o Teorema de Transporte de Reynolds, chega-se na equação de Balanço de Momento.

8.A.1 Hipóteses geométricas e cinemáticas

Usando como referência a Figura 8.3, as seguintes considerações são impostas

- O volume V é limitado pelos planos espaciais S_1 e S_2 , fixos e paralelos ao plano $x_1 - x_2$. As coordenadas x_3 destes planos são arbitrárias e adotadas como $x_3 = 0$ e $x_3 = L$, para S_1 e S_2 , respectivamente. Logo $x_3 \in [0, L]$.
- A superfície lateral de V , chamada de ∂V_t , pode variar com o tempo, mais não necessariamente representa uma parede material com respeito ao fluido, i.e. o fluido pode atravessar a parede do vaso.
- A área luminal de V no ponto x_3 no instante t é chamada de $S(x_3, t)$ ou simplesmente S .
- S_1 e S_2 representam $S(0, t)$ e $S(L, t)$, as seções transversais em $x_3 = 0$ e $x_3 = L$ respectivamente para o instante t .
- A superfície total que limita o volume V é chamada de ∂V e composta por ∂V_t , S_1 e S_2 . Logo $\partial V = \partial V_t \cup S_1 \cup S_2$.
- v_1 , v_2 e v_3 representam as componentes do vetor velocidade do fluido nas direções x_1 , x_2 e x_3 respectivamente, sendo $\mathbf{v} = (v_1, v_2, v_3)$.

²Já que o fluido é incompressível, como mencionado a seguir.

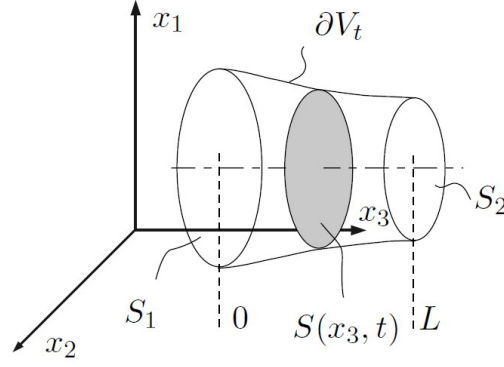


Figura 8.3: Modelo cinemático e geométrico simplificado do SCVH.

- Como é assumido que o sangue é um fluido incompressível, tem-se que

$$\text{div } \mathbf{v} = 0 \quad \text{em } V. \quad (8.10)$$

- Seja \mathbf{n} a normal unitária saliente em ∂V . Quando \mathbf{n} é usado como sub-índice de um vetor, interpreta-se como a componente do vetor na direção da normal, i.e. $\mathbf{v}_{\mathbf{n}} = \mathbf{v} \cdot \mathbf{n}$ é a componente normal da velocidade do fluido na superfície. Define-se também $\mathbf{u}_{\mathbf{n}}$ como sendo a velocidade normal de ∂V_t . É fácil ver que

$$\mathbf{u}_{\mathbf{n}} = 0 \quad \text{em } S_1 \text{ e } S_2 \quad (8.11)$$

onde

$$\mathbf{n} = (0, 0, -1) \quad \text{em } S_1 ; \quad \mathbf{n} = (0, 0, 1) \quad \text{em } S_2 \quad (8.12)$$

e, por definição, as componentes de \mathbf{u} na direção x_3 são nulas³.

Feitas as hipóteses necessárias para o modelo, estas serão utilizadas no Teorema de Transporte de Reynolds para obter uma expressão mais simples.

8.A.2 Teorema de Transporte de Reynolds

Sob as hipóteses da Mecânica do Contínuo clássica [82], o Teorema de Transporte de Reynolds permite calcular a derivada total (ou material) da descrição espacial de um campo escalar arbitrário ξ definido em um domínio V arbitrário

$$\frac{d}{dt} \int_V \xi \, dV = \int_V \frac{\partial \xi}{\partial t} \, dV + \int_{\partial V} \xi \mathbf{u}_{\mathbf{n}} \, d\partial V. \quad (8.13)$$

Pode-se reescrever o primeiro membro de (8.13) como

$$\frac{d}{dt} \int_V \xi \, dV = \frac{d}{dt} \int_0^L \left(\int_S \xi \, dA \right) dx_3. \quad (8.14)$$

³Nos planos S_1 e S_2 não existe movimento na direção x_3 .

Define-se o valor médio de ξ na área da seção transversal como sendo $\bar{\xi}$

$$\bar{\xi} = \frac{1}{S} \int_S \xi \, dA \Rightarrow \bar{\xi} S = \int_S \xi \, dA. \quad (8.15)$$

Usando (8.15) em (8.14) tem-se que

$$\frac{d}{dt} \int_V \xi \, dV = \frac{d}{dt} \int_0^L \bar{\xi} S \, dx_3. \quad (8.16)$$

Define-se também a velocidade normal de ∂V com respeito ao fluido como sendo $\mathbf{w}_n = \mathbf{u}_n - \mathbf{v}_n$.

Trabalhando no último termo de (8.13), e seja $\int_{\partial V} \xi \mathbf{u}_n \, d\partial V$, tem-se que

$$\int_{\partial V} \xi \mathbf{u}_n \, dA = \int_{\partial V} \xi (\mathbf{w}_n + \mathbf{v}_n) \, dA = \int_{\partial V} \xi \mathbf{w}_n \, dA + \int_{\partial V} \xi \mathbf{v}_n \, dA.$$

Usando o Teorema da Divergência e o fato que $\text{div}(\xi \mathbf{v}) = \nabla \xi \cdot \mathbf{v} + \xi \text{div} \, \mathbf{v}$, a expressão anterior resulta

$$\begin{aligned} \int_{\partial V} \xi \mathbf{u}_n \, dA &= \int_{\partial V_t} \xi \mathbf{w}_n \, dA + \int_{S_1 \cup S_2} \underbrace{\xi (\mathbf{u}_n - \mathbf{v}_n)}_{=0} \, dA + \int_V \text{div}(\xi \mathbf{v}) \, dV \\ &= \int_{\partial V_t} \xi \mathbf{w}_n \, dA - \left(\int_{S_1} \xi \mathbf{v}_{n_1} \, dA + \int_{S_2} \xi \mathbf{v}_{n_2} \, dA \right) \\ &\quad + \int_V \underbrace{\xi \text{div}(\mathbf{v})}_{=0} \, dV + \int_V \nabla \xi \cdot \mathbf{v} \, dV. \end{aligned}$$

Usando a Eq. (8.10), considerando que $\mathbf{n}_1 = -\mathbf{n}_2$ e tomando $\mathbf{n} = \mathbf{n}_2$, pode-se reescrever a equação anterior como

$$\int_{\partial V} \xi \mathbf{u}_n \, dA = \int_{\partial V_t} \xi \mathbf{w}_n \, dA + \int_V \nabla \xi \cdot \mathbf{v} \, dV - \left(\int_{S_2} \xi \mathbf{v}_n \, dA - \int_{S_1} \xi \mathbf{v}_n \, dA \right).$$

Usando (8.15) pode-se simplificar esta expressão para obter

$$\int_{\partial V} \xi \mathbf{u}_n \, dA = \int_{\partial V_t} \xi \mathbf{w}_n \, dA + \int_V \nabla \xi \cdot \mathbf{v} \, dV - (S_2(\overline{\xi \mathbf{v}_n}) - S_1(\overline{\xi \mathbf{v}_n})). \quad (8.17)$$

O último termo da equação anterior pode ser reescrito da seguinte maneira

$$S_2(\overline{\xi \mathbf{v}_n}) - S_1(\overline{\xi \mathbf{v}_n}) = \int_0^L \frac{\partial}{\partial x_3} [S(\overline{\xi v_3})] \, dx_3.$$

Introduzindo na Eq. (8.17) tem-se

$$\int_{\partial V} \xi \mathbf{u}_n \, dA = \int_{\partial V_t} \xi \mathbf{w}_n \, dA + \int_V \nabla \xi \cdot \mathbf{v} \, dV - \int_0^L \frac{\partial}{\partial x_3} [S(\overline{\xi v_3})] \, dx_3. \quad (8.18)$$

Usando (8.13), (8.16) e (8.18), tem-se

$$\frac{d}{dt} \int_0^L \bar{\xi} S \, dx_3 = \int_V \frac{\partial \xi}{\partial t} dV + \int_{\partial V_t} \xi \mathbf{w}_n \, dA + \int_V \nabla \xi \cdot \mathbf{v} \, dV - \int_0^L \frac{\partial}{\partial x_3} (S(\bar{\xi} v_3)) \, dx_3,$$

mas por outro lado $\int_V (\frac{\partial \xi}{\partial t} + \nabla \xi \cdot \mathbf{v}) \, dV = \int_V \dot{\xi} \, dV$, logo

$$\int_0^L \frac{\partial}{\partial t} (S \bar{\xi}) \, dx_3 + \int_0^L \frac{\partial}{\partial z} [S(\bar{\xi} v_3)] \, dx_3 = \int_V \dot{\xi} \, dV + \int_{\partial V_t} \xi \mathbf{w}_n \, dA.$$

Tendo em vista que

$$\int_{\partial V_t} (\cdot) \, dA = \int_0^L \left(\oint_{C(x_3, t)} (\cdot) \, dl \right) dx_3,$$

sendo $C(x_3, t)$ a curva que delimita a seção transversal de $S(x_3, t)$, obtém-se uma expressão do Teorema de Transporte de Reynolds específica para o problema em consideração

$$\frac{\partial}{\partial t} (S \bar{\xi}) + \frac{\partial}{\partial x_3} [S(\bar{\xi} v_3)] = \int_S \dot{\xi} \, dA + \oint_{C(x_3, t)} \xi \mathbf{w}_n \, dx_3. \quad (8.19)$$

Eq. (8.19) é a forma do Teorema do Transporte de Reynolds que inclui as hipóteses cinemáticas e geométricas consideradas para este modelo. Usando esta expressão a seguir serão derivadas as equações de conservação de momento e conservação de massa correspondentes. É importante ressaltar que nas hipóteses até aqui consideradas, não foi assumido em momento algum simetria do domínio.

8.A.3 Conservação da Massa

Assumindo agora que $\xi = 1 \, \forall t$ na Equação (8.19), tem-se que

$$\frac{\partial S}{\partial t} + \frac{\partial}{\partial x_3} S \bar{v}_3 = \oint_{C(x_3, t)} \mathbf{w}_n \, dx_3$$

Dado que \mathbf{w}_n é a velocidade relativa da parede com respeito ao fluido, $-\mathbf{w}_n$ é a velocidade do fluido com relação à parede. Define-se então o fluxo de saída (através da parede) por unidade de longitude e por unidade de tempo como sendo

$$\psi = - \oint_{C(x_3, t)} \mathbf{w}_n \, dx_3 \quad (8.20)$$

Usando este resultado na Eq. (8.20) tem-se que

Conservação da Massa

$$\frac{\partial S}{\partial t} + \frac{\partial}{\partial x_3} S \bar{v}_3 + \psi = 0. \quad (8.21)$$

8.A.4 Conservação do Momento

No trabalho de Hughes e Lubliner [95], os autores partem da equação de Navier-Stokes e deduzem uma equação de evolução para v_3 . Neste caso, foi seguido um caminho ligeiramente diferente, já que para chegar nas equações de estado foram utilizados princípios clássicos do Cálculo Variacional.

No contexto da Mecânica do Contínuo clássica e considerando o PPV, é de fato verdadeiro que $\mathbb{T} \in \mathcal{W}'$ equilibra $f \in \mathcal{U}'$ se

- $\text{PE} = \langle f, \hat{\mathbf{v}} \rangle = 0 \quad \forall \hat{\mathbf{v}} \in \text{Var}_{\mathbf{v}} \cap \mathcal{N}(\mathcal{D}),$
- $\text{PI} + \text{PE} = -(\mathbb{T}, \mathcal{D}\hat{\mathbf{v}}) + \langle f, \hat{\mathbf{v}} \rangle = 0 \quad \forall \hat{\mathbf{v}} \in \text{Var}_{\mathbf{v}}.$

Sendo $\mathcal{D}(\mathbf{v}) \equiv (\nabla(\mathbf{v}))^s$, diz-se também que $\mathcal{D}(\mathbf{v})$ é compatível se $\exists \mathbb{D} \in \mathcal{W}$ tal que $\mathcal{D}(\mathbf{v}) = \mathbb{D}$ ($\mathbb{D} \in \mathcal{W}$ o espaço de taxas de deformação). Chama-se $\mathcal{N}(\mathcal{D})$ ao núcleo do operador $\mathcal{D}(\cdot)$, o espaço das ações de movimento rígidas. $\text{Var}_{\mathbf{v}}$ é o subespaço das ações de movimento virtuais. Sejam também \mathcal{W}' o espaço dual de \mathcal{W} e \mathcal{V}' o espaço dual de \mathcal{V} .

Para o caso de equilíbrio dinâmico, f é substituído por $f^* = f - \rho \dot{\mathbf{v}}$, onde $\dot{\mathbf{v}}$ é a aceleração *real* do corpo. Em um corpo tridimensional, a descrição espacial (ou Euleriana) do PPV é dada por

$$\int_V \mathbb{T} \cdot \mathcal{D}(\hat{\mathbf{v}}) dV = \int_V \mathbf{b} \cdot \hat{\mathbf{v}} dV + \int_{\partial V_{\mathbf{a}}} \bar{\mathbf{a}} \cdot \hat{\mathbf{v}} dA - \int_V \rho \dot{\mathbf{v}} \cdot \hat{\mathbf{v}} dV \quad \forall \hat{\mathbf{v}} \in \text{Var}_{\mathbf{v}}, \quad (8.22)$$

sendo $\bar{\mathbf{a}}$ a força de superfície prescrita em $\partial V_{\mathbf{a}}$. Esta força atua na superfície onde \mathbf{v} não esta prescrita. Também tem-se que

$$\text{Var}_{\mathbf{v}} = \{\hat{\mathbf{v}} \in \mathcal{U}_{\mathbf{v}} ; \hat{\mathbf{v}} = 0 \text{ em } \partial \mathcal{V}_{\mathbf{v}} ; \partial V_{\mathbf{a}} \cap \partial V_{\mathbf{v}} = \emptyset\}.$$

Da Mecânica do Contínuo sabe-se que

$$\dot{\mathbf{v}} = \mathbf{v}' + \nabla(\mathbf{v})\mathbf{v}.$$

Pode-se reescrever (8.22) como

$$\int_V \mathbb{T} \cdot \mathcal{D}(\hat{\mathbf{v}}) dV = \int_V \mathbf{b} \cdot \hat{\mathbf{v}} dV + \int_{\partial V} \bar{\mathbf{a}} \cdot \hat{\mathbf{v}} dS - \int_V \rho \left(\frac{\partial \mathbf{v}}{\partial t} + \nabla(\mathbf{v})\mathbf{v} \right) \cdot \hat{\mathbf{v}} dV \quad \forall \hat{\mathbf{v}} \in \text{Var}_{\mathbf{v}}.$$

Supõe-se agora que o fluido estudado é Newtoniano. Pode-se então introduzir a equação constitutiva

$$\mathbb{T} = -p\mathbb{I} + 2\mu \left(\mathbb{D} - \frac{1}{3} \text{tr}(\mathbb{D})\mathbb{I} \right),$$

ou equivalentemente

$$\mathbb{T} = -p\mathbb{I} + \mathbb{T}^D. \quad (8.23)$$

Sendo μ a viscosidade, \mathbb{T}^D a parte desviadora do tensor \mathbb{T} e $\mathcal{D}(\mathbf{v}) = \mathbb{D}$. Para simplificar a

notação será utilizado $\mathcal{D}(\hat{\mathbf{v}}) = \hat{\mathbb{D}}$ e $\frac{1}{3}tr(\mathbb{D})\mathbb{I} = \mathbb{D}^H$. Tem-se então de (8.23) que

$$\int_V (-p\mathbb{I} + \mathbb{T}^D) \cdot \hat{\mathbb{D}} dV = \int_V \mathbf{b} \cdot \hat{\mathbf{v}} dV + \int_{\partial V_a} \hat{\mathbf{a}} \cdot \hat{\mathbf{v}} dS - \int_V \rho \dot{\mathbf{v}} \cdot \hat{\mathbf{v}} dV \quad \forall \hat{\mathbf{v}} \in Var_{\mathbf{v}}.$$

Operando nesta última expressão, tem-se

$$\int_V \left(-p \operatorname{div}(\hat{\mathbf{v}}) + \mathbb{T}^D \cdot \hat{\mathbb{D}} \right) dV = \int_V \mathbf{b} \cdot \hat{\mathbf{v}} dV + \int_{\partial V_a} \hat{\mathbf{a}} \cdot \hat{\mathbf{v}} dA - \int_V \rho \dot{\mathbf{v}} \cdot \hat{\mathbf{v}} dV \quad \forall \hat{\mathbf{v}} \in Var_{\mathbf{v}}. \quad (8.24)$$

Trabalhando agora nos termos do lado esquerdo de (8.24), tem-se

$$- \int_V p \operatorname{div}(\hat{\mathbf{v}}) dV = - \int_V \operatorname{div}(p\hat{\mathbf{v}}) dV + \int_V \hat{\mathbf{v}} \cdot \nabla p dV = - \int_{\partial V} p \hat{\mathbf{v}} \cdot \mathbf{n} dA + \int_V \hat{\mathbf{v}} \cdot \nabla p dV.$$

Operando no segundo termo, obtém-se

$$\begin{aligned} \int_V \mathbb{T}^D \cdot \hat{\mathbb{D}} dV &= \int_V \mathbb{T}^D \cdot \nabla \hat{\mathbf{v}} dV \\ &= \int_V \operatorname{div}(\mathbb{T}^D \hat{\mathbf{v}}) dV - \int_V \operatorname{div} \mathbb{T}^D \cdot \hat{\mathbf{v}} dV \\ &= \int_{\partial V} \mathbb{T}^D \mathbf{n} \cdot \hat{\mathbf{v}} dA - \int_V \operatorname{div} \mathbb{T}^D \cdot \hat{\mathbf{v}} dV. \end{aligned}$$

Voltando em (8.24), o lado esquerdo desta equação fica

$$- \int_{\partial V} p \mathbf{n} \cdot \hat{\mathbf{v}} dA + \int_V \nabla p \cdot \hat{\mathbf{v}} dV + \int_{\partial V} \mathbb{T}^D \mathbf{n} \cdot \hat{\mathbf{v}} dA - \int_V \operatorname{div} \mathbb{T}^D \cdot \hat{\mathbf{v}} dV.$$

Substituindo este resultado na equação (8.24) tem-se

$$\begin{aligned} \int_{\partial V} (-p\mathbb{I} + \mathbb{T}^D) \mathbf{n} \cdot \hat{\mathbf{v}} dA + \int_V (\nabla p - \operatorname{div} \mathbb{T}^D) \cdot \hat{\mathbf{v}} dV + \int_V \rho \dot{\mathbf{v}} \cdot \hat{\mathbf{v}} dV \\ = \int_V \mathbf{b} \cdot \hat{\mathbf{v}} dV + \int_{\partial V_a} \bar{\mathbf{a}} \cdot \hat{\mathbf{v}} dA \quad \forall \hat{\mathbf{v}} \in Var_{\mathbf{v}}. \end{aligned}$$

Assim, o Problema Variacional pode ser colocado como

Problema 2. Problema Variacional 1 - Escoamento de Fluido Newtoniano

Dado $\bar{\mathbf{a}}$ e \mathbf{b} , encontrar $\mathbf{v} \in Kin_{\mathbf{v}}$ tal que:

$$\begin{aligned} \int_{\partial V} (-p\mathbb{I} + \mathbb{T}^D) \mathbf{n} \cdot \hat{\mathbf{v}} dA + \int_V (\nabla p - \operatorname{div} \mathbb{T}^D) \cdot \hat{\mathbf{v}} dV + \int_V \rho \dot{\mathbf{v}} \cdot \hat{\mathbf{v}} dV = \\ \int_V \mathbf{b} \cdot \hat{\mathbf{v}} dV + \int_{\partial V_a} \bar{\mathbf{a}} \cdot \hat{\mathbf{v}} dA \quad \forall \hat{\mathbf{v}} \in Var_{\mathbf{v}}. \end{aligned} \quad (8.25)$$

Para $Kin_{\mathbf{v}}$ e $Var_{\mathbf{v}}$ definidos como:

$$\begin{aligned} Kin_{\mathbf{v}} &= \{ \mathbf{v} \in \mathcal{U} ; \mathbf{v} = \bar{\mathbf{v}} \text{ em } \partial V_{\bar{\mathbf{v}}} \}, \\ Var_{\mathbf{v}} &= \{ \hat{\mathbf{v}} \in \mathcal{V} ; \hat{\mathbf{v}} = 0 \text{ em } \partial V_{\bar{\mathbf{v}}} \}. \end{aligned}$$

Este é o Princípio das Potências Virtuais para escoamento de fluido Newtoniano. Assume-se também a hipótese de incompressibilidade, i.e., $\text{div}(\mathbf{v}) = 0$. Logo, pode-se reescrever $\text{div}(\mathbb{T}^D)$ como

$$\text{div}(\mathbb{T}^D) = \text{div} \left(2\mu \left[\mathbb{D} - \underbrace{\frac{1}{3} \text{tr}(\mathbb{D}) \mathbb{I}}_{=0} \right] \right) = \text{div} \left(2\mu \left(\frac{\nabla \mathbf{v} + \nabla \mathbf{v}^T}{2} \right) \right) = \mu [v_{i,j} + v_{j,i}]_{,j}.$$

Como $v_{j,ij} = v_{j,ji} = 0$ (já que o $\text{div}(\mathbf{v}) = v_{j,j} = 0$), a expressão anterior fica

$$\text{div}(\mathbb{T}^D) = \mu v_{i,jj}.$$

Tendo em conta que no modelo sob consideração somente é de interesse a componente na direção do eixo longitudinal da artéria (x_3), considera-se $\hat{\mathbf{v}} = \hat{\mathbf{v}}_{x_3} = \hat{v}_3 \mathbf{e}_3$ ⁴, logo

$$\text{div}(\mathbb{T}^D) \cdot \hat{\mathbf{v}}_{x_3} = \mu v_{i,jj} \hat{v}_3 \delta_{3i} = v_{3,jj} \hat{v}_3.$$

Algo similar acontece com ∇p

$$\nabla p \cdot \hat{\mathbf{v}}_{x_3} = p_{,3} \hat{v}_3.$$

Considerando ρf_3 como sendo a componente na direção de \mathbf{e}_3 da força de corpo \mathbf{b} , tem-se

$$\mathbf{b} \cdot \hat{\mathbf{v}}_{x_3} = \rho f_3 \hat{v}_3.$$

Dado que $\hat{\mathbf{v}}$ é arbitrário, escolhe-se $\hat{v}_3 = 0$ em $\partial V_{\mathbf{a}}$, logo

$$\mathbf{a} \cdot \hat{\mathbf{v}}_{x_3} = 0 \text{ em } \partial V_{\mathbf{a}}.$$

Considera-se também que não existe fluxo de pressão nem de tensão na fronteira. Logo, a formulação variacional para o modelo 1D fica

Problema 3. Problema Variacional 2 - Escoamento de Fluido Newtoniano Incompressível dentro de um tubo

Dado f_3 ⁵, achar $v_3 \in \text{Kin}_{v_3}$ tal que

$$\int_V (p_{,3} - \mu v_{3,jj}) \hat{v}_3 dV + \int_V \rho \dot{v}_3 \hat{v}_3 dV = \int_V \rho f_3 \hat{v}_3 dV \quad \forall \hat{v}_3 \in \text{Var}_{v_3}.$$

Para Kin_{v_3} e Var_{v_3} definidos como

$$\begin{aligned} \text{Kin}_{v_3} &= \{v_3 \in \mathcal{U} ; v_3 = \bar{v}^3 \text{ em } \partial V_{\bar{v}^3} ; v_3 = 0 \text{ em } \partial V_{\mathbf{a}}\}, \\ \text{Var}_{v_3} &= \{\hat{v}_3 \in \mathcal{V} ; \hat{v}_3 = 0 \text{ em } \partial V_{\bar{v}^3} ; \hat{v}_3 = 0 \text{ em } \partial V_{\mathbf{a}}\}. \end{aligned}$$

Do Teorema Fundamental do Cálculo das Variações chega-se na equação de Euler-

⁴Entende-se por $\hat{\mathbf{v}}_{x_3}$ uma velocidade que só tem componente na direção \mathbf{e}_3 .

⁵A força de corpo por unidade de massa na direção \mathbf{e}_3 .

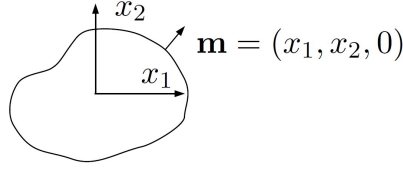


Figura 8.4: Vetor normal a curva que limita a seção transversal do lúmen do vaso.

Lagrange, para Fluido Newtoniano Incompressível dentro de um tubo

$$\rho \dot{v}_3 = -\frac{\partial p}{\partial x_3} + \mu v_{3,jj} + \rho f_3 \quad \text{em } V. \quad (8.26)$$

Dividindo ambos lados de (8.26) por ρ e integrando na seção transversal tem-se

$$\int_S v_3 dA = - \int_S \frac{1}{\rho} \frac{\partial p}{\partial x_3} dA + \int_S \frac{\mu}{\rho} v_{3,jj} dA + \int_S f_3 dA. \quad (8.27)$$

Assume-se agora a seguinte hipótese

$$p = \bar{p}(x_3, t) = \frac{1}{S} \int_S p dA, \quad \text{i.e., o perfil de pressão é plano,}$$

para o segundo termo do lado direito da equação (8.27) tem-se

$$\int_S v_{3,jj} dA = \int_S (v_{3,11} + v_{3,22}) dA + \int_S v_{3,33} dA.$$

Usando a **Identidade de Green**, sejam f e $g \in C^2(\Omega) \cap C^1(\Omega)$ então

$$\int_\Omega f \Delta g d\Omega + \int_\Omega \nabla f \cdot \nabla g d\Omega = \int_{\partial\Omega} f \frac{\partial g}{\partial \mathbf{m}} dA,$$

e para $g = v_3$ e $f = 1$ tem-se que

$$\int_S (v_{3,11} + v_{3,22}) dA = \oint_{C(x_1, x_2)} \frac{\partial v_3}{\partial \mathbf{m}} dx_3,$$

sendo $\mathbf{m} = (x_1, x_2, 0)$ o vetor unitário normal ao contorno de $C(x_1, x_2)$ como mostrado na Figura (8.4). Logo, reescreve-se (8.27) como

$$\int_S v_3 dA = - \int_S \frac{1}{\rho} \frac{\partial p}{\partial x_3} dA + \oint_{C(x_1, x_2)} \frac{\mu}{\rho} \frac{\partial v_3}{\partial \mathbf{m}} dx_3 + \int_S \frac{\mu}{\rho} v_{3,33} dA + \int_S f_3 dA. \quad (8.28)$$

Assume-se também que a componente v_3 da velocidade pode ser decomposta como

$$v_3(x_1, x_2, x_3, t) = \phi(x_1, x_2) v(x_3, t).$$

Da condição de não deslizamento tem-se que

$$\phi|_C = 0.$$

Sem perda de generalidade, adota-se ϕ de maneira a satisfazer

$$\begin{aligned} \frac{1}{S} \int_S \phi(x_1, x_2) dA &= 1, \\ \bar{v}_3^2 - v^2 &= \bar{\delta} v^2. \end{aligned} \quad (8.29)$$

Logo

$$\bar{v}_3^2 = \frac{1}{S} \int_S \phi^2 v^2 dA \Rightarrow v_3^2 - v^2 = v^2 \left[\frac{1}{S} \int_S \phi^2 dA - 1 \right] = \bar{\delta} v^2,$$

sendo $\delta = \phi^2 - 1$. Usando-se a equação (8.19) para $\xi = v_3$, tem-se

$$\frac{\partial}{\partial t}(S\bar{v}_3) + \frac{\partial}{\partial x_3}[S(\bar{v}_3^2)] = \int_S v_3 dA + \oint_{C(x_3, t)} v_3 \mathbf{w}_n dx_3, \quad (8.30)$$

e da equação 8.28

$$\int_S v_3 dA = -\frac{1}{\rho} \int_S \frac{\partial p}{\partial x_3} dA + \frac{\mu}{\rho} \left[\oint_{C(x_1, x_2)} \frac{\partial v_3}{\partial \mathbf{m}} dx_3 + \int_S v_{3,33} dA \right] + S\bar{f}_3, \quad (8.31)$$

sendo

$$\bar{v}_3 = \frac{1}{S} \int_S v_3 dA = \frac{1}{S} \int_S \phi v dA = v \cdot \underbrace{\frac{1}{S} \int_S \phi dA}_{=1} = v.$$

Trabalhando com os termos de (8.30) e (8.31) obtém-se

$$\frac{\partial}{\partial x_3} S(\bar{v}_3^2) = \frac{\partial}{\partial x_3} S(\phi^2 v^2) = \frac{\partial}{\partial x_3} v^2 \int_S \phi^2 dA = \frac{\partial}{\partial x_3} [v^2 S(1 + \bar{\delta})];$$

$$\frac{\partial}{\partial t}(S\bar{v}_3) = \frac{\partial}{\partial t} \int_S \phi v dA = \frac{\partial}{\partial t}(vS);$$

$$\oint_{C(x_1, x_2)} \frac{\partial v_3}{\partial \mathbf{m}} dx_3 = \oint_{C(x_1, x_2)} v \frac{\partial \phi}{\partial \mathbf{m}} dx_3;$$

$$-\frac{1}{\rho} \int_S \frac{\partial p}{\partial x_3} dA = -\frac{S}{\rho} \frac{\partial p}{\partial x_3}.$$

O terceiro termo do lado esquerdo de (8.31)

$$\int_{S(x_3, t)} \frac{\partial^2 v_3}{\partial x_3^2} dS,$$

onde $S(x_3, t)$ é a seção transversal da artéria para o instante t no ponto x_3 do eixo da

mesma, pode ser reescrito utilizando a regra de Leibnitz, ou seja

$$\frac{\partial}{\partial z} \int_{a(z)}^{b(z)} f(x, z) dx = \int_{a(z)}^{b(z)} \frac{\partial f(x, z)}{\partial z} dx + f(b(z), z) \frac{\partial b}{\partial z} - f(a(z), z) \frac{\partial a}{\partial z}. \quad (8.32)$$

Por sua vez pode-se escrever em coordenadas cilíndricas a componente da velocidade do sangue na direção do eixo \mathbf{e}_3

$$v_3(x_1, x_2, x_3, t) = \tilde{v}_3(r, \theta, x_3, t).$$

Tem-se então que

$$\begin{aligned} \frac{\partial}{\partial x_3} \int_{S(x_3, t)} v_3 dS &= \frac{\partial}{\partial x_3} \int_0^{2\pi} \int_0^{R(\theta, x_3, t)} \tilde{v}_3(r, \theta, x_3, t) r dr d\theta \\ &= \int_0^{2\pi} \frac{\partial}{\partial x_3} \int_0^{R(\theta, x_3, t)} \tilde{v}_3(r, \theta, x_3, t) r dr d\theta. \end{aligned}$$

Aplicando (8.32) tem-se:

$$\begin{aligned} \frac{\partial}{\partial x_3} \int_{S(x_3, t)} v_3 dS &= \underbrace{\int_0^{2\pi} \{ \tilde{v}_3 r \}_{|_{R(\theta, x_3, t)}} \frac{\partial R}{\partial x_3} |_{R(\theta, x_3, t)} d\theta}_{=0 \text{ já que } \tilde{v}_3|_R = 0} \\ &+ \int_0^{2\pi} \int_0^{R(\theta, x_3, t)} \frac{\partial \tilde{v}_3(r, \theta, x_3, t)}{\partial x_3} r dr d\theta \\ &= \int_{S(x_3, t)} \frac{\partial v_3}{\partial x_3} dS. \end{aligned}$$

Logo, dado que $\tilde{v}_3|_R = 0$, tem-se

$$\frac{\partial}{\partial x_3} \int_{S(x_3, t)} v_3 dS = \int_{S(x_3, t)} \frac{\partial v_3}{\partial x_3} dS.$$

Como $\tilde{v}_3|_R \equiv 0 \quad \forall x_3 \Rightarrow \frac{\partial \tilde{v}_3}{\partial x_3}|_R = 0$ pode-se aplicar um raciocínio idêntico ao anterior para obter

$$\frac{\partial^2}{\partial x_3^2} \int_{S(x_3, t)} v_3 dS = \int_{S(x_3, t)} \frac{\partial^2 v_3}{\partial x_3^2} dS. \quad (8.33)$$

Logo, usando (8.33) no terceiro termo do lado direito de (8.31), tem-se que

$$\frac{\mu}{\rho} \int_S v_{3,33} dA = \frac{\mu}{\rho} \frac{\partial}{\partial x_3} \int_S v_3 dA = \frac{\mu}{\rho} \frac{\partial}{\partial x_3} (S \bar{v}_3).$$

Finalmente, recordando que $\bar{v}_3 = v$, a equação de Balanço do Momento para o problema sob consideração é dada por

Balanco do Momento

$$\begin{aligned}
 \frac{\partial}{\partial t}(Sv) + \frac{\partial}{\partial x_3}[v^2 S(1 + \bar{\delta})] + \frac{S}{\rho} \frac{\partial p}{\partial x_3} &= S\bar{f}_3 + vN + \frac{\mu}{\rho} \frac{\partial^2}{\partial x_3^2}(Sv) \\
 \text{sendo} \quad N &= \frac{\mu}{\rho} \oint_{C(x_3, t)} \frac{\partial \phi}{\partial \mathbf{m}} dx_3 \\
 \bar{\delta} &= \frac{1}{S} \int_S \phi^2 dA - 1.
 \end{aligned} \tag{8.34}$$

Com isto completa-se o sistema de equações correspondente ao modelo unidimensional de fluxo em vasos de grande porte do SCVH, incluindo fluxos de saída devido a ramificações e/ou permeabilidade das paredes arteriais.

Tem-se então que o sistema é dado pelas equações de **Balanco de Massa** (Eq. (8.21)) e **Balanco de Momento** (Eq. (8.34)). Logo, o sistema fica

$$\begin{aligned}
 \frac{\partial S}{\partial t} + \frac{\partial}{\partial x_3} Sv + \psi &= 0 \\
 \frac{\partial}{\partial t}(Sv) + \frac{\partial}{\partial x_3}[v^2 S(1 + \bar{\delta})] + \frac{S}{\rho} \frac{\partial p}{\partial x_3} &= S\bar{f}_3 + vN + \frac{\mu}{\rho} \frac{\partial^2}{\partial x_3^2}(Sv) \\
 \text{para } \bar{\delta} &= \frac{1}{S} \int_S \phi^2 dA - 1 \\
 N &= \frac{\mu}{\rho} \oint_{C(x_3, t)} \frac{\partial \phi}{\partial \mathbf{m}} dx_3.
 \end{aligned}$$

Pode-se perceber que o sistema tem três incógnitas (i.e., S , v e p), mas somente duas equações. Para completar este sistema é introduzida mais uma equação constitutiva, relacionando a pressão com a área da seção transversal

$$p(x_3, t) = p(S(x_3, t), x_3, t).$$

Assumindo, por exemplo, um modelo elástico que relacione a área luminal S e a pressão p , pode-se adotar a relação constitutiva linear utilizada em [193], ou seja

$$p = p_0 + \frac{Eh_0}{R_0} \left(\sqrt{\frac{S}{S_0}} - 1 \right).$$

sendo h_0 , R_0 e A_0 a espessura, o raio e a área da seção transversal do vaso respectivamente sob a ação da pressão referencial p_0 , completando com isso o sistema de equações correspondentes ao modelo.

Modelo 1D

Fluxo Unidimensional em Vasos de Grande Porte - Sistema de Equações

$$\begin{aligned}
 \frac{\partial S}{\partial t} + \frac{\partial}{\partial x_3} S v + \psi &= 0 \\
 \frac{\partial}{\partial t}(S v) + \frac{\partial}{\partial x_3}[v^2 S(1 + \bar{\delta})] + \frac{S}{\rho} \frac{\partial p}{\partial x_3} &= S f_3 + v N + \frac{\mu}{\rho} \frac{\partial^2}{\partial x_3^2}(S \bar{v}) \\
 p &= p_0 + \frac{E h_0}{R_0} \left(\sqrt{\frac{S}{S_0}} - 1 \right) \\
 \text{sendo } N &= \frac{\mu}{\rho} \oint_{C(x_3, t)} \frac{\partial \phi}{\partial \mathbf{m}} dx_3 \\
 \text{e } \bar{\delta} &= \frac{1}{S} \int_S \phi^2 dA - 1. \tag{8.35}
 \end{aligned}$$

No caso, uma equação constitutiva visco-elástica foi utilizada para relacionar a pressão e a áreas da seção transversal, ou seja

$$p = p_0 + \frac{E h_0}{R_0} \left(\sqrt{\frac{S}{S_0}} - 1 \right) + \frac{k h_0}{R_0} \frac{1}{2 \sqrt{S_0 S}} S_{,t}. \tag{8.36}$$

Capítulo 9

HeMoLab - Laboratório de Modelagem em Hemodinâmica

O aumento na complexidade das intervenções cirúrgicas tem exigido um maior uso de técnicas precisas que permitam prever o resultado dos procedimentos médicos. O atual grau de desenvolvimento alcançado pelas técnicas de modelagem computacional unida ao crescimento da performance dos computadores permite o estudo, desenvolvimento e solução de modelos computacionais capazes de avaliar a resposta do corpo humano a estes procedimentos. A modelagem e simulação computacional, computação gráfica e realidade virtual fornecem as ferramentas necessárias para representar, com alto grau de detalhe, os fenômenos que acontecem dentro do organismo de um paciente de maneira não invasiva. Levando em conta essas idéias, foi desenvolvida a ferramenta computacional HeMoLab - Laboratório de Modelagem em Hemodinâmica, que reúne as mais novas técnicas de modelagem do Sistema Cardiovascular Humano com o objetivo de se tornar útil na difícil tarefa de modelar este sistema. Desenvolvimento de modelos personalizados do sistema arterial completo, processamento de imagens médicas para obtenção de geometrias dos pacientes, geração de modelos tridimensionais de distritos arteriais específicos, simulação e ajuste do sistema arterial integral para reproduzir o observado em um paciente, integração destes dados com as geometrias extraídas das imagens do paciente são algumas das funcionalidades mais destacáveis desta ferramenta.

No Capítulo 8 foram apresentados os modelos e metodologias necessárias na modelagem computacional do SCVH. O criação e desenvolvimento destes tipos de modelos é um processo altamente complexo e requer o tratamento de informações heterogêneas. Edição de árvores arteriais, processamento de imagens médicas e geometrias são somente algumas das ferramentas necessárias para gerar os modelos. Até o presente momento, a geração de modelos do SCVH era feita de maneira artesanal, um processo delicado que envolve diferentes ferramentas e dados em diversos formatos. Neste processo resulta difícil evitar a introdução de erros humanos assim como a sua detecção. Por este motivo surge a necessidade de desenvolver ferramentas computacionais para os especialistas (sejam engenheiros, técnicos ou médicos) reunindo os vários passos da modelagem em um único ambiente de trabalho, padronizando as informações tratadas.

Uma das potenciais aplicações desta tecnologia é no planejamento terapêutico e cirúrgico das mais variadas doenças cardiovasculares. Neste caso, é fundamental o uso de modelos matemáticos que representem a dinâmica SCVH, assim como a sua simulação computacional. Diferentes características desta dinâmica são de interesse, e.g., campos de velocidade do sangue dentro das artérias, propagação das ondas de pressão na árvore arterial, estado de tensões nas paredes arteriais, tensões oscilatórias, etc. Os modelos devem permitir ainda a incorporação de informações específicas dos pacientes. Estas informações podem chegar de diferentes maneiras e formatos, podendo ser até incompletas

em alguns casos (imagens médicas, propriedades mecânicas dos tecidos, leituras de pressão ou fluxo em determinados pontos da anatomia, etc.).

Algumas das aplicações mais destacáveis desta ferramenta na medicina cardiovascular são:

- *Auxiliar no treinamento e aprimoramento de técnicas cirúrgicas:* permitindo a estudantes aprender os procedimentos ou a cirurgiões experientes aperfeiçoar as suas técnicas e habilidades.
- *Estudar o resultado de diferentes intervenções cirúrgicas:* permitindo determinar o melhor procedimento a ser utilizado em um determinado paciente, a melhor maneira de realizar este procedimento e até avaliar os potenciais resultados.
- *Representar fenômenos como absorção, difusão e transporte de substâncias bioquímicas nos tecidos da parede arterial:* reproduzindo, por exemplo, o que acontece ao utilizar *stents* revestidos com fármacos.
- *Indicar risco de ruptura de aneurismas e desenvolvimento de hiperplasia:* diferentes estudos indicam que fatores hemodinâmicos (tensões oscilatórias e a conseqüente fadiga dos tecidos, altos gradientes de tensão nas paredes arteriais, regiões de recirculação, elevados tempos de residência das partículas) estão intimamente relacionados ao risco de ruptura de aneurismas assim como ao desenvolvimento de hiperplasia. Os modelos apresentados no Capítulo 8 permitem estudar este tipo de fatores.

A abrangência e complexidade dos modelos mencionados faz com que o desenvolvimento desta ferramenta seja uma tarefa complexa que requer interação de especialistas de diferentes áreas (engenharia, medicina, computação, entre outras). Desta interação surgirá uma série de requisitos básicos que esta ferramenta deveria atender assim como o *framework* mais apropriado para desenvolvimento da ferramenta e os diferentes módulos que a mesma deveria disponibilizar. Nas próximas seções serão apresentados em detalhe estes aspectos.

9.1 Análise de requisitos

A criação de modelos do SCVH exige integração de diferentes funcionalidades e dados. Pode ser de interesse analisar em detalhe a hemodinâmica (fluxo, pressão, estado de tensões, etc.) em uma região do sistema arterial de um paciente. Para isto é necessário reconstruir geometrias a partir de imagens médicas do paciente para reproduzir o fluxo sanguíneo nos vasos daquele indivíduo. Uma outra necessidade será parametrizar uma árvore arterial padrão para conseguir aproximar com a maior precisão possível o observado em um paciente. Para isto, devem-se fornecer ferramentas para modificar as propriedades mecânicas de uma determinada árvore, assim como adicionar ou eliminar segmentos arteriais nas regiões que tenham maior ou menor relevância, respectivamente. Aparecem então diferentes requerimentos a serem considerados no desenvolvimento:

- *Processamento de imagens:* para poder extrair a geometria das artérias de um determinado paciente é necessário fornecer ferramentas de processamento e visualização destas imagens. Uma vez processadas as imagens, a geometria das artérias ou estruturas de interesse nelas presentes devem ser reconstruídas tridimensionalmente (e.g., triangulação das superfícies).
- *Visualização/Edição da árvore 1D:* devem-se fornecer ferramentas que permitam visualizar e editar os parâmetros da árvore arterial 1D de maneira a incluir as características e comportamentos específicos de cada paciente.
- *Visualização/Edição da geometrias 3D:* uma vez obtidas as geometrias (superfícies) a partir das imagens dos pacientes, as mesmas devem ser processadas/refinadas para, a partir delas, gerar as malhas volumétricas que serão utilizadas para calcular as aproximações numéricas das equações de fluidos/sólidos correspondentes.
- *Acoplamentos entre os diferentes modelos:* para fazer com que os modelos funcionem de maneira integrada, é necessário acoplar os modelos 1D e 3D.
- *Solução numérica de equações diferenciais:* os modelos mencionados anteriormente fornecem uma série de equações cujas soluções correspondem aos campos de interesse (velocidade, pressão, etc.). As soluções destas equações podem ser aproximadas numericamente.
- *Visualização dos resultados:* os resultados do cálculo numérico são gigantescos volumes de dados. Para poder interpretá-los, estes dados devem ser apresentados de uma maneira amigável utilizando técnicas de visualização científica.

Estes requerimentos foram separados em diferentes módulos. Na sequência são analisados os requerimentos de cada um deles.

9.1.1 Módulo 1: Processamento de Imagens

O processamento de imagens tem sido largamente utilizado na medicina [91, 187]. Talvez esta seja a maneira mais efetiva de utilizar dados reais em simulações computacionais. O objetivo deste módulo é permitir ao usuário ler e processar imagens médicas. Geralmente as imagens médicas são recebidas no formato padrão DICOM [26] (o mais utilizado pelos aparelhos de aquisição hoje em dia). No entanto, algumas vezes estes dados podem ser fornecidos em outros formatos, por exemplo como sequências de imagens BMP, JPG, PNG ou em uma única imagem que contém os diferentes planos. Por este motivo, este módulo deve permitir a leitura de diferentes formatos de imagens.

Existem diversos filtros usualmente utilizados para processar uma imagem [17, 27, 64, 75, 113]. No caso de imagens médicas, estas costumam apresentar ruído, o que dificulta o processo de identificação das diferentes estruturas. Filtros de suavização Gaussiano, suavização anisotrópico, filtros morfológicos, dentre outros, são usualmente utilizados para remover ruído e melhorar a qualidade de uma imagem degradada. Uma vez que a imagem foi filtrada, é necessário identificar as diferentes regiões que a compõem. Este processo,

chamado de segmentação, será utilizado neste caso para identificar artérias ou outras estruturas relacionadas ao sistema cardiovascular presentes nas imagens (paredes arteriais, aneurismas, etc.). Existem na bibliografia uma variedade de métodos para segmentar imagens, variando de acordo com a aplicação específica [30, 112, 114, 121, 124, 125, 208]. Tanto os filtros de imagens quanto os métodos de segmentação, possuem parâmetros de controle que devem ser calibrados de acordo com cada aplicação. Assim, deve-se permitir esse ajuste de forma sistemática. Além do mais, processamento de imagens médicas é na atualidade uma das áreas mais ativas de produção científica. Por este motivo, devem-se considerar mecanismos que permitam incorporar métodos novos que possam ser desenvolvidos no futuro como, por exemplo os desenvolvidos neste trabalho (Parte I), além dos já existentes.

Finalmente, uma vez selecionada a região de interesse dentro da imagem, a geometria (triangulação) que a representa deve ser gerada utilizando algum método de reconstrução tridimensional (e.g., *Marching Cubes* [122] e os métodos de segmentação propostos neste trabalho, Parte II). A partir desta primeira versão da geometria do vaso (ou estrutura segmentada), será gerada a malha de elementos finitos correspondente (Seção 9.1.3).

9.1.2 Módulo 2: Modelo 1D

A maior parte do sistema arterial é representada utilizando uma simplificação unidimensional das equações de Navier-Stokes, este é chamado de Modelo 1D [61, 95] (apresentados na Seção 8.1.1).

Baseada na topologia da árvore arterial proposta por Avolio [19], foi desenvolvida uma geometria “padrão” composta de 128 segmentos arteriais que representam as maiores artérias do corpo humano. O primeiro requerimento deste módulo é a edição deste tipo de árvores (e outras nela baseadas) que se encontram armazenadas em arquivos de texto formatados (o formato destes arquivos é o definido pelo SolverGP, Seção 9.3.6).

Existem nestas árvores dois tipos de elementos: segmentos e terminais *Windkessel*. Cada segmento representa uma artéria, fornecendo informações de fluxo, pressão e área em todos os pontos. Os terminais condensam em um ponto (modelo 0D, baseados no modelo proposto em [66]) características de uma região completa, permitindo simplificar a árvore de maneira que não é necessário fornecer propriedades dos distritos simplificados, porém os mesmos não fornecem detalhes de fluxo e pressão nas regiões condensadas.

Dois tipos de informações caracterizam este modelo:

- *geométricas* e
- *paramétricas*.

Os dados *geométricos* definem a topologia da árvore representada. Ao incorporar maior quantidade de elementos nesta árvore espera-se uma maior precisão nos resultados a ela associados. Para modificar a geometria da árvore, será necessário fornecer ferramentas de edição que permitam agregar e eliminar elementos desta árvore, assim como modificar os já existentes.

Os dados *paramétricos* são aqueles que definem as características mecânicas das artérias, quais sejam: diâmetro da artéria¹; espessura da parede arterial; percentual de elastina, colágeno e músculo liso na parede, pressão externa no vaso, permeabilidade da parede, curva de ejeção do coração, dentre outras. Estes parâmetros devem assumir valores dentro de determinados intervalos, caso contrário o modelo não teria sentido físico. A verificação deste valores evitará então inconsistências (numéricas e físicas) nos modelos.

9.1.3 Módulo 3: Modelo 3D

Outra funcionalidade importante é permitir a simulação do fluxo sanguíneo em determinados trechos da árvore arterial permitindo analisar em detalhe as características do fluxo naquela região.

As superfícies geradas a partir das imagens médicas costumam ser de má qualidade (por ter triângulos tipo “agulhas” onde a área se aproxima de zero mas o comprimento das arestas é maior a zero, ou muito pequenos onde todas as arestas são da mesma ordem mas a área do elemento é próxima de zero). Por este motivo, são necessárias ferramentas² que permitam melhorar as superfícies, eliminando este tipo de característica indesejável. Uma malha de boa qualidade se caracteriza por estar composta de triângulos praticamente equiláteros (todos os seus lados são quase iguais). A precisão dos resultados (associados aos modelos) dependerá diretamente da qualidade das malhas e do tamanho dos seus elementos.

Também existe a necessidade de incorporar informações específicas dos pacientes tais como espessura das paredes arteriais, percentual de colágeno, elastina e músculo liso que irão influir na resposta do modelo do sistema arterial e enriquecê-los. Pode ser de interesse também modelar o comportamento do fluido utilizando diferentes relações constitutivas (fluido Newtoniano ou não-Newtoniano).

Todas estas informações deverão ser salvas de maneira persistente (i.e., em arquivos). Em particular, no formato definido pelo SolverGP, o qual será utilizado para calcular as soluções destes modelos.

9.1.4 Módulo 4: Acoplamento

Uma vez gerada a árvore 1D e a(s) geometria(s) 3D, e determinados os correspondentes parâmetros, estas informações devem ser reunidas de maneira a gerar um único modelo acoplado. Os passos seguidos são:

- identificar/selecionar os segmentos arteriais do modelo 1D que serão substituídos por uma geometria 3D,
- remover estes segmentos da árvore 1D,
- associar os pontos de entrada/saída da geometria 3D (identificados por grupos de elementos) com os terminais correspondentes na árvore 1D.

¹O diâmetro é uma característica geométrica da artéria, mesmo assim este entra como um parâmetro no modelo matemático simplificado, por este motivo é considerado um dado paramétrico.

²Filtros que operam sobre a malha modificando a posição dos pontos e a conectividade dos triângulos

Para isto será necessário criar grupos de elementos no modelo 3D que caracterizem os pontos de conexão entre ambos modelos. As informações correspondentes aos modelos 1D, 3D e ao acoplamento devem ser salvas no formato do SolverGP.

9.1.5 Módulo 5: Simulação Numérica

O *solver* numérico SolverGP é um *framework* pensado para atender diferentes necessidades. O problema modelado é descrito utilizando uma série de arquivos (os quais contém informação da geometria, acoplamentos, parâmetros e condições iniciais e de contorno associadas ao problema) em um formato definido especificamente para este *framework*. Nestes arquivos encontra-se a informação necessária para montar o sistema de equações correspondente ao problema modelado.

Os parâmetros utilizados na resolução dos sistemas de equações (*solver* numérico direto ou iterativo, esquema de integração temporal, etc.) de cada problema são também definidos nestes arquivos. Logo, o problema encontra-se completamente definido em termos dos arquivos de entrada do SolverGP. Por este motivo, a leitura/escrita destes arquivos é um ponto fundamental no aproveitamento desta ferramenta.

9.1.6 Módulo 6: Visualização e Análise

Uma vez realizada a simulação numérica, a solução do problema modelado (e.g., campos escalares no caso da pressão, campos vetoriais no caso da velocidade e deslocamento, etc.) é representada pelo valor (numérico) destes campos em cada ponto do domínio. Estas informações representam uma quantidade enorme de dados os quais não podem ser analisados diretamente sem pós processamento. Para interpretar estes dados, são utilizadas técnicas de visualização científica [73, 83, 175, 176] que permitem representar a informação de maneira compreensível. Por exemplo, o campo de velocidades pode ser representado utilizando *glyphs* que indiquem a direção e intensidade do campo nos diferentes pontos do domínio. O deslocamento das paredes arteriais provocado pela variação da pressão no interior dos vasos pode ser visualizado aplicando uma deformação na geometria consistente com os deslocamentos em cada ponto. Existem inúmeras maneiras de representar estes dados dependendo da informação sob análise. Para isto, são necessárias estruturas de dados (para armazená-los), filtros (para processar esta informação) e bibliotecas gráficas especialmente desenhadas para visualização de grandes volumes de dados atendendo a boa performance e tirando proveito da tecnologia disponível (e.g., OpenGL[152]).

9.2 O *framework* utilizado

Nesta seção é analisado o ambiente de trabalho e ferramentas utilizadas no desenvolvimento. Em seguida é feita uma breve revisão da arquitetura do software ParaView, o qual foi utilizado como base para o desenvolvimento.

9.2.1 Ambiente de trabalho

O desenvolvimento do HeMoLab está sendo feito no sistema operacional Linux, na distribuição Ubuntu 6.06.1 LTS [191]. A linguagem de desenvolvimento é ANSI-C++ [184], e os compiladores utilizados são c++/gcc [71] (correspondentes ao GCC versão 4.0.3).

O IDE (*Integrated Development Environment*) escolhido é o Eclipse SDK [178] (versão 3.1.2), com os módulos (*plugins*) Eclipse C/C++ *Development Tooling* - CDT [189] (versão 3.0.2) para edição de arquivos c/c++ e Subclipse [185] (versão 0.9.105) para interação com o repositório. O controle de versões é feito utilizando um repositório Subversion [186] (SVN - 1.3.1). Na configuração de projeto é utilizado o CMake [123] (versão 2.2 - *patch* 3).

Para gerenciar o projeto é utilizado o dotProject [47] (versão 2.0.1), e para gerenciar as tarefas está sendo utilizado o Bugzilla [34] (versão 2.16.7).

9.2.2 ParaView

Para suprir os requerimentos mencionados na seção anterior, está sendo desenvolvido um software que reúne as ferramentas mencionadas dentro de um mesmo ambiente de trabalho. O ambiente escolhido para desenvolver as ferramentas foi o ParaView [98], o qual será o marco destas funcionalidades.

ParaView utiliza como base a biblioteca *Visualization ToolKit* (VTK [99]), uma das mais utilizadas para visualização e renderização 3D. VTK foi implementado em C++ e usa primitivas OpenGL para a renderização. Já a interface de usuário (*User Interface* - UI) do ParaView foi escrita na linguagem TCL. O ParaView foi criado pela Kitware Inc. juntamente com os Laboratórios de Los Alamos, Sandia e Lawrence Livermore. Este software é uma aplicação originalmente desenvolvida para visualização de dados científicos, com interface gráfica simples e que pode ser parametrizada pelos próprios usuários. Este software também incorpora métodos de processamento de imagens, além de proporcionar suporte para computação distribuída e portabilidade para vários sistemas operacionais. Também possui a capacidade de se comunicar através de diferentes formatos de arquivos. Dependendo da arquitetura em que as aplicações funcionam, o tamanho dos conjuntos de dados que o ParaView suporta pode variar.

Embora o ParaView seja, em boa parte, um programa interpretado, este apresenta o rendimento de um programa compilado, com as facilidades e extensibilidade que uma linguagem interpretada oferece. Este *software* pode ser estendido por meio do uso de arquivos XML que descrevam novos módulos, aproveitando rotinas escritas na linguagem TCL que por sua vez utiliza bibliotecas compiladas feitas em C++.

ParaView apresenta uma arquitetura modular, baseada na idéia que o processamento de dados, a renderização e o controle de interface de usuário podem funcionar em processos separados, em diferentes computadores e até em diferentes plataformas. Talvez o aspecto mais significativo do ParaView é a separação da interface gráfica de usuário (*Graphical User Interface* - GUI) dos servidores de processamento. Para atender estas necessidades foi necessária a criação de um Gerenciador de servidores (chamado *Server*

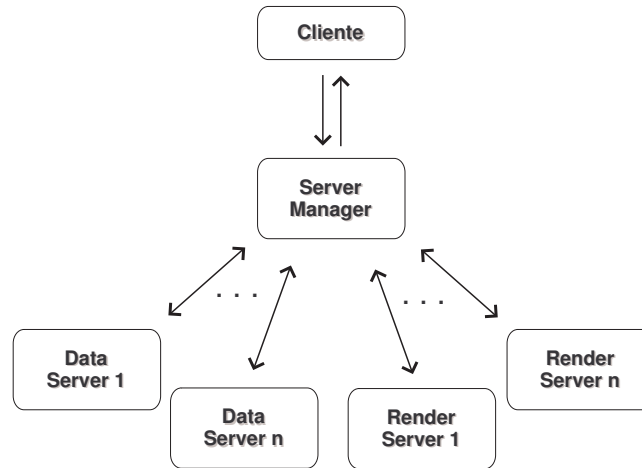


Figura 9.1: Arquitetura Cliente-Servidor do ParaView.

Manager, responsável pela comunicação entre os diferentes processos) e de uma linguagem para comunicar objetos em diferentes processos (a Linguagem Cliente-Servidor, foi especialmente criada para chamar métodos em objetos VTK de maneira remota).

As características pelas quais o ParaView resultou a melhor alternativa para ser o ambiente de desenvolvimento das funcionalidades acima descritas são resumidas a seguir:

- *Aplicação Open-Source:* o ParaView é uma aplicação *Open Source* sob a licença GNU, implicando que qualquer pessoa ou grupo tem o direito de usar e/ou distribuir o software do modo que este é ou introduzindo as modificações que ele ou eles bem entendam necessárias sem pagar por direitos autorais ou patentes.
- *Desenvolvido na linguagem ANSI-C++:* o que o faz eficiente e robusto, além de permitir que o mesmo seja recompilado em diferentes plataformas.
- *Extensibilidade:* a arquitetura do ParaView permite estender facilmente as suas funcionalidades, permitindo agregar novas rotinas de maneira simples.
- *Paralelismo:* esta ferramenta foi desenvolvida de maneira a poder trabalhar em vários processos e inclusive em vários computadores ao mesmo tempo. Esta característica torna-se muito atraente quando o volume dos dados é muito grande.
- *Versatilidade:* permite tratar diferentes formatos de dados (e incluir novos) de maneira independente sem ser esta uma limitação na sua performance.
- *Baseado em VTK:* esta biblioteca é amplamente utilizada e conhecida, o que facilita a sua manipulação e o desenvolvimento de novos módulos.

9.2.3 Arquitetura do ParaView

O ParaView baseia-se em uma arquitetura Cliente-Servidor. Sua arquitetura foi desenvolvida de forma modular, permitindo a separação do processamento de dados, a visualização e a interface de usuário em diferentes processos/computadores. Desta maneira, não resulta difícil montar um cenário no qual o resultado de uma simulação (geometria + dados), processada em um supercomputador, é visualizada em um computador com menor quantidade de processadores mas com um hardware de renderização 3D de alta performance enquanto a interface de usuário é apresentada em um computador pessoal simples.

Neste contexto existem quatro componentes que integram o ParaView:

- *Client* (CL);
- *Server Manager* (SM);
- *Render Server* (RS) e
- *Data Server* (DS).

Cada um destes componentes tem funções específicas, as mesmas são descritas a seguir.

9.2.4 *Client* - CL

O Cliente é responsável por passar os valores/dados, fornecidos pelo usuário, para o *Server Manager*. Estes dados são lidos utilizando controles de interface (abas) comuns, e.g., caixas de texto, botões, etc. Também é responsabilidade do cliente manter consistentes as informações mostradas na interface com os dados que se encontram no *Server Manager*.

As GUI dos módulos de leitura/escrita/filtragem são especificadas utilizando arquivos XML. Existem dois tipos de arquivos XML: a) XML Cliente: especificam as abas necessárias na GUI para controlar os leitores/escritores/filtros que serão utilizados pelo usuário para introduzir dados (e.g., parâmetros dos filtros); b) XML Servidor: relacionam as classes no *Server Manager* (*proxies*) com os correspondentes leitores/escritores/filtros no servidor (objetos VTK). Utilizando estes *proxies* as classes de interface do cliente se comunicam com as classes nos diferentes servidores.

Para cada aba que se deseje agregar na interface de um determinado módulo, existe um marcador (*tag*) correspondente no XML. Estes controles podem ser abas simples (e.g., um botão, caixa de texto, etc.) ou componentes mais complexos compostos agregando várias abas simples (e.g., um painel que contém vários campos, barras de rolagem, e botões). Desta maneira, ao criar novos filtros, podem ser criados controles que permitam ao usuário introduzir as informações específicas daquele filtro.

9.2.5 *Server Manager* - SM

O *Server Manager* (ou Gerenciador de Servidores) é o módulo responsável pela comunicação entre o *Render Server*, o *Data Server* e o Cliente. Também é responsável por manter uma cópia do estado dos servidores por meio do uso de propriedades. Nele existem

referências a *proxies* (objetos locais) que se comunicam com os objetos VTK (criados remotamente) nos servidores, fornecendo também uma interface para criação e gerenciamento de objetos naqueles processos possivelmente em paralelo. O *Server Manager* possui os seguintes componentes:

- *Proxies* - **vtkSMProxy** e suas sub-classes, fornecem uma interface local para se comunicar com os objetos VTK, que residem no(s) servidor(es). As suas responsabilidades são as seguintes: criar os objetos VTK no servidor; manter referências aos objetos VTK no servidor usando os *Ids Cliente Servidor(CSID)*³; manter uma cópia do estado dos objetos que se encontram no lado servidor;
- Propriedades - **vtkSMProperty** e suas sub-classes, fornecem o acesso do cliente à interface (API - *Application Programming Interface*) dos objetos no lado servidor representado pelos *proxies*;
- Domínios - **vtkSMDomain** e suas sub-classes, representam uma coleção/intervalo de possíveis valores que as propriedades podem assumir.

Os diferentes *proxies* são gerenciados pelo *Proxy Manager* (**vtkSMProxyManager**). O *Proxy Manager* é uma classe *Singleton*[68] (existe uma única instância dela na memória) responsável por manter referências aos *proxies* e suas correspondentes propriedades de acordo ao especificado nos arquivos XML do servidor (os quais serão descritos em breve). Cada *proxy* está relacionado a um objeto no servidor por meio do CSID. O CSID e o objeto no servidor são criados pelo *Process Module* (**vtkProcessModule**, também *Singleton*), responsável por atribuir um CSID único no sistema a cada objeto criado através dele e de obter a referência desse objeto dado o CSID.

A configuração do SM é armazenada em arquivos XML. Estes arquivos irão especificar com que objetos no servidor cada *proxy* se comunica e quais são as propriedades a ele associadas. Desta maneira, o usuário ou o desenvolvedor tem um mecanismo dinâmico de associar *proxies* (no SM) com objetos VTK (nos servidores). Enquanto os arquivos XML do cliente especificam que controles devem aparecer na interface para permitir ao usuário alterar os dados e parâmetros dos diferentes módulos e filtros, o XML do servidor determina qual será o *proxy* responsável por tratar/enviar estes dados ao servidor e quais serão as propriedades associadas e esse *proxy*.

Dentro dos arquivos XML do servidor existe um único *tag* XML chamado **ServerManagerConfiguration**, por sua vez este contém *tags* do tipo **ProxyGroup**. Este marcador é utilizado para agrupar *proxies* de acordo com a sua funcionalidade (e.g., leitores/escritores, filtros, *widgets* 3D, etc.), também define no seu atributo **name** o grupo ao qual *proxies* definidos dentro dele pertencem, alguns exemplos podem ser “*sources*”, “*filters*” e “*3d.widgets*”. Um breve exemplo é apresentado na Fig. 9.2.

Um exemplo simples da chamada de um *proxy* via XML é o caso dos leitores, fontes e filtros (todos estes tipos de filtros são considerados fontes de dados ou *sources*).

³O id cliente servidor é um valor único a nível de sistema que permite identificar objetos no(s) servidor(es).

```

<ServerManagerConfiguration>
  <ProxyGroup name="sources">
    <SourceProxy name="CubeSource" class="vtkCubeSource">
      [...]
    </SourceProxy>
  </ProxyGroup>
  <ProxyGroup name="filters">
    <SourceProxy name="SurfaceSmooth" class="vtkSmoothPolyDataFilter">
      [...]
    </SourceProxy>
  </ProxyGroup>
</ServerManagerConfiguration>

```

Figura 9.2: Exemplo código XML com *proxies* em diferentes grupos.

Cada *source* é representado (no XML) pelo elemento `SourceProxy` (um sub-marcador do `ProxyGroup`). O marcador `SourceProxy` tem os seguintes atributos: `name`, que é um identificador **único** desse *source proxy*, o mesmo não pode estar repetido em outro *source proxy*; e `class`, que é o nome da classe *source* que será criada no servidor ao se criar esse *proxy*. Dentro destes *tags*, encontram-se definidas as propriedades associadas a este *proxy*, como mostrado na Fig. 9.3. Desta maneira é feita uma associação única entre um *source proxy* e uma classe VTK (*source*) no servidor. O *proxy* cumpre a função de estabelecer a comunicação entre o processo no qual o *proxy* se encontra e a classe correspondente em um servidor. Este exemplo considera o caso no qual o *source proxy* supre as funcionalidades necessárias para dialogar com um objeto do tipo `CubeSource` no servidor. De uma maneira semelhante são associados outros tipos de *proxies* com outras classes VTK (e.g., filtros mais complexos, *widgets* 3D, etc.).

As propriedades são utilizadas para representar a chamada a um método (e qualquer argumento a ele associado) de um objeto VTK armazenado no CL, SM, DS ou RS. As propriedades têm um estado interno (atributos que representam os parâmetros de entrada do método) o qual pode ser enviado para o objeto VTK associado a ela. Geralmente, o desenvolvedor não cria diretamente instâncias das propriedades, elas são criadas e atribuídas quando o arquivo e/ou *strings* de configuração XML são processados pelo ParaView. Na Fig. 9.3 é apresentado um exemplo simples de configuração de um *source proxy* com propriedades. Neste exemplo também é utilizado o domínio, o qual especifica a faixa de valores permitidos para esta propriedade.

9.2.6 *Render Server* - RS

Este módulo representa todo o processamento requerido para mostrar e renderizar um ou vários conjuntos de dados. Toda vez que um conjunto de dados é carregado ou gerado pelo ParaView, ele é primeiramente filtrado para gerar uma representação poligonal. Posteriormente, é criada uma série de *mappers* (transformam coordenadas reais em coordenadas na tela), atores, propriedades e tabela de cores (LookUpTables - LUTs) que permitirão visualizar a representação poligonal destes dados.

```

<SourceProxy name="CubeSource" class="vtkCubeSource">
  <DoubleVectorProperty
    name="XLength"
    command="SetXLength"
    number_of_elements="1"
    default_values="1.0">
    <DoubleRangeDomain name="range" min="0"/>
  </DoubleVectorProperty>
  <DoubleVectorProperty
    name="YLength"
    command="SetYLength"
    number_of_elements="1"
    default_values="1.0">
    <DoubleRangeDomain name="range" min="0"/>
  </DoubleVectorProperty>
</SourceProxy>

```

Figura 9.3: Exemplo código XML com propriedades e domínio.

Existe no ParaView uma API abstrata para simplificar o processo de agregar novos algoritmos de renderização. O *Render Module* (representado pela classe `vtkRenderModule` e a sua GUI) é o objeto de renderização de mais alto nível no ParaView. O *Render Module* é responsável pela criação e gerenciamento de todas as janelas de renderização, manejo dos eventos de renderização e algoritmos de renderização em paralelo. No caso de estudo sob consideração, os algoritmos de renderização disponibilizados pelo ParaView serão suficientes.

9.2.7 Data Server - DS

O *Data Server* é o módulo do ParaView responsável por carregar ou criar conjuntos de dados e aplicar filtros nesses dados. Os conjuntos de dados podem ser partidos para ser manipulados em diferentes processos do DS. Caso o RS não exista, cada nó do *Data Server* é capaz de renderizar os resultados da sua porção do conjunto de dados para que uma imagem composta de todas as partes possa ser mostrada no CL (processo no qual reside a GUI).

Este módulo contém leitores, fontes e filtros VTK. Nele são lidos e processados os dados para finalmente criar as representações geométricas necessárias para a renderização. Cada DS tem um *pipeline* VTK idêntico, e cada processo é indicado a ler a parte dos dados que a ele corresponde manipular.

9.2.8 Linguagem Cliente-Servidor

A Linguagem Cliente-Servidor do VTK, é uma linguagem independente da plataforma capaz de criar, utilizar (executar métodos) e destruir objetos VTK. Mensagens criadas pelo **cliente** são enviadas ao **servidor** onde são interpretadas e executadas⁴. Desta

⁴Os termos cliente e servidor neste caso são utilizados de uma maneira sutilmente diferente. Por **cliente** entende-se o processo, módulo ou classe que utiliza esta linguagem para chamar um método em um objeto VTK, o qual é chamado de **servidor**.

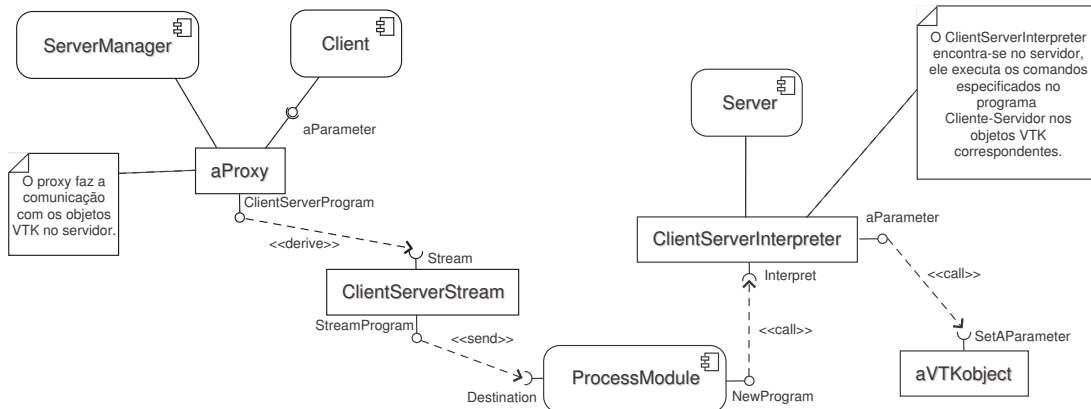


Figura 9.4: Comunicação entre Cliente e Servidor.

maneira, o cliente e o servidor podem-se encontrar no mesmo espaço de memória, em diferentes processos, em diferentes computadores ou até diferentes plataformas. Dentro do ParaView, o SM faz o papel do cliente e os RS, DS e CL cumprem o de servidor.

Os programas Cliente-Servidor VTK são escritos nos *Streams Cliente-Servidor*, os quais são posteriormente interpretados pelo *Intérprete Cliente-Servidor*⁵. Existe uma instância deste interprete em cada processo do ParaView, incluído o SM. O *Process Module* (`vtkProcessModule`) no SM tem uma instância do stream no qual o programa Cliente-Servidor será escrito. Os objetos no SM (geralmente os *proxies*), irão escrever estes programas utilizando a linguagem, e posteriormente pedirão ao *Process Module* para enviar o programa escrito ao servidor correspondente. Estes programas serão recebidos pelos processos correspondentes, onde cada interprete os executa localmente (Fig. 9.1).

Instâncias do *Process Module* são criadas no CL e nos Servidores (SM, RS, DS) com diversos propósitos. Elas mantêm referências locais aos interpretes de *streams* e provêem uma interface para chamar métodos em objetos remotos usando a linguagem Cliente-Servidor. Podem também reunir informação dos nós servidores usando objetos de tipo *information* (classe `vtkPVIInformation`). Também provêem métodos para obter o número de “partições” e o id de cada uma delas, assim como métodos que geram e retornam CSIDs únicos.

9.2.8.1 *Streams* Cliente-Servidor

Os *Streams* Cliente-Servidor (classe `vtkClientServerStream`) são utilizados para comunicar os diferentes processos do ParaView, por meio da linguagem Cliente-Servidor do VTK. Cada instrução nesta linguagem é chamada de mensagem, onde cada uma delas tem um comando, zero ou mais argumentos e um *token End* (Tab. 9.1). As mensagens são armazenados em um formato binário independente de plataforma.

⁵Nas classes `vtkClientServerStream` e `vtkClientServerInterpreter` respectivamente.

	comando	argumentos	fim
mensagem 0	New	<i>string</i> (vtkObject),id(5)	End
mensagem 1	Invoke	id(7), <i>string</i> (SetXResolution)	End
mensagem 2	Delete	id(5)	End

Tabela 9.1: Exemplo de comandos na linguagem Cliente-Servidor do VTK

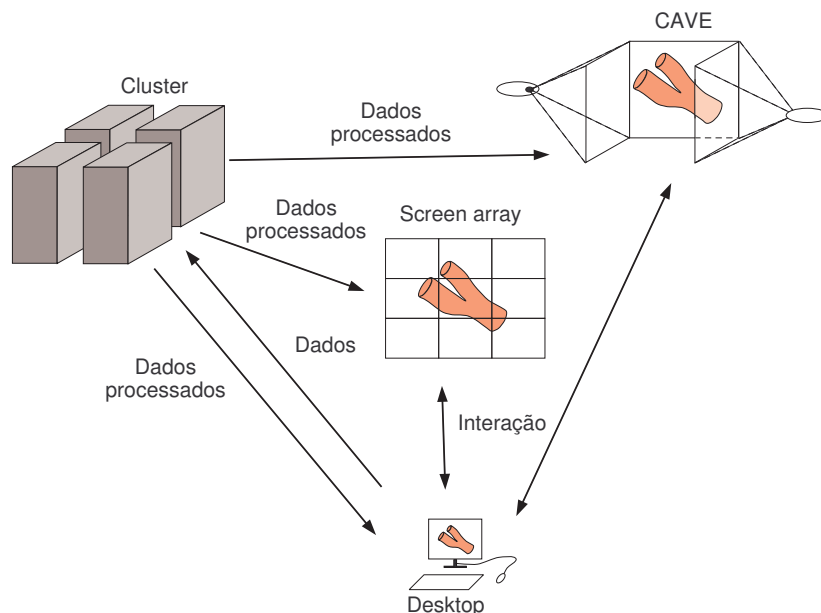


Figura 9.5: Possível configuração de *deployment* do ParaView.

9.2.8.2 Configuração - Deployment

O ParaView oferece a possibilidade de ser operado de diferentes maneiras de acordo com o tamanho dos dados sendo tratados e com o hardware disponível. Como mencionado, este software permite o processamento de grandes volumes de dados em paralelo por meio do uso de um *cluster* de computadores utilizando mais de um *Data Server*. Também permite realizar a renderização utilizando uma tela simples, um *screen array* ou um CAVE configurando o *Render Server* para tal fim. Dependendo do hardware disponível e do uso que se deseje dar ao software, diferentes configurações podem ser utilizadas. Em todos os casos, o Cliente tem a função de permitir a interação com o usuário. Estes cenários são apresentados na Fig. 9.5.

9.3 HeMoLab - Laboratório de Modelagem em Hemodinâmica

O nome HeMoLab, significa Laboratório de Modelagem em Hemodinâmica. Este software agrupa dentro de um mesmo ambiente diferentes ferramentas que permitem criar modelos do SCVH. Seguindo os requerimentos mencionados na Seção 9.1, estas funcionalidades foram implementadas como módulos do software ParaView. Na seção seguinte é descrita a organização dos pacotes e módulos desenvolvidos.

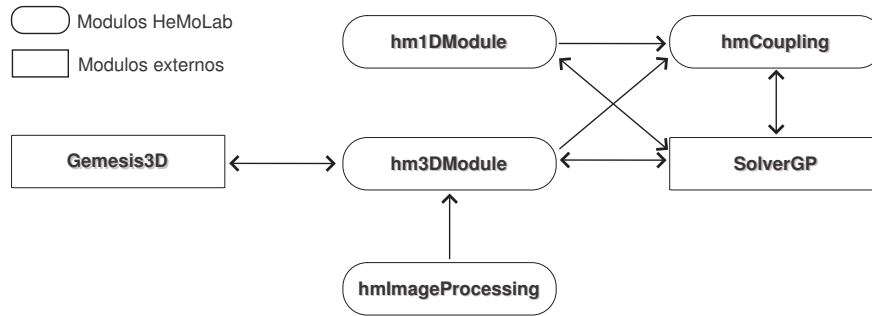


Figura 9.6: Módulos do HeMoLab e sua interação.

9.3.1 Pacotes e Módulos

Devido a complexidade dos requerimentos do HeMoLab foi necessário separá-los de acordo com as necessidades de cada um deles. Assim, as funcionalidades foram separadas de duas maneiras: por pacote e por módulo. Os pacotes seguem a mesma filosofia utilizada pelo VTK para separar classes de acordo ao tipo de funcionalidade que eles provêm, sejam:

- **Common:** agrupa funcionalidades comuns, são classes de infraestrutura que controlam as regras do negócio.
- **Filtering:** reúne filtros implementados especificamente para o HeMoLab, estes podem ser aplicados em diferentes tipos de dados (superfície, volume, etc.).
- **Graphics:** classes utilizadas para representar geometrias complexas (árvore 1D, malhas de superfície/volume 3D, etc.).
- **GUIClient:** define componentes utilizados na interface de usuário.
- **Imaging:** agrupa os diferentes filtros de imagens.
- **IO:** agrupa as classes de entrada/saída (leitura/escrita).
- **Server Manager:** reúne as classes responsáveis por comunicar os diferentes processos (*proxies*).
- **Widgets:** classes utilizadas na interação do usuário com as estruturas de dados através de eventos de *mouse* e de teclado.

Neste contexto, cada pacote oferece funcionalidade agrupadas pelo seu tipo. Estas funcionalidades são utilizadas pelos distintos módulos para oferecer ferramentas mais complexas. Os seguintes módulos foram criados dentro do HeMoLab:

- hmImageProcessing,
- hm1DModule,
- hm3DModule,

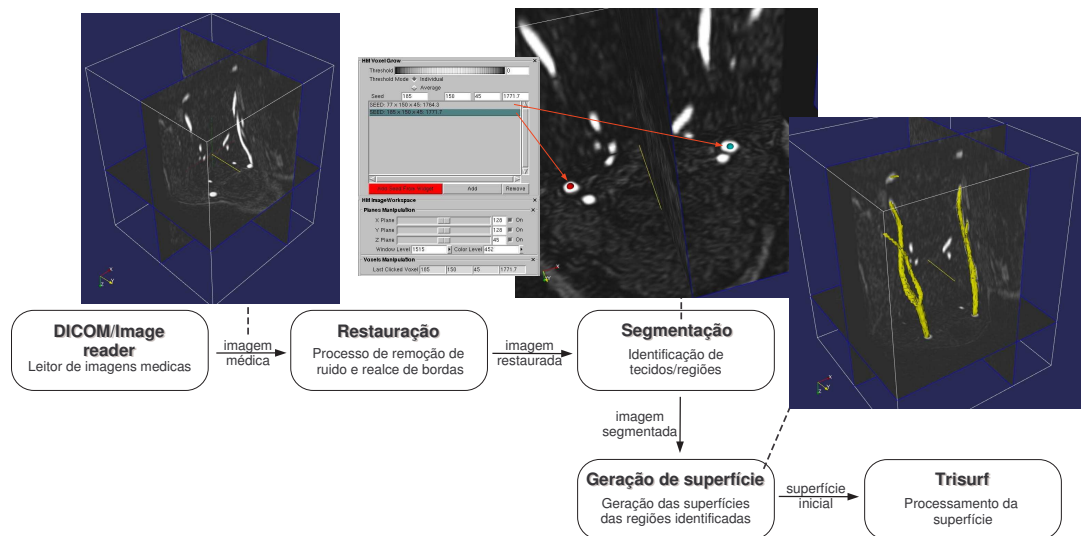


Figura 9.7: Processo de obtenção de geometrias a partir de imagens médicas.

- hmCoupling e
- SolverGP.

Estes módulos podem trabalhar de maneira independente ou compartilhar informações segundo seja necessário. Nas seções seguintes estes módulos são estudados em detalhe.

9.3.2 Módulo de Processamento de Imagens - hmImageProcessing

No processo de modelagem do SCVH, é de interesse reproduzir o comportamento deste sistema sob diferentes condições. Para gerar informação que seja útil para os médicos, os modelos devem ser ajustados para reproduzir o comportamento específico de um determinado paciente. Neste processo será necessário incorporar informação personalizada. Uma importante fonte deste tipo de dados são as imagens médicas, as quais podem ser adquiridas de diferentes fontes (MRI, CT, Ultrassom, dentre outras). Serão necessárias, então, ferramentas que permitam trabalhar com imagens médicas de diferentes tipos e em diferentes formatos.

O módulo hmImageProcessing tem por objetivo fornecer as ferramentas a partir das quais, por exemplo, as geometrias das artérias serão obtidas (Fig. 9.7). Neste caso, uma imagem médica deve ser processada [29, 37, 75, 97, 27] utilizando diferentes filtros de melhoramento e restauração de imagens para posteriormente identificar as diferentes regiões nela presentes (processo chamado de segmentação [88, 112]). As ferramentas providas permitem:

- *Leitura/escrita:* leitores de imagens em diferentes formatos são incorporados permitindo carregar imagens médicas em formatos padrão (e.g., DICOM). Para agregar novos formatos, é simplesmente necessário implementar o leitor/escritor como uma classe VTK e agregar a chamada a ele no arquivo XML correspondente a este módulo.

<i>Componente</i> (Classe)	Responsabilidade
<i>Image Workspace</i> (<i>vtkPVHMIImageWorkspaceWidget</i>)	Interface de usuário do visualizador de imagens.
<i>Image Workspace Proxy</i> (<i>vtkSMHMIImageWorkspaceWidgetProxy</i>)	Comunica o CL com um widget 3D no RS.
<i>Image Workspace Widget</i> (<i>vtkHMIImageWorkspaceWidget</i>)	Permite visualizar os planos da imagem nas direções x, y e z .

Tabela 9.2: Classes do Módulo de Processamento de Imagens

Os escritores (*writers*) podem ser utilizados para salvar imagens processadas (e.g., o usuário deseja salvar uma imagem depois de ter aplicado uma série de filtros). O próprio VTK fornece *writers* de diferentes formatos, os quais foram incorporados usando os arquivos XML.

- *Filtros*: as imagens podem ser processadas utilizando diferentes filtros, por exemplo:
 - Suavizado Gaussiano: convolução de uma imagem com um *kernel* Gaussiano. Este tipo de filtro “suaviza” a imagem e é normalmente utilizado para remoção de ruído [3].
 - Suavizado Anisotrópico: semelhante ao caso anterior, com a particularidade de preservar as bordas na imagem. Este tipo de filtro é utilizado quando se deseja eliminar ruído da imagem mas preservando as características [64, 113, 27, 164].
 - Filtro da Média e da Mediana: são normalmente utilizados para remover outros tipos de ruído, como ruído de impulso, para o qual os outros filtros apresentam dificuldades [79].
 - Filtro Limiar ou *Threshold*: este filtro permite selecionar os pixels/voxels da imagem que satisfazem uma determinada condição (i.e., que se encontram dentro de uma faixa de valores) [97]. Desta maneira, um valor v_1 é atribuído aos pixels que satisfazem esta condição e um outro valor v_2 ao resto.
 - Filtros de restauração R_{DT} : Os filtros de restauração apresentados no capítulo 3 se mostraram eficientes para remover ruído de imagens médicas, sendo os seus parâmetros de fácil ajuste.
- *Segmentação*: é de interesse também, selecionar dentro de uma imagem diferentes objetos ou regiões (segmentar a imagem). No caso estes objetos serão as artérias, as suas paredes, aneurismas, etc. Existem diferentes métodos que permitem realizar este tipo de operação, alguns dos quais são geralmente aplicados em imagens médicas [21, 24, 30]. Os mais utilizados foram contemplados no módulo de processamento de imagens. Também foi incorporado o método de segmentação baseado na derivada topológica S_{DT} -Discrete descrito no capítulo 6 [110].

Diferentes pacotes foram utilizados para proporcionar estas funcionalidades, a interação das classes de interface, *proxies*, filtros e classes VTK, são especificadas utilizando

arquivos XML (`hmImagingClient.xml` e `hmImagingServer.xml`) seguindo o mecanismo proposto no ParaView. Este módulo está composto por classes dos pacotes `GUIClient`, `Imaging`, `Server Manager` e `Widgets` utilizados da seguinte maneira:

- **GUIClient:** Este pacote provê as classes de interface que permitem aos usuários modificar parâmetros e interagir com os filtros no servidor. Para visualizar os planos da imagem, é utilizado o *image workspace*, o qual mantém uma referência a um *image workspace proxy* que realiza a comunicação com *widget 3D* (*image workspace widget*) responsável por mostrar o plano selecionado. O *widget 3D* pode-se encontrar em um outro processo. O *image workspace* permite ao usuário selecionar/visualizar um plano da imagem nas direções principais (i.e., x , y e z). Este componente da GUI é adicionado quando se deseja visualizar o resultado de um determinado filtro. Além do *image workspace*, cada filtro irá ter associada a ele uma ou mais abas que permitam controlar os seus parâmetros.
- **Imaging:** Os filtros de processamento de imagens desenvolvidos especificamente como parte deste projeto se encontram neste pacote, dentre eles: *D_T Restoration* (remoção de ruído por meio da derivada topológica), *VoxelGrow* (Segmentação por crescimento de regiões), *D_T Segmentation* (segmentação utilizando a derivada topológica). Eles são implementados como filtros VTK e posteriormente adicionados empregando arquivos XML.
- **Server Manager:** O *image workspace* utiliza um *image workspace proxy* para se comunicar com a classe responsável pela visualização da imagem em 3D. Assim, este *proxy* realiza a conexão entre o CL e o RS (no qual se encontra o *widget 3D*).
- **Widgets:** Para visualizar a imagem, foi desenvolvido um *widget 3D* específico. O *image workspace widget* utiliza três instâncias da classe `vtkImagePlaneWidget` para mostrar cortes da imagem nas direções principais (x , y e z). O único objetivo deste *widget* é a visualização da imagem (por planos) e ele não produz nenhum tipo de processamento/filtrado nos dados. A associação deste *widget* a uma imagem qualquer, e.g., a saída de um filtro aplicado em outra imagem, é feita no arquivo `hmImagingClient.xml` ao se associar a um filtro um *image workspace*. Assim, a saída do filtro será automaticamente visualizada usando este *widget 3D*. Caso o mesmo não seja utilizado, somente a *bounding box* da imagem é visualizada.

Pretende-se incorporar novos filtros de processamento de imagens neste módulo. Existe também a possibilidade de se agregar filtros de outras bibliotecas (e.g., ITK [96]). Nesta biblioteca existem implementados alguns dos filtros de imagens e algoritmos de segmentação mais utilizados na atualidade.

9.3.3 Módulo 1D - `hm1DModule`

Os modelos 1D [61, 95, 149, 193] permitem representar de maneira simplificada o fluxo sanguíneo nas maiores artérias do corpo humano. Desta maneira, pode ser de

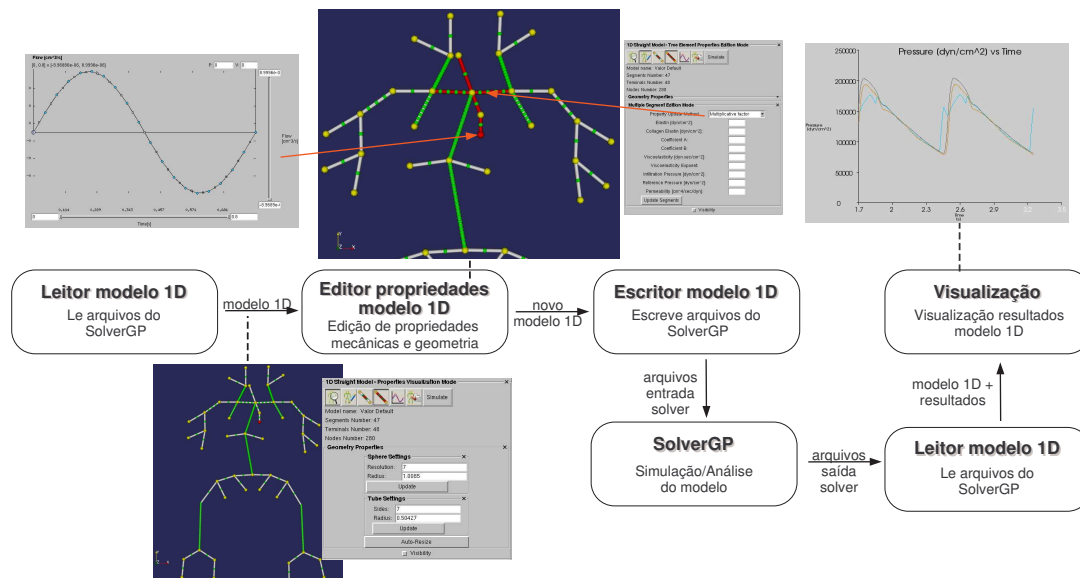


Figura 9.8: Processo de edição/geração de modelos 1D.

interesse para os usuários gerar modelos específicos 1D para estudar a resposta do comportamento do SCVH em diferentes condições (e.g., ao se modificar propriedades mecânicas do modelo, ao se modificar a geometria da árvore, etc.). Para suprir estes requerimentos, é necessário permitir a geração de modelos 1D personalizados assim como modificar os já existentes. Estes modelos serão a entrada do SolverGP (*solver* numérico, descrito em maior detalhe na Seção 9.3.6), o qual realizará a correspondente simulação (independentemente da existência de uma geometria 3D associada). Isto evidencia a existência de uma relação direta entre o hm1DModule e o SolverGP (leitura/escrita do formato dos arquivos do *solver* numérico).

O módulo hm1DModule permite ler, editar propriedades, modificar a geometria e salvar modelos da árvore 1D no formato do SolverGP (Fig. 9.8). Esta característica permite a geração de novos modelos assim como a edição dos já existentes.

Os pacotes e correspondentes componentes utilizados por este módulo são determinados nos arquivos `hm1DModuleClient.xml` e `hm1DModuleServer.xml`. No arquivo `hm1DModuleClient.xml` é chamado o componente da interface de usuário utilizado para controlar a interação com as classes no servidor, e as propriedades por ele utilizadas. O arquivo `hm1DModuleServer.xml` especifica as relações entre *proxies* e objetos VTK no servidor assim como as propriedades associadas a estes *proxies*.

As diferentes componentes deste módulo encontram-se distribuídas de acordo à funcionalidade nos diferentes pacotes. Os pacotes utilizados por este módulo são Common, Graphics, GUIClient, IO, Server Manager e Widgets.

- **Common:** Para representar a estrutura de dados de uma árvore 1D diferentes componentes foram desenvolvidos. O *1D straight model*, criado no DS, é responsável por gerenciar a maneira que as árvores 1D são editadas ou criada. Nesta classe existem implementadas as regras que controlam a criação e inserção de novos seg-

<i>Componente</i> (Classe)	Responsabilidade
<i>1D Straight Model</i> (<i>vtkHM1DStraightModel</i>)	Armazena a estrutura de árvore e gerencia a maneira na qual ela é criada.
<i>Segment</i> (<i>vtkHM1DSegment</i>)	Armazena informação correspondente ao segmento (incluindo seus elementos e nós).
<i>Terminal</i> (<i>vtkHM1DTerminal</i>)	Armazena informação correspondente ao terminal.
<i>1D Straight Model Grid</i> (<i>vtkHM1DStraightModelGrid</i>)	Armazena a representação geométrica da árvore 1D.
<i>1D Straight Model Source</i> (<i>vtkHM1DStraightModelSource</i>)	Gera, a partir do 1D Straight Model, o grid que o representa.
<i>1D Straight Model Reader</i> (<i>vtkHM1DStraightModelReader</i>)	Gerencia a leitura dos arquivos de dados do <i>solver</i> numérico.
<i>1D Straight Model Writer</i> (<i>vtkHM1DStraightModelWriter</i>)	Gerencia a escrita dos arquivos de dados do <i>solver</i> numérico.
<i>Segment Proxy</i> (<i>vtkSMHM1DSegmentProxy</i>)	Comunica o CL com um segmento no DS.
<i>Terminal Proxy</i> (<i>vtkSMHM1DTerminalProxy</i>)	Comunica o CL com um terminal no DS.
<i>Straight Model Widget Proxy</i> (<i>vtkSMHM1DStraightModelWidgetProxy</i>)	Comunica o CL com um widget 3D no RS.
<i>Segment Widget</i> (<i>vtkPVHMSegmentWidget</i>)	Oferece a interface correspondente aos parâmetros dos segmentos.
<i>Terminal Widget</i> (<i>vtkPVHMTerminalWidget</i>)	Oferece a interface correspondente aos parâmetros dos terminais.
<i>Element Widget</i> (<i>vtkPVHMElementWidget</i>)	Oferece a interface correspondente aos parâmetros dos elementos.
<i>Heart Widget</i> (<i>vtkPVHMHeartWidget</i>)	Oferece a interface correspondente aos parâmetros do coração.
<i>Straight Model Source Widget</i> (<i>vtkPVStraightModelSourceWidget</i>)	Oferece a interface correspondente aos parâmetros da árvore 1D.
<i>Straight Model Source 3D Widget</i> (<i>vtkHM1DStraightModelWidget</i>)	Apresenta a árvore 1D ao usuário.
<i>XY Plot Widget</i> (<i>vtkKWHMXYPlotWidget</i>)	Visualiza as curvas dos resultados no gráfico.

Tabela 9.3: Classes do Módulo 1D

mentos arteriais e terminais na árvore. Seguindo a natureza da árvore arterial, a mesma foi desenhada utilizando o modelo *Composite* [68], este permite tratar os elementos da árvore independentemente do seu tipo (terminais ou segmentos). Os segmentos e terminais armazenam as informações que correspondem ao modelo, i.e., propriedades mecânicas e geométricas da árvore e dos seus elementos e informações próprias da malha de elementos finitos (utilizada para achar as soluções das equações 1D correspondentes).

Existem também restrições na maneira em que novos segmentos e terminais podem ser adicionados na árvore. No caso de agregar um novo segmento, este deve ter no

seu extremo (aquele que não se encontra conectado ao resto da árvore) um terminal⁶. Este é incorporado automaticamente pelo *1D straight model*. Quando são conectados dois segmentos com diferentes diâmetros, é colocado entre eles um terminal. Neste caso o terminal utilizado para modelar a queda de pressão que ocorre entre os dois segmentos. No caso dos dois segmentos ter o mesmo diâmetro, o terminal não é necessário. Estas restrições são levadas em consideração no momento de adicionar segmentos e terminais pelo *1D straight model*.

- **Graphics:** Para visualizar a árvore, foi necessário criar uma representação geométrica dele. Esta representação é montada a partir das informações existentes no *1D straight model*, o qual armazena comprimento e posição dos segmentos e terminais da árvore. Desta maneira a árvore arterial é percorrida para gerar as representações dos segmentos (compostos por elementos⁷ e nós) e terminais. Esta representação é armazenada pelo *1D straight model grid* (extensão da classe *polygonal data* de VTK), e representa os diferentes componentes da árvore (segmentos, terminais, nós e elementos) como pontos e linhas dentro de um *cell array*. A estas linhas e pontos é atribuída uma propriedade (escalar) indicando o seu tipo. O *1D straight model source* é o responsável por gerar o *1D straight model grid* a partir do *1D straight model*. Desta maneira, a dependência entre os dados e sua representação fica centralizada nesta classe.
- **GUIClient:** As classes neste pacote são responsáveis por mostrar propriedades dos componentes da árvore assim como permitir ao usuário modifica-las. Neste caso as abas utilizadas são *element widget*, *segment widget*, *heart widget*, *terminal widget* e *straight model source widget*. Quando uma árvore é aberta (ou criada), um *straight model source widget* é criado no CL (esta associação é especificada no arquivo XML do cliente, descritos na Seção 9.2.5). Este irá mostrar as propriedades correspondentes (usando instâncias das abas mencionadas, i.e., *element widget*, *segment widget*, *heart widget* e *terminal widget*) dependendo do modo de operação no qual ele se encontre (edição ou visualização) e do componente (terminal, segmento, etc.) que ele tenha selecionado. Estas classes são painéis compostos com *labels*, *text boxes* e botões dentre outras abas simples que permitem editar e visualizar os valores correspondentes aos diferentes componentes. Uma vez modificados os valores, estes são armazenados em variáveis internas.

O *heart widget* permite visualizar/editar as propriedades do coração. Neste tipo de modelo o coração é representado com um terminal especial que “injeta” no sistema uma quantidade de sangue determinada por uma curva. Esta curva de ejeção muda de um indivíduo para outro. Assim, é permitido ao usuário ajustar esta curva de acordo com o observado no paciente.

Quando se deseja enviar estas modificações ao servidor, a informação é passada utilizando *proxies* ou propriedades. Um *proxy* é utilizado quando a operação é de

⁶Este terminal representa a resistência oferecida pelo resto do sistema depois daquele ponto.

⁷Por elemento entende-se elemento finito, já que as propriedades destes podem ser alteradas. Os nós armazenam informações geométricas complementares como diâmetro do segmento naquele ponto.

maior complexidade (e.g., se requer efetuar uma sequência de chamadas a métodos). As propriedades são utilizadas quando a interação é simples.

- **IO:** Para poder armazenar de maneira persistente árvores criados ou poder ler árvores já existentes, estas informações devem poder ser salvas de maneira persistente (i.e., em arquivos). Diferentes formatos podem ser adotados para salvar estas informações (XML, CSV, etc.), no entanto, em uma primeira etapa o principal objetivo é compartilhar estas informações com o SolverGP (Sec. 9.3.6). Este software, pensado para solução de problemas discretos, permite um uso simples de diferentes tipos de elementos finitos (i.e., diferentes modelos). Neste contexto, a definição das malhas de elementos finitos, das suas propriedades, condições de contorno e condições de iniciais é feita através de arquivos de texto formatados. Para ler/escrever estes arquivos foram desenvolvidas diferentes classes no pacote IO. O componente principal para a leitura é o *1D straight model reader* (extensão do *data reader* do VTK), responsável por coordenar a leitura dos diferentes arquivos (feita por classes específicas) e gerar o correspondente *1D straight model*. De forma equivalente existe um componente responsável pela escrita (*1D straight model writer*, o qual gera os arquivos do modelo com geometria, propriedades mecânicas, condições iniciais e configuração do algoritmo de resolução). Desta maneira é feita a interação entre o módulo 1D e o SolverGP.

- **Server Manager:** A comunicação entre o cliente e os diferentes servidores (RS e DS) é feita pelo SM. Como mencionado anteriormente, no SM coexistem as classes responsáveis por esta comunicação, os *proxies*. A comunicação entre o CL e o DS, é feita pelos *proxies* correspondentes aos segmentos e terminais (*segment proxy* e *terminal proxy* respectivamente) os quais permitem trocar informações entre estes processos. Assim, estas classes são utilizadas para se comunicar com os componentes da árvore que se encontram no *1D straight model* do servidor. As informações são enviadas pelos *proxies* utilizando *streams*. Para identificar o objeto no qual um método deve ser chamado, é utilizado o *CSID* daquele objeto. Desta maneira, no cliente somente são criados um *segment proxy* e um *terminal proxy*, os quais são “apontados” para diferentes segmentos ou terminais usando o *CSID* correspondente.

Por outro lado, existe a necessidade de comunicar o CL com o RS (para identificar as interações do usuário com o *widget* 3D da árvore e os seus componentes). Esta comunicação é responsabilidade do *straight model widget proxy*, o qual envia informações (principalmente de configuração da visualização) da árvore ao correspondente *widget* no RS. Também, esta classe armazena referências aos *segment proxies* e *terminal proxies*, utilizados para interagir com elementos da árvore no DS.

- **Widgets:** A representação visual da árvore 1D é feita utilizando um *widget* 3D especial. Este *widget*, chamado *straight model source 3D widget*, é responsável por controlar a interação com a árvore 1D (e.g., seleção de elementos da árvore como segmentos, terminais, elementos e nós) assim como da configuração e gerenciamento dos gráfico para visualização de resultados (gráficos 2D correspondentes a área, pressão,

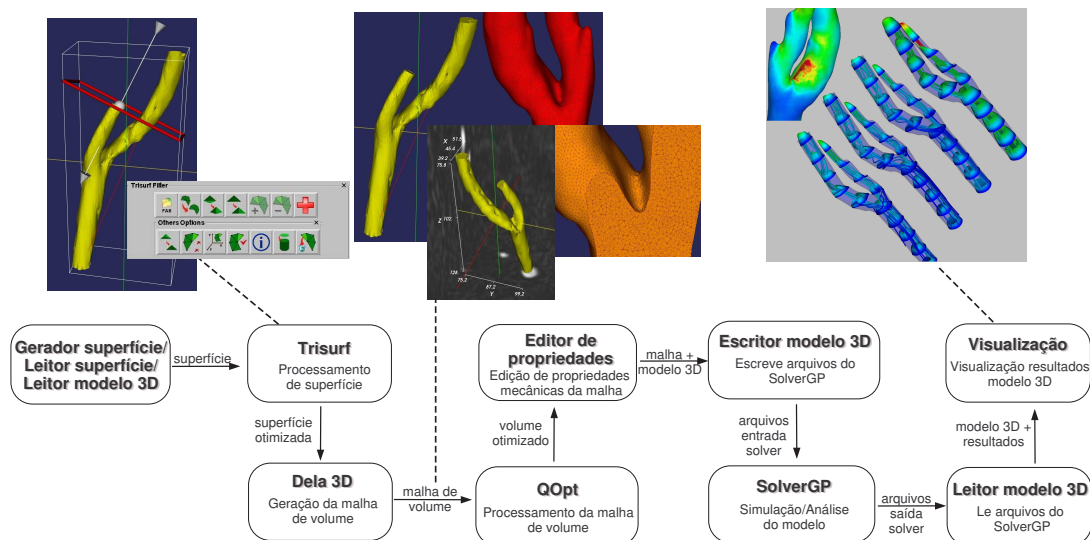


Figura 9.9: Processo de geração de malhas e modelos 3D.

fluxo, etc.). Para obter/retornar a informação de propriedades de um segmento, terminal ou outro componente utilizou-se o esquema de *callbacks* (implementando o esquema *Observer* [68]) do VTK, permitindo ao usuário selecionar qualquer componente da árvore. Quando um determinado componente (representado por um ator) é selecionado na área de visualização, o seu CSID é identificado. Desta maneira é possível “dialogar” com todos os elementos da árvore, podendo-se obter e/ou modificar as suas informações/propriedades. O *widget* 3D permite a seleção de segmentos (incluindo seus nós e elementos) e terminais. Ao selecionar um segmento, por exemplo, as suas informações são mostradas na interface permitindo visualizar e/ou modificar o seu valor.

Este *widget* 3D gerencia também a visualização dos resultados do processamento da árvore. O *xy plot widget* é utilizado pelo *straight model source 3D widget* para mostrar as curvas correspondentes a pressão, fluxo, área, etc., permitindo selecionar trechos do gráfico para estudar os detalhes destas curvas.

Desta maneira, o módulo *hm1DModule* é responsável pela criação\edição\visualização de resultados de modelos 1D. Estes modelos são lidos\armazenados em um formato específico. Além da leitura de arquivos de dados, é feita a leitura dos resultados da simulação, permitindo a sua visualização e análise.

9.3.4 Módulo 3D - *hm3DModule*

O objetivo do Módulo 3D é prover ferramentas que permitam, a partir da geometria da artéria ou região de interesse, gerar uma malha de elementos finitos de boa qualidade adequadamente refinada para finalmente gerar os arquivos correspondentes prontos para a simulação (utilizando o SolverGP, Seção 9.3.6).

Neste módulo são reunidos diferentes filtros de processamento de superfície os quais permitem processar a triangulação de uma geometria para melhorar sua qualidade. Assim,

<i>Componente</i> (Classe)	Responsabilidade
<i>Full Model</i> (<code>vtkHMFullModel</code>)	Armazenar informações correspondentes a um modelo completo.
<i>Mesh</i> (<code>vtkHMMesh</code>)	Armazenar a geometria da malha (superfície ou volume).
<i>Surface Reader</i> (<code>vtkHMSurfaceReader</code>)	Leitor de arquivos ".sur" (pontos e conectividade).
<i>Surface Writer</i> (<code>vtkHMSurfaceWriter</code>)	Escritor de arquivos ".sur" (pontos e conectividade).
<i>Volume Reader</i> (<code>vtkHMVolumeReader</code>)	Leitor de arquivos ".vwm" (pontos e conectividade).
<i>Volume Writer</i> (<code>vtkHMVolumeWriter</code>)	Escritor de arquivos ".vwm" (pontos e conectividade).
<i>Trisurf Widget</i> (<code>vtkPVHM3DTrisurfFilter</code>)	Oferece a interface correspondente dos filtros/parâmetros do Trisurf.
<i>Dela 3D Widget</i> (<code>vtkPVHM3DDelaFilter</code>)	Oferece a interface correspondente aos parâmetros do Dela3D.
<i>Qopt Widget</i> (<code>vtkPVHM3DQoptFilter</code>)	Oferece a interface correspondente aos parâmetros do Qopt.

Tabela 9.4: Classes do Módulo 3D

uma vez que a região de interesse foi isolada em uma imagem médica, a superfície (triangulação) que a caracteriza é construída. Esta primeira triangulação da geometria costuma ser de má qualidade e precisa de um processamento adequado. Para se obter as malhas de elementos finitos, foram incluídos diferentes métodos clássicos de geração e otimização de malhas não estruturadas (triangulares\tetraédricas). Estes algoritmos encontram-se implementados em um software chamado Gemesis3D [197]. As funcionalidades oferecidas por este software foram adicionadas e disponibilizadas dentro do módulo 3D. Depois de melhorar a superfície, é gerada uma malha de volume (utilizando o algoritmo de Delaunay [20] para geração de malhas de volume), e posteriormente esta malha (composta de tetraedros) é otimizada [196].

Obtidas as malhas de elementos finitos, pode ser de interesse para o usuário realizar a simulação computacional somente do modelo 3D (impondo condições iniciais e de contorno adequadas, é possível realizar a simulação do modelo 3D isolado, Fig. 9.9). Isto pode ser útil em diferentes circunstâncias, e.g., geração das condições iniciais do modelo acoplado. Neste caso devem ser fornecidas as condições no contorno (i.e., entradas\saídas da geometria). Assim, o módulo `hm3DModule` pode gerar os arquivos do modelo 3D independente da árvore 1D (arquivos de entrada do `SolverGP`).

As classes e pacotes utilizadas para fornecer estas funcionalidades são especificadas nos arquivos `hm3DModuleClient.xml` e `hm3DModuleServer.xml`. No arquivo do cliente são definidas as classes de interface utilizadas para aplicar filtros VTK nos dados e controlar seus parâmetros. O arquivo XML do servidor irá relacionar os diferentes *proxies* com os filtros e classes VTK no servidor.

As funcionalidades deste módulo são implementadas utilizando classes dos pacotes

Common, Filtering, Graphics, GUIClient, IO e Server Manager.

- **Common:** Neste pacote encontra-se o componente responsável por manter e armazenar informações correspondentes a um modelo 3D (ou modelo completo), chamado *full model*. Por modelo completo entende-se que as equações correspondentes (fluido o sólido) são consideradas completas e sem hipóteses simplificativas que reduzem a dimensionalidade do problema, como no caso do modelo 1D. Um exemplo de isto são as equações de Navier-Stokes utilizadas para modelar o comportamento de um fluido em um distrito arbitrário. Estas equações foram introduzidas na Seção 8.1.4. No caso, o *full model* armazena a informação da malha, grupos de elementos e parâmetros associados ao modelo. Logo, cada grupo tem informação de quais elementos correspondem ao grupo e dos campos escalares\vetoriais\tensoriais que descrevem as propriedades associadas. Pode-se observar que este modelo se mantém genérico e flexível, permitindo tanto representar a malha de elementos finitos da parede arterial (neste caso as mesmas poderiam utilizar um modelo de membrana, de placa ou de sólido conforme o caso) como a correspondente ao fluido (no caso, para as equações de Navier-Stokes).
- **Filtering:** Como o funcionamento do Gemesis3D é baseado em um esquema de dados e filtros aplicados nos dados, foi relativamente fácil incorporar as suas funcionalidades para trabalhar em um ambiente baseado em VTK. Os filtros deste módulo podem ser diferenciados de acordo com o tipo de malha processada, as que podem ser de superfície (malhas triangulares) ou de volume (malhas tetraédricas).

Os filtros de superfície incluem funcionalidades que permitem otimizar a qualidade da malha de acordo com diferentes critérios (estes filtros encontram-se no módulo Trisurf do Gemesis 3D). Estes filtros introduzem modificações na forma e topologia da malha, sendo eles: filtro de suavizado, divisão de triângulos obtusos, inserção de nós (refinamento), colapsar triângulos pequenos (satisfazendo um certo critério de tamanho), dentre outros. A aplicação sucessiva (com bom critério por parte do usuário) destes filtros na malha irá melhorar a sua qualidade, até se obter uma malha de superfície com qualidade suficiente.

Uma vez que a malha de superfície foi melhorada, uma malha de volume (tetraedros) é gerada a partir dela. Para gerar esta malha volumétrica, são utilizados dos algoritmos diferentes: método Delaunay [43, 198], método Frontal [161, 162] e método de *Octrees* [67, 180, 207] (módulo Dela3D do Gemesis 3D). Estes métodos são amplamente utilizados para geração de malhas não estruturadas de elementos finitos e se mostram eficientes na geração deste tipo de malhas. No entanto os resultados destes métodos não necessariamente resultam em malhas de volume de boa qualidade. Aparece então a necessidade de otimizar estas malhas. Para isto é utilizado um *software* para otimizar malhas de volume [196] (módulo Qopt do Gemesis 3D). Estas funcionalidades são implementadas no software Gemesis 3D e foram adaptadas utilizando o esquema *Adapter* [68] de maneira a prover uma interface transparente para acessar estas funcionalidades.

- **Graphics:** As classes de representação de geometria são incluídas neste pacote. Em particular o componente *mesh* (é uma especialização do *unstructured grid* do VTK), irá armazenar a malha formada por coordenadas e conectividade. Ao aplicar um filtro nesta malha, esta será modificada gerando-se uma nova malha (processada) na saída. As classes neste pacote são responsáveis por armazenar a representação da malha (seja de superfície ou de volume).
- **GUIClient:** Para permitir ao usuário escolher a operação/filtro a se realizar na geometria foram desenvolvidos 3 componentes: *trisurf widget*, *dela 3D widget* e *qopt widget*. No caso do *trisurf widget* oferece a interface para escolher uma das diferentes operações possíveis em uma superfície (e.g., suavizado de superfície, divisão de triângulos obtusos, inserir nós para refinar a malha, dentre outros). Cada uma destas funções têm diferentes parâmetros, os quais são requeridos ao usuário pelo componente. O *dela 3D widget* permite definir tamanho característico dos tetraedros de acordo ao grupo. Com estas informações a malha de volume é gerada. O *qopt widget* permite ver e analisar de maneira quantitativa a qualidade da malha em valores médios (qualidade média dos elementos) e absolutos (pior elemento). O filtro Qopt irá analisar a malha para intentar melhorá-la operando por *clusters* de elementos, o número máximo de elementos neste *cluster* é o parâmetro que controla o algoritmo.
- **IO:** O software Gemesis3D tem definido um formato de arquivos para representar superfícies. Estes arquivos baseiam-se simplesmente em descrever os pontos e conectividade da malha. Além disto, estes arquivos permitem definir grupos de elementos, permitindo por exemplo, agrupar os triângulos da parede arterial e separadamente as faces de entrada/saída da região estudada. Para poder utilizar geometrias já existentes e exportar geometrias para ser processadas com outras versões do Gemesis 3D, foram implementados módulos de leitura/escrita deste formato. Estes são o *surface reader/writer* para leitura e escrita de superfícies (arquivos “.sur”) e volume reader/writer para leitura e escrita de volumes (arquivos “.vwm”). Além de permitir ler/escrever arquivos com as geometrias, foi implementada a funcionalidade de ler/escrever os modelos completos (incluindo condições de contorno, propriedades mecânicas, condições iniciais e configuração do algoritmo de resolução) no formato do SolverGP para que estes possam ser analisados (i.e., simulação).
- **Server Manager:** Neste caso, o *source proxy* já implementado no ParaView foi suficiente para comunicar o cliente com os servidores, sendo apenas incorporadas as propriedades necessárias para controlar os parâmetros dos filtros.

Como no caso do módulo 1D, o módulo 3D pode trabalhar de maneira independente dos outros módulos, o que dá maior versatilidade à ferramenta. Para a visualização de resultados destas simulações são atualmente utilizados os mecanismos providos pelo ParaView (e.g., visualização de campos escalares por meio de tabelas de cores, representação de campos vetoriais com *glyphs* ou *stream tubes*, representação de deslocamentos

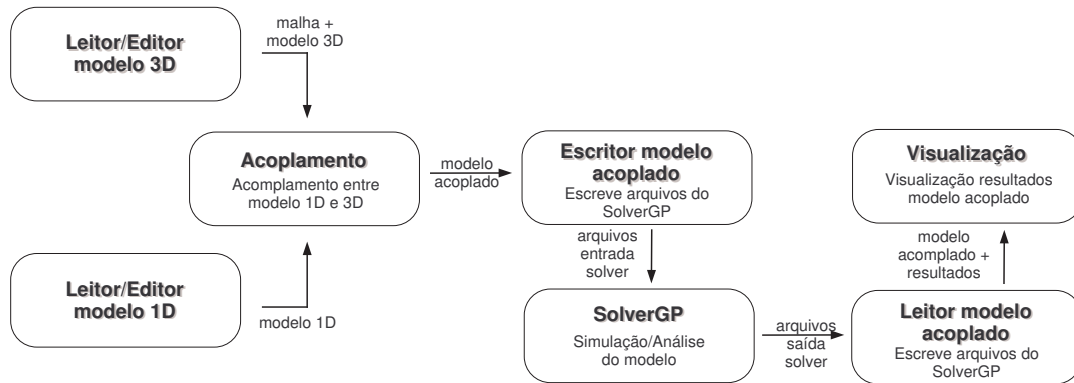


Figura 9.10: Processo de acoplamento entre modelos 1D e 3D e a sua simulação.

por meio de deformações na malha, dentre outros). Mesmo assim novos mecanismos podem ser facilmente adicionados. O cálculo de tensões nas paredes arteriais (*Wall Shear Stress* - WSS) e tensões oscilatórias (*Oscillating Shear Stress Index* - OSI) deverá ser incorporado posteriormente.

9.3.5 Módulo de Acoplamento - hmCoupling

O fluxo do sangue no corpo humano é reproduzido (de maneira aproximada) utilizando dois modelos com características diferentes: o modelo 1D e o modelo 3D. Cada modelo tem características e informações próprias que, ao ser unidos, representam de maneira integral o comportamento do sistema cardiovascular. Para gerar modelos personalizados acoplados 1D-3D, foi criado o módulo hmCoupling. Este tipo de modelos acoplados, fornece informações locais da hemodinâmica (e.g., detalhe do fluxo sanguíneo na bifurcação da carótida, estado de tensões nas paredes arteriais, etc.) consistentes com as condições providas pelo resto do sistema arterial (fluxo, pressão, etc. que vem do modelo 1D). Os modelos acoplados (ou multidimensionais [60, 63, 107, 193]) requerem associar terminais do modelo 1D com grupos de elementos no modelo 3D. Para realizar esta associação, o módulo hm3DModule permite criar grupos de elementos que serão posteriormente associados a terminais no modelo 1D. Armazenar esta informação, assim como salvá-la em forma de um modelo personalizado integrado, é responsabilidade do módulo hmCoupling. O formato utilizado para salvar o modelo é o do SolverPG (que permite a simulação de modelos acoplados 1D-3D).

Todas estas funcionalidades são implementadas utilizando os pacotes Common, GUIClient, IO e Server Manager (Tabela 9.5).

- **Common:** A informação própria do acoplamento é armazenada utilizando o *full model to straight model coupling*. Para cada acoplamento é armazenada a informação correspondente (no caso, direção da conexão, referência ao *full model* e o número de grupo, referência ao *1D straight model* e o id do terminal). Esta é toda a informação necessária para gerar a especificação do modelo acoplado.

<i>Componente</i> (Classe)	Responsabilidade
<i>Full Model to Straight Model Coupling</i> (<code>vtkHMFullModelTo1DStraightModelCoupling</code>)	Armazenar informações correspondentes a um modelo completo.
<i>Full Model to Straight Model Coupling Widget</i> (<code>vtkPVHMFullToStraightModelCoupling</code>)	Oferece a interface para associar grupos no modelo 3D e terminais no modelo 1D.
<i>Full Model to Straight Model Coupling Reader</i> (<code>vtkHMFullModelTo1DStraightModelCouplingReader</code>)	Leitor de modelos acoplados.
<i>Full Model to Straight Model Coupling Writer</i> (<code>vtkHMFullModelTo1DStraightModelCouplingWriter</code>)	Escritor de modelos acoplados.
<i>Full Model to Straight Model Coupling Proxy</i> (<code>vtkSMHMCouplingProxy</code>)	Comunica o CL e o DS.

Tabela 9.5: Classes do Módulo de Acoplamento

- **GUIClient:** Para associar grupos no *full model* e terminais no *1D straight model* é utilizado o *full model to straight model coupling widget*. Assim, são escolhidos o *1D straight model* e o *full model*. Logo, por cada acoplamento deve ser relacionado um terminal no *1D straight model* e um grupo no *full model*. Esta informação é enviada ao servidor utilizando o *proxy* correspondente (*full model to straight model coupling proxy*).
- **IO:** Uma vez especificado o modelo acoplado o mesmo deve ser salvo no formato do SolverGP. Para isto foram criados dois componentes, um de leitura e um de escrita para salvar as informações do modelo.
- **Server Manager:** As informações fornecidas pelo usuário são enviadas às classes correspondentes no servidor (*full model to straight model coupling*, responsável por armazenar as informações dos acoplamentos). De forma semelhante ao feito nos outros módulos, esta responsabilidade corresponde aos *proxies*. O componente *full model to straight model coupling proxy* irá comunicar estas informações às classes no servidor. Uma vez que as informações estão completas o sistema permite salvar a especificação do modelo acoplado.

Os modelos multidimensionais cobram uma grande importância devido à sua versatilidade. Este tipo de formulação permite obter informação detalhada do fluxo sanguíneo em qualquer região do corpo mantendo a consistência com o resto do sistema cardiovascular.

9.3.6 Solver numérico - SolverGP

A simulação do fluxo sanguíneo é feita usando Dinâmica dos Fluidos Computacional (*Computational Fluid Dynamics* - CFD [46, 74, 188]) e o Método dos Elementos Finitos (*Finite Element Method* - FEM). Para achar uma aproximação numérica dos modelos do SCVH, é utilizado um software de cálculo numérico chamado SolverGP [194]. Este software

pode ser estendido para considerar problemas não-lineares, interação fluido estrutura, acoplamento entre modelos 1D e 3D, dentre outras.

O SolverGP foi pensado como uma arquitetura genérica, a qual permite implementar facilmente *solvers* numéricos para métodos como FEM, FDM (*Finite Difference Method*), FVM (*Finite Volume Method*), assim como *solvers* numérico para outros métodos discretos como redes elétricas e redes neurais, etc., tanto para problemas estacionários como transientes. Além disso, é possível implementar qualquer método que resulte em um sistema de equações algébricas que possa ser construído a partir de contribuições sucessivas de operadores locais (matrizes elementares, i.e., em uma matriz global são somados subconjuntos de coeficientes correspondentes a contribuição dos diferentes elementos). A versatilidade deste aplicativo baseia-se na capacidade de um ensamblador simbólico⁸ e numérico que opera com matrizes elementares de qualquer tipo com um número arbitrário de graus de liberdade por nó, alcançando um alto grau de reutilização. A única tarefa delegada ao usuário é a implementação das matrizes elementares e a sua estrutura de acoplamento. Para o caso de simulação do SCVH, os elementos utilizados já existem implementados no SolverGP.

Uma vez definidas as equações utilizadas por um determinado modelo, as mesmas são empregadas nos elementos correspondentes da malha. Todas estas configurações são feitas através de arquivos de texto. Os arquivos de entrada deste *software* estão divididos pelos dados que eles contém, sendo estes:

- Parâmetros básicos - (*Basparam.txt*): Contém parâmetros básicos do *solver* numérico, os mesmos encontram-se definidos por grupo de elementos. Neste arquivo existe o chamado aos elementos correspondentes a cada grupo cada um dos quais estão especificados na biblioteca de elementos do *solver* numérico [194]. Também contém as configurações do *solver*.
- Geometria - (*Mesh.txt*): Este arquivo define a geometria do problema. Contém a malha de elementos finitos.
- Parâmetros da malha - (*Param.txt*): Armazena os parâmetros utilizados por cada elemento. Estes parâmetros mudam de acordo com o modelo associado ao elemento.
- Condições iniciais - (*Inifile.txt*): Indica para cada nó da malha a condição inicial associada a ele. Utilizado para problemas transientes.
- Resultados - (*Dataout.txt*): Arquivo de saída. Contém os resultados da simulação gerados pelo SolverGP.

9.3.7 Cenários

O sistema HeMoLab foi pensado de maneira modular. Sendo assim, e dependendo do objetivo do usuário, este poderá utilizar um ou mais módulos do sistema, sem estar

⁸Componente dentro do sistema que monta a partir das matrizes elementares uma matriz global.

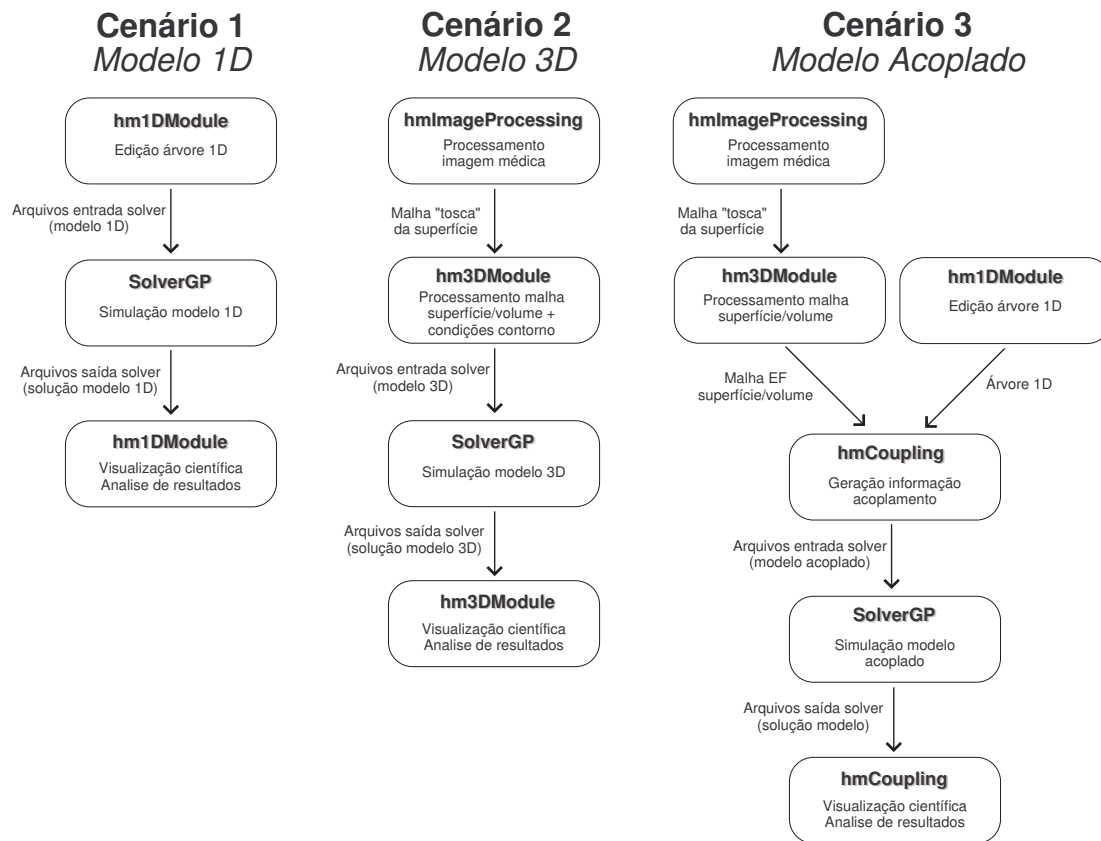


Figura 9.11: Cenários - três exemplos.

restrito à utilização de um único tipo de modelo. Na operação do sistema estes módulos podem ser utilizados em 3 diferentes cenários (Fig. 9.11):

9.3.7.1 Cenário 1 - Modelo 1D

Caso o usuário deseje simplesmente estudar a resposta do sistema cardiovascular a uma perturbação determinada (na geometria, nas propriedades mecânicas, etc.), este pode realizar a modificação desejada e analisar o modelo⁹. Por exemplo, o médico deseja ajustar o modelo a um determinado paciente. Neste caso, diversas medidas (pulso, pressão) são feitas em diferentes pontos do corpo do paciente. Ajustando os parâmetros e propriedades mecânicas/geométricas do modelo, o especialista procura a combinação de valores que melhor ajusta ao observado no paciente.

Neste caso, o hm1DModule é utilizado para edição da árvore 1D. Uma vez feitas as modificações são gerados os arquivos de entrada do SolverGP (contendo o modelo). Assim, o SolverGP computa a solução do modelo gerando os resultados. Estes resultados são finalmente analisados pelo hm1DModule.

⁹Por análise do modelo, entende-se realizar a simulação.

9.3.7.2 Cenário 2 - Modelo 3D

Se o único interesse é estudar condições do fluxo em um distrito arterial localizado, a mesma pode ser analisada de maneira independente. A geometria pode ser gerada a partir de uma imagem médica do paciente ou lida de um arquivo já existente (e.g., “.sur”). Isto é de interesse, por exemplo, para gerar condições iniciais para um modelo acoplado. A geometria é processada (melhoramento de superfície, geração da malha de volume, melhoramento da malha de volume), posteriormente são determinadas as suas propriedades mecânicas e gerados os arquivos correspondentes ao modelo 3D (entrada do SolverGP).

O hm3DModule é utilizado para gerar\otimizar a malha de elementos finitos e gerar a especificação do modelo pronto para o SolverGP calcular a sua solução. Dependendo do caso, os resultados podem ser diretamente analisados ou utilizados como condição inicial no caso acoplado (Cenário 3).

9.3.7.3 Cenário 3 - Modelo Acoplado

Talvez o cenário mais interessante seja o que permite estudar de maneira integrada o sistema arterial. Esta opção permite estudar o comportamento detalhado das condições do fluxo (e.g., regiões de re-circulação, tensões de corte e tensões oscilatórias, etc.) em um determinado distrito do SCVH consistente com as condições fornecidas pelo resto do sistema. Neste caso, devem ser definidos os pontos nos quais os modelos de diferente dimensionalidade estão conectados. Uma vez incorporada a informação do acoplamento, os arquivos podem ser gerados para simular o modelo.

Este modelo pode ser utilizado com diversas finalidades, e.g., para estudar o risco de ruptura de aneurisma. Neste caso, as imagens do paciente são analisadas para extrair a geometria das artérias e do aneurisma. Esta geometria é processada utilizando o hm3DModule para gerar uma malha de elementos finitos adequada e condições iniciais para o campo de pressão e velocidade dentro daquele distrito. Paralelamente, utilizando o hm1DModule a árvore 1D é ajustada para reproduzir as condições observadas no paciente. Estas duas informações são agregadas utilizando o módulo hmCoupling e a simulação é feita. Estes resultados irão complementar a informação da qual o especialista dispõe para tomar suas decisões.

9.4 Comentários

A cada dia é mais importante o uso de ferramentas de predição em áreas como biologia e medicina. A modelagem e simulação computacional junto às técnicas de visualização gráfica vem sendo crescentemente utilizadas para estes fins, já que permitem representar com alto grau de detalhe os fenômenos químicos e mecânicos envolvidos. Assistência no treinamento de médicos, predição do resultado de diferentes intervenções cirúrgicas, simulação do transporte de fármacos e substâncias pelo fluxo sanguíneo, predição do risco de ruptura de aneurismas, caracterização de materiais são somente algumas das potenciais aplicações destas técnicas em medicina.

Visando estas possibilidades foi desenvolvido o sistema HeMoLab. Esta ferramenta foi implementada sobre o ParaView, um software de visualização científica que utiliza computação paralela para processar grandes volumes de informação. Completamente desenvolvido em ANSI-C++, este software pode ser executado em diferentes plataformas trabalhando em paralelo.

Para modelar os fenômenos mencionados anteriormente, são utilizados modelos 1D, modelos 3D e modelos acoplados. Foram desenvolvidos módulos diferentes que permitem criar modelos personalizados de cada tipo, sejam: `hm1DModule`, `hm3DModule`, `hmCoupling`. Para poder criar modelos personalizados, é necessário incorporar informações específicas dos pacientes. Este tipo de informação geralmente vem no formato de imagens. Para isto foi criado também um módulo de processamento de imagens (`hmImageProcessing`). Desta maneira as funcionalidades ficam claramente separadas em módulos independentes. Esta característica permite trabalhar separadamente com cada módulo ou de maneira integral com vários módulos (cenários apresentados na Fig. 9.11). A utilização deste software no ambiente médico irá fornecer informações mais precisas e muito valiosas na hora do diagnóstico.

Para informações atualizadas sobre este sistema computacional visite o site **`http://hemolab.lncc.br`** ou via email com `nacho@lncc.br` ou `feij@lncc.br`.

Apêndice Capítulo 9

9.A Terminologia

Nesta seção são introduzidas as definições de alguns termos utilizados ao longo do texto com o simples objetivo de assistir ao leitor.

Elemento: Na discretização de um domínio contínuo, a mínima porção continua (e.g., em uma triangulação um triângulo, em uma malha de volume tetraédrica um tetraedro, em uma malha regular de quadriláteros um quadrado, etc.)

Id Cliente Servidor (CSID): identificador único a nível de sistema que permite identificar objetos no servidor.

Malha: conjunto de figuras geométricas (triângulos, quadriláteros, tetraedros, prismas, etc.) que se compõem para gerar uma geometria mais complexa. Geralmente são especificadas como um conjunto de pontos e a conectividade entre eles.

Método dos Elementos finitos (FEM): método usado para encontrar a solução aproximada de um problema variacional ou das equações em derivada parciais a ele associado.

Modelo: conjunto de hipóteses sobre a estrutura ou o comportamento de um sistema físico pelo qual se procuram explicar ou prever, dentro de uma teoria científica, as propriedades do sistema.

Modelo personalizado: modelo correspondente a um individuo particular, o qual considera as suas características mecânicas e geométricas.

Nó: ponto nodal. Locais no quais os valores das incógnitas (pressão, velocidade, deslocamento, etc.) serão aproximados.

Segmento (arterial): trecho da árvore 1D que representa uma artéria. As maiores artérias do corpo humano são representadas usando segmentos.

Capítulo 10

Uso de Modelos na Simulação do Sistema Cardiovascular Humano

Para mostrar a potencialidade do ambiente computacional proposto neste trabalho, no presente Capítulo são apresentados alguns resultados obtidos com o Sistema HeMoLab, referenciando as técnicas numéricas utilizadas em cada caso e os aspectos referentes a condições de contorno e de acoplamento. Para o modelo multidimensional da bifurcação da artéria carótida são apresentadas também as geometrias padrão e real (proveniente de um paciente) utilizadas.

Em particular, os modelos apresentados no Capítulo 8 implementados no Sistema HeMoLab descrito no Capítulo 9 são utilizados para analisar as condições de fluxo na bifurcação da artéria carótida sob diferentes situações. Em um primeiro estudo, é analisada a influência de diferentes graus de estenose (gerados artificialmente) no fluxo sanguíneo. Com isto pretende-se mostrar como o Sistema HeMoLab pode ser empregado para estudar, por exemplo, se existe uma relação entre o grau de estenose de um indivíduo e curvas de pressão e fluxo observadas em diferentes pontos da árvore arterial. Também, foi comparada uma geometria padrão obtida a partir de dados bibliográficos (sem estenose) com uma geometria real para estudar como a geometria local influencia os fatores hemodinâmicos.

Uma outra aplicação do Sistema HeMoLab aqui apresentada corresponde ao estudo de como a resposta hemodinâmica é afetada pelo comportamento do sangue. Por um lado, é possível considerar o sangue como um fluido Newtoniano para o qual se tem um modelo mais simples, conseguindo-se uma boa aproximação dos dados obtidos experimentalmente nas artérias mais importantes do corpo. Por outro lado, um valor não constante para a viscosidade dinâmica pode ser utilizado para modelar o comportamento do sangue, e.g., o modelo de Casson. Neste caso também é comparada a influência dos diferentes modelos constitutivos no fluxo sanguíneo para duas geometrias diferentes da artéria carótida.

10.1 Metodologia

São descritas a seguir as geometrias utilizadas na árvore 1D e nas regiões 3D, assim como os parâmetros utilizados em cada modelo.

10.1.1 Árvore unidimensional

A árvore arterial completa foi construída de 128 segmentos (ver Fig. 8.1(a), e Tabelas 10.1 e 10.2). Os parâmetros geométricos e mecânicos baseiam-se nos propostos por Avolio [19], e estão apresentadas na Tabela 10.1. Na Tabela 10.2 são apresentadas as propriedades dos terminais Windkessel (Modelo 0D) empregados na caracterização da árvore arterial.

Nro	Nome	Cumprimento [cm]	Raio proximal [cm]	Raio distal [cm]	Eh [dy.cm ⁻¹]
1	Aorta ascendente A	1.0	1.46	1.46	741500
2	Aorta ascendente B	3.0	1.45	1.45	741500
3	Arco aórtico A	2.0	1.12	1.12	741500
4	Arco aórtico B	3.9	1.07	1.07	576200
5	Aorta torácica A	5.2	1.0	1.0	545640
6	Aorta torácica B	10.4	0.675	0.675	394000
7	Aorta abdominal A	5.3	0.61	0.61	370500
8	Aorta abdominal B e C	2.0	0.6	0.6	348000
9	Aorta abdominal D	10.6	0.58	0.58	352400
10	Aorta abdominal E	1.0	0.52	0.52	252500
11,31	Íliaca comum	5.8	0.37	0.37	368150
12,32	Íliaca externa	14.4	0.32	0.32	148700
13,33	Femoral	44.3	0.26	0.26	230900
14,34	Tibial posterior	33.1	0.25	0.25	667500
15	Sem nome	3.4	0.62	0.62	377000
16,17	Subclávia A	3.4	0.423	0.423	288700
18,19	Subclávia B	42.2	0.403	0.403	1170000
20,21	Ulnária A	6.7	0.215	0.215	679100
22,23	Ulnária B	17.1	0.203	0.203	717664
24	Carótida comum direita	20.8	0.37	0.37	264000
25	Carótida comum esquerda	19.87	0.37	0.37	264000
26	Carótida externa direita	17.7	0.177	0.177	259000
27	Carótida externa esquerda	15.48	0.21	0.21	302504
28,35	Tibial anterior	34.3	0.13	0.13	513145
29,30	Radial	23.5	0.174	0.174	682580
36	Carótida interna direita	17.7	0.177	0.177	259000
37	Carótida interna esquerda	14.59	0.2627	0.2627	378130

Tabela 10.1: Parâmetros geométricos e mecânicos.

Nro	Nome	R_1	R_2	C
		dyn.seg.cm ⁻² .ml ⁻¹		ml.cm ² .dyn ⁻¹
1	Coronária	10.00×10^3	41.00×10^3	0.7900×10^{-5}
2	Intercostal	2.78×10^3	11.12×10^3	0.1638×10^{-4}
3	Gástrica, Hepática e Esplênica	2.54×10^3	10.17×10^3	0.2967×10^{-3}
4	Renal(dois)	1.26×10^3	5.04×10^3	0.1235×10^{-3}
5	Mesentérica Superior	1.92×10^3	7.68×10^3	0.1726×10^{-3}
6	Mesentérica Inferior	16.62×10^3	66.46×10^3	0.7400×10^{-4}
7	Íliaca Interna	17.04×10^3	68.17×10^3	0.6750×10^{-4}
8	Femoral Profunda	11.60×10^3	46.39×10^3	0.5030×10^{-5}
9	Tibial Anterior	56.15×10^3	224.61×10^3	0.4170×10^{-5}
10	Tibial Posterior	9.54×10^3	38.16×10^3	0.3900×10^{-5}
11	Vertebral	16.65×10^3	66.60×10^3	0.9880×10^{-4}
12	Interóssea	211.74×10^3	846.96×10^3	0.3107×10^{-6}
13	Ulnária	10.56×10^3	42.24×10^3	0.3520×10^{-5}
14	Radial	10.56×10^3	42.24×10^3	0.3520×10^{-5}
15	Carótida	6.31×10^3	25.55×10^3	0.1330×10^{-5}

Tabela 10.2: Terminais *Windkessel*.

Quanto a viscosidade parietal, define-se o ângulo $\phi = \arctg\left(\frac{\omega k}{E}\right)$, onde $\omega = \frac{2\pi}{T}$, sendo T o período do batimento cardíaco. Este parâmetro é uma medida da magnitude relativa da força da viscosidade parietal elástica. Como condição de contorno de entrada (*inflow*), foi considerado o fluxo ejetado pelo ventrículo esquerdo na raiz da aorta. Na Fig. 8.1(b) é

apresentada a curva utilizada neste trabalho (originalmente proposta por Stettler [183]). Nas bifurcações, foi considerada continuidade na pressão e no fluxo [150]. O modelo 1D é composto de 1212 nós e 1326 elementos lineares. A solução do sistema de Eqs. (8.1) foi aproximada utilizando o Método dos Elementos Finitos (FEM), com uma formulação de *Galerkin-Least Squares* nas linhas características correspondendo às equações normais associadas ao sistema hiperbólico quando $k = 0$. Este esquema é operado com um parâmetro de estabilização correspondente ao valor ótimo do método SUPG[94].

10.1.2 Geometrias 3D

O FEM foi utilizado para achar uma aproximação das soluções dos sistemas de equações mencionados em domínios 3D. O domínio é composto de elementos tetraédricos P1b-P1 com funções de forma tipo bolha para o campo de velocidades e funções lineares para o campo da pressão [167]. A solução é estabilizada utilizando o método SUPG. As derivadas temporais são resolvidas com um esquema de diferenças finitas tipo θ -Euler, utilizando iterações de Picard para tratar a não linearidade do termo convectivo. A solução do problema completo (i.e., cada passo de tempo) é dividida em três passos: no primeiro, os graus de liberdade correspondentes aos nós bolha são eliminados por uma substituição direta e o problema 3D é resolvido utilizando os valores de pressão obtidos do modelo 1D na iteração anterior. No segundo passo, a solução para o modelo 1D é computada com a taxa de fluxo obtida do modelo 3D no primeiro passo. Finalmente, são achados os graus de liberdade correspondentes aos nós bolha para ser utilizados no passo seguinte. Este esquema de acoplamento não requer o cálculo dos invariantes de Riemann em contraste com o proposto em [62]. A posição dos nós da malha dentro do domínio é atualizada baseada na posição dos nós na superfície (obtidos utilizando Eq. (8.5)) achando a solução para um problema de Laplace para cada coordenada nodal.

10.1.3 Geometria 3D padrão

A geometria padrão (NCG), na qual três dos quatro casos apresentados estão baseados, foi proposta por Bharadvaj et. al. [25]. Esta geometria foi escalonada para ser consistente com o diâmetro da carótida comum no modelo 1D. Desta geometria, duas diferentes classes de estenoses foram consideradas como apresentadas nas Figs. 10.1(a) e 10.1(b). Estes casos foram chamados SCGI e SCGII, respectivamente. Na Tabela 10.3 as dimensões de cada uma delas é especificada. Em ambos casos a diminuição da área efetiva alcança 80% na seção SS3. A malha de elementos finitos tem 57009 nós e 323711 elementos para o caso NCG, 40373 nós e 198154 elementos para o caso SCGI e 38433 nós e 208196 elementos para o caso SCGII.

10.1.4 Geometria 3D real

A geometria correspondente a carótida real (caso RCG) foi obtida de uma imagem de CT. A imagem de entrada foi segmentada utilizando o método de segmentação baseado na Derivada Topológica S_{DT} -Discrete [110]. A partir desta segmentação e utilizando o

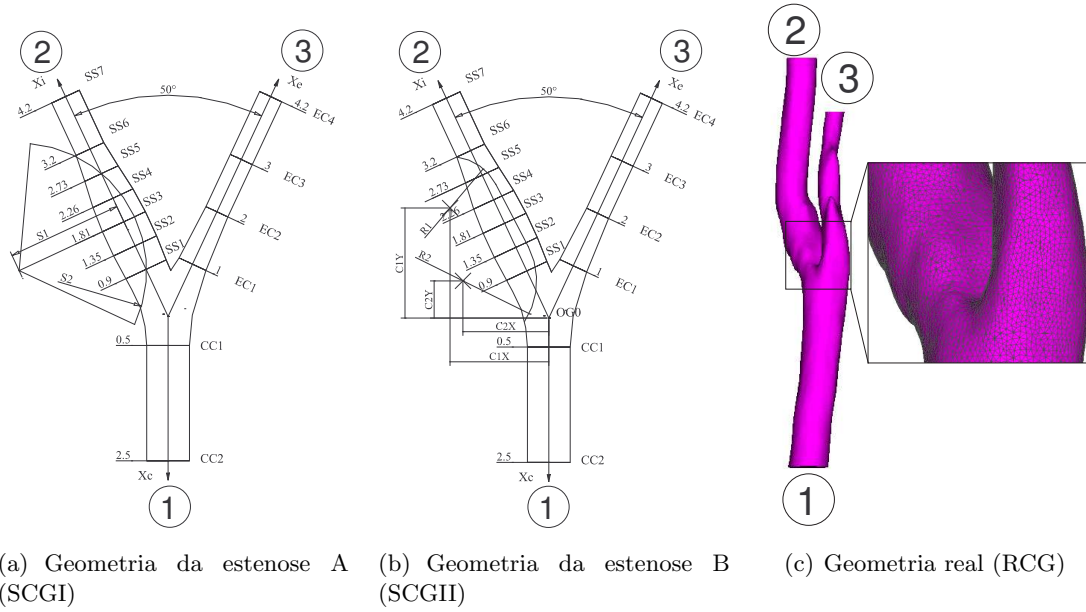


Figura 10.1: Geometrias utilizadas.

Seção	SS1	SS2	SS3	SS4	SS5	SS6	SS7
Diâmetro[cm]	0.77182	0.8214	0.8214	0.76368	0.6364	0.5254	0.5254
Seção	CC1	CC2	EC1	EC2	EC3	EC4	
Diâmetro[cm]	0.74	0.74	0.51356	0.42032	0.42032	0.42032	
SCGI	S1	S2					
S[cm]	2	2.20535					
SCGII	R1	R2	CX1	CY1	CX2	CY2	
R/CX/CY[cm]	1.00555	3.11198	-1.711571	1.785855	-3.35534	0.375378	

Tabela 10.3: Dimensões das duas geometrias estenosadas.

algoritmo de *Marching Cubes*, a geometria 3D da parede interna da artéria carótida foi reconstruída. Esta triangulação inicial foi pós-processada para obter a malha final de superfície e, utilizando o método Delaunay [195], a malha tetraédrica de volume foi gerada. A malha de elementos finitos resultante (94732 nós, 638128 elementos) é apresentada na Fig. 10.1(c).

10.2 Teste 1 - Sensibilidade a geometria

Nesta seção são apresentados os resultados numéricos correspondentes a comparação entre fluxo e pressão das diferentes geometrias padrão (Tabela 10.3). Para todos os modelos os cálculos foram feitos utilizando um tamanho de passo no tempo de $\Delta t = 1.25 \times 10^{-3} s$, logo o ciclo cardíaco foi dividido em 640 passos de tempo. Posteriormente, é feita a comparação entre a geometria padrão e a geometria real. Para correlacionar a possibilidade de desenvolvimento de estenose com as características do fluxo, o OSI (*oscillating shear*

stresses index [106]) foi calculado

$$OSI = \frac{1}{2} \left[1 - \frac{\left| \int_0^T \tau dt \right|}{\int_0^T |\tau| dt} \right] \quad (10.1)$$

sendo T o período cardíaco e τ o tensor de tensões na parede.

10.2.1 Diferentes geometrias de estenose

Na comparação entre a NCG e nas duas geometrias estenosadas SCGI e SCGII (Fig. 10.1), são apresentadas as curvas de pressão e fluxo em diferentes pontos: na Artéria Carótida Comum (CCA, marcada como ①) e depois da bifurcação na Artéria Carótida Interna (ICA, marcada como ②) e na Artéria Carótida Externa (ECA, marcada como ③). Pode-se apreciar que estas curvas não sofrem perturbações significativas.

10.2.2 Geometria real vs. geometria padrão

Nesta seção é comparada a NCG com a RCG (Fig. (10.2.2)). Para cada caso a pressão e fluxo foi comparada nos mesmos pontos mencionados anteriormente. Neste caso, não se observam diferenças na forma da curva, mas sim nos valores máximos e mínimos no fluxo e pressão, sendo no caso da geometria real levemente menores.

Finalmente, a distribuição do OSI para diferentes geometrias é apresentada nas Figuras 10.2.2 e 10.2.2. Deve-se ressaltar que fortes variações da direção do fluxo sanguíneo estão associadas a valores do OSI próximos de 0.5.

10.3 Teste 2 - Sensibilidade ao modelo constitutivo

Novamente aqui os cálculos foram realizados utilizando um tamanho de passo no tempo de $\Delta t = 1.25 \times 10^{-03} \text{ sec}$, logo o ciclo cardíaco foi dividido em 640 passos de tempo. Nas seções seguintes, são apresentados os resultados mostrando diferenças entre os modelos Newtoniano e de Casson. Para cada modelo, o OSI (*oscillating shear stresses index* [106], apresentado na seção anterior) e o WSS (*wall shear stress*) são computados. O WSS quantifica a tensão de corte média na parede arterial, e é obtido utilizando a seguinte equação

$$WSS = \frac{1}{T} \int_0^T |\tau_w| dt. \quad (10.2)$$

Estes indicadores são importantes já que diversos estudos mostram que podem indicar os locais com potencial para desenvolver acumulação de placa de ateroma, com o conseqüente crescimento da estenose [13, 72, 106].

10.3.1 Geometria padrão

Na Fig. 10.6 são apresentados os vetores de velocidade na seção transversal SS2 e no plano de simetria da geometria ideal NCG (Fig. 10.1(a)). O ponto A indica a parede arterial interna e o ponto B é o correspondente a parede externa. Em seções transversas é possível observar que a dinâmica dos vórtices ao se considerar o modelo de Casson é diferente da que corresponde ao modelo Newtoniano. No modelo Newtoniano, são observados quatro vórtices, dois principais e dois muito pequenos do lado da parede interna para $t = 0.3 s$ (Fig. 10.6(a)). Este comportamento não é observado para o modelo de Casson onde somente dois vórtices aparecem (ver Fig. 10.6(b)). Também, pode-se apreciar que os vórtices têm uma largura menor no caso de Casson, enquanto para o modelo Newtoniano estes permanecem circulares. Considerando as seções longitudinais, depois da fase sistólica, é observado que a área efetiva no seno da carótida é maior no modelo Casson, ou de maneira equivalente, a zona de re-circulação no modelo Newtoniano é maior. Isto também pode ser observado nos resultados para $t = 0.4 s$ (Fig. 10.6(c) e 10.6(d)) e $t = 0.5 s$ (Fig. 10.6(e) e 10.6(f)).

Na Fig. 10.7 são apresentados os índices OSI e WSS no seno da artéria carótida. Para o modelo Newtoniano observa-se uma distribuição em forma de anel no OSI, como resultado de uma maior zona de re-circulação. Pelo contrário, para o modelos de Casson esta distribuição não é observada, aparecendo uma distribuição mais concentrada para o OSI. Quanto ao índice WSS, para o modelo Newtoniano existem tensões mais elevadas que tendem a permanecer na parede interna da bifurcação da carótida. Isto é uma consequência direta da menor área efetiva pela qual ocorre o fluxo no caso Newtoniano comparado com o caso Casson.

Para complementar estas informações, na Fig. 10.8 são apresentadas as mesmas linhas de fluxo para $t = 0.3 s$ obtidas com ambos modelos. Pode-se observar uma importante diferença entre ambos modelos, evidenciando uma maior zona de re-circulação no caso Newtoniano.

10.3.2 Geometria real

Pela dependência existente entre a geometria e as características locais do fluxo, neste caso resulta mais difícil estabelecer uma comparação quantitativa como a realizada no caso anterior. No entanto, é possível observar algumas características próprias a cada modelo. A distribuição do OSI e WSS foram computadas (Fig. 10.9). Com relação a dinâmica dos vórtices, a Fig. 10.10 apresenta o campo de velocidade na seção transversal. Nesta figura pode-se observar a diferença entre o modelo Casson e Newtoniano, cuja dinâmica é muito mais complexa no segundo caso. Logo, resultados correspondentes a OSI e WSS (Fig. 10.9) são facilmente justificados. O índice OSI parece alcançar valores muito pequenos no caso Casson devido a um fluxo muito melhor comportado (muito mais laminar). Desconsiderando os valores máximos, a distribuição do WSS permanece similar. Neste caso, o valor máximo da tensão de cisalhamento é em ambos casos localizada na parede interior a bifurcação (Fig. 10.9(c) e 10.9(d)).

10.4 Comentários

Neste trabalho foram utilizados modelos 1D-3D acoplados para comparar diferentes graus de estenose na bifurcação da artéria carótida utilizando uma geometria padrão e uma geometria obtida diretamente a partir de imagens de um paciente. Um dos objetivos desta comparação foi quantificar a sensibilidade de fatores hemodinâmicos tais como fluxo, pressão e OSI. Como apresentado nas Fig. 10.2.1 e 10.2.2 as curvas de fluxo e pressão não são alteradas de maneira significativa ao se mudar o nível de estenose. Por outro lado, o fluxo depende fortemente da geometria como evidenciado pela comparação dos OSI apresentados nas Figs. 10.2.2 e 10.2.2.

Também foi feito um estudo da sensibilidade do fluxo a diferentes leis constitutivas utilizando os indicadores OSI e WSS. No caso Newtoniano, foram observados maiores regiões de re-circulação, tendo como consequência uma diferença no índice OSI. Mesmo que este índice não apresente grandes mudanças quantitativas, a sua distribuição no modelo de Casson é modificada.

Por outro lado, ao se considerar geometrias reais, a utilização do modelo Casson aparece como a escolha mais adequada. Além disso, estes resultados mostram que, mesmo que a análise em geometrias padrão pode levar a importantes conclusões, existem diferenças consideráveis com resultados obtidos a partir de geometrias baseadas em dados reais devido a forte dependência geométrica dos resultados. O desenvolvimento de modelos personalizados é justificado quando se objetiva entender a dinâmica em um caso particular.

Considerando o tempo de cálculo, é importante ressaltar que o tempo requerido para o modelo Casson não é maior que o correspondente ao caso Newtoniano. Logo, a adoção do primeiro modelo parece ser a alternativa mais adequada. Com isto em mente, é importante lembrar que no momento de regularizar o modelo Casson, um cuidado especial deve ser tomado na escolha dos parâmetros para evitar obter soluções incorretas (como ressaltado na Seção 8.1.5).

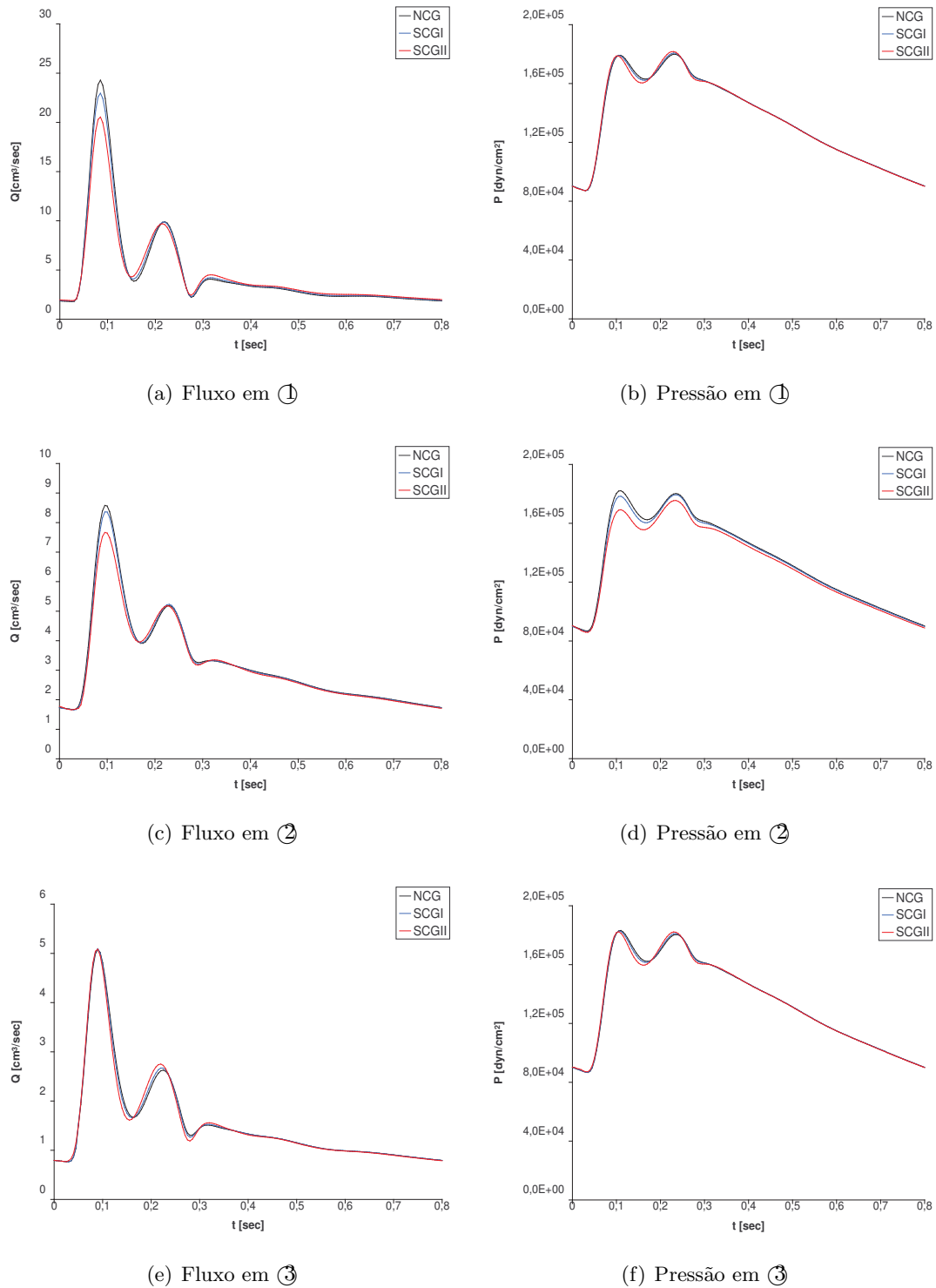


Figura 10.2: Fluxo e pressão em CCA ① , ICA ② e ECA ③ para a geometria padrão com diferentes estenoses (NCG, SCGI e SCGI).

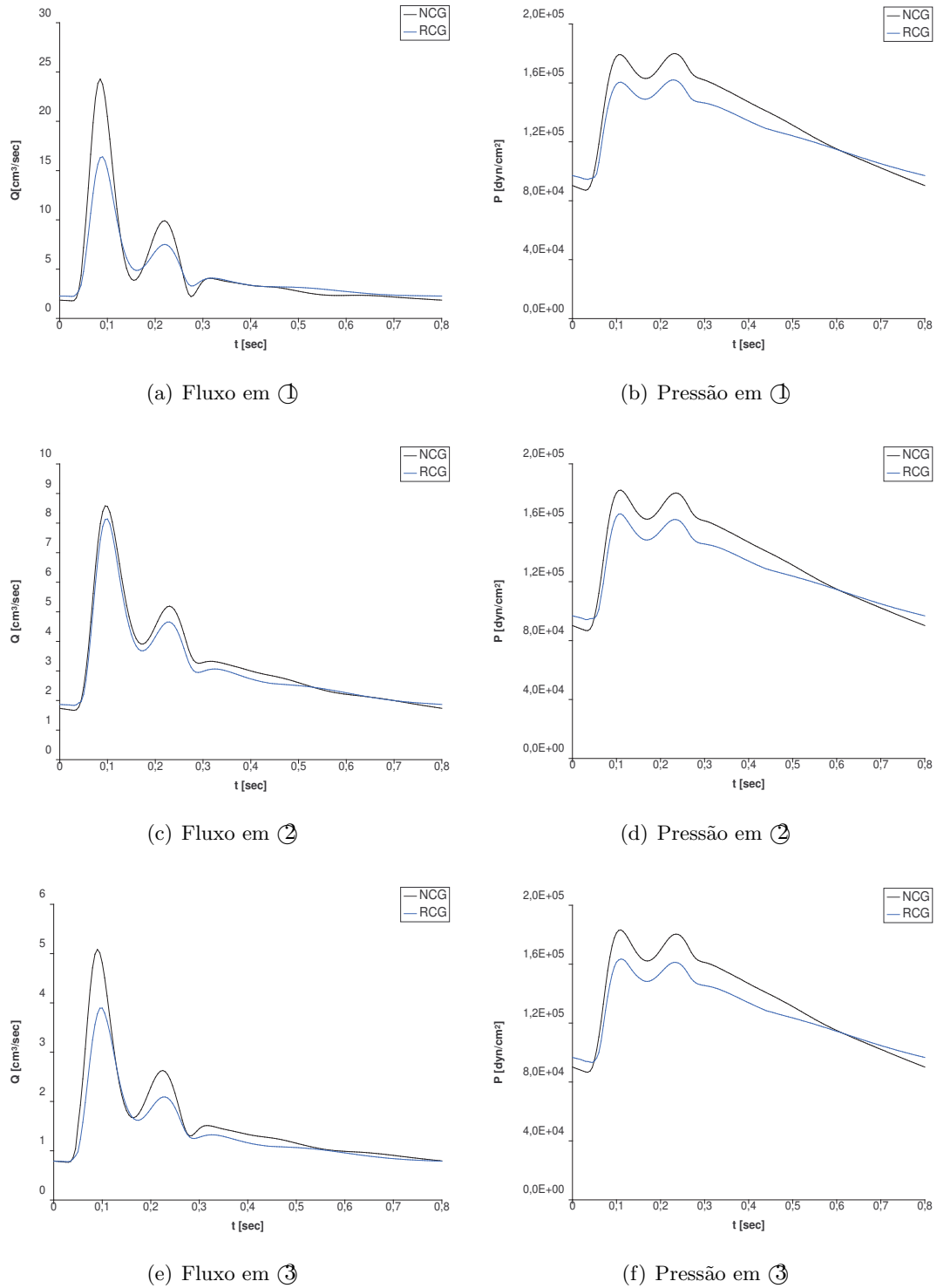


Figura 10.3: Fluxo e pressão em CCA ① , ICA ② e ECA ③ para a geometria padrão e real (NCG e RCG).

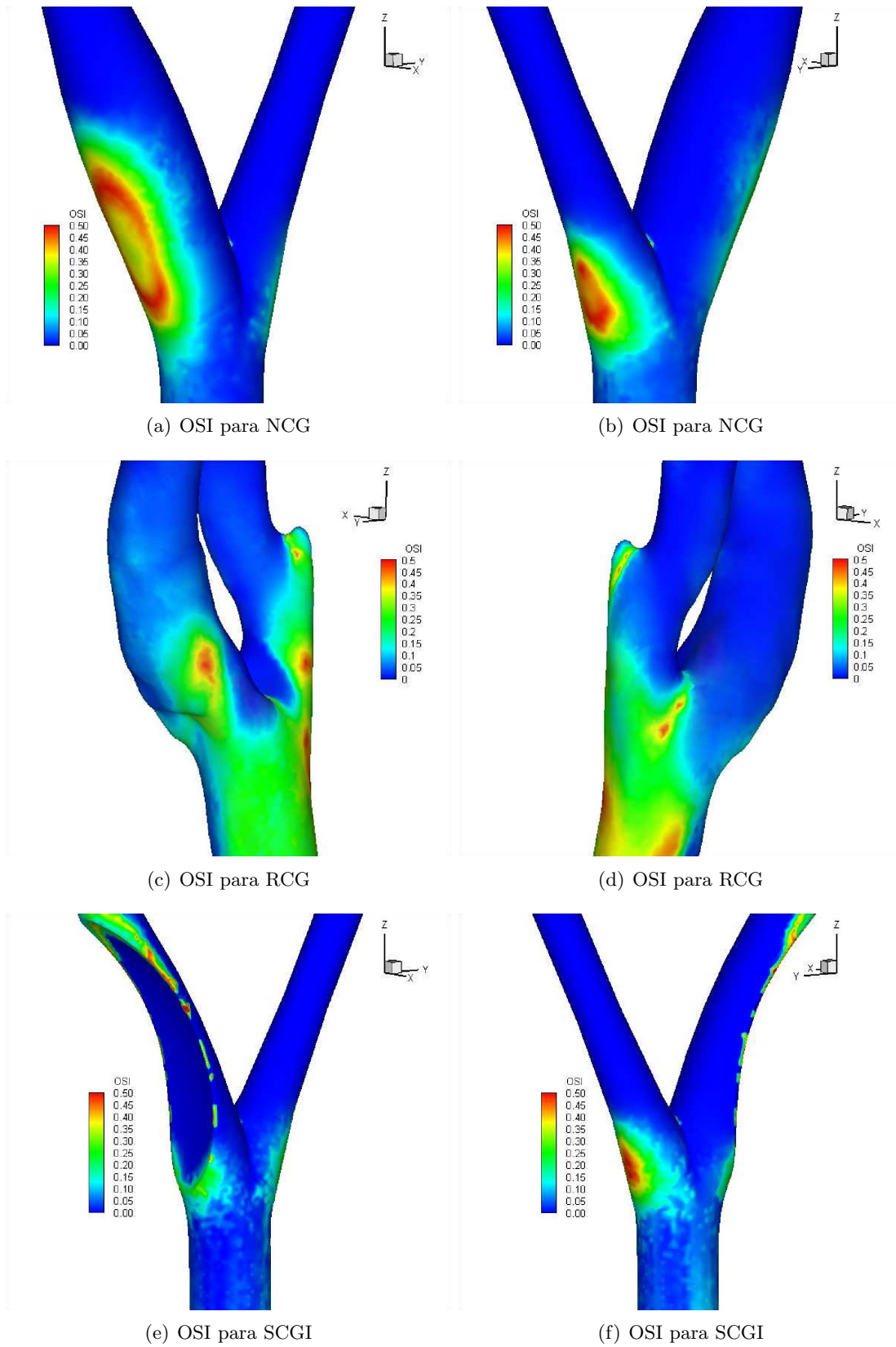


Figura 10.4: OSI para a geometria real e padrão.

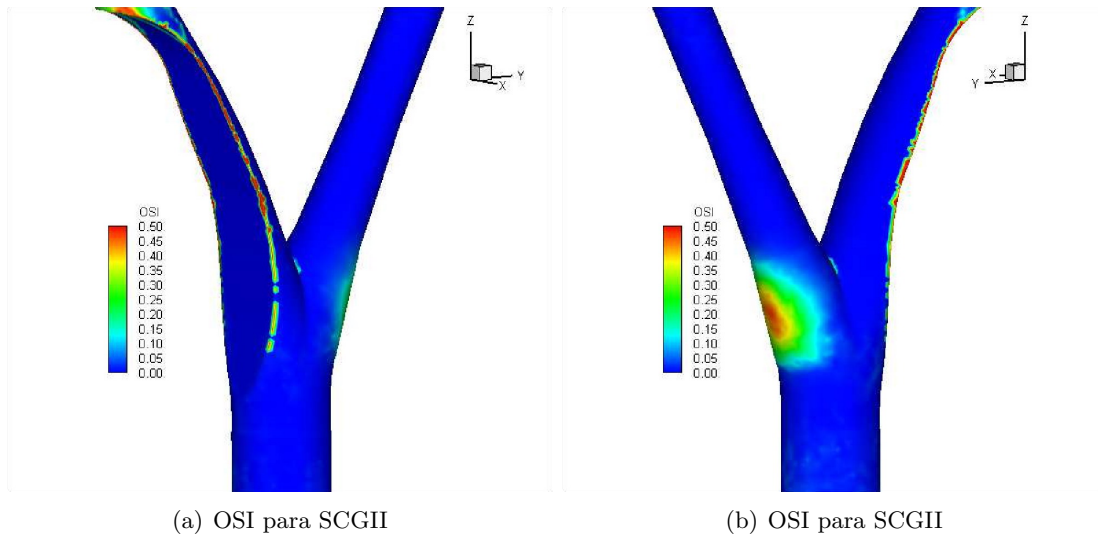


Figura 10.5: OSI para a geometria real e padrão (Cont.).

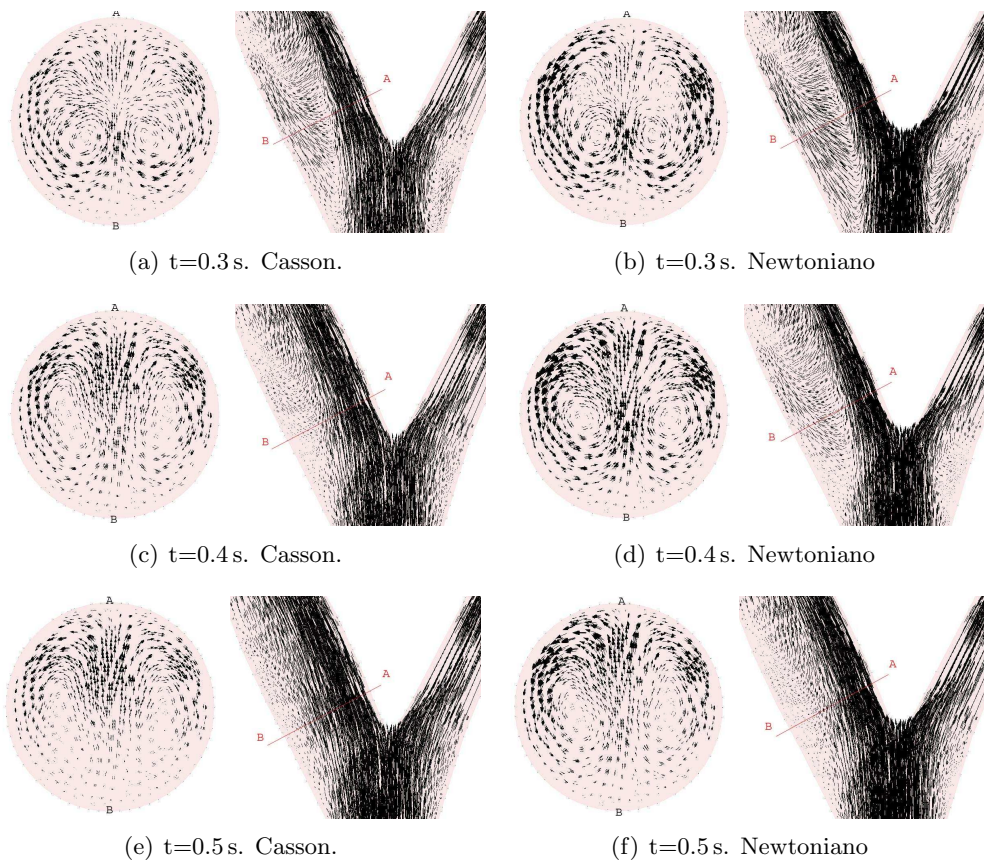


Figura 10.6: Fluxo transversal e longitudinal para a NCG.

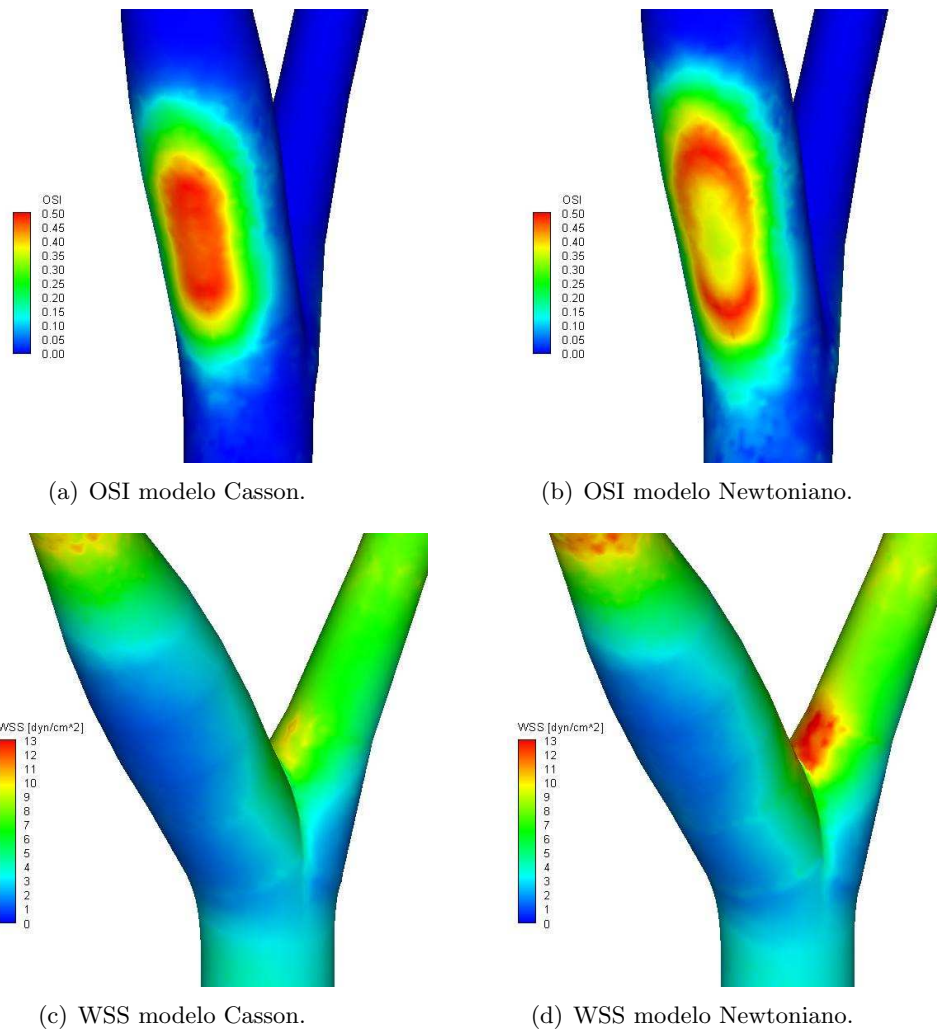


Figura 10.7: Distribuição do OSI e WSS para NCG.

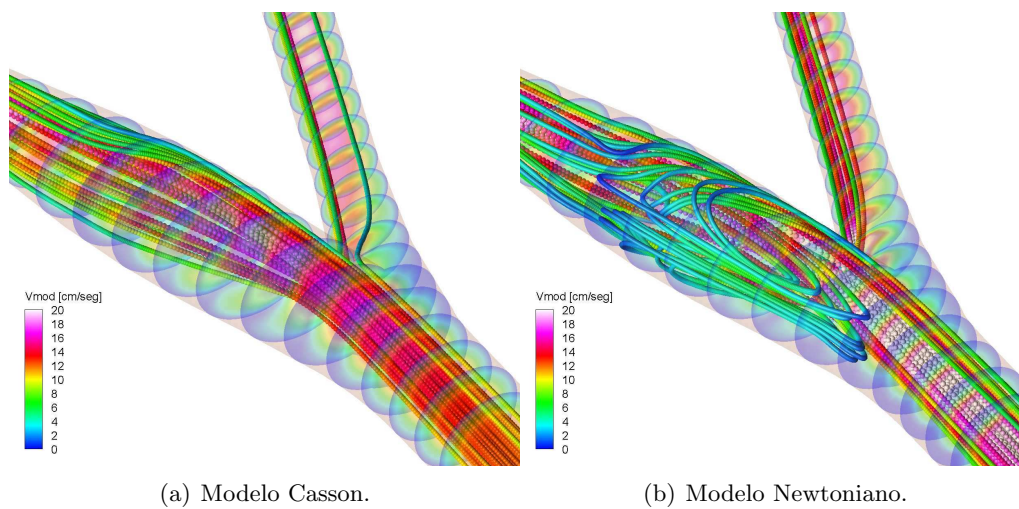


Figura 10.8: Linhas de fluxo para NCG em $t = 0.3$ s.

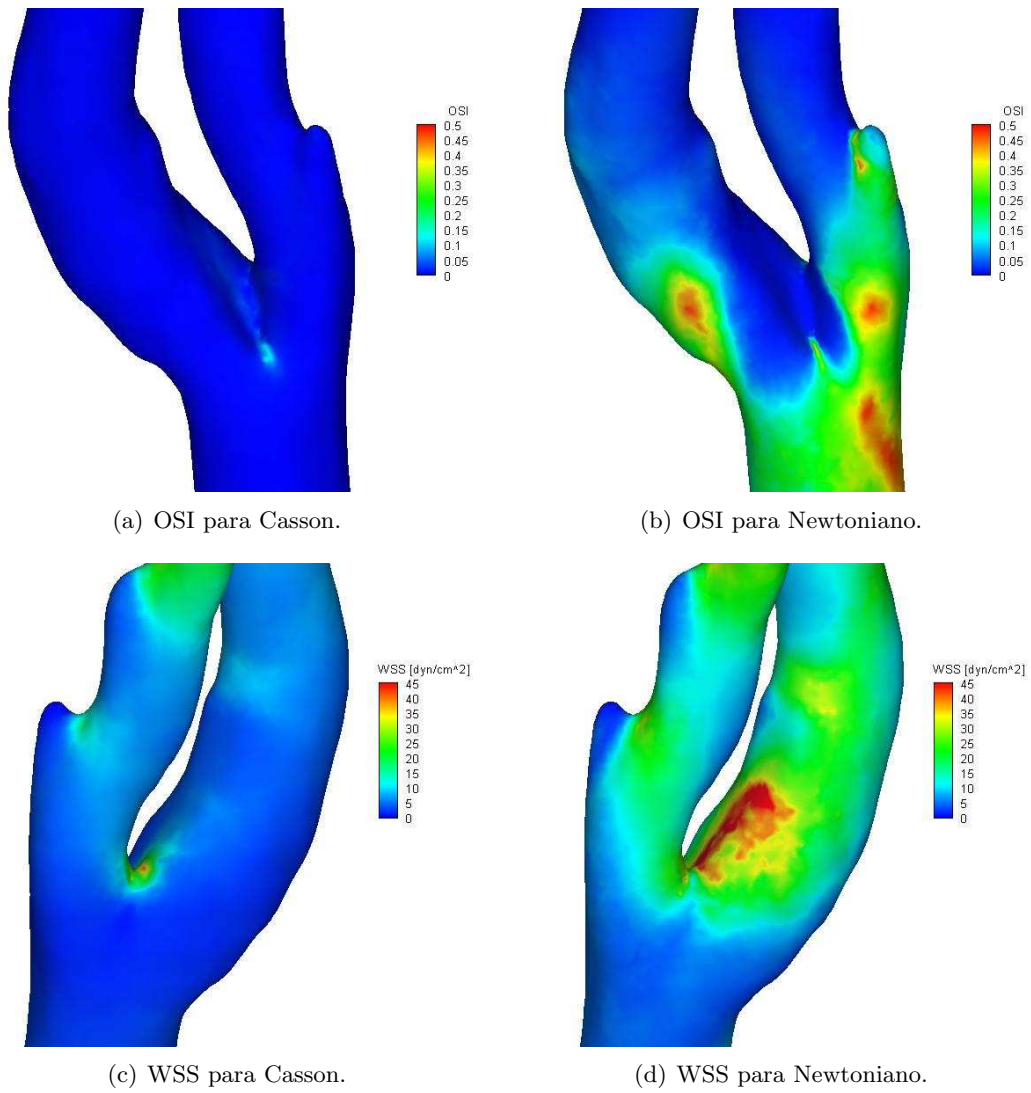


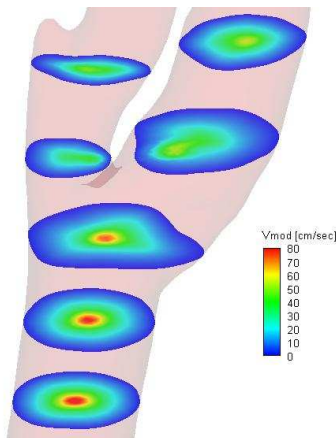
Figura 10.9: Distribuição de OSI e WSS para geometria real.



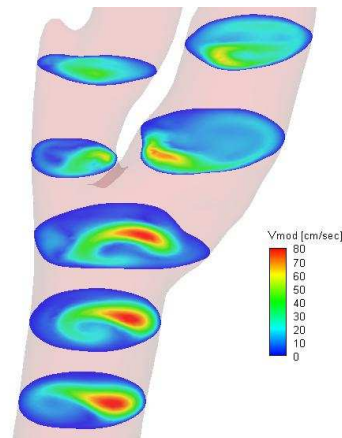
(a) Campo de velocidade transversal para modelo Casson.



(b) Campo de velocidade transversal para modelo Newtoniano.



(c) Norma da velocidade para modelo Casson.



(d) Norma da velocidade para modelo Newtoniano.

Figura 10.10: Fluxo na seção transversal para geometria real em $t = 0.3$ s.

Conclusões e comentários finais

Este trabalho teve por objetivo estudar a restauração e segmentação de imagens e sua aplicação na Modelagem do Sistema Cardiovascular Humano. O aumento da complexidade na ciência tem motivado o intercâmbio de conhecimento e técnicas entre diferentes disciplinas. Neste sentido a Derivada Topológica, originariamente concebida para tratar de maneira adequada problemas de otimização de topologia e problemas inversos, foi utilizada no processamento de imagens. Sendo assim, apresentam-se de maneira sucinta as contribuições deste trabalho nas áreas de processamento de imagens e modelagem computacional do Sistema Cardiovascular Humano. São apresentados também alguns problemas em aberto a serem estudados no futuro.

Contribuições deste trabalho

Neste trabalho foram estudados dois problemas do processamento de imagens: segmentação e restauração. O uso da derivada topológica nesta área da ciência tem mostrado resultados promissores, abrindo espaço para uma nova maneira de abordar esta classe de problemas.

Em restauração de imagens o objetivo é restaurar uma imagem que foi degradada por algum fenômeno. Neste trabalho, a derivada topológica foi utilizada para resolver este problema, sendo definida uma função custo para a qual esta derivada foi calculada. Como discutido, a derivada topológica permite obter uma expansão assintótica da função custo avaliada no domínio de definição do problema quando pequenas perturbações são introduzidas. Assim, esta derivada foi utilizada como função indicadora dos locais nos quais perturbações devem ser introduzidas de maneira que, na presença de difusão, a maior quantidade de ruído seja removida preservando as bordas. Com base nesta idéia, duas abordagens foram estudadas sendo uma delas contínua e a outra discreta. A contínua, originariamente proposta por Belaid *et al.* [22] e baseada na equação de difusão de calor estacionária, foi revisada. Neste caso, o cálculo da derivada foi apresentado em detalhe utilizando o *Topological-Shape Sensitivity Method*. Para mostrar as potencialidades do método de restauração proposto, foram apresentados alguns resultados numéricos, sendo avaliadas também algumas possíveis melhoras no algoritmo. Na abordagem discreta, foi proposto um novo algoritmo de restauração de imagens baseado em uma versão discreta da derivada topológica. Contrariamente ao caso contínuo, este é baseado na equação de difusão de calor evolutiva, apresentando também uma considerável melhora no custo computacional respeito do algoritmo contínuo. Da mesma forma, foram apresentados alguns resultados numéricos. De acordo com estes resultados, cabe ressaltar que ambos os métodos se mostraram muito eficientes na remoção de diferentes tipos de ruído, sendo robustos e de fácil ajuste dos parâmetros. Em particular, os métodos de restauração encontrados na literatura requerem um ajuste fino destes parâmetros dependendo do tipo e intensidade de ruído. Por outro lado, os métodos baseados na derivada topológica

se mostram superiores, aumentando consideravelmente o SNR das imagens respeito dos resultados obtidos com os outros métodos e sem requerer de ajustes nos parâmetros. A principal diferença dos métodos clássicos com os baseados na D_T está na forma como as bordas são detectadas. No primeiro caso, utiliza-se apenas a informação do gradiente para determinar os locais onde existem descontinuidades na imagem. Por outro lado, os métodos D_T estudam a sensibilidade de uma função custo apropriadamente definida a mudanças na topologia ou propriedades materiais, o que os faz muito mais robustos.

Em segmentação de imagens o objetivo é decompor uma imagem nos objetos ou regiões que a compõem. De maneira semelhante ao caso de restauração, e partindo do funcional de Mumford-Shah [133], foi definida uma função custo que, dada uma imagem e um conjunto de classes a ela associada, quantifica a qualidade de uma segmentação. Assim, a sensibilidade desta função custo, à introdução de pequenas perturbações na imagem segmentada, dada pela derivada topológica, foi calculada. Para este problema foi calculada a derivada topológica contínua também aplicando o *Topological-Shape Sensitivity Method*. A partir da informação fornecida pela derivada topológica, foi proposto um algoritmo de segmentação de imagens. Neste método (chamado S_{D_T} -Continuous) a solução de uma equação de estado é necessária para calcular a derivada topológica. Posteriormente, uma segunda abordagem foi proposta utilizando uma simplificação da idéia anterior, mais simples na sua implementação e de menor custo computacional, a qual utiliza uma versão discreta da derivada topológica. Neste segundo método (chamado S_{D_T} -Discrete), não é necessário resolver uma equação de estado, obtendo-se um algoritmo rápido e que permite uma implementação eficiente em 3D. Além disso, para o caso em que o valor das classes não é conhecido *a priori*, uma simples técnica de otimização destes valores foi proposta mostrando excelentes resultados. Estes dois novos algoritmos de segmentação foram comparados com outros amplamente utilizados na literatura, mostrando-se competitivos e robustos na maioria das situações. De fato, a robustez dos métodos propostos na presença de diferentes intensidade de ruído foi analisada dando excelentes resultados. Cabe ressaltar que, em contrapartida ao seu custo computacional, o S_{D_T} -Continuous mostra-se extremadamente robusto, dando resultados de melhor qualidade mesmo para altos níveis de ruído. Por outro lado com S_{D_T} -Discrete resultados de ótima qualidade são obtidos a um custo computacional muito menor.

Cabe destacar que tanto os métodos propostos baseados na derivada topológica como os existentes na literatura utilizados nas comparações (tanto de restauração como de segmentação) foram implementados computacionalmente pelo autor utilizando o software Matlab 6.5 [190]. Os arquivos correspondentes aos métodos propostos (baseados na derivada topológica) estão disponíveis no site Matlab Central.

Por último, foi estudado o uso de técnicas de modelagem na simulação computacional do Sistema Cardiovascular Humano. O objetivo deste estudo foi a integração das técnicas de processamento de imagens propostas com os modelos utilizados na simulação computacional do SCVH. Para criar este tipo de modelos, informações de diversas origem e natureza devem ser tratadas, requerendo-se o desenvolvimento de ferramentas computacionais que permitam operar com grandes volumes de dados. Visando estas necessidades, foi desen-

volvido o sistema HeMoLab. Esta ferramenta foi implementada sobre o ParaView, um software de visualização científica que utiliza computação paralela para processar grandes volumes de informação. Completamente desenvolvido em ANSI-C++, este software pode ser executado em diferentes plataformas podendo até trabalhar em paralelo.

Para modelar os fenômenos mencionados anteriormente, foram utilizados modelos multidimensionais (i.e., 0D, 1D, 3D) tratados como módulos independentes, quais sejam: hm1DModule, hm3DModule, hmCoupling. Estes módulos podem também ser empregados de maneira conjunta permitindo o desenvolvimento de modelos do sistema arterial completo. Para poder criar modelos personalizados, é necessário incorporar informações específicas dos pacientes. Estes tipos de informações geralmente vêm em forma de imagens médicas. Para isto foi criado também um módulo de processamento de imagens hmImageProcessing. Desta maneira funcionalidades diferentes ficam separadas em módulos independentes. Esta característica permite trabalhar com cada módulo separadamente ou de forma integrada.

Esta ferramenta computacional permite desenvolver novos procedimentos de diagnóstico, assim como estudar de maneira detalhada, tratar e acompanhar a evolução de doenças associadas ao SCVH. Com o uso deste software é também possível prever o resultado de importantes intervenções cirúrgicas como *bypass* coronários, implantação de *stents*, *coils* e *clips* em aneurismas.

Comentários finais

Como de forma alguma os métodos propostos e as aplicações estudadas esgotam as potencialidades do uso da derivada topológica em processamento de imagens, existem ainda uma variedade de problemas a serem considerados desde esta perspectiva. Efetivamente, várias questões do processamento de imagens podem ser abordadas desta ótica, sendo algumas delas:

- O uso da derivada topológica no problema de **reconstrução de imagens** a partir de projeções. Este é de fato um problema inverso, área na qual a derivada topológica tem mostrado muita utilidade. Em particular, resultados preliminares [116] evidenciam a potencialidade desta ferramenta no referido problema.
- O acompanhamento de objetos em seqüências de imagens (*object tracking*) requer achar a função característica da região por ele ocupada.
- Decomposição de imagens em textura e geometria, onde uma imagem pode ser interpretada como uma soma destas componentes. Neste caso, no processo de decomposição, a geometria pode ser identificada pela sua função característica.
- O problema de *In-painting*, onde o objetivo é recuperar partes perdidas de uma imagem, é mais um problema desafiador para ser considerado.

Assim sendo, o uso da derivada topológica no processamento de imagens é de fato uma promissória área de pesquisa, tanto do ponto de vista teórico quanto de aplicações.

Por outro lado, na modelagem computacional de sistemas biológicos, como é o caso do Sistema Cardiovascular Humano, tem muito a ser feito. Alguns destes desafios são: desenvolvimento de novos modelos simplificados que permitam capturar com maior detalhe o fluxo nas artérias principais do corpo; melhorar as ferramentas de refinamento de malhas de elementos finitos; utilidades e ferramentas específicas para predição de ruptura de aneurismas; simulação de implantes de *stents* e simulação de *bypass*. O uso deste tipo de tecnologia na medicina tem pela frente uma série de metas a serem cumpridas, dentre elas:

- Especialistas devem ser treinados para o uso destes tipos de técnicas.
- Devem ser definidos protocolos para a sua utilização em paciente reais.
- Os modelos utilizados devem ser constantemente aprimorados para conseguir ajustar com maior precisão o observado no sistema cardiovascular.
- A aplicação das técnicas vindas da mecânica e engenharia no estudo de outras áreas da medicina e da saúde, como por exemplo traumatologia, onde técnicas de análise estrutural podem ser de grande utilidade.

Estes fatos fazem da modelagem computacional de Sistemas Biológicos uma área muito atraente e com inúmeras linhas de pesquisa a serem exploradas.

Como levantado no início deste texto, nos últimos anos as doenças cardiovasculares têm-se tornado a principal causa de morte no país e no mundo. Estes fatos têm motivado à comunidade científica e médica a desenvolver novas técnicas e procedimentos para diagnosticar, tratar e acompanhar de maneira mais eficiente a evolução deste males nos seres humanos. Paralelamente, técnicas de modelagem e simulação computacional permitem estudar em detalhe os fenômenos físicos e químicos que originam e favorecem a evolução destas doenças. Estas técnicas de modelagem em muito podem contribuir no desenvolvimento de novas metodologias de medicina preventiva, assistência no treinamento de médicos, avaliação de risco de ruptura de aneurismas, simulação do transporte de fármacos e substâncias pelo fluxo sanguíneo, simulação de diferentes tipos de intervenções como *bypass* coronários, planejamento cirúrgico e implantação de *stents*, *coils* e *clips* em aneurismas entre outras. Finalmente, espera-se que a utilização desta tecnologia no ambiente médico forneça informações precisas e valiosas aos especialistas, provocando um impacto significativo na saúde e qualidade de vida das pessoas.

Apêndice A

Discretização de Equações Diferenciais

Com o intuito de deixar este trabalho auto-contido, em este apêndice serão apresentados algumas das aproximações em diferenças finitas e elementos finitos utilizadas na área de restauração e segmentação de imagens. Entretanto para o leitor interessado os seguintes livros básicos são recomendados:

- C. Hirsch, *Numerical Computation of Internal and External Flows - Volume 1: Fundamentals of Numerical Discretization*. John Wiley and Sons, Brussels, Belgium, 1989, ISBN 0471917621.
- O. C. Zienkiewicz and R. L. Taylor, *Finite Element Method: Volume 1- The Basis*. Butterworth Heinemann, London, 2000, 712 pp, ISBN 0750650494.
- T. J. R. Hughes, *The Finite Element Method - Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, 2000, ISBN 0486411818.

A.1 Diferenças Finitas

A aproximação por diferenças finitas é o mais antigo método utilizado para obter soluções numéricas de equações diferenciais. A idéia deste método é bastante simples e corresponde a estimar o valor das derivadas por meio de um quociente de duas diferenças de acordo com a definição da derivada [90].

Para uma função $u(x)$ a derivada no ponto x é definida como

$$u_{,x} \equiv \frac{\partial u}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{u(x + \Delta x) - u(x)}{\Delta x} \quad (\text{A.1})$$

Sendo Δx finito e pequeno a expressão do lado direito da equação é uma aproximação do valor exato de $u_{,x}$. A aproximação é melhorada ao se reduzir Δx , mas para todo valor finito de Δx é introduzido um erro (chamado de erro de truncamento) o qual vai para zero quando Δx vai para zero. A potência de Δx com a qual este erro vai para zero é chamada de ordem da aproximação e pode ser obtida a partir da expansão em séries de Taylor de $u(x + \Delta x)$ ao redor do ponto x . O conceito de aproximação por diferenças finitas se baseia em propriedades das expansões de Taylor. Desenvolvendo $u(x + \Delta x)$ em séries de Taylor, obtém-se

$$u(x + \Delta x) = u(x) + \Delta x u_{,x}(x) + \frac{\Delta x^2}{2} u_{,xx}(x) + \dots \quad (\text{A.2})$$

e logo, para uma ordem maior de Δx ,

$$\frac{u(x + \Delta x) - u(x)}{\Delta x} = u_{,x}(x) + \frac{\Delta x}{2} u_{,xx}(x) + \dots \quad (\text{A.3})$$

Esta aproximação para $u_{,x}(x)$ é dita de “primeira ordem” em Δx e pode se escrever como

$$\frac{u(x + \Delta x) - u(x)}{\Delta x} = u_{,x}(x) + \mathcal{O}(\Delta x) \quad (\text{A.4})$$

indicando que o erro de truncamento $\mathcal{O}(\Delta x)$ vai para zero como a primeira potência em Δx .

Ao se considerar o espaço unidimensional, eixo x , onde a discretização foi feita de maneira que o espaço contínuo foi substituído por N pontos discretos x^i , $i = 1 \dots N$. Sem perda de generalidade pode se considerar $x^i = i\Delta x$.

O seguinte esquema de aproximação por diferenças finitas pode ser definido para a primeira derivada $u_x(i) \equiv \frac{\partial u}{\partial x}(x^i)$.

$$\begin{aligned} u_{,x}^+(i) &\equiv \frac{\partial u}{\partial x}(x^i) = \frac{u^{i+1} - u^i}{\Delta x} + \mathcal{O}(\Delta x) && \text{diferença adiantada} \\ u_{,x}^-(i) &\equiv \frac{\partial u}{\partial x}(x^i) = \frac{u^i - u^{i-1}}{\Delta x} + \mathcal{O}(\Delta x) && \text{diferença atrasada} \end{aligned} \quad (\text{A.5})$$

Uma aproximação de segunda ordem é obtida a partir da diferença centrada

$$u_{,x}^c(i) \equiv \frac{\partial u}{\partial x}(x^i) = \frac{u^{i+1} - u^{i-1}}{2\Delta x} + \mathcal{O}(\Delta x^2) \quad \text{diferença centrada} \quad (\text{A.6})$$

Um grande número de aproximações por diferenças finitas pode ser obtido para as derivadas de funções. Utilizando estes esquemas é possível aproximar derivadas no espaço e no tempo.

A.1.1 Discretização no tempo e no espaço

Nos problemas sendo estudados, o domínio do espaço corresponde ao plano $x - y$, logo considera-se uma malha regular de tamanho $\Delta x \times \Delta y$. No tempo é considerado um tamanho de passo Δt .

Como exemplo considera-se a equação não-linear de calor, dada por

$$u_{,t} = \text{div}(c(|\nabla u|)\nabla u) \quad \text{em } \Omega \times [0, T], \quad (\text{A.7})$$

onde o termo da divergência pode ser reescrito como

$$\text{div}(c(|\nabla u|)\nabla u) = \text{tr} \begin{bmatrix} (c(|\nabla u|)u_{,x})_{,x} & (c(|\nabla u|)u_{,x})_{,y} \\ (c(|\nabla u|)u_{,y})_{,x} & (c(|\nabla u|)u_{,y})_{,y} \end{bmatrix}. \quad (\text{A.8})$$

Logo, a Eq. (A.7) é equivalente a

$$u_{,t} = (c(|\nabla u|)u_{,x})_{,x} + (c(|\nabla u|)u_{,y})_{,y} \quad \text{em } \Omega \times [0, T] \quad (\text{A.9})$$

Aplicando o esquema de diferenças anteriormente descrito na derivada temporal na Eq.

(A.9) tem-se

$$u_{,t} = \frac{u_{t+1}^{i,j} - u_t^{i,j}}{\Delta t} \quad (\text{A.10})$$

Aplicando o mesmo esquema nas derivadas em x e em y obtém-se

$$\begin{aligned} u_{,x}^+ &= \frac{u_t^{i+1,j} - u_t^{i,j}}{\Delta x} \\ u_{,y}^+ &= \frac{u_t^{i,j+1} - u_t^{i,j}}{\Delta y} \end{aligned} \quad (\text{A.11})$$

A derivada atrasada (a qual utilizaremos para calcular derivadas segundas) é dada por

$$\begin{aligned} u_{,x}^- &= \frac{u_t^{i,j} - u_t^{i-1,j}}{\Delta x} \\ u_{,y}^- &= \frac{u_t^{i,j} - u_t^{i,j-1}}{\Delta y} \end{aligned} \quad (\text{A.12})$$

Para calcular a aproximação do termo $c(|\nabla u|u_{,x})_{,x}$ utiliza-se uma combinação das derivadas atrasada e adiantada, seja

$$\begin{aligned} (c(|\nabla u|)u_{,x})_{,x} &= \frac{c^{i+1/2,j} u_{,x}^+ - c^{i-1/2,j} u_{,x}^-}{\Delta x^2} \\ &= \frac{c^{i+1/2,j} (u_t^{i+1,j} - u_t^{i,j}) - c^{i-1/2,j} (u_t^{i,j} - u_t^{i-1,j})}{\Delta x^2}, \end{aligned}$$

sendo $c^{i,j} = c(|\nabla u(i,j)|)$. Operando de maneira semelhante, obtém-se $c(|\nabla u|u_{,y})_{,y}$

$$(c(|\nabla u|)u_{,y})_{,y} = \frac{c^{i,j+1/2} (u_t^{i,j+1} - u_t^{i,j}) - c^{i,j-1/2} (u_t^{i,j} - u_t^{i,j-1})}{\Delta y^2}.$$

Para o caso particular de imagens, consideram-se Δx e Δy como sendo o tamanho do pixel, ou seja $\Delta x = \Delta y = 1$. Juntado os diferentes termos obtém-se a versão discreta de (A.7)

$$\begin{aligned} \frac{u_{t+1}^{i,j} - u_t^{i,j}}{\Delta t} &= c^{i,j+1/2} (u_t^{i,j+1} - u_t^{i,j}) - c^{i,j-1/2} (u_t^{i,j} - u_t^{i,j-1}) \\ &+ c^{i+1/2,j} (u_t^{i+1,j} - u_t^{i,j}) - c^{i-1/2,j} (u_t^{i,j} - u_t^{i-1,j}) \end{aligned} \quad (\text{A.13})$$

A maneira natural para achar a solução desta equação de maneira iterativa, é escrever o valor de $u_{t+1}^{i,j}$ como uma função $u_t^{i,j}$, seja

$$\begin{aligned} u_{t+1}^{i,j} &= u_t^{i,j} + \Delta t \left(c^{i,j+1/2} (u_t^{i,j+1} - u_t^{i,j}) - c^{i,j-1/2} (u_t^{i,j} - u_t^{i,j-1}) \right. \\ &\quad \left. + c^{i+1/2,j} (u_t^{i+1,j} - u_t^{i,j}) - c^{i-1/2,j} (u_t^{i,j} - u_t^{i-1,j}) \right) \end{aligned} \quad (\text{A.14})$$

Esta expressão pode ser escrita de maneira mais simples ao se introduzir uma pequena mudança na notação. Substitui-se, então, o par de índices i, j (para $i = 1 \dots N$, $j = 1 \dots M$) por s (para $s = 1 \dots N * M$). Considera-se também o conjunto n_s , con-

junto de vizinhos do pixel s (denotado na notação anterior como i, j), como sendo os pixels nas direções $\{\mathbf{north}, \mathbf{south}, \mathbf{east}, \mathbf{west}\}$ do pixel s (o que na notação anterior seria $\{i, j+1; i, j-1; i-1, j; i+1, j\}$). Usando esta nova notação, a equação anterior pode ser reescrita como

$$u_{t+1}^s = u_t^s + \Delta t \sum_{p \in n^s} c(|\hat{\Delta} u_t^{s,p}|) \hat{\Delta} u_t^{s,p} \quad (\text{A.15})$$

sendo u^s a intensidade da imagem discretizada no pixel s numa malha 2D regular, t o instante de tempo, Δt o tamanho de passo no tempo, e $\hat{\Delta} u_t^{s,p}$ é dado por:

$$\hat{\Delta} u_t^{s,p} = u^p - u_t^s \quad (\text{A.16})$$

Este tipo de discretização permite achar a solução do sistema de equações de maneira iterativa.

A.1.2 Análise de estabilidade

Procura-se uma condição que garanta que a solução numérica se aproxima da solução da equação diferencial sendo discretizada. Seja então \bar{u}_t^{ij} a solução exata no ponto discreto ij da malha e u_t^{ij} a solução numérica, logo pode-se considerar

$$u_t^{ij} = \bar{u}_t^{ij} - \varepsilon_t^{ij}, \quad (\text{A.17})$$

sendo ε_t^{ij} o erro no ponto ij no tempo t . Por simplicidade, é considerado o problema evolutivo linear com coeficientes constantes (i.e., Eq. (2.19)). Como \bar{u}_t^{ij} satisfaz exatamente o esquema, é possível provar que ε_t^{ij} evolui no tempo na mesma forma que a solução numérica u_t^{ij} , i.e.

$$\frac{\varepsilon_{t+1}^{ij} - \varepsilon_t^{ij}}{\Delta t} = \frac{(\varepsilon_t^{ij+1} - 2\varepsilon_t^{ij} + \varepsilon_t^{ij-1})}{\Delta y^2} + \frac{(\varepsilon_t^{i+1j} - 2\varepsilon_t^{ij} + \varepsilon_t^{i-1j})}{\Delta x^2}$$

Ao se considerar o problema periódico, tanto a solução como o erro podem ser expressados em série de Fourier em cada instante de tempo, obtendo-se uma soma finita de harmônicas, dada por

$$\varepsilon_t^{ij} = \sum_{i'=-N}^N \sum_{j'=-M}^M E_t^{i'j'} e^{I i i' \frac{\pi}{N} + I j j' \frac{\pi}{M}}$$

sendo $I = \sqrt{-1}$ e $E_t^{i'j'}$ a amplitude da harmônica $i'j'$. Pode-se considerar também

$$\psi_x = \frac{i' \pi}{N}$$

$$\psi_y = \frac{j' \pi}{M}$$

Tem-se então que uma única harmônica do erro é dada por $E_t^{i'j'} e^{I i i' \frac{\pi}{N} + I j j' \frac{\pi}{M}}$, e sua evolução no tempo é dada pelo mesmo esquema numérico que a solução u_t^{ij} . Logo, desconsiderando

os índices $i'j'$, obtém-se:

$$\begin{aligned} \frac{(E_{t+1} - E_t)}{\Delta t} e^{I(\psi_x + j\psi_y)} - E_t \left[\frac{e^{I(\psi_x + (j+1)\psi_y)} - 2e^{I(\psi_x + j\psi_y)} + e^{I(\psi_x + (j-1)\psi_y)}}{\Delta y^2} \right. \\ \left. - \frac{e^{I((i+1)\psi_x + j\psi_y)} - 2e^{I(\psi_x + j\psi_y)} + e^{I((i-1)\psi_x + j\psi_y)}}{\Delta x^2} \right] = 0 \end{aligned} \quad (\text{A.18})$$

no caso estudado o espaço é discretizado utilizando os pixels da imagem, tem-se então que $\Delta x = \Delta y = 1$, logo

$$\begin{aligned} \frac{(E_{t+1} - E_t)}{\Delta t} e^{I(\psi_x + j\psi_y)} - E_t \left[e^{I(\psi_x + (j+1)\psi_y)} + e^{I(\psi_x + (j-1)\psi_y)} + e^{I((i+1)\psi_x + j\psi_y)} \right. \\ \left. + e^{I((i-1)\psi_x + j\psi_y)} - 4e^{I(\psi_x + j\psi_y)} \right] = 0 \end{aligned} \quad (\text{A.19})$$

Para assegurar a estabilidade, deve-se garantir que o erro não cresça ao longo do tempo, i.e. $|G| = \left| \frac{E_{t+1}}{E_t} \right| \leq 1$. Logo, a Eq. (A.19) pode ser reescrita como

$$(E_{t+1} - E_t) e^{I(\psi_x + j\psi_y)} - E_t \Delta t e^{I(\psi_x + j\psi_y)} \left[e^{I\psi_x} + e^{-I\psi_x} + e^{I\psi_y} + e^{-I\psi_y} - 4 \right] = 0 \quad (\text{A.20})$$

Assim, a seguinte condição deve ser satisfeita

$$|G| = |1 + 2 \Delta t [\cos \psi_x + \cos \psi_y - 2]| \leq 1 \quad (\text{A.21})$$

Operando em (A.21), obtém-se

$$\begin{aligned} -1 &\leq 1 + 2 \Delta t [\cos \psi_x + \cos \psi_y - 2] \leq 1 \\ -2 &\leq 2 \Delta t [\cos \psi_x + \cos \psi_y - 2] \leq 0 \\ -1 &\leq \Delta t [\cos \psi_x + \cos \psi_y - 2] \leq 0, \end{aligned}$$

multiplicando por -1 tem-se

$$\begin{aligned} 1 &\geq \Delta t [2 - \cos \psi_x - \cos \psi_y] \geq 0 \\ \frac{1}{2 - \cos \psi_x - \cos \psi_y} &\geq \Delta t \geq 0. \end{aligned}$$

Considerando que $-1 \leq \cos \psi_x \leq 1$ e $-1 \leq \cos \psi_y \leq 1$ tem-se

$$\frac{1}{4} \geq \Delta t \geq 0,$$

obtendo-se então a condição de estabilidade (também conhecida como condição de Courant-Friedrichs-Levy) para o problema sob consideração.

A.2 Discretização via Elementos Finitos

Seja o seguinte problema de minimização:

Problema 4. *Achar u que minimize o funcional:*

$$\mathcal{F}(u) = \frac{1}{2} \left(\int_{\Omega} |u_0 - Ru|^2 d\Omega + 2\lambda \int_{\Omega} |\nabla u|^2 d\Omega \right) \text{ para } u \in \mathcal{U} \quad (\text{A.22})$$

sendo,

$$\mathcal{U} = H^1(\Omega) = \{u \in L^2(\Omega); \nabla u \in L^2(\Omega)\}.$$

O mínimo deste problema satisfaz a seguinte condição:

$$\delta\mathcal{F}(u; \eta) = \int_{\Omega} (R^*Ru - R^*u_0)\eta \, d\Omega + 2\lambda \int_{\Omega} \nabla u \cdot \nabla \eta \, d\Omega \quad \forall \eta \in \mathcal{V}, \quad (\text{A.23})$$

sendo $\delta\mathcal{F}(u; \eta)$ a primeira variação do funcional $\mathcal{F}(u)$.

A Eq. (A.23) é a forma variacional do problema de mínimo associado ao funcional (A.22). Devido à dificuldade de achar a solução exata desta equação, procura-se uma aproximação da mesma pelo Método de Galerkin.

A.3 Método de aproximação de Galerkin

Tem-se então o seguinte problema:

Problema 5. *Achar $u \in \mathcal{U}$ tal que:*

$$\int_{\Omega} (R^*Ru - R^*u_0)\eta + 2\lambda(\nabla u \cdot \nabla \eta) \, d\Omega = 0 \quad \forall \eta \in \mathcal{U} \quad (\text{A.24})$$

O primeiro passo no desenvolvimento do método é construir uma aproximação de dimensão finita do espaço \mathcal{U} [209]. Este conjunto de funções será chamado de \mathcal{U}^h . O supra-índice refere-se à associação de \mathcal{U}^h a uma **malha** ou **discretização** do domínio Ω , parametrizada pelo tamanho característico h . Pretende-se assumir que \mathcal{U}^h é um subconjunto de \mathcal{U} , seja:

$$\mathcal{U}^h \subset \mathcal{U} \quad (\text{A.25})$$

Utilizando então as funções $u^h \in \mathcal{U}^h$, pode-se reescrever o Problema 5 na sua forma (fraca) aproximada, ou seja:

Problema 6. *Achar $u^h \in \mathcal{U}^h$ tal que:*

$$\int_{\Omega} (R^*Ru^h - R^*u_0) \cdot v^h + 2\lambda(\nabla u^h \cdot \nabla v^h) \, d\Omega = 0 \quad \forall v^h \in \mathcal{U}^h \quad (\text{A.26})$$

O Problema 6 é a forma de Galerkin do Problema 5.

A.3.1 Forma algébrica

A forma de Galerkin do problema leva a um sistema linear de equações algébricas acopladas. De fato, construindo \mathcal{U}^h a partir de todas as combinações lineares das funções dadas por $N_A : \bar{\Omega} \rightarrow \mathbb{R}$, sendo $A = 1, 2, \dots, n$. Isto significa que se $v^h \in \mathcal{U}^h$, existem constantes v_A , $A = 1, 2, \dots, n$, tal que:

$$\begin{aligned} v^h &= \sum_{A=1}^n v_A N_A \\ &= v_1 N_1 + v_2 N_2 + \dots + v_n N_n. \end{aligned} \quad (\text{A.27})$$

As funções N_A 's são chamadas de **funções de forma**. O mesmo pode ser feito com u^h , a solução do problema:

$$\begin{aligned} u^h &= \sum_{B=1}^n u_B N_B \\ &= u_1 N_1 + u_2 N_2 + \dots + u_n N_n. \end{aligned} \quad (\text{A.28})$$

Ao substituir estas expressões em (A.26) e reorganizar tem-se:

$$\sum_{B=1}^n \sum_{A=1}^n \left(\int_{\Omega} R^* R N_B N_A + 2\lambda (\nabla N_B \cdot \nabla N_A) d\Omega \right) u_B - \sum_{A=1}^n \left(\int_{\Omega} R^* u_0 N_A d\Omega \right) = 0 \quad (\text{A.29})$$

já que o anterior é valido $\forall v^h \in \mathcal{U}^h$, em particular é verdade $\forall v_A$. Considerando:

$$\begin{aligned} K_{AB} &= \left(\int_{\Omega} R^* R N_B N_A + 2\lambda (\nabla N_B \cdot \nabla N_A) d\Omega \right) \\ f_A &= \int_{\Omega} R^* u_0 N_A d\Omega \\ \mathbf{K} &= [K_{AB}] \\ \mathbf{f} &= [f_A] \\ \mathbf{u} &= [u_B] \end{aligned} \quad (\text{A.30})$$

pode-se reescrever:

$$\mathbf{K} \mathbf{u} = \mathbf{f} \quad (\text{A.31})$$

Pode-se então, neste ponto, reescrever o Problema 6 na sua forma matricial:

Problema 7. Dado \mathbf{K} e \mathbf{f} , achar \mathbf{u} tal que:

$$\mathbf{K} \mathbf{u} = \mathbf{f} \quad (\text{A.32})$$

Este problema tem sido largamente estudado, existindo diversos métodos para achar a solução \mathbf{u} [15]. A matriz \mathbf{K} pode ser calculada através da contribuição de cada uma das matrizes \mathbf{K}^e , sendo \mathbf{K}^e a expressão (A.29) calculada no domínio Ω^e de um dado elemento e . Isto será estudado com mais detalhe na Seção A.3.3.

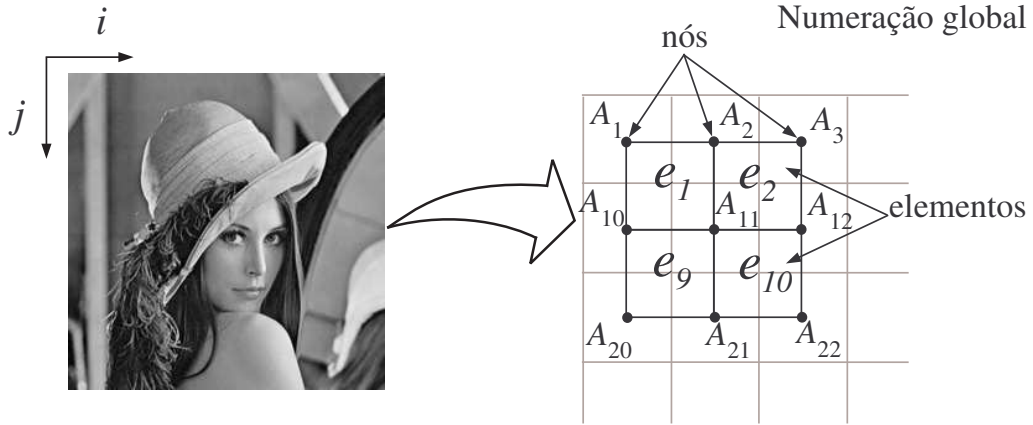


Figura A.1: Discretização de Elementos Finitos utilizada e numeração global.

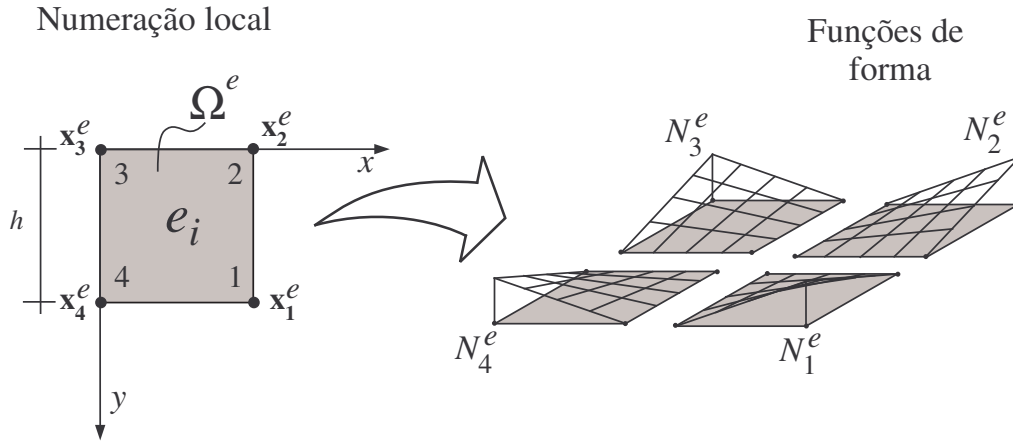


Figura A.2: Domínio do elemento quadrilátero bilinear, numeração local e funções de forma.

A.3.2 Família de elementos

Neste caso, o MEF será utilizado para construir as funções bases do espaço de aproximação \mathcal{U}^h . Por este motivo, será considerada como discretização natural os pixels presentes na imagem. Mesmo assim o super-índice h não será descartado para identificar o espaço \mathcal{U}^h . No caso, o domínio foi discretizado com elementos quadriláteros bilineares como apresentado na Figura A.1. Por simplicidade, os nós da malha foram centrados nos pixels. Desta maneira, cada grau de liberdade (u_B) está associado à intensidade de um pixel.

O domínio de um elemento quadrilátero é definido pela localização dos seus quatro nós \mathbf{x}_a^e , para $a = 1 \dots 4$. Assume-se que os nós estão, a nível do elemento e , ordenados (incidência do elemento) no sentido anti-horário (Fig. A.2). Foi escolhido um sistema de coordenadas consistente com o utilizado no contexto de processamento de imagens. Em

uma imagem, a origem (o ponto $(0, 0)$) encontra-se no canto superior esquerdo da mesma, sendo o eixo y associado a vertical e crescente para baixo (associado aos índices j dos pixels na imagem) e o eixo x no sentido horizontal crescente para a direita (associado aos índices i dos pixels na imagem). Tem-se então, que neste caso particular os elementos da malha encontram-se “alinhados” com os pixels da imagem. Assim, e como todos os elementos finitos tem as mesmas dimensões, $h \times h$, foi adotado a nível do elemento e o sistema de coordenadas indicado na Figura A.2.

A.3.3 Funções de forma e matrizes elementares

São apresentadas a seguir as funções de forma associadas ao elemento e (Fig. A.2), quais sejam

$$\begin{aligned} N_1^e(x, y) &= \frac{1}{h^2} xy \\ N_2^e(x, y) &= \frac{1}{h^2} x(h - y) \\ N_3^e(x, y) &= \frac{1}{h^2} (h - x)(h - y) \\ N_4^e(x, y) &= \frac{1}{h^2} (h - x)y, \text{ para } (x, y) \in \Omega^e. \end{aligned} \quad (\text{A.33})$$

Estas funções de forma satisfazem a seguinte condição:

$$N_b^e(\mathbf{x}_a) = \delta_{ab} \text{ para } a, b = 1 \dots 4. \quad (\text{A.34})$$

É fácil verificar que

$$\begin{aligned} \nabla N_1^e(x, y) &= \frac{1}{h^2} \begin{Bmatrix} y \\ x \end{Bmatrix} \\ \nabla N_2^e(x, y) &= \frac{1}{h^2} \begin{Bmatrix} h - y \\ -x \end{Bmatrix} \\ \nabla N_3^e(x, y) &= -\frac{1}{h^2} \begin{Bmatrix} h - y \\ h - x \end{Bmatrix} \\ \nabla N_4^e(x, y) &= \frac{1}{h^2} \begin{Bmatrix} -y \\ h - x \end{Bmatrix}, \text{ para } (x, y) \in \Omega^e. \end{aligned} \quad (\text{A.35})$$

Da linearidade do operador integral, é possível separar as integrais colocadas na Eq. (A.30) em integrais no domínio em cada elemento Ω^e . Para relacionar a numeração elementar e global dos nós, considera-se a função $IEN(e, a)$ [93] que, dado o número de elemento e e o número de nó $a = 1 \dots 4$, retorna o índice do nó global A correspondente.

Define-se então a matriz elementar \mathbf{K}^e como

$$\mathbf{K}^e = \begin{pmatrix} K_{11}^e & K_{12}^e & K_{13}^e & K_{14}^e \\ K_{21}^e & K_{22}^e & K_{23}^e & K_{24}^e \\ K_{31}^e & K_{32}^e & K_{33}^e & K_{34}^e \\ K_{41}^e & K_{42}^e & K_{43}^e & K_{44}^e \end{pmatrix} \quad (\text{A.36})$$

cujos coeficientes K_{ab}^e são dados por

$$\begin{aligned} K_{11}^e &= \int_{\Omega^e} \left(2\lambda \nabla N_1^e(x, y) \cdot \nabla N_1^e(x, y) + N_1^e(x, y) N_1^e(x, y) \right) d\Omega^e, \\ K_{12}^e &= \int_{\Omega^e} \left(2\lambda \nabla N_1^e(x, y) \cdot \nabla N_2^e(x, y) + N_1^e(x, y) N_2^e(x, y) \right) d\Omega^e, \\ K_{13}^e &= \int_{\Omega^e} \left(2\lambda \nabla N_1^e(x, y) \cdot \nabla N_3^e(x, y) + N_1^e(x, y) N_3^e(x, y) \right) d\Omega^e, \\ K_{14}^e &= \int_{\Omega^e} \left(2\lambda \nabla N_1^e(x, y) \cdot \nabla N_4^e(x, y) + N_1^e(x, y) N_4^e(x, y) \right) d\Omega^e. \end{aligned}$$

Como as funções N_a^e são conhecidas, estas integrais podem ser calculadas explicitamente, logo:

$$\begin{aligned} K_{11}^e &= \frac{1}{9} (h^2 + 12\lambda), \\ K_{12}^e &= \frac{1}{18} (h^2 - 6\lambda), \\ K_{13}^e &= \frac{1}{36} (h^2 - 24\lambda), \\ K_{14}^e &= \frac{1}{18} (h^2 - 6\lambda). \end{aligned}$$

Da simetria do problema tem-se que:

$$\begin{aligned} K_{11}^e &= K_{22}^e = K_{33}^e = K_{44}^e, \\ K_{12}^e &= K_{21}^e = K_{34}^e = K_{43}^e = K_{14}^e = K_{23}^e = K_{32}^e = K_{41}^e \text{ e} \\ K_{13}^e &= K_{24}^e = K_{31}^e = K_{42}^e. \end{aligned}$$

Da mesma forma são calculados os coeficientes do vetor \mathbf{f} . Nos problemas estudados foi considerado que os pixels tem tamanho unitário (i.e., $h = 1$ a nível dos pixels¹).

¹Se for necessário refinar a malha de elementos finitos, h varia.

Referências Bibliográficas

- [1] F. Alonso, M.E. Algorri, and F. Flores-Mangas. Composite index for the quantitative evaluation of image segmentation results. In *Proceedings of the 26th Annual International Conference of the IEEE EMBS*, pages 1794–1797, San Francisco, CA, USA, 2004.
- [2] L. Alvarez, F. Guichard, P.L. Lions, and J-M. Morel. Axioms and fundamental equations of image processing. Technical Report 9216, Ceremade, 1992.
- [3] L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel. Axiomatisation et nouveaux opérateurs de la morphologie mathématique. *Comptes Rendus de l'Académie des Sciences*, t. 315, Série I:265–268, 1992.
- [4] L. Alvarez, P.L. Lions, and J-M. Morel. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal on Numerical Analysis*, 29:845–866, 1992.
- [5] L. Ambrosio. A compactness theorem for a new class of functions of bounded variation. *Bolletino della Unione Matematica Italiana*, VII(4):857–881, 1989.
- [6] L. Ambrosio. Variational problems in sbv and image segmentation. *Acta Applicandae Mathematicae*, 17:1–40, 1989.
- [7] S. Amstutz. *Aspects théoriques et numériques en optimisation de forme topologique*. Phd thesis, Institut National des Sciences Appliquées de Toulouse, France, 2003.
- [8] S. Amstutz. The topological asymptotic for the navier-stokes equations. *ESAIM: Control, Optimisation and Calculus of Variations*, 11(3):401–425, 2005.
- [9] S. Amstutz. Topological sensitivity analysis for some nonlinear pde systems. *J. Math. Pures Appl.*, 85(4):540–557, 2006.
- [10] S. Amstutz and H. Andrä. A new algorithm for topology optimization using a level-set method. Technical report, Université Paul Sabatier, Toulouse - Cedex - France, 2005.
- [11] S. Amstutz and H. Andrä. A new algorithm for topology optimization using a level-set method. *J. Comput. Phys.*, 216(2):573–588, 2006.
- [12] S. Amstutz, I. Horchani, and M. Masmoudi. Crack detection by the topological gradient method. *Control and Cybernetics*, 34(1):119–138, 2005.
- [13] A.S. Anayiotos, S.A. Jones, D.P. Giddens, S. Glagov, and C.K. Zarins. Shear stress at a compliant model of the human carotid bifurcation. *J. Biomech. Engn.*, 117:98–106, 1994.
- [14] American Heart Association. American heart association, heart and stroke statistical update, 2007.
- [15] K. Atkinson and W. Han. *Elementary Numerical Analysis, 3rd Edition*. Wiley, 2003.

- [16] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing*, volume 147 of *Applied Mathematical Sciences*. Springer-Verlag, 2002.
- [17] G. Aubert and L. Vese. A variational method in image recovery. *SIAM Journal of Numerical Analysis*, 34(5):1948–1979, 1997.
- [18] D. Auroux, M. Masmoudi, and L. Belaid. Image restoration and classification by topological asymptotic expansion. *Variational Formulations in Mechanics: Theory and Applications - CIMNE, Barcelona, Spain 2006 (In press)*, 2006.
- [19] A. P. Avolio. Multi-branched model of the human arterial system. *Med. Biol. Eng. Comp.*, 18:709–718, 1980.
- [20] T. Baker. Automatic mesh generation for complex three dimensional regions using a constrained delaunay triangulation. *Engineering Computations*, 5:161–175, 1989.
- [21] C. Banga and F. Ghorbel. Optimal bootstrap sampling for fast image segmentation: application to retina images. In *IEEE ICASP 93*, volume 5, pages 638–641, Minneapolis USA, 1993.
- [22] L. J. Belaid, M. Jaoua, M. Masmoudi, and L. Siala. Application of the topological gradient to image restoration and edge detection. *To appear on Special Issue on "Shape and Topological Sensitivity Analysis: Theory and Application" of the Engineering Analysis with Boundary Element Journal*, 2007.
- [23] M. Berger and B. Gostiaux. *Differential Geometry: Manifolds, Curves and Surfaces*, volume 115 of *Graduate Texts In Mathematics*. Springer-Verlag, 1988.
- [24] J. C. Bezdek. *Pattern Recognition with fuzzy Objective Function Algorithms*. Plenum Press, New York, 1981.
- [25] B. K. Bharadvaj, R. F. Mabon, and D. P. Giddens. Steady flow in a model of the human carotid bifurcation-i. flow visualization. *J. of Biomechanics*, 15:349–362, 1982.
- [26] W.D. Bidgood. Clinical importance of the dicom structured reporting standard. *Int J Card Imaging*, 14(5):307–315, October 1998.
- [27] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger. Robust anisotropic diffusion. *IEEE Transactions on Image Processing*, 7(3):421–423, 1998.
- [28] P. Blanco, I. Larrabide, S. Urquiza, and R. Feijóo. Sensitivity of blood flow patterns to the constitutive law of the fluid. In *Proceedings of the ECCM 2006. LNEC - Lisbon, Portugal 5th-9th June*, 2006.
- [29] A. Bonnet. On the regularity of the edge set of mumford-shah minimizers. *Progress in Nonlinear Differential Equations*, 25:93–103, 1996.
- [30] R. Boscolo, M.S. Brown, and M.F. McNitt-Gray. Medical image segmentation with knowledge-guided robust active contours. *Radiographics*, 22(2):437–448, 2002.
- [31] B. Bourdin. Image segmentation with a finite element method. *M2AN - Mathematical Modeling Numer. Analysis*, 33(2):296–310, 1999.
- [32] H. Brezis. *Analyse Fonctionnelle - Théorie et Applications*. Masson, Paris - France, 1983.

- [33] M. Brühl, M. Hanke, and M.S. Vogelius. A direct impedance tomography algorithm for locating small inhomogeneities. *Numerische Mathematik*, 93:635–654, 2003.
- [34] Bugzilla. Bugzilla. <http://www.bugzilla.org/>.
- [35] M. Burger, B. Hackl, and W. Ring. Incorporating topological derivatives into level set methods. *Journal of Computational Physics*, 1(194):344–362, 2004.
- [36] A. P. Calderon. On an inverse boundary value problem. *Soc. Bras. de Math.*, pages 65–73, 1980.
- [37] J. F. Canny. Finding edges and lines in images. Technical Report AI-TR-720, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, 1983.
- [38] C. G. Caro, J. M. Fitz-Gerald, and R. C. Schroter. Atheroma and arterial wall shear dependent mass transfer mechanism for atherogenesis. *Proc. Royal Society of London, Biology*, 177:109–159, 1971.
- [39] P. Causin, J. F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. Technical Report 5084, Institut National de Recherche en Informatique et en Automatique - INRIA, 2004.
- [40] J. C  a, S. Garreau, P. Guillaume, and M. Masmoudi. The shape and topological optimizations connection. *Computer Methods in Applied Mechanics and Engineering*, 188(4):713–726, 2000.
- [41] J. R. Cebal and R. L  hner. From medical images to cfd meshes. In *Proceedings, 8th International Meshing Roundtable*, pages 321–331, South Lake Tahoe, CA, U.S.A., October 1999.
- [42] J. R. Cebal, P. J. Yim, R. L  hner, O. Soto, and P. L. Choyke. Blood flow modeling in carotid arteries with computational fluid dynamics and mr imaging. *Academic Radiology*, 9-11:1286–1299, 2002.
- [43] B. Delaunay. Sur la sphere vide. *Bull. Acad. Sci. USSR VII: Class. Scil, Mat. Nat.*, pages 793–800, 1934.
- [44] S. Deparis, M. A. Fernandez, and L. Formaggia. Acceleration of a fixed point algorithm for fluid-structure interaction using transpiration conditions. *Submitted to M2AN Math. Model. Numer. Anal*, 2005.
- [45] E. DiGiorgi, M. Carriero, and A. Leaci. Existence theorem for a maximum problem with a free discontinuity set. *Archive for Rational Mechanics and Analysis*, 108:195–218, 1989.
- [46] J. Donea and A. Huerta. *Finite Element Methods for Flow Problems*. John Wiley & Sons, Chichester, 2003.
- [47] dotProject. dotProject. <http://www.dotproject.net/>.
- [48] J. C. Dunn. A fuzzy relative of the isodata process and its use in detecting compact, well separated clusters. *Journal of Cybernetics*, 3:32–57, 1974.
- [49] V. Dutt and J. Greenleaf. Adaptive speckle reduction filter for log-compressed b-scan images. *IEEE Trans. Med. Imaging*, 15(6):802–813, 1996.
- [50] M. P. Bends  e. Optimal shape design as a material distribution problem. *Struct. Optim.*, 1:193–202, 1989.

- [51] M. P. Bendsøe and N. Kikuchi. Generating optimal topologies in structural design using an homogenization method. *Comput. Methods Appl. Mech. Engrg.*, 71:197–224, 1988.
- [52] H. A. Eschenauer, V. V. Kobelev, and A. Schumacher. Bubble method for topology and shape optimization of structures. *Structural Optimization*, 8:42–51, 1994.
- [53] H. A. Eschenauer and N. Olhoff. Topology optimization of continuum structures: A review. *Appl. Mech. Rev.*, 54:331–390, 2001.
- [54] J. D. Eshelby. The elastic energy-momentum tensor. *J. Elasticity* 5, pages 321–335, 1985.
- [55] C. Eveland, K. Konolige, and R. C. Bolles. Background modeling for segmentation of video-rate stereo sequences. In *International Conference on Computer Vision and Pattern Recognition*, pages 266–272, 1998.
- [56] T. Guinzani F., E. A. Fancello, and A. A. Novotny. An iterative reconstruction method based on the topological derivative for solving an inverse conductivity problem. In *Proceedings of the XXVI Iberian Latin-American Congress on Computational Methods in Engineering CILAMCE 2005 Brazilian Assoc. for Comp. Mechanics & Latin American Assoc. of Comp. Methods in Engineering*, Guarapari, Espírito Santo, Brazil, 19th to 21st October, 2005.
- [57] R. A. Feijóo, A. A. Novotny, E. Taroco, and C. Padra. The topological derivative for the poisson’s problem. *Mathematical Models and Methods in Applied Sciences*, 13-12:1–20, 2003.
- [58] R.A. Feijóo, A.A. Novotny, C. Padra, and E. Taroco. The topological-shape sensitivity method and its application in 2d elasticity. *Journal of Computational Methods in Sciences and Engineering*, 2004.
- [59] L. Formaggia, J. F. Gerbeau, F. Nobile, and A. Quarteroni. On the coupling of 3d and 1d navier-stokes equations for flow problems in compliant vessels. *Comp. Meth.App. Mech & Eng.*, 191:561–582, 2001.
- [60] L. Formaggia, J. F. Gerbeau, F. Nobile, and A. Quarteroni. Numerical treatment of defective boundary conditions for the navier-stokes equations. *SIAM J. Numer. Anal.*, 40(1):376–401, 2002.
- [61] L. Formaggia, D. Lamponi, and A. Quarteroni. One-dimensional models for blood flow in arteries. *Journal of Engineering Mathematics*, 47:251–276, 2003.
- [62] L. Formaggia, F. Nobile, J. F. Gerbeau, and A. Quarteroni. Numerical treatment of defective boundary conditions for the navierstokes equations. Technical Report 20, EPFL-DMA Analyse et Analyse Numerique Report, 2000.
- [63] L. Formaggia, F. Nobile, A. Quarteroni, and A. Veneziani. Multiscale modelling of the circulatory system: a preliminary analysis. *Comput. Visual Sci.*, 2:75–83, 1999.
- [64] A. S. Frangakis and R. Hegerl. Noise reduction in electron tomographic reconstructions using nonlinear anisotropic diffusion. *J. Struct. Biol.*, 1(135):239–250, 2001.
- [65] A. F. Frangi. *Three-Dimensional Model-Based Analysis of Vascular and Cardiac Images*. PhD thesis, Image Sciences Institute, University Medical Center Utrecht, 2001.

- [66] O. Frank. Die grundform des arteriellen pulses. *Z. Biol.*, 37:483–526, 1899.
- [67] P. Frey, B. Sarter, and M. Gautherie. Fully automatic mesh generation for 3d domains based upon voxel sets. *International Journal for Numerical Methods in Engineering.*, 37:2735–2753, 1994.
- [68] E. Gamma, R. Helm, and R. Johnson. *Design Patterns. Elements Of Reusable Object-Oriented Software*. Addison-Wesley Professional Computing Series. Addison-Wesley, 1995. Gam E 95:1 1.Ex.
- [69] S. Garreau, P. Guillaume, and M. Masmoudi. The topological gradient. Technical report, UFR MIG, Université Paul Sabatier, Toulouse - France, 1998.
- [70] S. Garreau, P. Guillaume, and M. Masmoudi. The topological asymptotic for pde systems: the elasticity case. *SIAM J. Control Optim.*, 39(6):1756–1778, 2001.
- [71] GCC. GCC, the GNU Compiler Collection. <http://gcc.gnu.org/>.
- [72] D.P. Giddens, C.K. Zarins, and S. Glagov. The role of fluid mechanics in the localization and detection of atherosclerosis. *J. Biomech. Engn.*, 115:588–594, 1993.
- [73] R. E. Gillilan and F. Wood. Visualization, virtual reality, and animation with the data flow model of computing. *Computer Graphics*, 29(2):55–58, May 1995.
- [74] V. Girault and P.-A. Raviart. *Finite Element Methods for Navier-Stokes Equations. Theory and Algorithms*. Springer-Verlag, Berlin, 1986.
- [75] C. R. Gonzalez and R. E. Woods. *Digital Image Processing - Second Edition*. Prentice Hall, 2001.
- [76] P. Gravel, G. Beaudoin, and J. A. De Guise. A method for modeling noise in medical images. *IEEE Transactions On Medical Imaging*, 23(10):1221–1232, 2004.
- [77] R. M. Gray and Y. Linde. Vector quantization and predictive quantizer for gauss-markov sources. *IEEE Transaction on Communications*, 30:381–389, 1982.
- [78] W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee. Using adaptive tracking to classify and monitor activities in a site. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 22–31, Santa Barbara - California, June 1998. IEEE Computer Society.
- [79] F. Guichard and J.-M. Morel. Image analysis and p.d.e’s, 2001.
- [80] P. Guillaume and K. Sid Idris. The topological asymptotic expansion for the dirichlet problem. *SIAM J. Control Optim.*, 41(4):1042–1072, 2002.
- [81] P. Guillaume and K. Sid Idris. Topological sensitivity and shape optimization for the stokes equations. *SIAM J. Control Optim.*, 43(1):1–31, 2004.
- [82] M. O. Gurtin. *An Introduction to Continuum Mechanics*. Academic Press, 1981.
- [83] R. B. Haber and D. A. McNabb. Visualization idioms: A conceptual model for scientific visualization systems. In *Visualization in Scientific Computing - IEEE Computer Society Press*, pages 74–93, 1990.
- [84] J. Hadamard. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton University Bulletin*, 1:49–52, 1902.

- [85] E. J. Haug and J. Céa. Optimization of distributed parameters structures. In *Iowa - USA*, 1981.
- [86] E. J. Haug, K. K. Choi, and V. Komkov. *Design Sensitivity Analysis of Structural Systems*. Academic Press, 1986.
- [87] L. He and S. Osher. Solving the chan-veese model by multiphase level set algorithm based on the topological derivative. Technical Report 56, UCLA - CAM, 2006.
- [88] K. Held, E. R. Kops, B. J. Krause, W. M. Wells 3rd, R. Kikinis, and H. W. Muller-Gartner. Markov random field segmentation of brain mr images. *IEEE Trans. Med. Imaging*, 16(6):878–886, 1997.
- [89] M. Hintermüller. Fast level set based algorithms usind shape and topological sensitivity. *Control and Cybernetics*, 34(1):305–324, 2005.
- [90] C. Hirsch. *Numerical Computation of Internal and External Flows - Volume 1: Fundamentals of Numerical Discretization*. John Wiley and Sons, 1989.
- [91] K. H. Hohne, H. Fuchs, and S. M. Pizer. *3D imaging in medicine: algorithms, systems, applications*. Springer-Verlag, New York, 1990.
- [92] P. J. Huber. *Robust Statistics*. Wiley - New York, 1981.
- [93] T. J. R. Hughes. *The Finite Element Method - Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, 2000.
- [94] T. J. R. Hughes and A. N. Brooks. A theoretical framework for petrov-galerkin methods with discontinuous weighting functions: Application to the streamline-upwind procedure. *Finite Elements in Fluids IV*, pages 46–65, 1982.
- [95] T. J. R. Hughes and J. Lubliner. On the one-dimensional theory of blood flow in larger vessels. *American Elseiver Publishing Company - Mathematical Biosciences*, 18:161–170, 1973.
- [96] L. Ibanez, W. Schroeder, L. Ng, and J. Cates. *The ITK Software Guide*. Kitware, Inc. ISBN 1-930934-15-7, <http://www.itk.org/ItkSoftwareGuide.pdf>, second edition, 2005.
- [97] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.
- [98] Kitware. ParaView - Kitware Inc. <http://www.paraview.org>.
- [99] Kitware. Vtk: The visualization toolkit - kitware inc. <http://www.vtk.org>.
- [100] I. Koch and G. Marshall. Bootstrap coverage plots for image segmentation. In *IEEE Proceedings of ICPR 1996*, pages 447–451, 1996.
- [101] J. J. Koenderink. The structure of images. *Biol. Cybern.*, 50:363–370, 1984.
- [102] R. Kohn and M. Vogelius. Relaxation of a variational method for impedance computed tomography. *Comm. Pure Appl. Math.*, 40(6):745–777, 1987.
- [103] P. Kornprobst, R. Deriche, and G. Aubert. Nonlinear operators in image restoration. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 325–331, Puerto-Rico, June 1997. IEEE.

- [104] K. Krissian, R. Kikinis, C-F. Westin, and K. Vosburgh. Speckle-constrained filtering of ultrasound images. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 547–552, Washington, DC, USA, 2005. IEEE Computer Society.
- [105] K. Krissian, K. Vosburgh, R. Kikinis, and C.-F. Westin. Speckle-constrained anisotropic diffusion for ultrasound images. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition'05)*, San Diego CA, USA, June 2005.
- [106] D. N. Ku, D. P. Giddens, C. K. Zarins, and S. Glagov. Pulsatile flow and atherosclerosis in the human carotid bifurcation. positive correlation between plaque location and low oscillating shear stress. *Atherosclerosis*, 5:293–302, 1985.
- [107] D. N. Lamponi. *One dimensional and multiscale models for blood flow circulation*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2004.
- [108] I. Larrabide, P.J. Blanco, S.A. Urquiza, and R.A. Feijóo. Sensitivity of blood flow in stenosed carotid bifurcation. In *Proceedings of ICCB2005 - II International Conference on Computational Bioengineering, Rodrigues et al. eds. ,Lisbon, Portugal*, 2005.
- [109] I. Larrabide, R. Feijóo, E. Taroco, and A. Novotny. Configurational derivative as a tool for image segmentation. In *Proceedings of the ECCM 2006. LNEC - Lisbon, Portugal 5th-9th June*. European Committee of Computational Solid and Structural Mechanics - ECCSM, 2006.
- [110] I. Larrabide, R. A. Feijóo, A. A. Novotny, E. Taroco, and M. Masmoudi. An image segmentation method based on a discrete version of the topological derivative. In *World Congress Structural and Multidisciplinary Optimization 6, Rio de Janeiro*. International Society for Structural and Multidisciplinary Optimization, 2005.
- [111] I. Larrabide, R. A. Feijóo, A. A. Novotny, and E. A. Taroco. Topological derivative: A tool for image processing. *Computers & Structures - An International Journal*. Editors: K. J. Bathe & B. H. V. Topping. Guest Editor for the Special Issue: C. A. Mota Soares, M. Bendsoe, K. K. Choi and J. Herskovits, 2006. Accepted for publication - December 2006.
- [112] I. Larrabide, S. Fiorentini, and M. J. Vénere. Voxel grow: A region growing segmentation technique. In *International Conference on Computer Science, Software Engineering, Information Technology, e-Business, and Applications (CSITeA03)*, 2003.
- [113] I. Larrabide, A. A. Novotny, R. A. Feijóo, and E. Taroco. A medical image enhancement algorithm based on topological derivative and anisotropic diffusion. In *Proceedings of the XXVI Iberian Latin-American Congress on Computational Methods in Engineering - CILAMCE 2005 - Guarapari, Espírito Santo, Brazil*, 2005.
- [114] I. Larrabide, A. A. Novotny, R. A. Feijóo, and E. Taroco. Topological derivative as a tool for image processing: Image segmentation. Technical Report 15/2006, LNCC - Laboratório Nacional de Computação Científica, 2006.

- [115] I. Larrabide, A. A. Novotny, M. Masmoudi, R. A. Feijóo, and E. A. Taroco. Topological derivative as a tool for image processing - part i: Image segmentation. *IEEE Transactions in Image Processing*, 2006. Submitted - December 2006.
- [116] I. Larrabide, A.A. Novotny, and R.A. Feijóo. A simple method for tomography reconstruction based on a discrete version of the topological gradient. In *Accepted for publication in Inverse Problems, Design and Optimization Symposium (IPDO-2007) - April 16-18 - 2007, Miami, Florida, USA*, 2007.
- [117] I. Larrabide and R.A.Feijóo. Hemolab - laboratório de modelagem em hemodinâmica. Technical Report 13/2006, LNCC, 2006.
- [118] I. Larrabide and R.A.Feijóo. Hemolab - hemodynamics modeling laboratory, an application for modelling the human cardiovascular system. In *Submitted to CGI 2007. May 30th - June 2nd 2007. Petrópolis - RJ - Brasil*, 2007.
- [119] T. Lewiński and J. Sokołowski. Topological derivative for nucleation of non-circular voids. Technical Report 3798, INRIA, Lorraine, 1999.
- [120] T. Lewiński and J. Sokołowski. Energy change due to the appearance of cavities in elastic solids. *Int. J. Solids Struct*, 40:1765–1803, 2003.
- [121] H. Li, R. Deklerck, B. DeCuyper, A. Hermanusa, E. Nyssen, and J. Cornelis. Object recognition in brain ct-scans: knowledge-based fusion of data from multiple feature extractors. *IEEE Trans. Med. Imaging*, 14:212–229, 1995.
- [122] W. Lorensen and H. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *In Proc. of ACM SIGGRAPH*, pages 163–169, 1987.
- [123] CMake Cross Platform Make. CMake - Cross Platform Make. <http://www.cmake.org/>.
- [124] R. Malladi and J. A. Sethian. Image processing: Flows under min/max curvature and mean curvature. *Graphical Models and Image Processing*, 58(2):127–141, 1996.
- [125] R. Malladi and J. A. Sethian. Level set methods for curvature flow, image enhancement, and shape recovery in medical images. In *Proc. of Conf. on Visualization and Mathematics*, pages 329–345, Germany, 1997. Springer-Verlag.
- [126] H. A. Mang, F.G. Rammerstorfer, and J. Eberhardsteiner, editors. *Fifth World Congress on Computational Mechanics - <http://wccm.tuwien.ac.at>*, Viena - Austria, 2002.
- [127] D. Marr and E. Hildreth. Theory of edge detection. In *Proceedings of the Royal Society London*, volume 207, pages 187–217, 1980.
- [128] N. D. A. Mascarenhas, L. F. Costa, and F. N. S. Medeiros. Speckle noise filtering in sar images by map approach. Technical report, UFSCar - DC, 1999.
- [129] M. Masmoudi. The topological asymptotic. *Computational Methods for Control Applications, Ed. H.Kawarada and J.Périaux, International Series GAKUTO*, 2002.
- [130] M. Masmoudi, J. Pommier, and B. Samet. The topological asymptotic expansion for the maxwell equations and some applications. *Inverse Problems*, 21:547–564, 2005.

- [131] O. V. Michailovich and A. Tannenbaum. Despeckling of medical ultrasound images. *IEEE Transactions On Ultrasonics, Ferroelectrics, And Frequency Control*, 53(1):64–78, 2006.
- [132] D. Mumford and J. Shah. Boundary detection by minimizing functionals. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 22–26, San Francisco, CA, June 1985. IEEE.
- [133] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42:577–684, 1989.
- [134] F. Murat and J. Simon. *Sur le Contrôle par un Domaine Géométrique*. PhD thesis, Université Pierre et Marie Curie, Paris - France, 1976.
- [135] T.R. Rogge N. Stergiopulos, D.F. Young. Computer simulation of arterial flow whit applications to arterial and aortic stenoses. *J. of Biomechanics*, 25:1477–1488, 1992.
- [136] S. A. Nazarov and J. Sokołowski. The topological derivative of the dirichlet integral due to formation of a thin ligament. *Siberian Math. J.*, 45(2):341–355, 2004.
- [137] S. A. Nazarov and J. Sokołowski. Singular perturbations in shape optimization for the dirichlet laplacian. *C. R. Mecanique*, 333:305–310, 2005.
- [138] S. A. Nazarov and J. Sokołowski. Self-adjoint extensions for the neumann laplacian and applications. *To appear at Acta Mathematicae Applicatae Sinica*, 22(3):879–906, 2006.
- [139] W. W. Nichols and M. F. O’Rourke. *McDonalds’s Blood Flow in Arteies - Theoretical, experimental and clinical principles*. Arnold, 1998.
- [140] A. A. Novotny. *Análise de Sensibilidade Topológica*. Phd thesis, Laboratorio Nacional de Computação Científica, Petrópolis - RJ - Brazil, 2003.
- [141] A. A. Novotny, R. A. Feijóo, C. Padra, and E. Taroco. Topological sensitivity analysis applied to topology design of kirchhoff’s plate bending problem. *Control and Cybernetics*, 34(1):339–361, 2005.
- [142] A. A. Novotny, R. A. Feijóo, E. Taroco, and C. Padra. Topological-shape sensitivity method: Theory and applications. In *IUTAM Symposium on Topological Design Optimization*, Denmark, 2005.
- [143] A. A. Novotny, R. A. Feijóo, C. Padra, and E. Taroco. Derivada topológica via análise de sensibilidade á mudança de forma na otimização topológica. *Revista Internacional de Métodos Numéricos para o Cálculo y Diseño en Ingenieria*, 18(4):499–519, 2002.
- [144] A. A. Novotny, R. A. Feijóo, C. Padra, and E. Taroco. Topological optimization via shape sensitivity analysis applied in 2d elasticity. In *Mang et al. [126]*, 2002.
- [145] A. A. Novotny, R. A. Feijóo, E. Taroco, M. Masmoudi, and C. Padra. Topological sensitivity analysis for a nonlinear case: the p-poisson problem. In *6th World Congress on Structural and Multidisciplinary Optimization*, Rio de Janeiro - Brazil, 2005.

- [146] A. A. Novotny, R. A. Feijóo, E. Taroco, and C. Padra. Topological sensitivity analysis. *Computer Methods in Applied Mechanics and Engineering*, 192:803–829, 2003.
- [147] A. A. Novotny, R. A. Feijóo, E. Taroco, and C. Padra. Topological sensitivity analysis for three-dimensional linear elasticity problem. In *6th World Congress on Structural and Multidisciplinary Optimization*, Rio de Janeiro - Brazil, 2005.
- [148] USC SIPI (University of Southern California Signal & Image Processing Institute). Imagens obtidas de <http://sipi.usc.edu/database/>.
- [149] M. S. Olufsen. *Modeling the Arterial System with reference to an Anesthesia Simulator*. PhD thesis, Roskilde Universitetscenter-IMFUSA, 1998.
- [150] M. S. Olufsen. Structured tree outflow condition for blood flow in large systemic arteries. *Am. J. Physiol.*, 276:257–268, 1999.
- [151] M. M. Onana, G. Mercier, V. P. Onana, J.P. Rudant, E. Tonye, and H. Trebossen. A combined speckle noise reduction and compression of sar images using a multiwavelet based method to improve codec performance. In *IGARSS '01*, 2001.
- [152] OpenGL. OpenGL. <http://opengl.org>.
- [153] WHO World Health Organization. <http://www.who.int/whosis/whostat2006/en/index.html>. WHO - World Health Organization, 2002.
- [154] J. Sokołowski. Optimality conditions for simultaneous topology and shape optimization. *SIAM J. Control Optim.*, 42(4):1198–1221, 2003.
- [155] J. Sokołowski and A. Żochowski. On the topological derivative in shape optimization. *SIAM Journal on Control and Optimization*, 37(4):1251–1272, 1999.
- [156] J. Sokołowski and A. Żochowski. Modelling of topological derivatives for contact problems. *Numerische Mathematik*, 102:14–179, 2005.
- [157] J. Sokołowski and A. Żochowski. Topological derivatives for obstacle problems. In *IFIP Proceedings*, 2005.
- [158] J. Sokołowski and J. P. Zolésio. *Introduction to Shape Optimization - Shape Sensitivity Analysis*. Springer-Verlag, 1992.
- [159] T. N. Pappas. An adaptive clustering algorithm for image segmentation. *IEEE Transactions on Signal Processing*, 40:901–914, April 1992.
- [160] N. Paragios and R. Deriche. A pde-based level set approach for detection and tracking of moving objects. In *Proceedings of the 6th International Conference on Computer Vision*, pages 1139–1145, Bombay, India, January 1998. IEEE Computer Society, IEEE Computer Society Press.
- [161] J. Peraire, J. Peiró, L. Formaggia, K. Morgan, and O. Zienkiewicz. Finite element euler computations in three dimensions. *International Journal for Numerical Methods in Engineering*, 26:2135–2159, 1988.
- [162] J. Peraire, J. Peiró, and K. Morgan. Adaptive remeshing for three-dimensional compressible flow computations. *Journal of Comp. Physics*, 103:449–466, 1991.

- [163] K. Perktold and G. Rappitsch. *Mathematical modeling of local arterial flow and vessel mechanics*. Pitman, 1994.
- [164] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(7):629–639, 1990.
- [165] K. Pertold and G. Rappitsch. Computer simulation of local blood flow and vessel mechanics in a compliant carotid artery bifurcation model. *Journal of Biomechanics*, 28:845–856, 1995.
- [166] O. Pironneau. *Optimal Shape Design for Elliptic Systems*. Springer-Verlag, 1984.
- [167] O. Pironneau. *Finite Element Methods for Fluids*. John Wiley & Sons, Chichester, 1989.
- [168] G. Pólya and G. Szegő. *Isoperimetric Inequalities in Mathematical Physics*. Annals of Mathematical Studies Number 27 - Princeton University Press, 1951.
- [169] J. M. S. Prewitt. *Picture Processing and Psychopictorics*, pages 75–149. Academic Press, New York, 1970.
- [170] A. Quarteroni. *Modeling the cardiovascular system: a mathematical challenge - Mathematics Unlimited - 2001 and Beyond*, B. Engquist and W. Schmid Eds., pages 961–972. Springer-Verlag, 2001.
- [171] A. Quarteroni, M. Tuveri, and A. Veneziani. Computational vascular fluid dynamics: problems, models and methods. *Comp. Vis. Science*, 2:163–197, 2000.
- [172] R. Malladi and J.A. Sethian. Level set methods for curvature flow, image enhancement, and shape recovery in medical images. In *Proc. of Conf. on Visualization and Mathematics*, pages 329–345, Germany, 1997. Springer-Verlag.
- [173] L. G. Roberts. *Optical and Electrooptical Information processing*, chapter Machine perception of three-dimensional solids, pages 159–197. MIT Press, 1965.
- [174] B. Samet, S. Amstutz, and M. Masmoudi. The topological asymptotic for the helmholtz equation. *SIAM Journal on Control and Optimization*, 42(5):1523–1544, 2003.
- [175] R. Scateni, editor. *Scientific Visualization '95: proceedings of the International Symposium, Cagliari, Italy, 27–29 September, 1995*, Singapore; Philadelphia, PA, USA; River Edge, NJ, USA, 1995. World Scientific Publishing Co.
- [176] W. J. Schroeder, L. S. Avila, and W. Hoffman. Visualizing with vtk: A tutorial. *IEEE Comput. Graph. Appl.*, 20(5):20–27, 2000.
- [177] A. Schumacher. *Topologieoptimierung von Bauteilstrukturen unter Verwendung von Lochpositionierungskriterien*. PhD thesis, Universität-Gesamthochschule-Siegen, 1995.
- [178] Eclipse SDK. Eclipse. <http://www.eclipse.org/>.
- [179] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999.
- [180] M. Shephard and M. Georges. Automatic three-dimensional mesh generation by the finite octree technique. *International Journal for Numerical Methods in Engineering*, 32:709–749, 1991.

- [181] I. Sobel. *Machine Vision for Three-Dimensional Scenes*, pages 376–379. H. Freeman editor, 1990.
- [182] J. Sokolowski and A. Żochowski. Topological derivatives for elliptic problems. *Inverse Problems*, 15:123–134, 1999.
- [183] J. C. Stettler, P. Niederer, and M. Anliker. Theoretical analysis of arterial hemodynamics including the influence of bifurcations. In *Annals of Biomedical Engineering*, 1980.
- [184] A. Stevens and C. Walnum. *Standard C++ Bible*. Hungry Minds, 2000.
- [185] Subclipse. Subclipse. <http://subclipse.tigris.org/>.
- [186] Subversion. Subversion. <http://subversion.tigris.org/>.
- [187] J. Suri, S. Singh, and K. Setarehdan, editors. *Advanced algorithmic approaches to medical image segmentation: state-of-the-art applications in cardiology, neurology, mammography and pathology*. Springer-Verlag, 2001.
- [188] R. Temam. *Navier-Stokes Equations. Theory and Numerical Analysis*. North Holland, New York, 1979.
- [189] C/C++ Development Tooling CDT. Eclipse CDT. <http://www.eclipse.org/cdt/>.
- [190] Inc. The MathWorks. *Using MATLAB*. The MathWorks, Inc., 1999.
- [191] Ubuntu. Ubuntu: Linux for human beings. <http://www.ubuntu.com>.
- [192] S. Urquiza, P. J. Blanco, G. Lombera, M. J. Vénere, and R. A. Feijóo. Blood flow finite element solution through the carotid artery. In *Mecánica Computacional*, volume 22, Bahia Blanca - Argentina, 2003.
- [193] S. A. Urquiza, P. J. Blanco, M. J. Vénere, and R. A. Feijóo. Multidimensional modelling for carotid artery blood flow. *Computer Methods in Applied Mechanics and Engineering*, 195(33-36):4002–4017, July 2006.
- [194] S. A. Urquiza and M. J. Vénere. An application framework architecture for fem and other related solvers. In S.R.Idelsohn, V.E.Sonzogni, and A.Cardona, editors, *Mecánica Computacional*, volume 22, pages 3099–3109, Santa Fé - Paraná, Argentina, October 2002.
- [195] M. J. Vénere. Procedimientos para la generación de mallas tridimensionales de elementos finitos. *Revista internacional de Métodos numéricos para cálculo y diseño en ingeniería*, 12:3–16, 1996.
- [196] M. J. Vénere. Optimización de la calidad de mallas de elementos finitos mediante cambios localizados de topología. *Rev. Int. de Mét. Num. para Cálculo y Diseño en Ing.*, 13:3–13, 1997.
- [197] M. J. Vénere. *Técnicas adaptativas para el Método de Elementos Finitos en dos y tres dimensiones*. PhD thesis, Instituto Balseiro, Universidad Nacional de Cuyo, Noviembre 1996.
- [198] M. J. Vénere and E. A. Dari. Análisis comparativo de algoritmos para obtener triangulaciones delaunay. *Mecánica Computacional*, 10:491–506, 1990.

- [199] A. Veneziani and C. Vergara. Flow rate defective boundary conditions in haemodynamics simulations. *Int. J. Numer. Meth. Fluids*, 0:1–6, 2000.
- [200] J. Wan, B. Steele, S. A. Spicer, S. Strohband, G. R. Feijoo, T. J. Hughes, and C. A. Taylor. A one-dimensional finite element method for simulation-based medical planning for cardiovascular disease. *Comput. Methods Biomech. Biomed. Engin.*, 5(3):195–206, 2002.
- [201] 2004 WHO World Health Organization. The world health report 2004 - changing history, "annex table 2: Deaths by cause, sex and mortality stratum in who regions, estimates for 2002". WHO - World Health Organization, 2004.
- [202] A. P. Witkin. Scale-space filtering. In *Proc. Int. Joint Conf. Artif. Intell.*, pages 1019–1021, 1983.
- [203] A. Wrangsjö. A bayesian framework for image denoising. Lic. Thesis LiU-Tek-Lic-2004:38, Linköping University, Sweden, April 2004. Thesis No. 1109, ISBN 91-85295-07-8.
- [204] C. R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfunder: real time tracking of human body. *IEEE Tran. on Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.
- [205] C. Xu, D. L. Pham, and J. L. Prince. *Medical Image Segmentation Using Deformable Models*, pages 129–174. SPIE Handbook on Medical Imaging - Volume III: Medical Image Analysis, 2000.
- [206] C. Xu, D. L. Pham, M. E. Rettmann, D. N. Yu, and J. L. Prince. Reconstruction of the human cerebral cortex from magnetic resonance images. *IEEE Transactions on Medical Imaging*, 18(6):467–480, 1999.
- [207] M. Yerry and M. Shephard. Automatic three dimensional mesh generation by the modified octree technique. *International Journal for Numerical Methods in Engineering*, 20:1965–1990, 1984.
- [208] Y. Zhang, M. Brady, and S. Smith. Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1):45–57, 2001.
- [209] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method: Volume 1- The Basis*, volume 1. McGraw-Hill Book Company, fourth edition, 2000.
- [210] P. Zijdenbos, B. M. Dawant, R. A. Margolin, and A. C. Palmer. Morphometric analysis of white matter lesions in mr images: Method and validation. *IEEE Trans. Med. Imag.*, 13(4):716–724, December 1994.
- [211] J. P. Zolésio. *The material derivative (or speed) method for shape optimization*, in: *Optimization of Distributed Parameter Structures*, volume II(50), pages 1089–1151. NATO Adv. Study Inst. Ser. E, Appl. Sci., 1981.



Ignacio Larrabide nasceu em Argentina no ano 1978. Viveu em Tandil, Provincia de Buenos Aires até o ano 2002 quando se formou em Engenharia de Software na Universidad Nacional del Centro. Em 2003 mudou-se para Petrópolis - RJ - Brasil onde desenvolveu este trabalho dentro do Programa de Pós - Graduação em Modelagem Computacional do Laboratório Nacional de Computação Científica - LNCC/MCT. Seus interesses de pesquisa incluem diversos aspectos de processamento de imagens baseados em modelos, princípios variacionais, métodos numéricos, processamento de imagens médicas e desenvolvimento de software.

Este trabalho tem por objetivo estudar a restauração e segmentação de imagens médicas e sua aplicação na Modelagem do Sistema Cardiovascular Humano. O aumento da complexidade na ciência tem motivado o intercâmbio de conhecimento e técnicas entre diferentes disciplinas.

Neste sentido a Derivada Topológica, originariamente concebida para tratar de maneira adequada problemas de otimização de topologia e problemas inversos, é utilizada no processamento de imagens. No caso de restauração, são estudadas duas alternativas baseadas na equação de difusão de calor, sendo uma abordagem estacionária e outra evolutiva. Para estudar a performance dos respectivos algoritmos propostos diversas imagens são restauradas e os resultados são quantitativamente comparados com resultados obtidos empregando outros algoritmos amplamente encontrados na literatura. Também é discutido neste trabalho o problema de segmentação que consiste em identificar objetos e regiões em uma imagem. Este problema também é abordado utilizando a derivada topológica, sendo apresentadas duas técnicas inovadoras. A primeira baseada na derivada topológica contínua e a outra em simplificações desta, chegando-se em uma versão discreta, de menor custo computacional. A performance dos respectivos algoritmos é testada segmentando diferentes imagens e comparando os resultados com os de outros métodos usualmente utilizados na literatura. Em ambas aplicações (restauração e segmentação), é realizada uma análise da influência dos parâmetros associados a cada algoritmo nos resultados do processamento.

Por último, é estudado o uso de técnicas de modelagem na simulação computacional do Sistema Cardiovascular Humano (SCVH). O objetivo deste estudo é a integração das técnicas de processamento de imagens propostas com os modelos necessários na simulação computacional do SCVH. Cabe ressaltar que, na atualidade, modelos multidimensionais que integram representações das diversas partes do sistema cardiovascular com diferente nível de detalhe, são amplamente utilizados. Para criar estes tipos de modelos, informações de diversas origens e variada natureza devem ser tratadas, requerendo-se o desenvolvimento de ferramentas computacionais que permitam operar com grandes volumes de dados. No entanto, não existe na atualidade uma ferramenta deste tipo. Sendo assim, e com o intuito de suprir esta deficiência, foi desenvolvido um sistema chamado HeMoLab - Laboratório de Modelagem em Hemodinâmica, o qual é descrito neste trabalho. Por último, são apresentados alguns resultados obtidos com este sistema computacional.

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)