**Automatização de oráculos de teste para imagens médicas de
modelos tridimensionais**

Misael Costa Júnior

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências
de Computação e Matemática Computacional (PPG-CCMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Misael Costa Júnior

Automatização de oráculos de teste para imagens médicas de modelos tridimensionais

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Márcio Eduardo Delamaro

USP – São Carlos
Abril de 2018

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

C837a Costa Júnior, Misael
Automatização de oráculos de teste para imagens
médicas de modelos tridimensionais / Misael Costa
Júnior; orientador Márcio Eduardo Delamaro. -- São
Carlos, 2018.
154 p.

Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática
Computacional) -- Instituto de Ciências Matemáticas
e de Computação, Universidade de São Paulo, 2018.

1. Teste de software. 2. Oráculos de teste. 3.
Angiografia. 4. Redes vasculares sintéticas
tridimensionais. I. Eduardo Delamaro, Márcio ,
orient. II. Título.

Misael Costa Júnior

**Test oracle automation for medical images of
three-dimensional models**

Master dissertation submitted to the Institute of
Mathematics and Computer Sciences – ICMC-USP,
in partial fulfillment of the requirements for the
degree of the Master Program in Computer Science
and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Prof. Dr. Márcio Eduardo Delamaro

USP – São Carlos
April 2018

*Dedico este trabalho à minha querida família, por todo incentivo e apoio durante essa
desafiadora caminhada.*

AGRADECIMENTOS

A Deus, por me oferecer oportunidades de amadurecimento e me proteger durante essa trajetória. Hoje entendo o significado de muitas das renúncias às quais tive que fazer e o agradeco imensamente por essa oportunidade.

Aos meus pais, Luiza Maria e Misael Costa, por todo carinho e conselhos oferecidos durante essa jornada. Agradeço-os também pelo sacrifício, dedicação, abdicação de tempo e de muitos projetos pessoais para que eu tivesse a oportunidade de estudar e ter uma boa formação profissional e pessoal. A conclusão deste trabalho não teria sido possível sem o apoio incondicional desde o momento em que decidi enfrentar esse grande desafio. Às minhas irmãs, Dayanne Lises e Danielly Tamires, que, por mais difícil que fossem as circunstâncias, sempre tiveram paciência e compreensão. Em muitas ocasiões a busca de conselhos a vocês foi a minha única fonte de motivação.

Ao meu orientador, Prof. Dr. Márcio Delamaro (ICMC-USP), pela oportunidade e confiança oferecidas a mim durante essa jornada. Agradeço-o também pelos conselhos, ajuda e contribuições para a conclusão deste trabalho. Além disso, seu exemplo profissional, paciência e ética fizeram dele fonte de admiração para mim. Ao Prof. Dr. Rafael Oliveira (UTFPR), pela amizade e companhia durante esse período. Seus conselhos, amizade e contribuições foram essenciais para o meu amadurecimento pessoal e profissional. À Prof^a. Dr^a. Fátima Nunes (EACH-USP) pela ajuda e contribuição durante o desenvolvimento deste trabalho.

Aos amigos do Laboratório de Engenharia de Software do ICMC (LabES-ICMC), alunos e professores, pelo companheirismo nos momentos de convivência no laboratório. Aos amigos que conquistei dos demais laboratórios de pesquisa e aos que pude conhecer em São Carlos. A convivência com todos vocês me ofereceu momentos de muita alegria e, em algumas ocasiões, serviram de grande motivação para que eu pudesse continuar confiante nessa jornada.

Aos professores e amigos da Universidade Estadual do Piauí (UESPI), que me ofereceram oportunidades e me fizeram acreditar no meu potencial. Em especial, ao Prof. Dr. Bringel Filho, exemplo de profissional e amigo, ao qual dispenso imensa estima e admiração.

Aos amigos Joelson Santos, Jonathan Ramos, José Filomen e Lucas Daniel, pela amizade construída durante esse período e pelos momentos tristes e felizes que passamos juntos. A amizade de vocês foi fundamental e uma das grandes conquistas que tive.

Por fim, à FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo), pelo financiamento do projeto por meio do processo de número 2015/11844-0.

“Em minhas preces de todo dia, sempre peço coragem e paciência. Coragem para continuar superando as dificuldades do caminho naqueles que não me compreendem. E paciência, para não me entregar ao desânimo diante das minhas fraquezas!”
(Chico Xavier)

RESUMO

JÚNIOR, M. C. **Automatização de oráculos de teste para imagens médicas de modelos tridimensionais**. 2018. 154 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

As atividades de teste automatizado contribuem significativamente para a redução de custos e a produtividade do projeto durante o processo de desenvolvimento de software. O teste automatizado reduz o esforço humano e aumenta a qualidade do produto final. No entanto, a alta complexidade da saída produzida pelos SUT (do inglês, *System Under Test*) contemporâneos limita, em alguns casos, a aplicação de estratégias de teste automatizadas. Sistemas com saídas gráficas/áudio, imagens tridimensionais, interfaces gráficas com o usuário e alguns aplicativos da Web são exemplos de sistemas com saídas complexas. A falta de oráculos automatizados de teste leva à aplicação de testes executados manualmente pelo próprio testador (oráculo humano), de modo informal, *ad-hoc* e improdutivo. Uma possível contribuição para aliviar esforços do testador é a implementação de oráculos de teste baseados na extração de características das saídas do SUT. Resultados de trabalhos anteriores mostram que tal abordagem contribui para o aumento da produtividade do teste, mitigando e complementando os esforços manuais. Este trabalho de mestrado propõe e avalia oráculos de teste automatizados para sistemas cujas saídas consistem em imagens médicas tridimensionais. Para tanto, é explorado o *framework* O-FIm/CO (do inglês, *Oracle for Images and Complex Outputs*) que utiliza conceitos de Recuperação de Imagens Baseada em Conteúdo (do inglês, *Content-Based Image Retrieval* – CBIR) como uma forma de automatizar oráculos de teste. Além de adaptações e extensões do *framework*, desenvolveram-se *plug-ins* que representam extratores de características de imagens médicas tridimensionais de vasos sanguíneos. Para evidenciar a eficácia da abordagem, foram conduzidos dois estudos experimentais objetivando avaliar a eficácia e a precisão dos oráculos de teste baseados em características na avaliação desse tipo de imagem. Além disso, realizou-se um estudo experimental comparando os oráculos de teste baseados em características e oráculos humanos. Os resultados evidenciam a eficácia da abordagem como uma estratégia promissora para automatizar atividades de teste, contribuindo para a redução de tempo e esforços gerados por abordagens manuais durante a avaliação da qualidade de sistemas geradores de imagens médicas tridimensionais.

Palavras-chave: Teste de software, Oráculos de teste, Angiografia, Redes vasculares sintéticas tridimensionais.

ABSTRACT

JÚNIOR, M. C. **Test oracle automation for medical images of three-dimensional models.** 2018. 154 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

Automated testing activities significantly contribute to cost reduction and project productivity during the software development process. Automated testing reduces human effort and increases the quality of the final product. However, the high complexity of contemporary SUT outputs limits, in some cases, the application of automated test strategies. Systems with graphical/audio outputs, three-dimensional imaging, graphical user interfaces and some web applications are examples of systems with complex outputs. The lack of automated test oracles leads to the application of manual tests performed by the tester (human oracle) in an informal, ad-hoc and unproductive manner. One possible contribution to alleviate the tester's efforts is to implement test oracles based on the extraction of characteristics from the SUT outputs. Results from previous work show that such an approach contributes to increase test productivity, mitigating and complementing manual efforts. This dissertation proposes and evaluates automated test oracles for systems whose outputs consist of three-dimensional medical images. To do so, the framework O-FIm/CO (Oracle for Images and Complex Outputs) that uses CBIR (Content-Based Image Retrieval) concepts is explored as a strategy to automate test oracles. In addition to adaptations and extensions of the framework, plug-ins were developed which represent extractors of three-dimensional medical image characteristics of blood vessels. To demonstrate the efficacy of the approach, two experimental studies were conducted to evaluate the efficacy and accuracy of feature-based test oracles in the evaluation of such images. Moreover, an experimental study was conducted to compare feature-based test oracles with human oracles. The results show the efficacy of the approach as a promising strategy to automate testing activities, contributing to the reduction of time and effort generated by manual approaches during the quality assessment of three-dimensional medical imaging systems.

Keywords: Software Testing, Test Oracles, Angiography, Three-Dimensional Synthetic Vascular Networks.

LISTA DE ILUSTRAÇÕES

Figura 1 – Atividades de teste de software	34
Figura 2 – Exemplo do GFC da função main do programa <i>Identifier</i>	37
Figura 3 – Fluxo genérico de teste com oráculos	40
Figura 4 – Cubo 3D usando o conceito de <i>voxels</i> e pontos estruturados.	45
Figura 5 – Vasos sintéticos tridimensionais – antes e depois da segmentação	46
Figura 6 – String de busca do MS.	52
Figura 7 – Fluxo de trabalho para identificação de estudos do MS.	54
Figura 8 – Análise de publicações por ano entre 1990 e 2016.	55
Figura 9 – Estudos práticos vs estudos teóricos.	58
Figura 10 – Programas <i>toy</i> versus programas reais.	59
Figura 11 – Programas <i>toy</i> versus programas reais: análise ano por ano.	59
Figura 12 – <i>Bar chart</i> de abordagens de validação de sistemas biomédicos.	62
Figura 13 – Taxonomia de abordagens de validação de sistemas biomédicos.	62
Figura 14 – Origem de abordagens de validação em sistemas biomédicos: academia, indústria e colaboração.	63
Figura 15 – Academia vs Indústria: análise ano por ano.	64
Figura 16 – Características relacionadas à análise de qualidade e teste de software em sistemas biomédicos.	65
Figura 17 – <i>Map chart</i> de países com impacto na pesquisa.	66
Figura 18 – Publicações de abordagens de validação em sistemas biomédicos: periódicos vs conferência.	67
Figura 19 – Estrutura genérica de um sistema CBIR.	73
Figura 20 – Abordagem oráculos de teste baseado em características.	74
Figura 21 – Estrutura genérica do O-FIm/CO.	77
Figura 22 – Exemplo de um oráculo gráfico para o <i>framework</i> O-FIm/CO.	78
Figura 23 – Exemplo de imagem de vaso sanguíneo após a segmentação (a) e após a esqueletização (b).	82
Figura 24 – Exemplo de imagens com o respectivo valor de densidade.	83
Figura 25 – Exemplo de imagens com o respectivo número de bifurcações.	83
Figura 26 – Exemplo de imagens com o respectivo número de pontos terminais.	84
Figura 27 – Exemplo de imagens com o respectivo número de segmentos.	84
Figura 28 – Exemplo de imagens com o respectivo valor do tamanho médio dos segmentos.	85

Figura 29 – Exemplo de características extraídas por meio dos extratores desenvolvidos – Estudo 1.	89
Figura 30 – Imagens 2D - Comparação da média da precisão e revocação para o conjunto de valores de limiares.	91
Figura 31 – Imagens 3D - Comparação da média da precisão e revocação para o conjunto de valores de limiares.	91
Figura 32 – Etapas para condução do estudo 1.	95
Figura 33 – Descritor de oráculo do estudo 1.	96
Figura 34 – Fluxo de atividades para categorização das imagens.	97
Figura 35 – Exemplo de características obtidas por meio dos extratores desenvolvidos – Estudo 2.	104
Figura 36 – Exemplo de uma imagem de vaso sanguíneo com aneurisma após aplicação de seis diferentes níveis de ruído gaussiano.	105
Figura 37 – Média da precisão e revocação com variação do limiar – Estudo 2.	107
Figura 38 – Exemplo de um gráfico de Precisão <i>versus</i> Revocação	108
Figura 39 – Etapas para condução do estudo 2.	110
Figura 40 – Descritor de oráculo do estudo 2.	110
Figura 41 – Fluxo de atividades para avaliação das imagens.	111
Figura 42 – Gráfico PxR com o segundo conjunto de características – Limiar = th_2	114
Figura 43 – <i>Screenshots</i> das etapas de preenchimento do formulário.	119
Figura 44 – Gráfico de pizza com áreas de pesquisa dos oráculos humanos.	120
Figura 45 – Gráfico de pizza com o tempo de atuação na área de pesquisa indicada.	121
Figura 46 – Etapas para condução do estudo 3.	123
Figura 47 – Planilha de análise e registro dos dados do estudo.	123
Figura 48 – Matriz de correção de características com coeficientes de correlação	154

LISTA DE CÓDIGOS-FONTE

Código-fonte 1	– Função main do programa <i>Identifier</i> .	36
Código-fonte 2	– Classe abstrata para <i>plug-ins</i> do O-FIm/CO	80
Código-fonte 3	– Interface Java para função de similaridade do O-FIm/CO	80
Código-fonte 4	– Código-fonte do extrator de quantidade de bifurcações	139
Código-fonte 5	– Código-fonte do extrator de quantidade de pontos terminais	141
Código-fonte 6	– Código-fonte do extrator de quantidade de segmentos	143
Código-fonte 7	– Código-fonte do extrator de tamanho médio de segmentos	147

LISTA DE TABELAS

Tabela 1 – Teste manual vs Teste automatizado	41
Tabela 2 – Comparação entre teste manual e automatizado.	42
Tabela 3 – Bibliotecas de pesquisa utilizadas no MS.	52
Tabela 4 – Validação <i>ad-hoc</i> e não-sistemática em sistemas biomédicos: abordagens, ferramentas e modelos.	56
Tabela 5 – Validação sistemática e automatizada em sistemas biomédicos: ferramentas, software e <i>frameworks</i>	57
Tabela 6 – Número de domínio de sistema por categoria.	60
Tabela 7 – Categorias de abordagens de análise de qualidade e teste de software.	61
Tabela 8 – Análise da divulgação de estudos por país.	66
Tabela 9 – Número de estudos publicados por periódico.	67
Tabela 10 – Número de estudos publicado por conferência.	68
Tabela 11 – Métodos a serem implementados por um <i>plug-in</i> da O-FIm/CO	79
Tabela 12 – Categorias de imagens definidas a partir da similaridade entre o número de características.	96
Tabela 13 – Imagens com seus respectivos números de características – Estudo 1.	98
Tabela 14 – Resultados retornados pelo oráculo por meio das métricas de precisão, revocação e especificidade para imagens 3D – limiar = 0,350.	99
Tabela 15 – Resultados retornados pelo oráculo por meio das métricas de precisão, revocação e especificidade para imagens 2D – limiar = 0,350.	100
Tabela 16 – Teste binomial a partir das métricas coletadas no estudo 1.	101
Tabela 17 – Imagens com os respectivos número de características – Estudo 2.	112
Tabela 18 – Desempenho do oráculo por meio das métricas de precisão, revocação e especificidade, com o primeiro conjunto de características – Limiar = th_2	112
Tabela 19 – Desempenho do oráculo por meio das métricas de precisão, revocação e especificidade, com o segundo conjunto de características – Limiar = th_2	113
Tabela 20 – Teste binomial a partir das métricas coletadas no estudo 2.	115
Tabela 21 – Desempenho dos oráculos humanos em relação ao oráculo automatizado (th_2).	124
Tabela 22 – <i>T-test</i> entre o oráculo de teste baseado em características e os oráculos humanos.	125

LISTA DE ABREVIATURAS E SIGLAS

2D	Bidimensionais
3D	Tridimensionais
CAD	<i>Computer-Aided Diagnosis</i>
CAV	<i>Cardiac Allograft Vasculopathy</i>
CBIR	<i>Content-Based Image Retrieval</i>
CQ	Controle de Qualidade
CQUS	<i>Control Quality for Ultrasound B-mode equipment</i>
CRO	Sistema Contínuo de Monitoramento da Respiração do Olfatômetro
CSB	Detecção de respiração de <i>Cheyne-Stokes</i>
DCF	<i>Device Coordination Framework</i>
DE	Distância Euclidiana
Def-Uso	<i>Def-Use Graph</i>
DG	Distribuição Gaussiana
DSE	<i>Dynamic Symbolic Execution</i>
EEG	Eletroencefalograma
ES	Engenharia de Software
FG	Filtro Gaussiano
GFC	Grafo de Fluxo de Controle
GUI	<i>Graphical User Interface</i>
MITK	<i>Medical Imaging Interaction Toolkit</i>
MODA	<i>Mock Object Based Test Generation for Database Applications</i>
MS	Mapeamento Sistemático
NIfTI	<i>Neuroimaging Informatics Technology Initiative</i>
O-FIm/CO	<i>Oracle For Images and Complex Outputs</i>
PDI	Processamento Digital de Imagens
PLN	Processamento de Linguagem Natural
PxR	Precisão <i>versus</i> Revocação
RCS	Respiração <i>Cheyne-Stokes</i>
RSL	Revisão Sistemática da Literatura
RV	Realidade Virtual
SIH	Sistemas de Informação Hospitalar

SMDCSes	Sistemas de Computação Móvel
SOTF	<i>Schedulable Online Testing</i>
SUT	<i>System Under Testing</i>
TTS	<i>Text-To-Speech</i>
UML	<i>Unified Modeling Language</i>
VTK	<i>Visualization Toolkit</i>
VV&T	Verificação, Validação e Teste de Software

SUMÁRIO

1	INTRODUÇÃO	27
1.1	Contextualização e motivação	27
1.2	Metodologia e objetivos	29
1.3	Contribuições	31
1.4	Organização do trabalho	31
2	FUNDAMENTAÇÃO TEÓRICA	33
2.1	Considerações iniciais	33
2.2	Atividades de VV&T	33
2.2.1	<i>Técnicas e critérios de teste</i>	<i>35</i>
2.2.1.1	<i>Técnica funcional</i>	<i>36</i>
2.2.1.2	<i>Técnica estrutural</i>	<i>36</i>
2.2.1.3	<i>Teste baseado em defeitos</i>	<i>38</i>
2.3	Automatização de teste de software	39
2.4	Oráculos de teste	42
2.4.1	<i>Problema de oráculo: Falsos positivos e Falsos negativos</i>	<i>43</i>
2.5	Sistemas com saídas complexas	43
2.5.1	<i>Imagens sintéticas tridimensionais de vasos sanguíneos</i>	<i>45</i>
2.6	Considerações finais	47
3	TESTE DE SOFTWARE E ANÁLISE DE QUALIDADE EM SISTEMAS BIOMÉDICOS: UM MAPEAMENTO SISTEMÁTICO	49
3.1	Considerações iniciais	49
3.2	Planejamento	50
3.2.1	<i>Objetivos</i>	<i>50</i>
3.2.2	<i>Questões de pesquisa</i>	<i>50</i>
3.2.3	<i>Estratégia de busca e seleção</i>	<i>51</i>
3.2.4	<i>Critérios de seleção</i>	<i>52</i>
3.3	Condução	53
3.4	Análise dos resultados	54
3.5	Validação de imagens médicas	68
3.6	Considerações finais	69

4	ORÁCULOS DE TESTE BASEADOS EM CARACTERÍSTICAS PARA IMAGENS MÉDICAS 3D DE VASOS SANGUÍNEOS	71
4.1	Considerações iniciais	71
4.2	Definições da abordagem	71
4.2.1	<i>CBIR</i>	72
4.3	Oráculos de teste baseado em características – um passo a passo .	74
4.4	Limitações da abordagem	75
4.5	Framework O-Flm/CO	76
4.5.1	<i>API – O-Flm/CO</i>	78
4.6	O-Flm/CO para imagens médicas 3D	79
4.6.1	<i>Extensões na estrutura do framework O-Flm/CO</i>	79
4.7	Extratores de características para imagens médicas 3D	81
4.7.1	<i>Correlação entre os extratores de características</i>	85
4.8	Considerações finais	86
5	APLICAÇÃO DA ABORDAGEM EM IMAGENS MÉDICAS 3D DE VASOS SANGUÍNEOS	87
5.1	Considerações iniciais	87
5.2	Categorização de imagens sintéticas 3D e 2D de vasos sanguíneos	87
5.2.1	<i>Materiais e métodos</i>	88
5.2.1.1	<i>Extratores de características</i>	88
5.2.1.2	<i>Função de similaridade</i>	89
5.2.1.3	<i>Classificador randômico</i>	90
5.2.1.4	<i>Configuração do limiar</i>	90
5.2.2	<i>Questões de pesquisa e métricas</i>	91
5.2.3	<i>Formulação de hipóteses</i>	93
5.2.4	<i>Conjunto de dados</i>	94
5.2.5	<i>Condução do experimento</i>	95
5.2.6	<i>Resultados e discussão</i>	97
5.2.6.1	QP1	97
5.2.6.2	QP2	101
5.2.7	<i>Ameaças à validade</i>	102
5.2.8	<i>Considerações sobre o experimento 1</i>	102
5.3	Avaliação de imagens sintéticas 3D de vasos sanguíneos após aplicação de ruído gaussiano	103
5.3.1	<i>Materiais e métodos</i>	104
5.3.1.1	<i>Extratores de características e Função de similaridade</i>	104
5.3.1.2	<i>Aplicação do ruído gaussiano</i>	105
5.3.1.3	<i>Configuração do limiar</i>	106
5.3.2	<i>Questões de pesquisa e métricas</i>	106

5.3.3	<i>Formulação de hipóteses</i>	108
5.3.4	<i>Conjunto de dados</i>	109
5.3.5	<i>Condução do experimento</i>	109
5.3.6	<i>Resultados e discussão</i>	111
5.3.6.1	QP1	112
5.3.7	<i>Ameaças à validade</i>	115
5.3.8	<i>Considerações sobre o experimento 2</i>	116
5.4	Oráculos humanos para avaliação da qualidade de sistemas que geram imagens sintéticas 3D de vasos sanguíneos	117
5.4.1	<i>Materiais e métodos</i>	117
5.4.1.1	<i>Imagens sintéticas 3D de vasos sanguíneos</i>	118
5.4.1.2	<i>Formulário de pesquisa</i>	118
5.4.2	<i>Seleção de sujeitos</i>	119
5.4.3	<i>Questões de pesquisa e métricas</i>	120
5.4.4	<i>Formulação de hipóteses</i>	121
5.4.5	<i>Condução do experimento</i>	122
5.4.6	<i>Resultados e discussão</i>	123
5.4.6.1	QP1	124
5.4.7	<i>Ameaças à validade</i>	125
5.4.8	<i>Considerações sobre o experimento 3</i>	126
5.5	<i>Considerações finais</i>	127
6	CONCLUSÕES E TRABALHOS FUTUROS	129
6.1	<i>Contribuições</i>	129
6.2	<i>Discussão das limitações</i>	130
6.3	<i>Trabalhos futuros</i>	131
6.4	<i>Considerações finais</i>	132
REFERÊNCIAS		133
APÊNDICE A	CÓDIGO-FONTE DA FUNÇÃO PRINCIPAL DOS EX-TRATORES DE CARACTERÍSTICAS PARA IMAGENS MÉDICAS 3D	139
APÊNDICE B	MATRIZ DE CORRELAÇÃO ENTRE TODOS OS PA-RES DE CARACTERÍSTICAS	153

INTRODUÇÃO

1.1 Contextualização e motivação

A complexidade no desenvolvimento de software está relacionada ao processo evolutivo pelo qual a computação passou nas últimas décadas. O software é o ingrediente chave em muitos dos dispositivos e sistemas que permeiam a sociedade contemporânea (AMMANN; OFFUTT, 2008). Linguagens de programação em alto nível de abstração, a diversidade de ambientes de execução, erros de escrita de código e especificação incompleta são alguns dos fatores que dificultam o desenvolvimento de software confiável (DELAMARO; MALDONADO; JINO, 2016). Além disso, muitos fatores podem ser identificados como causas de problemas em sistemas de software, mas a maioria deles tem uma única origem: o erro humano durante a implementação dos sistemas (DELAMARO; MALDONADO; JINO, 2016). Diante desse cenário, atividades Verificação, Validação e Teste de Software (VV&T) são essenciais para garantir que o software que está sendo construído esteja em conformidade com a especificação (BERTOLINO, 2007; DELAMARO; MALDONADO; JINO, 2016).

O teste de software é um dos mais importantes componentes da área de Engenharia de Software (ES). Conceitualmente, teste de software é o conjunto de atividades dinâmicas que consistem na execução de um programa com algumas entradas específicas, visando verificar se o seu comportamento é condizente com sua especificação (BERTOLINO, 2007; DELAMARO; MALDONADO; JINO, 2016). O processo de teste é o meio mais importante pelo qual tais sistemas são submetidos para terem sua conformidade com especificações verificadas (HUNTER; STROOPER, 2001; AMMANN; OFFUTT, 2008). Negligenciar as atividades de teste pode reverter à liberação de software de má qualidade, gerando prejuízos econômicos e comprometendo a reputação dos envolvidos no projeto.

O teste de software requer até 50% dos custos de desenvolvimento de software (AMMANN; OFFUTT, 2008). Para aplicativos críticos de segurança esse custo é ainda maior. Nesse

cenário, o emprego de testes automatizados contribui significativamente para a redução de custos e tempo de projeto durante o processo de desenvolvimento (GLENFORD; SANDLER; BADGETT, 2004). Segundo Rafi *et al.* (2012), aumento da qualidade do produto final, alto índice de cobertura, redução do tempo de teste, aumento da confiança, diminuição de esforços humanos e redução de custos são alguns dos benefícios do teste automatizado. Por isso, e pela aceitação entre pesquisadores e engenheiros de software da ideia de que qualidade é um fator essencial no desenvolvimento de software, muito se tem investido em pesquisas na área de teste de software (DELAMARO; MALDONADO; JINO, 2016). Pesquisadores e profissionais da indústria procuram definir técnicas, critérios e ferramentas que possibilitem a aplicação de tais atividades de maneira sistemática, com alta qualidade e custo reduzido.

O mecanismo que se utiliza para julgar a correção da saída ou o comportamento esperado de uma execução de um determinado programa é conhecido como “oráculo” (BARR *et al.*, 2015). Dado um programa P , em um determinado ambiente de desenvolvimento e teste, o oráculo pode ser implementado de diversas formas e deve basear-se na especificação de P . Caso tal especificação seja, por exemplo, uma definição de requisitos feita, informalmente, provavelmente caberá ao testador desempenhar o papel de oráculo e decidir sobre o comportamento de P diante de suas especificações. Caso exista para P , por exemplo, um modelo formal a partir do qual seja possível extrair seu comportamento, é possível automatizar a função de oráculo implementando-se um comparador entre a saída produzida e o comportamento definido no modelo. Existem, ainda, os casos em que o oráculo é inexistente ou existe apenas de maneira aproximada (HUAI *et al.*, 2014).

Um problema recorrente associado à automatização de oráculos de teste ocorre quando a saída do SUT (do inglês, *System Under Testing*) configura um domínio de dados complexos (OLIVEIRA *et al.*, 2014). Atualmente, muitos são os exemplos de sistemas de saídas complexas como: (1) sistemas com GUIs (do inglês, *Graphical User Interface*) complexas cuja informação visual é refletida em diferentes resoluções de tela, tamanhos e orientações; (2) sistemas TTS (do inglês, *Text-To-Speech*) que convertem texto escrito em arquivos de áudio; (3) aplicações Web que devem executar corretamente em diversos navegadores; (4) aplicativos móveis que são executados em várias plataformas diferentes sob diferentes resoluções de tela, tamanhos e orientações; (5) ambientes de Realidade Virtual (RV); e (6) sistemas que manipulam imagens sintéticas tridimensionais (3D) .

Em sistemas biomédicos – exemplo de sistema cujas saídas são consideradas complexas – é essencial a aplicação de atividades que objetivem avaliar sua qualidade. Sistemas biomédicos são aqueles que têm a capacidade de apoiar a decisão clínica de profissionais da área médica (JETLEY; JONES; ANDERSON, 2008). Sistemas de diagnóstico de imagens de ultra-sonografia (FILHO *et al.*, 2014; JIANG; SHI; LIU, 2011; WANG; HELMINEN; JIANG, 2014), sistemas de processamento de linguagem (ZHENG *et al.*, 2015; COHEN; HUNTER; PALMER, 2013), sistemas de informações hospitalares (KAWAMURA; KIMURA; TSUMOTO, 2014; PAECH;

WETTER, 2008) e sistemas de diagnóstico radiológico (WARD *et al.*, 2009) são exemplos de sistemas biomédicos contemporâneos. Sistemas biomédicos englobam um conjunto de características fundamentais que necessitam de atividades de para oferecer qualidade ao sistema como um todo.

Sistemas que manipulem imagens sintéticas 3D de vasos sanguíneos exemplificam um tipo de sistema biomédico (Seção 2.5.1). Tais sistemas apoiam a área médica contribuindo na antecipação do diagnóstico de determinadas patologias (GALARRETA-VALVERDE *et al.*, 2013). No entanto, sistemas que manipulem imagens sintéticas 3D de vasos sanguíneos não têm sua qualidade mensurada devido à falta de estratégias de teste automatizado. Além disso, a alta complexidade dos dados do domínio de saída de tais sistemas dificulta a aplicação de atividades de teste. Devido a falta de estratégias que visem automatizar o processo de avaliação de qualidade de tais sistemas, estratégias *ad-hoc* e manuais são aplicadas. Uma estratégia promissora para avaliar a qualidade de sistemas que manipulem tais imagens é por meio de uma abordagem denominada “oráculos de teste baseados em características” (DELAMARO; NUNES; OLIVEIRA, 2013; OLIVEIRA, 2017). Oráculos de teste baseados em características é uma abordagem na qual as características extraídas das saídas do SUT são utilizadas como fonte de informação para o oráculo julgar a correção da saída do SUT.

Esta dissertação visa fornecer, avaliar e validar uma estratégia para suporte de atividades de teste para sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos por meio de oráculos de teste automatizados. Desse modo, dois principais fatores motivam o desenvolvimento do presente trabalho: (i) estratégias tradicionais de teste são ineficazes quando aplicadas em sistemas com saída complexa; e (ii) estratégias sistemáticas e/ou automatizadas de teste configuram como um novo campo de pesquisa em sistemas biomédicos.

1.2 Metodologia e objetivos

O presente estudo explora conceitos de Recuperação de Imagens Baseado em Conteúdo (do inglês, *Content-based image retrieval* – CBIR) para promover a automatização de oráculos de teste para domínios de saídas complexas. CBIR é uma técnica oriunda da área de Processamento de Imagem e Reconhecimento de Padrões que permite que um sistema recupere um conjunto de imagens a partir de um banco de imagens, tendo como chave de busca uma imagem de referência (DATTA *et al.*, 2008). Nesse cenário, o *framework* O-FIm/CO (OLIVEIRA; DELAMARO; NUNES, 2009; DELAMARO; NUNES; OLIVEIRA, 2013; OLIVEIRA *et al.*, 2014) (do inglês, *Oracle For Images and Complex Outputs*) emprega conceitos de CBIR para permitir que sejam criados oráculos que apoiem o teste automatizado em sistemas com saídas complexas como, no presente caso, programas de geração de imagens 3D de vasos sanguíneos.

O O-FIm/CO permite que testadores instalem extratores de características específicos para o domínio de saída de um SUT particular. Após isso, utilizando como referência saídas

previamente validadas como corretas, é possível configurar um ambiente de teste no qual as características das saídas do SUT são comparadas às características das saídas de referência. Em trabalhos anteriores, a estratégia foi aplicada para apoiar o teste em esquemas CAD (do inglês, *Computer-Aided Diagnosis*) de imagens mamográficas (DELAMARO; NUNES; OLIVEIRA, 2013), GUI (OLIVEIRA; DELAMARO; NUNES, 2010), sistemas TTS (OLIVEIRA *et al.*, 2014) e banco de dados médico composto por modelos 3D (BERGAMASCO *et al.*, 2015).

O escopo desta dissertação é a automatização de oráculos de teste para sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos, permitindo uma análise minuciosa das redes vasculares (GALARRETA-VALVERDE *et al.*, 2013; GALARRETA-VALVERDE, 2012). Tais sistemas podem atuar em duas funcionalidades distintas: (1) geração de imagens sintéticas; e (2) processamento de imagens sintéticas. Considerando a primeira funcionalidade, algoritmos são capazes de gerar imagens que simulem características naturais de vasos reais como o ângulo de bifurcação, comprimento, diâmetro médio e possibilita a simulação de anomalias vasculares. Considerando o segundo aspecto, a segmentação e processamento de imagens de redes vasculares a partir de tais imagens é uma das tarefas iniciais no diagnóstico de doenças vasculares como estenoses ou aneurismas (GALARRETA-VALVERDE *et al.*, 2013; GALARRETA-VALVERDE, 2012).

O objetivo geral deste projeto é o fornecimento de alternativas para uma avaliação criteriosa, sistemática e produtiva da qualidade de sistemas com saídas que processem ou gerem imagens sintéticas 3D de vasos sanguíneos. Para tanto, utiliza-se a abordagem de oráculos de teste baseados em características. Dado tal objetivo geral, definiram-se os seguintes objetivos específicos:

- definir o modo como extratores de características podem fornecer informações sobre a correção de sistemas que processem ou gerem imagens sintéticas 3D de vasos sanguíneos;
- documentar e generalizar características e particularidades de sistemas que processem ou gerem imagens sintéticas 3D de vasos sanguíneos que possam ser utilizados em atividades de teste de software;
- generalizar um *framework* para teste de sistemas de saídas complexas; e
- estabelecer um ambiente para que novos sistemas 3D sejam inseridos no contexto do O-FIm/CO.

Além dos objetivos específicos citados, o presente estudo propõe e avalia uma estratégia de oráculo automatizado para sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos.

1.3 Contribuições

A seguir são descritas algumas contribuições em decorrência da execução da pesquisa reportada por meio desta dissertação:

- aplicação da abordagem oráculos de teste baseados em características em um novo domínio de sistemas com saídas complexas;
- um mapeamento sistemático sobre abordagens e estratégias de teste de software e avaliação de qualidade em sistemas biomédicos;
- desenvolvimento e validação de extratores de características para imagens médicas 3D com seus código-fontes disponibilizados de modo *open-source*; e
- avaliações experimentais de oráculos de teste como uma estratégia sistemática para a avaliação de sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos, destacando suas vantagens e desvantagens.

1.4 Organização do trabalho

Além deste capítulo introdutório, o restante deste texto está organizado da seguinte forma:

- o Capítulo 2 apresenta algumas definições e conceitos relacionados às atividades de VV&T e oráculos de teste. Além disso, o Capítulo 2 apresenta aspectos técnicos de sistemas com saídas complexas, o que inclui sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos;
- no Capítulo 3 é apresentado um mapeamento sistemático acerca de estudos que apresentem métodos ou estratégias de teste de software e avaliação de qualidade em sistemas biomédicos, o que inclui sistemas que processem ou gerem imagens sintéticas 3D de vasos sanguíneos;
- o Capítulo 4 apresenta algumas definições e conceitos relacionados à abordagem de oráculos de teste baseados em características. Além disso, apresenta aspectos técnicos do *framework* O-FIm/CO, assim como extensões realizadas para suporte de atividades de teste em imagens médicas 3D;
- no Capítulo 5 são apresentados três estudos experimentais realizados com o objetivo de avaliar imagens sintéticas 3D de vasos sanguíneos; e
- no Capítulo 6 são apresentadas conclusões e observações finais sobre o trabalho de mestrado realizado.

FUNDAMENTAÇÃO TEÓRICA

2.1 Considerações iniciais

Neste capítulo são apresentados conceitos relacionados a atividades de VV&T (2.2), automatização de teste de software (2.3), oráculos de teste (2.4) e sistemas com saídas complexas (2.5).

2.2 Atividades de VV&T

O desenvolvimento de sistemas de software com qualidade e produtividade é um dos principais objetivos da ES. Dessa forma, atividades de VV&T são consideradas fundamentais (DELAMARO; MALDONADO; JINO, 2016). A Validação assegura que o produto que está sendo desenvolvido corresponde ao produto correto, conforme os requisitos coletados a partir do usuário. A Verificação tende a assegurar consistência, completude, correção do produto em cada fase e entre fases do ciclo de vida (DELAMARO; MALDONADO; JINO, 2016). Por sua vez, o Teste de Software é um processo em que um programa é executado com o objetivo de revelar falhas, aumentando a confiabilidade do sistema por meio da correção de defeitos (BERTOLINO, 2003).

As atividades de VV&T estão diretamente relacionadas à garantia de qualidade, objetivando a detecção de falhas em sistemas. Para fins conceituais, a IEEE 610.12 (IEEE, 1990) padroniza termos utilizados na área de teste de software:

- **Engano (*mistake*):** resultados incorretos encontrados no software causados pela influência da ação humana. Geralmente ocorre por meio de uma ação incorreta do programador;
- **Defeito (*fault*):** passo, processo ou definição de dados incorretos, incompletos, ausentes ou extras que, ao ser executado, pode produzir um erro no programa. Ocorre, por exemplo, com uma instrução ou comando incorreto;

- **Erro (*error*):** diferença entre valor obtido e o valor esperado, ou seja, qualquer estado na execução do programa, intermediário ou final, que seja inconsistente; e
- **Falha (*failure*):** evento notável em que o sistema viola suas especificações, ou seja, é a produção de uma saída ou comportamento incorretos com relação à especificação.

De acordo com Delamaro, Maldonado e Jino (2016), as atividades de VV&T compõem atividades caracterizadas como estáticas e dinâmicas. As atividades estáticas são as que não requerem a execução ou mesmo a existência de um programa ou modelo executável para serem conduzidas. Atividades dinâmicas são aquelas que se baseiam na execução de um programa ou de um modelo.

A atividade de teste é essencial em todas as fases do processo de desenvolvimento de software. A Figura 1 ilustra como os elementos associados ao teste de software interagem em um ambiente genérico de teste. Cada um dos itens apresentados na figura podem ser representados da seguinte forma:

- *I*: todo o domínio de entrada do programa *P*;
- *T*: conjunto de dados de teste selecionados a partir de *I*;
- *P*: programa em teste;
- *O*: todo o domínio de saída do programa *P*;
- *S*: conjuntos de saídas correspondentes às entradas selecionadas em *T*; e
- *Oráculo* (ator): mecanismo que estabelece se as saídas dos testes estão de acordo com o esperado.

Figura 1 – Atividades de teste de software

Fonte: Adaptada de Delamaro, Maldonado e Jino (2016).

De acordo com a Figura 1, um domínio de entradas *I* é definido e um conjunto de dados de teste *T* é selecionado a partir de *I*. O conjunto *T* representa a entrada para o programa *P* que, conseqüentemente, produzirá uma respectiva saída. Além disso, um domínio de saída *O* do programa *P* é definido e conjuntos de saídas correspondentes às entradas *T* são selecionados a partir de *O*. Dessa forma, o oráculo humano, que também pode ser representado por uma ferramenta, define se as saídas dos testes estão de acordo com o esperado.

As etapas de teste de software visam sistematizar o teste de software e devem ser aplicadas durante o desenvolvimento. De modo genérico, existem quatro principais fases de teste de software:

- **Teste de unidade:** concentra esforços nas unidades de projeto do software a fim de verificar se estas funcionam adequadamente. Cada unidade do programa é explorada separadamente, procurando-se identificar erros de lógica e de implementação nas mesmas. As unidades de um sistema podem ser, por exemplo, componentes, métodos, classes ou funções;
- **Teste de integração:** realizada após o teste de unidade, é a atividade sistemática aplicada durante a integração da estrutura do programa, visando a identificar erros associados às interfaces e passagem de mensagens entre os módulos do software. O objetivo é, a partir dos módulos testados no nível de unidade, construir a estrutura de programa que foi determinada pelo projeto. Além disso, o teste de integração visa assegurar a consistência do software após adição de novas unidades;
- **Teste de sistema:** visa identificar erros de funções e características de desempenho que não estejam de acordo com a especificação. O objetivo é assegurar que o software e os demais elementos que compõem o sistema (por exemplo, hardware, banco de dados, sistema operacional) combinam-se adequadamente e que a função/desempenho global desejada seja obtida. Dentre os testes executados nessa fase, podem ser citados: Teste de recuperação (PRESSMAN, 2011), Teste de segurança (PRESSMAN, 2011), Teste de estresse (PRESSMAN, 2011), Teste de desempenho (PRESSMAN, 2011), Teste de aceitação e Teste *end-to-end* (LIANG *et al.*, 2012); e
- **Teste de regressão:** técnicas de teste de regressão reutilizam subconjuntos do conjunto de teste existente. Sob outra perspectiva, a cada manutenção é preciso verificar se o software mantém suas características e se nenhum efeito colateral foi introduzido.

As fases apresentadas são aplicadas com a utilização de diferentes técnicas. As técnicas de teste são definidas de acordo com a fonte de informação utilizada como recurso para a criação de casos de testes (DELAMARO; MALDONADO; JINO, 2016). A Seção 2.2.1 apresenta detalhes acerca das técnicas de teste mais comuns e seus respectivos critérios.

2.2.1 Técnicas e critérios de teste

As atividades de teste devem aumentar a chance de revelar defeitos em um SUT, selecionando dados do domínio de entrada que sejam mais eficazes para tal tarefa. As técnicas e critérios de teste fornecem ao desenvolvedor uma abordagem sistemática e teoricamente fundamentada para a modelagem de casos de teste eficazes, construindo um mecanismo que pode auxiliar a avaliar a qualidade e a adequação da atividade de teste.

Segundo Fabbri, Vicenzi e Maldonado (2016), as técnicas de teste de software podem ser classificadas em: técnica funcional, técnica estrutural e técnica baseada em defeitos. Cada técnica possui seus respectivos critérios que estabelecem requisitos de teste para os quais valores específicos do domínio de entrada do programa devem ser definidos, com o intuito de exercitá-los. A seguir é apresentada cada técnica de teste de software e alguns dos seus critérios.

2.2.1.1 Técnica funcional

O teste funcional, também conhecido como técnica caixa-preta, considera o produto em teste como uma caixa da qual só se conhece a entrada e a saída (sem conhecimento da parte interna). Baseia-se na especificação do software para derivar os casos de teste, desconsiderando aspectos associados ao código da aplicação. Dessa forma, nessa técnica os detalhes de implementação não são considerados e o software é avaliado segundo o ponto de vista do usuário (PEZZE; ZHANG, 2014; AMMANN; OFFUTT, 2008).

De acordo com Fabbri, Vicenzi e Maldonado (2016), os principais critérios do teste funcional são: (i) particionamento em classes de equivalência; (ii) análise do valor limite; (iii) teste funcional sistemático; e (iv) grafo causa-efeito.

A técnica funcional pode ser utilizada em todas as fases de teste e não depende do paradigma ou linguagem de programação utilizados. A técnica funcional é eficaz em detectar determinados tipos de defeitos como, por exemplo, funcionalidades ausentes. Entretanto, o êxito na aplicação da técnica funcional está fortemente relacionado com a qualidade da especificação de requisitos.

2.2.1.2 Técnica estrutural

A técnica estrutural, também conhecida como caixa-branca (ao contrário do teste caixa-preta), baseia-se no conhecimento da estrutura interna (implementação) do programa. A maioria dos critérios dessa técnica utiliza uma representação do programa conhecida como grafo de fluxo de controle (GFC), que é uma representação que visa auxiliar a geração dos requisitos de teste. Os **Critérios baseados em fluxo de controle** utilizam características de controle de execução do programa para determinar os requisitos de teste com base em comandos e desvios de execução. Os critérios mais conhecidos dessa classe são: (i) Todos-Nós; (ii) Todas-Arestas; e (iii) Todos-Caminhos. O Código 1 apresenta a função main do programa *Identifier*, enquanto a Figura 2 ilustra o GFC dessa função. No GFC, cada vértice representa um bloco indivisível de comandos do programa (BARBOSA *et al.*, 2016; GLENFORD; SANDLER; BADGETT, 2004).

Código-fonte 1 – Função main do programa *Identifier*.

```
1: {                               /*01*/
2:   char   achar;                 /*01*/
3:   int     length, valid_id;     /*01*/
```

```

4:  length = 0;                                /*01*/
5:  printf ("Identificador: ");                 /*01*/
6:  achar = fgetc (stdin);                     /*01*/
7:  valid_id = valid_s(achar);                  /*01*/
8:  if (valid_id)                               /*01*/
9:      length = 1;                             /*02*/
10:  achar = fgetc (stdin);                     /*03*/
11:  while (achar != '\n')                      /*04*/
12:  {                                           /*05*/
13:      if (!(valid_f(achar)))                 /*05*/
14:          valid_id = 0;                     /*06*/
15:      length++;                             /*07*/
16:      achar = fgetc (stdin);                 /*07*/
17:  }                                           /*07*/
18:  if (valid_id && (length >= 1)                /*08*/
19:      && (length < 6) )                      /*08*/
20:      printf ("Valido\n");                   /*09*/
21:  else                                       /*10*/
22:      printf ("Invalido\n");                 /*10*/
23: }                                           /*11*/

```

Fonte: Elaborada pelo autor.

Figura 2 – Exemplo do GFC da função main do programa *Identifier*.

Fonte: Elaborada pelo autor.

Adicionalmente aos critérios supra-citados, o **Critério baseado na complexidade**, também conhecido como teste do caminho básico ou critério de McCabe (MCCABE, 1976), baseia-se na complexidade ciclomática, calculada a partir do GFC. A complexidade ciclomática define o número de caminhos independentes de um programa, que é qualquer caminho no grafo em que pelo menos uma aresta ainda não tenha sido incluída em outro caminho (BARBOSA *et al.*, 2016). McCabe (1976) define a complexidade ciclomática (C) como: $C = E - N + 2$, tal que E é o número de arestas e N é o número de vértices do GFC G. Nesse cenário, com base no GFC ilustrado na Figura 2, C seria igual a $14 - 11 + 2$, totalizando 5. Dessa forma, seguindo o que é estabelecido pelo critério, o testador deve escolher cinco caminhos independentes como requisito de teste a ser cumprido por um conjunto de teste.

Por fim, os **Critérios baseados em fluxo de dados** adicionam informação sobre a utilização de variáveis ao GFC. Uma família de critérios bastante conhecida é a de Rapps e Weyuker (RAPPS; WAYUKER, 1985), que utiliza um GFC estendido, chamado grafo Def-Uso (do inglês *Def-Use Graph*). De acordo com Barbosa *et al.* (2016), os critérios mais conhecidos nessa classe são: (i) Todas-Definições; (ii) Todos-Usos; e (iii) Todos-DU-Caminhos. Outra família de critérios dessa classe é os Potenciais-Uso. Nessa classe, os elementos são caracterizados independentemente da ocorrência explícita de uma referência (um uso) a uma definição de variável. Os critérios mais conhecidos dessa família são: (i) Todos-Potenciais-Usos; (ii) Todos-Potenciais-Usos/DU; e (iii) Todos-Potenciais-DU-Caminhos.

Critérios de teste da técnica estrutural identificam caminhos que devem ser percorridos no código do programa. Dessa forma, o GFC é a base a partir do qual os requisitos de testes são derivados. No entanto, a aplicação da técnica estrutural não pode garantir que o programa em teste comporta-se como esperado, devido ao problema dos caminhos ausentes. Caminhos ausentes ocorrem se o programa não implementa algumas funções, não havendo um caminho que corresponda àquela função; logo, nenhum dado de teste será requerido para exercitá-la.

2.2.1.3 Teste baseado em defeitos

Na técnica baseada em defeitos, os requisitos de teste são derivados a partir dos erros mais frequentes cometidos durante o processo de desenvolvimento do software. O teste de mutação, também chamado de análise de mutantes, é o critério mais conhecido da técnica baseada em defeitos (DELAMARO *et al.*, 2016; DEMILLO; LIPTON; SAYWARD, 1978).

No teste de mutação, programas alternativos, denominados mutantes, são criados a partir de sutis modificações do programa original. Tais modificações são efetuadas por meio da inserção de pequenos desvios sintáticos produzidos por regras conhecidas como operadores de mutação. A ideia é gerar várias versões modificadas do programa original, com o objetivo de reproduzir os defeitos mais comuns introduzidos nos programas pelos programadores (DEMILLO, 1989; JIA; HARMAN, 2011).

A tarefa do testador é modelar um conjunto de casos de teste que seja capaz de distinguir

o programa original de todos os seus mutantes. Desse modo, em um ambiente de teste de mutação, um mutante pode ser classificado de três maneiras distintas: (i) vivo, (ii) equivalente ou (iii) morto. A princípio, todos os mutantes são classificados como vivos, não tendo nenhuma distinção aparente do programa original. Um mutante morto é aquele cuja saída difere, para algum caso de teste, do programa original. Por fim, um mutante equivalente é aquele que não tem diferença de comportamento em relação ao programa original, produzindo as mesmas saídas para qualquer dado de entrada.

Uma forma de medir a adequação de um conjunto de teste T em relação ao teste de mutação é por meio do escore de mutação:

$$ms(P, T) = \frac{DM(P, T)}{M(P) - EM(P)} \quad (2.1)$$

onde:

- $DM(P, T)$ = número de mutantes mortos pelos conjunto de casos de teste em T ;
- $M(P)$ = número total de mutantes gerados;
- $EM(P)$ = número de mutantes gerados equivalentes a P .

2.3 Automatização de teste de software

Atividades de V&V tomam muito tempo de projeto e, na maioria das vezes, são dispendiosas quando aplicadas em processos de desenvolvimento (MYERS; SANDLER; BADGETT, 2011). Além da falta de ferramentas adequadas que apoiem a condução dessas atividades, o estabelecimento de estratégias de teste adequadas e sistemáticas é um desafio enfrentado por profissionais e pesquisadores da área. Tais desafios levam, geralmente, à aplicação de estratégias de teste manuais e *ad-hoc*, tornando as atividades de V&V improdutivas, ineficazes e parciais (MYERS; SANDLER; BADGETT, 2011).

Um recurso para aumentar a produtividade e a qualidade das atividades de V&V é a aplicação de estratégias de testes automatizados. De acordo com Oliveira (2014), as estratégias de teste automatizado envolvem quatro aspectos: (i) levando em consideração a especificação do SUT, selecionar dados de teste adequados (automaticamente ou manualmente); (ii) configuração e automação de um ambiente ou ferramenta para executar um SUT com os dados do teste; (iii) selecionar um método para determinar a correção das saídas; e (iv) verificar os resultados por meio de um relatório ou critérios de cobertura pré-definidos.

A Figura 3 apresenta uma estrutura genérica associada ao fluxo de tarefas do teste de software. Tal estrutura contempla diversas etapas que podem variar dependendo da natureza do sistema em teste. Entretanto, o fluxo do teste habitualmente se inicia a partir da análise do “Domínio de entradas” do SUT, que representa todos os possíveis dados de entrada. Por meio

dessa análise, explorando técnicas/critérios o projetista do teste seleciona algumas entradas específicas que irão compor os “*Dados de teste*”. Em geral, um projetista de teste seleciona tais dados (“*ent*”) cuidadosamente a partir de algum critério de alguma técnica de teste. Essa seleção de dados de teste é fundamental para a produtividade do teste, haja vista que muitas vezes o domínio de entradas do SUT é infinito e o teste do sistema com todos os elementos do SUT é impraticável.

Figura 3 – Fluxo genérico de teste com oráculos

Fonte: Adaptada de Oliveira (2017).

Em seguida, o SUT é executado a partir das entradas “*ent*”, produzindo uma saída qualquer comumente chamada de “*saída real*”. É exatamente nessa etapa do teste que agem os “*Oráculos de teste*” – responsáveis por avaliar as saídas do SUT diante de alguma referência.

O teste automatizado tem uma série de características particulares que o diferem do teste manual, cuja característica principal é a necessidade da figura humana conduzindo os testes (DUSTIN; GARRETT; GAUF, 2009). A automatização do teste requer o desenvolvimento de recursos e ferramentas específicas, sendo focada em substituir testes manuais complexos. O teste automatizado pode estar associado a uma ampla variedade de contextos dentro das atividades de teste: (i) condução, execução e geração de resultados apoiados por software; (ii) suporte à geração e seleção de casos de teste; e (iii) rápida reexecução e suporte ao teste de regressão. Entretanto, o teste automatizado não pode ser separado do teste manual, uma vez que as duas abordagens são complementares (OLIVEIRA *et al.*, 2014).

Oliveira (2012) apresenta uma análise relacionando as modalidades de teste manual e automatizada. A Tabela 1 apresenta uma série de características envolvendo as duas modalidades de teste perante aspectos importantes em um ambiente de desenvolvimento e teste de software como, por exemplo, custo e produtividade. De acordo com a Tabela 1, o teste automatizado permite uma série de vantagens em relação ao teste manual como, por exemplo: reexecução de casos de teste, suporte à regressão, tempo de projeto reduzido e uma maior produtividade são algumas delas. Por outro lado, o teste manual é menos complexa de ser implementada e tem custo reduzido a curto prazo (HOFFMAN, 2006).

Tabela 1 – Teste manual vs Teste automatizado

	Teste manual	Teste automatizado
Condução	humano	máquina
Reexecução	demorada/cansativa	simples/rápida
Suporte à regressão	não	sim
Implementação	simples	complexa
Custo	alto	baixo
Tempo de projeto	extenso	reduzido
Produtividade	baixa	alta

Fonte: Adaptada de Oliveira (2012).

Vários aspectos e cenários podem ser levados em consideração para esclarecer as diferenças entre o teste automatizado e manual. A Tabela 2 apresenta uma comparação em que as vantagens e desvantagens do teste automatizado e manual são enumeradas. Nessa comparação, são considerados os aspectos como, complexidade, eficiência, eficácia, produtividade e custo. Para a maioria desses aspectos, as abordagens automatizadas funcionam melhor do que as manuais. No entanto, uma vantagem notável do teste manual sobre o automatizado se dá pelo fato de que uma pessoa executando testes manualmente é capaz de identificar comportamentos inesperados que o teste automatizado não seria capaz de identificar.

Um ponto fundamental durante a condução de atividades de teste é a utilização de ferramentas que deem suporte à automatização dessas atividades. A produtividade de atividades de teste em processos da ES está intimamente associada à disponibilidade de ferramentas adequadas (PRESSMAN, 2011). Além disso, o uso de ferramentas gera abordagens mais sistemáticas, produtivas e confiáveis.

Diversas ferramentas de teste foram desenvolvidas nas últimas décadas, objetivando apoiar a condução de atividades de teste para diferentes plataformas e linguagens de programação. A literatura apresenta uma variedade de ferramentas que podem apoiar a condução de atividades de teste automatizado.

Tabela 2 – Comparação entre teste manual e automatizado.

Teste manual	
Vantagens	Desvantagens
<ul style="list-style-type: none"> • o monitoramento humano pode observar comportamentos inesperados • notificação de inconsistências em relação a diferentes execuções de dados, estados, configurações e ambientes 	<ul style="list-style-type: none"> • a eficiência na detecção de erros deve considerar a fadiga humana • dados de teste determinados, iterações e combinações podem exigir automação • certos aspectos podem não ter soluções manuais (ou seja, análise de desempenho) • quando os testadores esperam certos erros, após algumas repetições, isso pode prejudicar a detecção de defeitos
Teste automatizado	
Vantagens	Desvantagens
<ul style="list-style-type: none"> • mais complexos e eficientes • as ferramentas podem combinar quantidade, produtividade e eficiência • possibilidade de transferência de tecnologia entre a indústria e a academia • eficiência para detecção de erros em domínios específicos 	<ul style="list-style-type: none"> • alguns domínios complexos de teste fornecem dificuldade extrema para automação • verifique apenas as condições para as quais a automação foi projetada para investigar • dificuldades em definir quais condições devem ser verificadas para considerar um teste como bem sucedido • implementações para gerar casos de teste eficientes são complexas • dificuldade em prever a forma como um SUT deve se comportar e quais as consequências esperadas após uma execução

Fonte: Adaptada de Oliveira (2017).

2.4 Oráculos de teste

O mecanismo que se utiliza para julgar a correção da saída ou o comportamento esperado da execução de um determinado programa é conhecido como “oráculo” (HOFFMAN, 2001). Oráculos verificam se os resultados obtidos a partir da execução de um SUT com determinados dados de teste estão corretos. Oráculos automatizados são extremamente importantes na redução de custos de teste e melhoram a qualidade de verificação e validação do software (OLIVEIRA; KANEWALA; NARDI, 2014).

Diferentes recursos técnicos que podem ser explorados para a automatização de oráculos: (i) modelos formais executáveis que representem o comportamento do SUT (YIN; LIU; NI, 2012); (ii) versões anteriores confiáveis do SUT que possam ser uma fonte segura acerca dos resultados de teste (YU *et al.*, 2013); (iii) assertivas previamente computadas (CHEON, 2007); (iv) documentações UML (do inglês, *Unified Modeling Language*) (BRIAND; LABICHE, 2002); (v) relações metamórficas (ASRAFI; LIU; KUO, 2011); (vi) inteligência artificial (FROUNCHI *et al.*, 2011); (vii) saídas esperadas de casos de teste previamente computados (JAVED; STROOPER; WATSON, 2007); e (viii) o próprio testador verificando saídas e comportamentos do SUT, configurando um *oráculo humano* (MCMINN; STEVENSON; HARMAN, 2010).

A variabilidade dos mecanismos de oráculos são comprovadas em Oliveira, Kanewala e Nardi (2014), por meio de um vasto mapeamento no qual mais de 300 estudos acerca de oráculos de teste são analisados e categorizados. É importante destacar que o estudo apresentado por Oliveira, Kanewala e Nardi (2014) revela que o número de publicações cujo tema central é a automatização de oráculos de teste vem crescendo ano após ano, comprovando a relevância dessa área de pesquisa. Desse modo, o design adequado de oráculos de teste é uma abordagem promissora para apoiar atividades de testes automatizados em sistemas de saídas complexas (OLIVEIRA, 2014).

2.4.1 Problema de oráculo: Falsos positivos e Falsos negativos

Um problema conhecido pelos pesquisadores e amplamente investigado dedicando esforços na área de oráculos de teste é conhecido como “Problema de Oráculo” (do inglês, *Oracle Problem*). Problema de oráculo é configurado em casos nos quais, utilizando meios práticos, é impossível ou muito difícil julgar a correção de saídas geradas a partir dos dados válidos do domínio de entrada (WEYUKER, 1982; BARR *et al.*, 2015).

A disponibilidade de oráculos de teste é vasta, embora nem sempre exista um oráculo de teste ideal. Dessa forma, frequentemente podem-se encontrar pontos do domínio de um SUT para os quais os resultados e o uso de oráculos são imprevisíveis. No contexto do oráculo, os seguintes casos podem ocorrer:

- **Falsos positivos:** o resultado do teste é considerado inválido (*fail*), entretanto o sistema funciona de acordo com suas especificações e nenhum problema aparente se manifestou para o testador; e
- **Falsos negativos:** o resultado do teste é considerado válido (*pass*), entretanto a execução não corresponde ao comportamento esperado.

Dependendo do SUT, é extremamente difícil prever resultados esperados a serem comparados com resultados reais de testes. Em situações extremas, podem-se explorar relações metamórficas para lidar com a falta de oráculos de teste (ASRAFI; LIU; KUO, 2011; CAO; ZHOU; CHEN, 2013). Assim, oráculos podem ser implementados de maneiras diferentes: com base em assertivas, casos de teste, tabelas, máquinas de estado, ou mesmo o próprio ser humano um oráculo de teste.

2.5 Sistemas com saídas complexas

Nos dias atuais, é crescente o número de sistemas com saídas complexas e aplicações cujas saídas são dadas em formatos incomuns (OLIVEIRA, 2014). Sons (OLIVEIRA *et al.*, 2014), imagens processadas (DELAMARO; NUNES; OLIVEIRA, 2013), interfaces 3D de usuário ou ambientes virtuais (PRACHYABRUED; BORST, 2014), e GUIs (BANERJEE *et al.*,

2013; BANERJEE B. NGUYEN; MEMON, 2003) são alguns dos domínios contemporâneos de saída complexa.

Domínios de saída complexa limitam, severamente, a aplicabilidade das atividades de teste de software automatizado (CHAN; TSE, 2013). Devido à complexidade associada com os dados processados pelo SUT, técnicas de testes tradicionais são ineficientes e, muitas vezes, impraticáveis (OLIVEIRA *et al.*, 2014). Assim, estratégias para a automação de testes em sistemas com saídas não-triviais são um desafio. Além disso, a maior parte das abordagens existentes são definidas para um domínio específico, tendo pouca ou nenhuma padronização. Nesse contexto, ao testar os sistemas de saídas complexas, os testadores geralmente “ignoram” etapas de testes tradicionais, como a análise de domínio de entrada do SUT para projetar um conjunto de testes eficaz, prevendo os efeitos das entradas por meio de saídas explícitas, mitigando o compartilhamento de conhecimento neste campo. A complexidade associada à saída do SUT leva ao emprego de estratégias subjetivas e *ad-hoc* de teste (HINTERLEITNER *et al.*, 2011). De um modo geral, um ser humano exercita o SUT, observando as suas saídas e preenchendo um protocolo de aspectos, tornando as atividades de teste difíceis, tediosas e custosas.

Sistemas que processem imagens digitais exemplificam um domínio de sistema cujas saídas são em formatos incomuns. Processamento Digital de Imagens (PDI) não é uma tarefa simples, envolve atividades interconectadas que vão desde a captura da imagem ou iluminação ao processo de representação da imagem (duas ou mais dimensões) (GONZALEZ; WOODS, 2000).

Imagens digitais 3D são uma extensão dos conceitos de imagem digital monocromática e multibanda para uma terceira dimensão que pode ser espaço ou tempo. A amostragem e a quantização podem ocorrer em (x,y,z) ou (x,y,t) , onde x,y,z representam o espaço e t o tempo. Assim, uma imagem 3D é representada como uma sequência de imagens monocromáticas ou multibandas ao longo do eixo espacial z ou do eixo temporal t . Assim, as imagens 3D são interpretadas como conjuntos de *voxels* (i.e., elementos da imagem com comprimento, largura e profundidade, onde o volume pode ser interpretado como uma pilha de imagens bidimensionais). A Figura 4 apresenta um cubo 3D usando o conceito de *voxels* e pontos estruturados.

Dentre as diversas aplicações da visualização 3D ou volumétrica, destaca-se a área médica. A aplicação de visualização volumétrica a dados médicos tem o objetivo de gerar imagens 3D a partir de fatias bidimensionais (2D) oriundas de tomografias, ressonância magnética, ultrassom, medicina nuclear, etc. Tal aplicação auxilia o profissional a criar uma imagem 3D dos dados, facilitando as tarefas de interpretação do diagnóstico, decisão terapêutica ou conduta cirúrgica. Um dos exemplos de sistemas que utilizam os conceitos de geração de imagens 3D são sistemas que manipulam imagens sintéticas 3D de vasos sanguíneos. A seguir são apresentados mais detalhes em relação a sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos.

Figura 4 – Cubo 3D usando o conceito de *voxels* e pontos estruturados.

Fonte: Elaborada pelo autor.

2.5.1 Imagens sintéticas tridimensionais de vasos sanguíneos

Sistemas que manipulam imagens sintéticas 3D de vasos sanguíneos são fundamentais no auxílio à área médica, contribuindo na antecipação do diagnóstico de determinadas patologias (GALARRETA-VALVERDE *et al.*, 2013). Dessa forma, é essencial que as imagens processadas tenham alto grau de confiabilidade, fornecendo detalhes da imagem, facilitando o trabalho do profissional da área médica. No entanto, sistemas que geram ou manipulam imagens sintéticas 3D não têm sua qualidade mensurada devido à falta de estratégias de teste automatizado. Por isso, é essencial o uso de ferramentas automatizadas que auxiliem o profissional de saúde no diagnóstico de possíveis anomalias ou patologias no sistema circulatório, fator cada vez mais comum em países desenvolvidos e em desenvolvimento (CHAGAS *et al.*, 2009).

Em geral, a atividade para modelagem de objetos 3D em sistemas computacionais é complexa e, em consequência do número de detalhes, várias aplicações demandam a geração de modelos geométricos ou réplicas a partir das características das superfícies externas do objeto. Nesse contexto, por exemplo, Galarreta-Valverde *et al.* (2013) apresentam uma estratégia de geração de vasos sanguíneos sintéticos de forma mais realista em 3D, realizado em três etapas: i) geração da *String* descritiva do vaso; ii) geração do vaso sintético; e iii) discretização do vaso. Na primeira etapa é gerada, por meio da gramática de entrada, uma *String* que representa a execução de uma gramática de uma dada interação. Essa *String* representa uma sequência de instruções e cada caractere está associado a uma ação. Essas ações são executadas na segunda etapa, gerando outra sequência com as propriedades, por exemplo o diâmetro do vaso. Por fim, a terceira etapa de discretização cria a imagem do vaso sintético “imitando” uma imagem angiográfica real.

A Figura 5 apresenta uma série de imagens de quatro vasos sintéticos 3D gerados pela estratégia definida em Galarreta-Valverde *et al.* (2013). As imagens são apresentadas primeiramente de modo completo (parte superior da Figura 5) e depois de modo segmentado –

sem o fundo e somente com o “mapa” vascular representado (parte inferior da Figura 5).

Figura 5 – Vasos sintéticos tridimensionais – antes e depois da segmentação

Fonte: Adaptada de Galarreta-Valverde *et al.* (2013).

Imagens como as apresentadas na Figura 5 são fundamentais para o fornecimento de um “mapa” vascular, que facilita a localização de possíveis anormalidades dos vasos e permitem a antecipação do diagnóstico de determinadas patologias. Nesse contexto, as redes vasculares sintéticas tornam-se muito importantes, haja vista que novos algoritmos de detecção de patologias vêm sendo estudados por pesquisadores e imagens artificiais que representem vasos reais são necessários. É importante destacar que, apesar de sua importância, exames de angiografia – que produzem imagens reais – devem ser evitados pelo fato de gerarem riscos como ataque cardíaco ou derrames no paciente, devido à utilização de cateteres intravasculares. Sendo assim, quanto maior a qualidade dos sistemas que gerem ou processem essas imagens, maior é o suporte aos algoritmos que contribuam para diagnósticos precisos de determinadas patologias, aliviando esforços do profissional de medicina e acelerando o início do tratamento do paciente.

Em um contexto amplo, as avaliações da qualidade de sistemas cujas saídas são imagens médicas 3D são realizadas com a participação efetiva dos humanos, limitando a produtividade. Dessa forma, é essencial o desenvolvimento de estratégias que contribuam para aliviar os esforços humanos em testar sistemas desta natureza, contribuindo para a melhora na qualidade de algoritmos e sistemas que processam ou geram imagens sintéticas 3D de vasos sanguíneos.

O escopo do presente trabalho de mestrado é a utilização da abordagem oráculos de teste baseado em características para aliviar os esforços humanos durante o processo de VV&T de sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos.

2.6 Considerações finais

Este capítulo apresentou conceitos e informações essenciais para compreensão completa da pesquisa desenvolvida. Conceitos relacionados a atividades de VV&T, automatização de teste de software, oráculos de teste e sistemas com saídas complexas foram discutidos. Em seguida, serão apresentados e discutidos os resultados de um mapeamento sistemático conduzido para identificar estratégias de teste de software e avaliação de qualidade em sistemas biomédicos, o que inclui sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos.

TESTE DE SOFTWARE E ANÁLISE DE QUALIDADE EM SISTEMAS BIOMÉDICOS: UM MAPEAMENTO SISTEMÁTICO

3.1 Considerações iniciais

Este capítulo apresenta os resultados obtidos a partir de um Mapeamento Sistemático (MS) conduzido por meio da análise de 91 estudos publicados de 1990 a 2016. O MS objetiva identificar abordagens e estratégias para o teste de software e análise de qualidade em sistemas biomédicos, sintetizando a evidência disponível e identificando os principais problemas que envolvem avaliação de qualidade nesse domínio de pesquisa. Um MS é uma vertente de uma Revisão Sistemática da Literatura (RSL), na qual realiza-se uma revisão mais ampla dos estudos primários, objetivando identificar evidências, bem como identificar lacunas direcionando o foco de revisões sistemáticas futuras e identificar áreas nas quais mais estudos primários precisam ser conduzidos (KITCHENHAM, 2004).

Na literatura, encontram-se trabalhos relacionados a diversos domínios de sistemas biomédicos. Dessa forma, um MS promove a compreensão estruturada de abordagens e estratégias para atividades de teste de software e análises de qualidade em sistemas biomédicos. A partir da identificação de evidências, é possível relatar as dificuldades e limitações identificadas na validação de sistemas biomédicos.

Devido a questões de espaço, alguns detalhes do protocolo utilizado na condução do MS foram suprimidos. Para aumentar a replicabilidade do estudo, uma planilha online foi disponibilizada na *Web* apresentando informações específicas de cada estudo obtido por meio do MS¹. Além disso, um relatório técnico pode ser apreciado na *Web* com mais detalhes do

¹ <https://goo.gl/kSHAKs>

protocolo de condução do MS e seleção dos estudos primários².

O MS foi conduzido seguindo as diretrizes propostas em estudos de especialistas em estudos secundários (KITCHENHAM, 2004; PETERSON; VAKKALANKA; KUZNIARZ, 2015; KEELE, 2007), sendo conduzido seguindo três etapas: (i) planejamento, (ii) condução e (iii) apresentação dos resultados. A seguir, tais etapas são apresentadas e a metodologia utilizada na condução do MS é descrita.

3.2 Planejamento

Na fase de planejamento é definido o protocolo do MS objetivando guiar todo o processo de pesquisa. Assim, os objetivos, Questões de Pesquisa (QPs), critérios de inclusão e exclusão, estratégia para extração dos dados e análise dos resultados, são definidos. A seguir, os principais tópicos relacionados à fase de planejamento do MS são descritos.

3.2.1 Objetivos

O objetivo geral do MS é identificar estudos na literatura que relatam ou propõem métodos, abordagens ou estratégias de teste de software e/ou análise de qualidade em sistemas biomédicos. Para atingir tal objetivo geral, são definidos os seguintes objetivos específicos:

- identificar métodos, técnicas, critérios, estratégias e abordagens de testes de software e análise de qualidade em sistemas biomédicos;
- definir uma taxonomia de estratégias de teste de software e análise de qualidade em sistemas biomédicos;
- obter uma visão geral dos procedimentos experimentais mais adequados para a validação de sistemas biomédicos;
- relatar as principais limitações e dificuldades na validação de sistemas biomédicos;
- identificar a colaboração entre academia e indústria; e
- avaliar a maturidade da área por meio de dados visuais.

3.2.2 Questões de pesquisa

Como uma estratégia de seleção e busca de estudos que se enquadram nos objetivos do MS, as QP's foram formuladas como segue.

- **QP 1:** De acordo com a literatura, existem técnicas, critérios e abordagens de teste de software e análise de qualidade que são usados em sistemas biomédicos?
 - **QP 1.1:** Existem estratégias *ad-hoc* ou informais de teste de software e análise de qualidade aplicadas em sistemas biomédicos?

² <https://goo.gl/e73dRW>

- **QP 1.2:** Existem estratégias sistematizadas e/ou automatizadas, incluindo o uso de ferramentas e *frameworks*, de teste de software e análise de qualidade aplicadas em sistemas biomédicos?
- **QP 2:** Quais procedimentos experimentais são mais utilizados e adequados para a validação de sistemas biomédicos?
 - **QP 2.1:** As validações ocorrem em sistemas biomédicos reais ou *toys*?
 - **QP 2.2:** Quais são os domínios dos sistemas biomédicos envolvidos?
- **QP 3:** Quais estratégias e abordagens, *ad-hoc* ou sistematizadas, de teste de software e análise de qualidade, são mais utilizadas para validar domínios de sistemas biomédicos?
- **QP 4:** Existe colaboração entre Indústria e Academia no desenvolvimento de estratégias de validação para sistemas biomédicos?
- **QP 5:** Quais são os desafios e limitações de validação mencionados nos estudos primários?
- **QP 6:** Qual é o impacto da pesquisa relacionada à validação de sistemas biomédicos?
 - **QP 6.1:** Qual a localização das universidades e institutos de pesquisa que mais pesquisam na área e têm o maior impacto de pesquisas relacionadas à validação de sistemas biomédicos?
 - **QP 6.2:** Quais são os meios de divulgação utilizados para disseminar as estratégias desenvolvidas para validar domínios de sistemas biomédicos?

3.2.3 Estratégia de busca e seleção

Dadas as questões de pesquisa, decidiu-se então por um processo de seleção dos estudos primários realizado em duas etapas: (i) busca automática e (ii) busca manual. A busca automatizada consiste do uso de Strings de busca em pesquisas nas bases indexadas de artigos científicos. Em complemento ao processo automatizado, a busca manual está relacionada à realização do processo de busca em referência dos estudos primários, *Snowballing* (análise de referências bibliográficas de estudos selecionados), e estudos indicados por especialistas. A seguir, as etapas para busca e seleção dos estudos primários são descritas.

- Bibliotecas de pesquisa: a Tabela 3 apresenta as bibliotecas de pesquisa utilizadas para busca de estudos primários do MS. A escolha das bibliotecas de pesquisa está relacionada à experiência dos pesquisadores envolvidos no MS;
- Idioma dos estudos: optou-se por não considerar no MS estudos que estejam escritos em idiomas não-alfanuméricos (japonês, chinês, mandarim, cantonês e etc). A escolha do idioma dos estudos está relacionada à experiência dos pesquisadores envolvidos na condução do MS;
- Palavras-chave: as palavras-chave utilizadas para a busca dos estudos primários foram elaboradas a partir dos cinco principais termos da pesquisa: critérios ou abordagens, teste de software ou análise de qualidade, e sistemas biomédicos. A Figura 6 apresenta a String de busca definida a partir das palavras-chave;
 - **Critérios:** *technique, techniques, method, criterion, strategy e approach*;

Tabela 3 – Bibliotecas de pesquisa utilizadas no MS.

Bibliotecas	Endereço
ACM Digital Library	http://dl.acm.org/
IEEE Xplore	http://ieeexplore.ieee.org/
Compendex	http://www.engineeringvillage.com/
Scopus	http://www.scopus.com/

Fonte: Elaborada pelo autor.

- **Teste de software e análise de qualidade:** “*software testing*” e “*software quality*”;
- **Sistemas biomédicos:** *biomedics*, *biomedical*, “*medical software*”, “*medical systems*” e “*medical devices*”.

Figura 6 – String de busca do MS.

(“technique” OR “techniques” OR “method” OR “criterion” OR “strategy” OR “approach”)
AND
 (“software testing” OR “software quality”)
AND
 (“biomedics” OR “biomedical” OR “medical software” OR “medical systems” OR “medical devices”)

Fonte: Elaborada pelo autor.

- Trabalhos relacionados: após a seleção dos estudos primários, foi realizada uma pesquisa nos trabalhos relacionados a estes estudos. De acordo com Kitchenham (2004), a prática do *Snowballing*, com a busca de estudos nas referências dos estudos primários, pode “revelar” mais artigos relevantes; e
- Estudos sugeridos/indicados por especialistas: estudos sugeridos/indicados por especialistas, no domínio de teste de software, foram considerados como fonte de pesquisa no MS. Embora a escolha dessa opção possa causar um possível viés, é necessário considerar os estudos sugeridos/indicados objetivando evitar a perda de estudos que possam ser boa evidência para o MS. Além disso, tais estudos foram utilizados como base para calibração da String de busca.

3.2.4 Critérios de seleção

Um passo importante na seleção dos estudos primários do MS é a definição dos Critérios de Inclusão (CI) e dos Critérios de Exclusão (CE). A definição dos critérios de seleção é uma tarefa essencial para assegurar a viabilidade da pesquisa (KITCHENHAM, 2004). Dessa forma, os seguintes critérios de seleção foram definidos:

- **Critérios de inclusão:**

- *CII*: qualquer estudo primário que aborde ou discuta (direta ou indiretamente) métodos ou abordagens de teste de software ou análise de qualidade em sistemas biomédicos;
- *CI2*: qualquer estudo primário que relate ou proponha um novo método ou abordagem, seja *ad-hoc* ou automatizada, de teste de sistemas biomédicos ou análise de qualidade;
- *CI3*: qualquer estudo primário que apresente ou discuta procedimentos experimentais mais adequados para validar sistemas biomédicos; e
- *CI4*: estudos primários cujo manuscrito esteja completamente disponível na *Web*.

- **Critérios de exclusão:**

- *CE1*: estudos que lidam ou mencionam sistemas biomédicos, porém não apresentam avaliações formais ou informais da qualidade apresentada por eles;
- *CE2*: estudos escritos em idiomas não-alfanuméricos (japônes, chinês, mandarim, cantonês e etc) ou em idiomas não conhecidos pelos autores (polonês, alemão, etc);
- *CE3*: a versão completa do manuscrito não está disponível na *Web* (após solicitar o autor) ou não foi publicada;
- *CE4*: a pesquisa é uma versão anterior de um estudo mais completo. Ex: *conference paper* que foi expandido para *journal paper*; e
- *CE5*: o estudo é uma monografia, tese, dissertação ou um trabalho de conclusão de curso.

3.3 Condução

A fase de condução ou execução de uma RSL objetiva gerar os resultados finais e artefatos intermediários, tais como: identificação das fontes de pesquisa, seleção dos trabalhos, extração dos dados e monitoramento do progresso e síntese dos dados. Assim, é a fase na qual ocorre a coleta e análise dos estudos primários, ou seja, publicações e outras fontes de estudo relacionadas às QPs (KITCHENHAM, 2004).

A Figura 7 apresenta um fluxo de trabalho que representa o processo de seleção do MS com o número de estudos resultantes em cada etapa de seleção. Inicialmente, identificaram-se 366 estudos por meio da busca automatizada nas bibliotecas digitais. Os estudos duplicados foram excluídos, resultando em 225 estudos. Os critérios de seleção foram aplicados nos estudos por meio das análises de Título, Resumo e palavras-chave, resultando em 139 estudos. Em seguida, os critérios de seleção foram aplicados durante a leitura completa dos estudos, resultando em 82 estudos selecionados a partir da busca automatizada.

Após a busca automatizada, efetuou-se a busca manual de estudos por meio do processo conhecido como *Snowballing*, que consiste da análise de referências dos estudos incluídos. Adicionalmente a esse processo, incluíram-se alguns estudos indicados por especialistas. Nove estudos foram identificados por meio do *Snowballing* e quatro estudos foram indicados por especialistas em teste de software. Os critérios de seleção foram, então aplicados durante a leitura parcial e completa de tais estudos. Por fim, um catálogo de 91 estudos primários compuseram o MS.

Figura 7 – Fluxo de trabalho para identificação de estudos do MS.

Fonte: Elaborada pelo autor.

3.4 Análise dos resultados

Por meio dos dados coletados, fase de análise e síntese de dados, permitiu-se uma representação das informações extraídas dos estudos primários com análises qualitativas e quantitativas. A maioria dos trabalhos selecionados são recentes, mostrando que há interesse de pesquisa nesse domínio de pesquisa. A Figura 8 mostra o interesse na área de 1990 a 2016 por meio de um gráfico com o número de estudos publicados a cada ano. Esses aspectos quantitativos mostram uma evolução considerável de pesquisas relacionadas às abordagens de validação em sistemas biomédicos.

Figura 8 – Análise de publicações por ano entre 1990 e 2016.

Fonte: Elaborada pelo autor.

A seguir, são apresentadas as respostas para cada QP do MS com base em dados quantitativos e análises subjetivas.

QP 1: De acordo com a literatura, existem técnicas, critérios e abordagens de teste de software e análise de qualidade que são usados em sistemas biomédicos?

Identificaram-se estratégias e abordagens utilizadas para validar domínios de sistemas biomédicos. As estratégias de validação identificadas foram classificadas em categorias de acordo com o processo utilizado para validar o domínio em questão: (i) *ad-hoc* e informal ou (ii) sistemática e automatizada. Estratégias *ad-hoc* e informais referem-se a estratégias de validação manual, em que o processo não requer formas sistemáticas de validação e não requer o uso de ferramentas que automatizam o processo de validação. Estratégias sistemáticas e automatizadas referem-se à aplicação de processos sistemáticos no desempenho da atividade de teste de software ou análise de qualidade na validação de um sistema. Geralmente, abordagens sistemáticas usam ferramentas que automatizam a validação do sistema.

Entre os estudos primários identificados, 98% (89/91) aplicam ou discutem estratégias de validação que utilizam abordagens manuais ou automatizadas. Além disso, 2% (2/91) dos estudos primários discutem apenas o estado da arte do domínio de pesquisa. Essa questão principal visa discutir as duas estratégias de validação, analisando os procedimentos aplicados no processo de validação no campo de pesquisa de sistemas biomédicos. Em seguida, evidências sobre as QPs específicas dessa QP geral são discutidas.

QP 1.1: Existem estratégias *ad-hoc* ou informais de teste de software e análise de qualidade aplicadas em sistemas biomédicos?

19% (18/91) dos estudos primários realizam processos informais ou *ad-hoc* para a condução de atividade de teste de software ou análise de qualidade de um domínio de sistema biomédico. Uma análise realizada nos estudos que utilizam abordagens *ad-hoc* evidenciou o uso de modelos de desenvolvimento e práticas para garantir a qualidade no desenvolvimento de sistemas biomédicos. A Tabela 4 apresenta algumas dessas abordagens identificadas nos estudos primários.

Tabela 4 – Validação *ad-hoc* e não-sistemática em sistemas biomédicos: abordagens, ferramentas e modelos.

ID	Abordagem	Domínio do sistema
1	MIL-STD-498 e linguagem de programação Ada	Software médico em geral
2	Processo de Planejamento de produto (PPP)	
3	Padrões ISO 13485 e FDA CFR 820	
4	Análise de falhas em sistemas médicos	
5	Padrão IEC 62304	
6	Práticas de Medicina Familiar (FMF)	
7	Qualidade, Prazo e Confiabilidade (QPC)	
8	Med-Adept	
9	Modelo de qualidade de segurança	
10	Qualidade do equipamento de diagnóstico radiológico	Equipamento de diagnóstico radiológico
11	Engenharia de requisitos de qualidade orientada para o mal uso (ERQOMU)	Sistema de aconselhamento sobre drogas
12	Ferramenta de Pesquisa Médica (FPM)	Equipamento de eletrocardiograma
13	ISOgrayTM	Sistemas de simulação virtual
14	OpenDMP	Aplicações em mineração de texto
15	Formulário Web	Software biomédico <i>open-source</i>

Fonte: Elaborada pelo autor.

Com base nas abordagens apresentadas na Tabela 4, é possível notar uma tendência para proposições de abordagens para o software médico em geral, em detrimento a sistemas de software específicos. Por outro lado, para sistemas que processam imagens, simulações virtuais e outras aplicações de resultados complexos, notaram-se abordagens desenvolvidas especificamente para tais domínios.

QP 1.2: Existem estratégias sistematizadas e/ou automatizadas, incluindo o uso de ferramentas e *frameworks*, de teste de software e análise de qualidade aplicadas em sistemas biomédicos?

As estratégias sistemáticas e automatizadas se referem a um conjunto de atividades que visam à validação de um domínio específico de sistema biomédico com o uso de ferramentas ou *frameworks* que automatizem o processo de validação. As ferramentas e *frameworks* são essenciais para garantir qualidade e a produtividade no processo de validação. Assim, 62% (57/91) dos estudos aplicam procedimentos sistemáticos e automatizados durante a validação de um domínio específico do sistema biomédico. Uma análise realizada em estudos que mencionam ou realizam processos de validação sistemáticos em sistemas biomédicos propiciou a categorização das abordagens e estratégias.

As ferramentas e *frameworks* identificados na validação dos sistemas biomédicos tinham como principal objetivo automatizar a atividade de validação do sistema em análise, oferecendo mais produtividade e eficiência. Nesse cenário, foi possível evidenciar o desenvolvimento de ferramentas e *frameworks* para validar, especificamente, domínios de sistemas biomédicos. Além disso, para determinados domínios de sistemas biomédicos, sistemas que processem ou gerem imagens médicas, a geração de dados de teste são mais complexas. Desse modo, o uso de ferramentas para validar tais sistemas é essencial. A Tabela 5 apresenta a abordagem utilizada na validação do respectivo domínio de sistema biomédico. Tais abordagens incluem o uso de ferramentas e *frameworks*, objetivando a automatização e eficácia na condução da validação do respectivo domínio de sistema biomédico.

Tabela 5 – Validação sistemática e automatizada em sistemas biomédicos: ferramentas, software e *frameworks*.

ID	Abordagem	Domínio do sistema
1	DEVSimPy Environment	Sistemas GUI
2	Sikuli	
3	Robot Framework	
4	O-FIm (Oracle for Images)	
5	NeuroSista	Imagens médicas
6	Projeto Humano Visível	
7	<i>Medical Imaging Interaction Toolkit</i> (MITK)	
8	Cardiac T2*	
9	QUALIMAGIQ	
10	Visualization Toolkit (VTK)	Sistemas médicos embarcados
11	CBMC model checker	
12	SATABS Model Checker	
13	NuSMV2	
14	Device Coordination Framework (DCF)	
15	HYRES	
16	SATs	Banco de dados biomédicos
17	Dynamic Symbolic Execution (DSE)	
18	Mock Object Based Test Generation for Data-base Applications (MODA)	
19	EvIdent	Software biomédico
20	Labview Environment	
21	OntoReTest	
22	BedMaster	
23	Schedulable Online Testing (SOTF)	
24	MuJava	
25	QuickCheck	Exames de ultrassonografia
26	Control Quality for Ultrasound B-mode equipment (CQUS)	
27	GPU-Based Framework	
28	JMP Statistical Discovery Software	Dispositivos médicos
29	OntoReTest	
30	NUnit	Processamento de Linguagem Natural (PLN)
31	Turf - EHR Usability Toolkit	
32	NIH Image	Exames de raio-x
33	MATLAB	Electrocardiogram
34	The Generic PCA Reference Model	Bomba de Infusão

Fonte: Elaborada pelo autor.

QP 2: Quais procedimentos experimentais são mais utilizados e adequados para a validação de sistemas biomédicos?

Realizou-se uma análise para verificar os principais procedimentos experimentais aplicados no processo de validação de domínios de sistemas biomédicos. Para tanto, observou-se a natureza do estudo, prático ou teórico, o tipo de sistema biomédico, sistema real ou *toy*, e o domínio de sistema biomédico a ser validado. Inicialmente, foi realizada uma análise para verificar o número de estudos práticos comparados a estudos teóricos. Estudos práticos apresentam alguma atividade prática durante o processo de validação. Tais atividades práticas, envolvem a condução de um estudo de caso demonstrando a validação de domínio de sistema biomédico em específico. Estudos teóricos descrevem uma área ou tópico de pesquisa, mas não demonstram a condução de, por exemplo, um estudo de caso. Os autores consideraram estudos de caso (44/91), experiências controladas e análises empíricas em geral (45/91) durante essa análise. A Figura 9 apresenta um gráfico de pizza com uma comparação do número de estudos práticos e teóricos incluídos no MS.

Figura 9 – Estudos práticos vs estudos teóricos.

Fonte: Elaborada pelo autor.

Foi analisada a origem dos domínios de sistemas biomédicos identificados – reais ou *toys* – e suas respectivas predominâncias em pesquisas da área biomédica. Além disso, os autores apresentam uma análise comparativa dos domínios dos sistemas biomédicos de 1990 a 2016.

QP 2.1: As validações ocorrem em sistemas biomédicos reais ou *toys*?

Identificaram-se 74 estudos que apresentam estratégias de validação com programas reais ou *toy* explicitamente. 95% (70/74) usam programas reais, 4% (3/74) exploram programas *toy* e 1% (1/74) dos estudos usam programas *toy* e reais. A Figura 10 ilustra a comparação entre o impacto do uso de programas *toy* e reais. Além disso, a Figura 11 ilustra uma análise que diferencia o número de estudos com programas *toy* e reais, ano a ano.

Figura 10 – Programas *toy* versus programas reais.

Fonte: Elaborada pelo autor.

Figura 11 – Programas *toy* versus programas reais: análise ano por ano.

Fonte: Elaborada pelo autor.

QP 2.2: Quais são os domínios dos sistemas biomédicos envolvidos?

Realizou-se uma análise com o objetivo de verificar os principais domínios de sistemas biomédicos utilizados por pesquisadores e profissionais no processo de validação. A Tabela 6 mostra uma categorização dos domínios dos sistemas biomédicos. Os sistemas foram classificados em 19 categorias distintas: (1) sistemas que processam imagens médicas; (2) sistemas de Processamento de Linguagem Natural (PLN); (3) sistemas de diagnóstico de eletroencefalograma (EEG); (4) sistemas urológicos; (5) dispositivos de detecção de Respiração Cheyne-Stokes (RCS); (6) olfatômetro contínuo de sistemas de monitorização respiratória; (7) dispositivo clínico de detecção de neuropatias diabéticas; (8) Sistemas de Informação Hospitalar (SIH); (9) sistema de monitoramento remoto; (10) sistemas de aconselhamento de drogas; (11) dispositivo

de marcapasso; (12) sistemas de análise de dados biomédicos; (13) bombas de infusão; (14) sistemas de telemedicina; (15) dispositivo de assistência ventricular; (16) sistemas de código aberto biomédico; (17) Sistemas de Computação Móvel (SMDCSes); (18) sistemas de exame de eletrocardiograma; e (19) propostas gerais.

Tabela 6 – Número de domínio de sistema por categoria.

ID	Domínio do sistema	#	%
(1)	Imagens médicas (ultra-som, radiologia, radiografia, ressonância magnética, tomografia, raios-x, sistemas CAD)	22	24
(2)	Processamento de Linguagem Natural (PLN)	3	3,2
(3)	Sistemas de diagnóstico de eletroencefalograma (EEG)	1	1
(4)	Sistema urológico	1	1
(5)	Deteção de respiração de Cheyne-Stokes (CSB)	1	1
(6)	Sistema Contínuo de Monitoramento da Respiração do Olfatômetro (CRO)	1	
(7)	Deteção de Neuropatia Clínica por Diabética	1	1
(8)	Sistema de Informação Hospitalar (SIH)	1	
(9)	Sistema de monitoramento remoto	1	1
(10)	Sistemas de consultoria de drogas	1	1
(11)	Dispositivo de marcapasso	5	5,5
(12)	Sistemas de análise de dados biomédicos	3	3,2
(13)	Bombas de infusão	5	5,5
(14)	Sistema de telemedicina	1	1
(15)	Dispositivo de assistência Ventricular	1	1
(16)	Sistemas biomédicos <i>open-source</i>	1	1
(17)	Sistemas médicos de computação móvel (SMDCSes)	1	1
(18)	Sistemas de exame de eletrocardiograma	2	2,1
(19)	Propostas gerais	39	42,8

Fonte: Elaborada pelo autor.

Considerando o escopo do presente trabalho de mestrado, foram avaliados os processos de validação, verificação e teste de sistemas que manipulam imagens sintéticas tridimensionais de redes vasculares. Dentre o número de domínios de sistemas que processam imagens médicas (22/91), apenas um estudo relata alguma abordagem de validação em imagens de redes vasculares. D’Errico *et al.* (2008) apresentam um método eficiente para análise de imagens intravasculares objetivando capturar mecanismos complexos e mudanças relacionadas ao diagnóstico precoce de CAV (do inglês, *Cardiac Allograft Vasculopathy*) na geometria coronária. Para alcançar tal objetivo, os autores compararam medidas obtidas por meio de duas técnicas e avaliaram a variabilidade interobservador da técnica do software, repetindo, aleatoriamente, remotamente. Concluiu-se que uma análise de software com base na detecção semi-automatizada de bordas da imagem proporcionou uma melhor precisão e repetibilidade de medições em relação a um método manual.

QP 3: Quais estratégias e abordagens, *ad-hoc* ou sistematizadas, de teste de software e análise de qualidade são mais usadas para validar domínios de sistemas biomédicos?

Os estudos selecionados foram classificados em duas categorias principais, de acordo com a estratégia de validação discutida: (i) análise de qualidade; e (ii) teste de software. A análise

de qualidade refere-se a estudos que focam a abordagem utilizada para avaliar a qualidade do sistema biomédico sem utilizar abordagens específicas da área de teste de software. Além disso, nessa categoria, é predominante o uso de padrões de qualidade na validação do respectivo domínio de sistema biomédico.

Por outro lado, a categoria de teste de software está relacionada a estudos que utilizam técnicas, critérios e estratégias de teste de software fundamentados na ES. Observou-se uma predominância do uso de abordagens que avalia o uso do sistema por meio do usuário (Teste de usabilidade), estratégias que verificam a saída do sistema com base na especificação (Teste funcional), estratégias que avaliam a cobertura da implementação do sistema por meio do código-fonte (Teste estrutural), estratégias que avaliam a qualidade do sistema durante a operação (Teste de operação), estratégias que validam o sistema com base na especificação formal do software (Teste formal), etc. A Tabela 7 mostra as diferentes categorias com uma descrição que distingue cada abordagem.

Tabela 7 – Categorias de abordagens de análise de qualidade e teste de software.

ID	Categoria	Descrição
P1	Análise de qualidade	Estratégias que envolvem garantia de qualidade por meio de metodologias de desenvolvimento de software, padrões de qualidade, etc.
P2	Teste funcional	Estratégias que avaliam o comportamento do sistema de acordo com as especificações.
P3	Teste estrutural	Estratégias que avaliam o software através do código-fonte.
P4	Teste baseado em defeitos	Estratégias que alteram trechos no código fonte para comparar com o programa original.
P5	Teste formal	Estratégias que avaliam o sistema de acordo com a especificação formal.
P6	Teste de usabilidade	Estratégias que avaliam o comportamento do sistema através do usuário.
P7	Teste de unidade	Avaliações em módulos do sistema.
P8	Teste de integração	Estratégia em que os módulos do sistema são testados em grupos.
P9	Teste de regressão	Estratégias que realizam os testes após correções.

Fonte: Elaborada pelo autor.

A análise e a classificação realizadas durante a leitura dos estudos primários permitiram a identificação de nove abordagens distintas de validação: (i) abordagens que avaliam a qualidade do software (37/91), (ii) abordagens que utilizam testes funcionais (41/91), (iii) abordagens que utilizam testes estruturais (8/91), (iv) abordagens que utilizam conceitos de teste baseado em erro (4/91), (v) abordagens que utilizam a especificação formal ou o modelo formal do software (10/91), (vi) teste de usabilidade (11/91), (vii) testes de unidade (13/91), (viii) testes de integração (5/91) e (ix) teste de regressão (3/91). Importante ressaltar que as abordagens que avaliam a qualidade do software demonstram ou propõem o uso de padrões de qualidade e metodologias de desenvolvimento de software como uma solução de garantia de qualidade. Além disso, nessa categoria, os autores não utilizam técnicas e estratégias da área de teste de software.

Com base na Tabela 7 as abordagens utilizadas em cada estudo primário foram identificadas e analisadas. A Figura 12 apresenta o número de abordagens identificadas nos estudos primários.

Figura 12 – Bar chart de abordagens de validação de sistemas biomédicos.

Fonte: Elaborada pelo autor.

A Tabela 7 apresenta as diferentes categorias de abordagens de validação identificadas durante a análise dos estudos primários do MS. A partir da definição dessas abordagens, definiu-se uma taxonomia que é essencial para apresentar as abordagens de análise de qualidade e teste de software prevalentes em diferentes domínios de sistemas biomédicos. A Figura 13 apresenta a taxonomia definida com base na análise dos estudos primários.

Figura 13 – Taxonomia de abordagens de validação de sistemas biomédicos.

Fonte: Elaborada pelo autor.

Durante a análise dos estudos primários, observou-se que alguns estudos não mencionavam claramente a estratégia, o critério ou a abordagem do teste de software ou análise de qualidade utilizada. Assim, identificaram-se as características da abordagem ou cenário de utilização da estratégia de validação com base nas diretrizes da ES. A definição da taxonomia foi realizada identificando a abordagem utilizada na validação do domínio do sistema biomédico e o contexto do processo de validação utilizado.

QP 4: Existe colaboração entre Indústria e Academia no desenvolvimento de estratégias de validação para sistemas biomédicos?

Uma das análises realizadas no MS foi a avaliação da participação da indústria e academia no desenvolvimento de abordagens de validação em sistemas biomédicos. Classificaram-se os estudos em três categorias: (i) Academia; (ii) Indústria; e (iii) Colaboração. Estudos com origem na Academia são aqueles cujos autores têm alguma filiação com universidades. Por outro lado, estudos classificados como provenientes da Indústria, fazem referência a institutos de investigação e pesquisa. Além disso, em alguns estudos, um ou mais autores tinham uma cooperação entre a academia e a indústria.

Durante essa análise, os autores identificaram que 81% (74/91) dos estudos primários são de origem da Academia, 7% (6/91) envolvem colaboração entre Academia e Indústria, e 12% (11/91) são de origem da indústria. A Figura 14 compara as três classificações por meio de um gráfico de pizza. Além disso, um gráfico de linha com uma comparação ano a ano entre as três classificações é apresentado na Figura 15. De acordo com as análises, há uma maior proporção de estudos provenientes da academia, principalmente entre 2009 e 2016. Por sua vez, os estudos da Indústria têm uma maior proporção nos últimos anos. Finalmente, estudos que envolvem colaboração entre Academia e Indústria ainda são pouco explorados, identificados apenas entre 2009 e 2015.

Figura 14 – Origem de abordagens de validação em sistemas biomédicos: academia, indústria e colaboração.

Fonte: Elaborada pelo autor.

Figura 15 – Academia vs Indústria: análise ano por ano.

Fonte: Elaborada pelo autor.

QP 5: Quais são os desafios e limitações de validação mencionados nos estudos?

Identificaram-se características relacionadas a desafios e limitações na validação de domínios de sistemas biomédicos. Nesse contexto, os autores analisaram as características mencionadas nos estudos primários, verificando as mais frequentes no processo de validação. É essencial identificar as características como uma oportunidade de avaliar os desafios de validação nessa área de pesquisa.

As seguintes características expressadas como desafios foram identificadas: confiabilidade (C1); segurança (C2); eficácia do sistema (C3); defeitos de software (C4); dificuldades no campo de aplicação (C5); esforço humano (C6); exposição ao paciente (C7); gerenciamento de requisitos (C8); gerenciamento de projeto (C9); atividades de V&V (C10); dificuldade de operação do sistema (C11); usabilidade (C12); integração do sistema (C13); interoperabilidade (C14); adaptabilidade (C15). A Figura 16 ilustra as características citadas por meio de um gráfico de barras.

A característica de *confiabilidade e segurança*, são características que fazem referência a atividades que priorizam manter a qualidade dos sistemas em análise e manter a integridade dos dados após a execução do sistema. A característica de *eficácia do sistema* faz referência à capacidade do sistema em produzir o esperado com eficiência. A característica de *defeitos de software* é citada nos estudos como um desafio na realização das atividades de validação em explorar e notificar defeitos no software. A característica de *dificuldades no campo de aplicação* faz referência aos desafios e limitações a serem considerados no domínio de aplicação do sistema em análise. *Esforço humano* é uma característica citada para fazer referência à participação do ser humano como uma ferramenta de validação do sistema em análise. A característica de *exposição ao paciente* faz referência ao risco em expor o paciente a determinados domínios

de sistemas biomédicos. As características de *gerenciamento de requisitos* e *gerenciamento de projeto* são características citadas nos estudos para evidenciar a importância em melhor produzir o software do sistema em análise. A característica de *atividades de V&V* fazem referência à importância na aplicação dessas atividades em sistemas da área médica. Por fim, *dificuldades de operação do sistema*, *usabilidade*, *integração do sistema*, *interoperabilidade* e *adaptabilidade*, são características citadas com uma análise das limitações e desafios dos sistemas após a sua execução.

Figura 16 – Características relacionadas à análise de qualidade e teste de software em sistemas biomédicos.

Fonte: Elaborada pelo autor.

QP 6: Qual é o impacto da pesquisa relacionada à validação de sistemas biomédicos?

Para essa QP, avaliaram-se os meios utilizados para disseminar a pesquisas que reportam estratégias de validação para sistemas biomédicos. Nesse contexto, os autores avaliaram e discutiram o impacto desses estudos em dois aspectos: (i) países onde os estudos foram realizados; e (ii) os meios utilizados para disseminar os resultados. A seguir, os dois aspectos são discutidos por meio de QPs mais específicas.

QP 6.1: Quais regiões têm o maior impacto de pesquisas relacionadas à validação de sistemas biomédicos?

Durante a condução do MS, foram identificados os países com maior incidência de estudos relacionados à validação de sistemas biomédicos. Para responder essa QP, foi conduzida uma análise demográfica considerando a filiação do primeiro autor do estudo primário com universidades/faculdades e empresas privadas. É importante observar que os autores identificaram o país onde o estudo foi realizado, não necessariamente a origem do primeiro autor.

A Tabela 8 apresenta os dados coletados por esta análise. Além disso, a Figura 17 apresenta um *map chart* ilustrando os países com estudos publicados relacionados à validação

de sistemas biomédicos.

Figura 17 – Map chart de países com impacto na pesquisa.

Fonte: Elaborada pelo autor.

Tabela 8 – Análise da divulgação de estudos por país.

País	#	%
USA	28	30,7
CAN	7	7,7
UK	7	7,7
FRA	6	6,6
AUS	5	5,5
CHN	5	5,5
BRA	4	4,4
GER	4	4,4
KOR	3	3,3
IND	3	3,3
THA	2	2,2
ITA	2	2,2
IRI	2	2,2
ESP	2	2,2
Outros	11	12

Fonte: Elaborada pelo autor.

QP 6.2: Quais são os meios utilizados para disseminar as estratégias desenvolvidas para validar domínios de sistemas biomédicos?

Essa análise tem como objetivo verificar os meios utilizados pelos pesquisadores na disseminação de estudos que relatam estratégias de validação de sistemas biomédicos. Foi realizada uma análise nos canais de divulgação de cada estudo primário. Como resultado, verificou-se que o número de estudos publicados em periódicos 32% (29/91) é duas vezes

menor do que os estudos publicados em conferências 68% (62/91). A Figura 18 apresenta uma comparação entre os estudos publicados em periódicos e conferências.

Figura 18 – Publicações de abordagens de validação em sistemas biomédicos: periódicos vs conferência.

Fonte: Elaborada pelo autor.

Com relação aos estudos publicados em periódicos, os autores identificaram que as três fontes mais utilizadas para disseminação das abordagens de teste de software e análise de qualidade são: *Journal Computer*, *Ultrasound in Medicine and Biology* e *Journal Computer Methods and Programs in Biomedicine*. Nos três periódicos citados, foram identificados 20% (6/29) dos estudos, enquanto que 80 % (23/29) dos estudos foram identificados em outros periódicos. A Tabela 9 apresenta a distribuição dos estudos por período.

Tabela 9 – Número de estudos publicados por periódico.

Periódico	#	%
Computer Methods and Programs in Biomedicine	2	7
Ultrasound in Medicine and Biology	2	7
Journal Computer Methods and Programs in Biomedicine	2	7
Outros	23	80

Fonte: Elaborada pelo autor.

Considerando os estudos publicados em conferências, identificou-se um amplo número de conferências em ES, teste de software e informática médica. As principais conferências utilizadas para divulgar estudos sobre garantia de qualidade e estratégias de teste de software aplicadas em sistemas biomédicos foram: CBMS (*Computer-Based Medical Systems*); BMEiCON (*Biomedical Engineering International Conference*); QR2MSE (*Quality, Reliability, Risk, Maintenance, and Safety Engineering*); ICSE (*International Conference on Software Engineering*); ICBE (*International Conference on Bioinformatics and Biomedical Engineering*); CinC (*Computers in Cardiology*); MIC (*Medical Imaging Conference*); ASS (*Advances in Systems Safety*); ICBBE (*International Conference on Bioinformatics and Biomedical Engineering*); e BMEI (*Biomedical Engineering and Informatics*). A Tabela 10 resume a distribuição de estudos nos procedimentos da conferência.

Tabela 10 – Número de estudos publicado por conferência.

Sigla	Conferência	#	%
CBMS	Computer-Based Medical Systems	4	6
BMEiCON	Biomedical Engineering International Conference	3	5
QR2MSE	Quality, Reliability, Risk, Maintenance, and Safety Engineering	3	5
ICSE	International Conference on Software Engineering	3	5
ICBE	International Conference on Bioinformatics and Biomedical Engineering	2	3
CinC	Computers in Cardiology	2	3
MIC	Medical Imaging Conference	2	3
ASS	Advances in Systems Safety	2	3
iCBBE	International Conference on Bioinformatics and Biomedical Engineering	2	3
BMEI	Biomedical Engineering and Informatics	2	3
Outros		37	59

Fonte: Elaborada pelo autor.

3.5 Validação de imagens médicas

O MS evidenciou abordagens de validação automatizadas em sistemas biomédicos que geram ou manipulam imagens médicas. 23% (21/91) dos estudos apresentam abordagens para sistemas que geram ou processam imagens médicas como, por exemplo, exames de ultra-sonografia, imagens radiológicas, imagens radiográficas, etc. Tal predominância, demonstra um grande interesse da literatura na validação de tais sistemas. Além disso, 85% (18/21) desses estudos discutem a complexidade do processo de validação nesse domínio de sistema, demonstrando a carência de processos que validem as saídas produzidas.

Filho *et al.* (2014) e Gibson, Duddley e Griffith (2001) avaliam a qualidade de imagens médicas por meio de abordagens automatizadas, demonstrando que tais abordagens são mais eficientes em comparação ao uso de métodos manuais. Filho *et al.* (2014) propõe automatizar o teste de CQ (Controle de Qualidade) em sistemas de diagnóstico por ultrassom, combinando esforços e erros humanos, por meio do desenvolvimento de uma ferramenta semi-automatizada. O tempo necessário para realizar teste de CQ em sistemas de diagnóstico por ultrassom é um fator crucial para um hospital que realiza uma grande quantidade de exames. Gibson, Duddley e Griffith (2001) apresentam um conjunto de programas para análise de imagens de exames de ultrassom para estimar uma série de parâmetros de desempenho, como resolução e contraste. Um estudo preliminar sobre o desempenho do método mostra que a reprodutibilidade das medidas é superior aos métodos manuais e visuais. Assim, a abordagem apresentada pelos autores demonstrou alta precisão em relação aos métodos realizados pelos humanos.

Oliveira *et al.* (2014) e Oliveira, Delamaro e Nunes (2009) apresentam uma estrutura extensível para apoiar o desenvolvimento de oráculos de teste para programas não testáveis com resultados complexos. A estrutura adapta os conceitos de CBIR para permitir que os testadores especifiquem os oráculos de teste. O *framework* O-FIm/CO combina conceitos de extração de

recursos de sinal, funções de similaridade e comparações de objetos para obter um programa Java que compara dois objetos, respondendo o quanto eles são semelhantes, de acordo com um limiar de proximidade.

3.6 Considerações finais

Este capítulo apresentou os estudos selecionados por meio de um MS realizado com o objetivo de identificar abordagens de análise de qualidade e teste de software de sistemas biomédicos. É importante ressaltar que a pesquisa conduzida e reportada na presente dissertação é inédita e não foram encontrados trabalhos similares, em que abordagens de teste de software sejam aplicadas em sistemas que processem imagens médicas 3D de vasos sanguíneos.

Por meio da condução do MS, 91 estudos foram identificados na área de teste de software e análise de qualidade em sistemas biomédicos. A partir da definição das QPs, análises quantitativas e qualitativas foram realizadas com o objetivo de sintetizar os principais tópicos de pesquisa explorados na validação de sistemas biomédicos. Como resultado principal, o MS demonstrou que o desenvolvimento de estratégias de validação para sistemas biomédicos ocorreu com mais frequência recentemente, demonstrando interesse da literatura neste domínio de pesquisa. Além disso, conceitos da área de ES ainda não estão claramente disseminados neste domínio de pesquisa. Com relação a sistemas biomédicos que processam ou geram imagens médicas, 23% (21/91) dos estudos primários apresentam abordagens para este domínio de sistema, por exemplo, exames de ultra-sonografia, imagens radiológicas, imagens radiográficas, etc. Tal predominância demonstra um grande interesse da literatura em propor abordagens de validação para sistemas biomédicos nesse domínio de pesquisa.

Os resultados do MS visam disseminar a maturidade das estratégias de validação neste domínio de pesquisa. As limitações e dificuldades identificadas, oferecem lacunas de pesquisa que podem ser exploradas pelos pesquisadores. Além disso, o MS estabelece uma base de comparação para futuras abordagens de validação em sistemas biomédicos, contribuindo de forma sólida para essa área de pesquisa.

O próximo capítulo apresenta definições e conceitos relacionados à abordagem de oráculos de teste baseados em características. Além disso, o próximo capítulo apresenta e discute adaptações e extensões do *framework* O-FIm/CO, e um catálogo de *plug-ins* que representam extratores de características de imagens médicas 3D.

ORÁCULOS DE TESTE BASEADOS EM CARACTERÍSTICAS PARA IMAGENS MÉDICAS 3D DE VASOS SANGUÍNEOS

4.1 Considerações iniciais

Este capítulo apresenta uma nova abordagem de oráculos de teste para auxiliar na condução de atividades sistemáticas de teste de software para sistemas que processem imagens sintéticas 3D de vasos sanguíneos. Além disso, detalhes em relação às adaptações e extensões do *framework* são apresentados e discutidos.

4.2 Definições da abordagem

Esta dissertação explora e avalia uma estratégia chamada oráculos de teste baseados em características (OLIVEIRA; DELAMARO; NUNES, 2010; DELAMARO; NUNES; OLIVEIRA, 2013; OLIVEIRA *et al.*, 2014; BERGAMASCO *et al.*, 2015). Oráculos de teste baseados em características exploram características extraídas de uma saída confiável indicada pelo testador como referência. Os valores das características extraídas são usadas na abordagem como fonte de informação para avaliar as saídas SUT. Então, ao lidar com a ausência de um resultado esperado que atende a todas as saídas do SUT, os testadores podem usar algum resultado confiável (modelado manualmente ou derivado a partir de um recurso alternativo) para automatizar as atividades de teste por meio de oráculos de teste (OLIVEIRA *et al.*, 2014). A partir da definição de uma referência por meio de valores das características confiáveis, os oráculos de teste baseados em características fornecem recursos que tornam possível a avaliação da diferença entre a saída do SUT e da referência.

Oráculos baseados em extração de características utilizam funções de similaridade

ou funções de distância para medir a diferença entre dois objetos em teste. Uma função de similaridade (por exemplo, Distância Euclidiana – DE) mostra, computacionalmente, por meio de um valor, o quanto dois objetos são semelhantes entre si. Valores de distância mais próximos do valor um (1) significam que os objetos são considerados menos similares. Por outro lado, distâncias mais próximas de zero (0), representam a máxima semelhança entre os dois objetos em comparação.

Oráculos de teste baseados em características são compostos de dois elementos básicos: (i) extração de características e (ii) comparações de objetos. De acordo com Oliveira (2017), para que oráculos de teste baseados em características possam ser projetados, é necessário que sejam explicitados os seguintes elementos: (i) o grupo de características relevantes para um determinado teste; (ii) a função de similaridade para medir a distância entre as saídas SUT e as referências predefinidas; e (iii) o valor máximo de distância aceitável entre a referência e as saídas SUT. Então, para fornecer um veredicto final (aprovação/falha), um oráculo de teste baseado em característica requer um valor de limiar que represente o máximo do valor de diferença entre as saídas de SUTs e as referências, considerando seus valores de recursos.

Estudos anteriores (OLIVEIRA; DELAMARO; NUNES, 2009; OLIVEIRA; DELAMARO; NUNES, 2010; DELAMARO; NUNES; OLIVEIRA, 2013; OLIVEIRA *et al.*, 2014) apontam que oráculos de teste baseados em características são recomendados nos seguintes casos: (i) quando as saídas SUT são complexas e não existe um método padrão para fornecer uma comparação adequada das saídas SUT com algum objeto semelhante; e (ii) quando é difícil definir uma saída esperada para a saída do SUT.

A abordagem utiliza conceitos de CBIR como uma estratégia eficaz para automatizar oráculos de teste. Para demonstrar amplamente os conceitos da abordagem explorada no presente trabalho, a Seção 4.2.1 apresenta o conceito de CBIR.

4.2.1 CBIR

O crescimento do número de dispositivos de aquisição de imagens e objetos em meios digitais, tanto para uso pessoal quanto equipamentos de uso profissional, levou à necessidade da criação de algoritmos eficientes de recuperação, indexação e técnicas de aquisição (GATO; NUNES; SCHIABEL, 2004). As imagens digitais, entretanto, constituem dados complexos cujo conteúdo pode ser interpretado de diversas maneiras. Nesse cenário, diversas áreas da ciência foram amplamente beneficiadas por tal evolução, com destaque especial para a área das ciências médicas. Por exemplo, em sistemas de auxílio a diagnósticos, conhecidos como sistemas CAD, o médico compõe um diagnóstico levando em consideração o resultado de processamento de um sistema computadorizado que executa baseando-se em processamentos e análises de imagens ou até em dados clínicos de pacientes (GATO; NUNES; SCHIABEL, 2004).

O CBIR é uma técnica oriunda da área de processamento de imagem e reconhecimento

de padrões que permite que um sistema recupere um conjunto de imagens de uma base, tendo como chave de busca outra imagem como referência (DATTA *et al.*, 2008). Sistemas CBIR permitem a recuperação de um conjunto finito de imagens similares a uma imagem de consulta (RORISSA, 2007). A similaridade das imagens é obtida por meio da comparação entre vetores de características extraídos de cada uma das imagens. As principais características que podem ser extraídas de uma imagem divididas em três grupos: cor, textura e forma. A comparação entre vetores de características é feita usando funções de similaridade, as quais, no formato mais simples, empregam funções de distância tal como a DE. Um sistema CBIR é composto pelos componentes especificados a seguir.

- Extratores de características: algoritmos que extraem características de imagens baseados, principalmente, em cor, forma e textura;
- Vetores de características: formados pelo conjunto de características extraídas da imagem. Representando, de forma numérica, uma imagem ou parte dela;
- Funções de similaridade: método capaz de comparar dois vetores de características e ter como retorno um valor, não negativo, que represente quão diferentes os vetores são. No formato mais simples, funções de similaridade empregam funções de distância; e
- Estruturas de indexação: são estruturas implementadas para armazenar e recuperar imagens de forma eficiente, facilitando consultas a imagens analisadas.

A Figura 19 apresenta um esquema CBIR, no qual é demonstrada como os componentes que compõem o contexto de um sistema CBIR podem ser representados. A forma de representação da Figura 19 é adaptada ao contexto do presente estudo.

Figura 19 – Estrutura genérica de um sistema CBIR.

Fonte: Adaptada de Delamaro, Nunes e Oliveira (2013).

Além do domínio de imagens, o conceito de CBIR pode ser aplicado em outros domínios como, por exemplo, sistemas que processam sons (OLIVEIRA *et al.*, 2014). Então, o conceito

de CBIR serve como alternativa para a automatização de oráculos de teste, possibilitando a definição de recursos para os extratores de características e função de similaridade.

4.3 Oráculos de teste baseado em características – um passo a passo

Para a condução da abordagem de oráculos de teste baseado em características, um fluxo de atividades é definido. O fluxo de atividades varia de acordo com o domínio de sistema da aplicação em teste. No entanto, algumas atividades podem ser consideradas como genéricas na aplicação da abordagem. Tais atividades são ilustradas na Figura 20 por meio de um *workflow*.

Figura 20 – Abordagem oráculos de teste baseado em características.

Fonte: Elaborada pelo autor.

Na Figura 20, as fases requeridas para a aplicação da abordagem de oráculos de teste baseados em extração de características são divididas em três macro-etapas. A seguir é apresentada uma breve descrição de cada uma dessas etapas:

- **Atividade 1:** inicialmente são definidas dois objetos: (i) imagem modelo; e (ii) imagem do SUT. A imagem modelo é o objeto a ser considerado como referência na condução do processo de teste e, conseqüentemente, fonte de informação para o oráculo de teste. Por sua vez, a imagem do SUT é o objeto a ser testado. Em outras palavras, a definição dos dois objetos devem levar em consideração que o objetivo do processo de teste é definir se a imagem do SUT é similar a imagem modelo;
- **Atividade 2:** na segunda etapa, são definidas as características que são relevantes para o contexto do teste a ser conduzido e os extratores de características são desenvolvidos para obter as respectivas características das imagens. As características podem ser, por

exemplo: densidade, pontos de bifurcação, ângulo de bifurcação, medida de tortuosidade, etc. Importante ressaltar que a definição de tais características é realizada, geralmente, com a supervisão de um especialista da área. Após extrair as características por meio dos extratores desenvolvidos, as características são armazenadas em vetores de características, representando a “essência” da imagem, segundo um aspecto específico. Para medir a similaridade entre a imagem modelo e a imagem do SUT, é definida uma função de similaridade. A função de similaridade define o grau de semelhança entre as imagens em análise. Alguns fatores são levados em consideração durante a definição de uma função de similaridade como o domínio do problema, o conjunto de dados a ser analisado e, em alguns casos, ter conhecimento do tipo de resultado que pode ser obtido de acordo com a medida de distância implantada. Uma das formas de se medir a similaridade entre os vetores de características é por meio de funções de distância; e

- **Atividade 3:** por fim, são realizadas atividades de configuração do oráculo de teste. Durante essa atividade, as características extraídas das imagens em análise são definidas como fonte de informação do oráculo para julgar a similaridade entre as imagens. Para o oráculo fornecer um veredicto final (**passou/falhou**), um valor de limiar é definido para representar o máximo do valor de diferença entre as imagens modelo e a imagem do SUT.

4.4 Limitações da abordagem

Uma série de limitações podem ser observadas e identificadas com a aplicação da abordagem oráculos de teste baseado em características. Tais limitações variam de acordo com o cenário do processo de validação conduzido e, principalmente, com o domínio do sistema a ser explorado no processo. Algumas das limitações da abordagem que podem ser identificadas durante a condução de validações empíricas são ilustradas a seguir:

- **Dependência da definição de objetos de referência:** para condução do processo de aplicação da abordagem é necessária a definição de objetos de referência a partir do qual as características possam ser extraídas. Sem a disponibilidade de uma funcionalidade que permita a definição de objetos de referência, o processo de teste se torna limitado e inviabiliza a aplicação da abordagem;
- **Estratégias para ajuste do valor do limiar:** A tarefa de definição do valor do limiar para medir o grau de similaridade entre dois objetos, não é uma tarefa trivial. Esta é uma característica que veio dos conceitos de CBIR, e não existe uma única solução para isso. Uma estratégia promissora para mitigar esse problema poderia ser definir intervalos de valor para comparação, como "não aceito - falhar", "verificação suspeita" e "aceito - passar". Além disso, a literatura dispõe de diversos métodos que auxiliam na definição de um limiar como, por exemplo, métodos estatísticos, métodos de observação empírica, etc. No entanto, na maioria dos métodos disponíveis para definição de um limiar, o conhecimento do testador da distribuição dos dados é essencial;

- **Disponibilidade de extratores de características:** dependendo da natureza dos objetos a serem analisados na atividade de teste, a disponibilidade de extratores de características pode ser bastante escassa. Além disso, imagens 3D e arquivos de áudio são considerados complexos, dificultando ainda mais o processo de implementação de extratores de características. Outro fator a ser observado é a disponibilidade de APIs que facilitam a implementação dos extratores. A disponibilidade de APIs para extração de características de imagens médicas, por exemplo, é escassa e varia de acordo com a extensão da imagem; e
- **Disponibilidade de conjuntos de dados:** a disponibilidade de conjuntos de dados de objetos a serem analisados em uma atividade de teste varia de acordo com a complexidade do objeto e, geralmente, da relevância do processo de teste a ser conduzido. A relevância do processo de teste está relacionada ao interesse de usuários em disponibilizarem conjuntos de dados como, por exemplo, de imagens médicas 3D de redes vasculares.

4.5 Framework O-FIm/CO

O *framework* O-FIm/CO (OLIVEIRA; DELAMARO; NUNES, 2009; DELAMARO; NUNES; OLIVEIRA, 2013; OLIVEIRA *et al.*, 2014) emprega conceitos de CBIR para configurar oráculos para apoiar testes em sistemas com saídas gráficas ou complexas como, por exemplo, uma imagem médica processada ou uma GUI. O O-FIm/CO apoia a comparação de objetos por meio da similaridade, particular a cada objeto, e automatiza a função de oráculos para domínios complexos de modo flexível e replicável. É importante destacar que apesar de serem complexos para atividades de teste automatizado, os domínios de saída apoiado pelo O-FIm/CO, em geral, são avaliados por oráculos humanos por meio de suas capacidades sensoriais como, por exemplo, visão e audição (DELAMARO; NUNES; OLIVEIRA, 2013; OLIVEIRA *et al.*, 2014). Dessa forma, o O-FIm/CO visa automatizar a função de oráculos humanos, automatizando a atividade de teste e aumentando a produtividade.

O O-FIm/CO permite que, em um ambiente de teste, sejam extraídas as características de um objeto produzido pelo SUT, permitindo assim que o cálculo da distância entre o objeto esperado e o objeto produzido pelo SUT sejam utilizados como oráculos de teste automatizado.

No contexto do O-FIm/CO, extratores de características específicos para o domínio de saída do SUT podem ser implementados ou, então, adaptados por testadores para que possam compor a automatização dos oráculos de teste dentro do contexto do *framework*. A estrutura genérica do O-FIm/CO permite ao testador interagir com o núcleo da estrutura diretamente, por meio de comandos específicos, ou indiretamente por meio de um *Wizard*. O núcleo tem a tarefa de reconhecer e instalar *plug-ins* (extratores ou funções de similaridade) para uso posterior na composição de oráculos. O testador deve definir um limiar de proximidade entre os dois objetos em comparação para que o oráculo automatizado possa estabelecer seus vereditos.

O testador pode, então, compor oráculos de teste utilizando as características mais

relevantes para o contexto do SUT. Tal composição pode ser descrita por meio de um arquivo de texto simples chamado de descritores de oráculos (OLIVEIRA, 2012). Um *parser* do O-FIm/CO é capaz de reconhecer um descritor de oráculos válidos e fazer chamadas às classes correspondentes, ajustando o sistema para a realização dos testes. Essa estrutura oferece ao testador um ambiente flexível de teste para sistemas com saídas complexas, automatizando oráculos de teste para aliviar o trabalho humano durante atividades de verificação e validação de sistemas com saídas não triviais. A Figura 21 apresenta a estrutura genérica do O-FIm/CO, destacando seus principais módulos.

Figura 21 – Estrutura genérica do O-FIm/CO.

Fonte: Adaptada de Oliveira *et al.* (2014).

Em consequência dos custos para realização das atividades de teste, o O-FIm/CO explora estratégias com uma arquitetura que atende, principalmente, requisitos como flexibilidade, usabilidade e facilidade de uso. No caso da flexibilidade, testadores podem criar oráculos em diferentes contextos, para diferentes saídas. No entanto, é necessário implementar os extratores e as funções de similaridade, definidos como *plug-ins* a serem utilizados pelo oráculo. Conforme foi mencionado, os *plug-ins* podem ser desenvolvidos para domínios específicos e, posteriormente, adicionados ao *framework*.

Por intermédio da utilização do O-FIm/CO, testadores obtêm um programa escrito em Java capaz de julgar se saídas/objetos complexos (imagens, sons, etc) produzidos pelo SUT são similares ou não, de acordo com as características e restrições inseridas diante do contexto do trabalho.

A Figura 22 ilustra um exemplo de definição de um oráculo gráfico para o *framework* O-FIm/CO. Um oráculo é criado usando dois extratores, **MyExtractor** e **OurExtractor**. O primeiro extrator possui uma propriedade chamada “color” que será definida com o valor (uma String) “red” e uma propriedade “alpha” cujo valor é um inteiro longo, 78. O segundo possui uma propriedade “scale” cujo valor será inicializado com um valor double, 1.33. Ambos possuem uma

propriedade chamada “rectangle” para a qual um vetor de inteiros deve ser usado. Além disso, é definida uma função de similaridade, **similarity Euclidean**, que será utilizada para medir a distância entre os objetos. Por fim, é definido o valor do limiar de proximidade, **precision**, cujo valor será inicializado com o valor double, 0.46.

Figura 22 – Exemplo de um oráculo gráfico para o *framework* O-FIm/CO.

```
-----
similarity Euclidean
extractor MyExtractor { color = "red" alpha = 78 rectangle = [100 100 30 40] }
extractor OurExtractor { rectangle = [0 0 128 64] scale = 1.33 }
precision = 0.46
-----
```

Fonte: Elaborada pelo autor.

O arquivo exibido na Figura 22 é utilizado pelo *parser* da ferramenta O-FIm/CO e indica ao *framework* como um oráculo gráfico deve realizar uma comparação durante a execução de determinado teste.

Dois pontos devem ser destacados acerca do uso do O-FIm/CO: (1) necessidade de uma base confiável a ser utilizada para comparação das saídas reais do SUT; e (2) amortização dos custos ao utilizar a estratégia em domínios de saída inéditos. Dessa forma, uma base confiável a ser considerada pelo O-FIm/CO pode ser um arquivo de imagem ou áudio considerado como modelo para a comparação e avaliação das saídas reais do SUT, dependendo do domínio de sistema a ser testado. Por sua vez, o uso de estratégias em domínios de saídas inéditas possibilitaria aliviar os esforços humanos e reduzir custos na avaliação das saídas de tais sistemas.

4.5.1 API – O-FIm/CO

A API do O-FIm/CO possibilita aos testadores a implementação de oráculos de teste para suporte da atividade de teste. A API do O-FIm/CO permite a criação de oráculos a partir de classes pré-definidas. A criação de oráculos de teste pode ser realizada de duas formas pela O-FIm/CO: (i) de forma mais manual por meio da implementação dos métodos necessários para composição do oráculos de teste; e (ii) de forma mais intuitiva por meio da interface gráfica do *wizard*. Dentre os métodos implementados pela API, a Tabela 11 apresenta quatro dos métodos mais úteis utilizados pela API.

Após a configuração do *framework* O-FIm/CO pelo testador para um ambiente de teste as seguintes atividades devem ser consideradas: (i) implementar uma aplicação, na qual o *framework* O-FIm/CO seja incluído como biblioteca; (ii) na aplicação do oráculo, os testadores devem definir a fonte de informações para o teste, o SUT e a descrição do oráculo (*parser*).

Tabela 11 – Métodos a serem implementados por um *plug-in* da O-FIm/CO

Assinatura do método	Especificação
<code>public void setDescOracle(String)</code>	define o caminho de uma descrição do oráculo para uma rodada de teste
<code>public void setComplexData(String)</code>	especifica o caminho de uma pasta (ou um arquivo) contendo as capturas de tela a serem usadas como linha de base para um conjunto de teste
<code>public void setComplexSUT(String)</code>	define o caminho de uma pasta (ou um arquivo) contendo as capturas de tela a serem usadas como fonte de informação para o conjunto de testes
<code>public double getOracleResult()</code>	retorna o (s) resultado (s) da aplicação do oráculo sobre o conjunto de testes e a linha de base previamente definida

Fonte: Adaptada de Oliveira (2017).

4.6 O-FIm/CO para imagens médicas 3D

Imagens médicas 3D ofereceram a possibilidade de grandes avanços na ciência e medicina moderna, à medida que imagens com maiores detalhes são produzidas. O surgimento de imagens 3D é devido ao progresso rápido e contínuo na visualização de imagens médicas computadorizadas e nos avanços em métodos de análise e diagnóstico assistido por computador (DOUGHERTY, 2011). De acordo com Dougherty (2011), o desafio é processar e analisar as imagens para extrair, quantificar e interpretar essas informações para obter compreensão e percepção da estrutura e função dos órgãos que estão sendo imaginados.

No contexto do presente trabalho de mestrado, foram desenvolvidos extratores objetivando extrair características relevantes de imagens médicas sintéticas 3D de vasos sanguíneos. Os extratores foram desenvolvidos de forma genérica, possibilitando a extração de características de outras imagens em modelos 3D. Em seguida, os extratores foram adaptados ao contexto do *framework* O-FIm/CO com o objetivo de permitir o uso adequado do O-FIm/CO no contexto da automação de oráculos de teste para imagens médicas em 3D. Assim, algumas alterações foram realizadas na estrutura do O-FIm/CO. A seguir são descritas as extensões adequadas na estrutura do O-FIm/CO para permitir o teste de sistemas que gerem e processem imagens médicas 3D.

4.6.1 Extensões na estrutura do *framework* O-FIm/CO

O *framework* O-FIm/CO oferece uma interface Java completa para sistemas com saídas complexas chamada OFImPlugin, apresentado no Código-fonte 2. Dessa forma, foi mantida a interface original, realizando uma pequena alteração no parâmetro do método **computeValue**. O método **computeValue** recebe como parâmetro um valor do tipo String. No presente trabalho de mestrado foi utilizada uma biblioteca para manipulação de imagens médicas em formato NIfTI¹, representando as informações dos pontos da imagem por meio de um *Object*. Assim, o método **computeValue** teve o seu parâmetro alterado para o tipo *Object*. Tal alteração pode ser ignorada de acordo com a biblioteca utilizada para manipulação da imagem do domínio do problema.

¹ Link para documentação da biblioteca NIfTI: <https://nifti.nimh.nih.gov/>

Código-fonte 2 – Classe abstrata para *plug-ins* do O-FIm/CO

```

1: package br.oracle.pluginInterfaces;
2: import java.io.IOException;
3: public abstract class OFImPlugin {
4:     public abstract String getName();
5:     public abstract void setProperty(String propertyName,
        Object propertyValue);
6:     public abstract Object getProperty(String propertyName);
7:     public abstract Object[] getProperties();
8:     public abstract String[] getPropertyNames();
9:     public abstract double computeValue(Object complexObject);
10: }

```

Fonte: Elaborada pelo autor.

A classe abstrata *OFImPlugin* possibilita a criação de *plug-ins* o que inclui extratores de características e funções de similaridade. Para o contexto do presente trabalho, foram criadas duas interfaces Java para adaptar *plug-ins* do O-FIm/CO: (i) *InterfaceImageExtractor*; e (ii) *InterfaceSimilarityFunction*. A partir da criação das classes abstratas, quando métodos específicos são necessários no contexto particular do *plug-in* em desenvolvimento, todos os métodos necessários serão definidos pela respectiva interface Java. Por exemplo, ao implementar uma função de similaridade, o testador deve explorar os métodos especificados por *InterfaceSimilarityFunction*. O Código-fonte 3 apresenta a interface Java para *plug-ins* que implementam funções de similaridade úteis em O-FIm/CO. Para essa interface Java, foi mantida a versão original apresentada por Oliveira (2017).

Código-fonte 3 – Interface Java para função de similaridade do O-FIm/CO

```

1: package br.oracle.pluginInterfaces;
2: import java.util.Vector;
3: public interface InterfaceSimilarityFunction {
4:     public String getSimilarityName();
5:     public void addExtractor(OFImPlugin oFImPlugin);
6:     public Vector<OFImPlugin> getExtractors();
7:     public double computeSimilarity(double[] vectModel, double
        [] vectSUT);
8:     public double[] getVectorSimilarity(String complexObject);
9: }

```

Fonte: Adaptada de Oliveira (2017).

A partir da definição dos *plug-ins* mais específicos, o testador deve apenas fazer suas

classes Java estenderem a classe abstrata *OFlmPlugin* e implementar alguma interface Java específica. Tais configurações são necessárias para a administração adequada de *plug-ins* pelo O-FIm/CO. Por meio dessas configurações, uma vez que um *plug-in* esteja instalado, a estrutura definida pelo testador pode ser explorada várias vezes em ambientes de teste, garantindo a flexibilidade pelo O-FIm/CO.

Dentre as interfaces Java de *plug-ins* implementadas por meio do *framework* O-FIm/CO, os extratores de características são um dos componentes mais importantes do O-FIm/CO. Além disso, a definição dos extratores varia de acordo com o domínio do problema e objetivo do testador. Caso o testador tenha desenvolvido um extrator para um domínio, por exemplo, de imagens médicas 3D, facilmente é possível adaptar o extrator para o O-FIm/CO. Para demonstrar a estrutura e o fluxo de implementação de um extrator para o *framework* O-FIm/CO, a Seção 4.7 apresenta detalhes dos *plug-ins* que representam extratores de características de imagens médicas 3D de vasos sanguíneos desenvolvidos no contexto do presente trabalho de mestrado.

4.7 Extratores de características para imagens médicas 3D

Para aplicar e avaliar a abordagem de oráculos de teste baseados em características, um catálogo de extratores de características foi desenvolvido objetivando a condução das avaliações experimentais no contexto do presente trabalho de mestrado. Os extratores de características desenvolvidos foram selecionados de acordo com o domínio do problema em cada estudo experimental. O processo de desenvolvimento dos extratores tem como base conceitos da área de processamento de imagens médicas. Além disso, todo o processo de desenvolvimento e validação de cada extrator teve a participação de dois especialistas da área de processamento de imagens médicas. Nesse cenário, foram implementados cinco extratores de características com uso da biblioteca NIfTI (*Neuroimaging Informatics Technology Initiative*): densidade, quantidade de pontos de bifurcação, quantidade de pontos terminais, quantidade de segmentos e tamanho médio de segmentos. A biblioteca NIfTI oferece métodos para manipulação de imagens médicas 3D na extensão .nii. No entanto, a biblioteca não dispõe de métodos pré-desenvolvidos que auxiliem na implementação de extratores, forçando a adaptação da biblioteca para os objetivos do trabalho a ser desenvolvido.

Os extratores foram disponibilizados em um repositório online como contribuição do presente trabalho de mestrado². Os códigos-fonte da função principal dos extratores são apresentados no Apêndice A. Em geral, os extratores de características são aplicados em imagens médicas 3D e, em seguida, usado para calcular o valor final dos extratores que é útil no contexto do O-FIm/CO.

² <https://goo.gl/6cy4wt>

Importante ressaltar que para aplicação dos extratores nas imagens é realizado um pré-processamento das imagens. Assim, foi realizado o processo de segmentação e esqueletização das imagens. A técnica de segmentação divide uma imagem digital em múltiplas regiões ou objetos (conjunto de *pixels/voxels*). Por outro lado, o algoritmo de esqueletização é frequentemente usado para gerar o esqueleto de uma região da imagem (GONZALEZ; WOODS, 2000). Ambos os processos foram realizados por meio da ferramenta Fiji³ do ImageJ com a supervisão de especialistas em processamento de imagens médicas. A ferramenta Fiji é um software *open-source* usado para análise de imagens, tanto de fotos quanto de vídeos. A ferramenta possui vários *plugins* que podem ter utilidade em diversas áreas de conhecimento. O processo de esqueletização é realizado por meio do algoritmo proposto por Lee, Jashyap e Chu (1994). A Figura 23 ilustra uma imagem sintética de vaso sanguíneo após o processo de segmentação (a) e a mesma imagem após o processo de esqueletização (b).

Figura 23 – Exemplo de imagem de vaso sanguíneo após a segmentação (a) e após a esqueletização (b).

Fonte: Elaborada pelo autor.

A seguir são apresentados os extratores desenvolvidos para condução do presente trabalho de mestrado:

Densidade: este extrator mede o valor da densidade do vaso sanguíneo em relação a imagem como um todo. A densidade significa o quanto o vaso sanguíneo preenche a imagem. Em outras palavras, o extrator de densidade retorna uma média dos valores dos pontos do vaso sanguíneo. Para tanto, foi realizada uma varredura na área pertencente ao vaso e calculada a média dos valores referente aos pontos que pertenciam apenas ao vaso sanguíneo. A Figura 24 apresenta um exemplo com duas imagens com o respectivo número de densidade. As imagens apresentadas pertencem a categorias diferentes de patologias de vasos sanguíneos: aneurisma e estenose.

³ Link contendo informações sobre a ferramenta Fiji: <http://imagej.net/Fiji>

Figura 24 – Exemplo de imagens com o respectivo valor de densidade.

Vaso sanguíneo com
aneurisma
densidade = 1.03×10^{-4}

Vaso sanguíneo com
estenose
densidade = 7.87×10^{-5}

Fonte: Elaborada pelo autor.

Pontos de bifurcação: este extrator identifica a quantidade de pontos de bifurcação de uma determinada imagem. Pontos de bifurcação são caracterizados pela separação de um ponto do vaso em dois ou mais ramos. Em outras palavras, pontos de bifurcação são os pontos na “árvore sanguínea” onde ocorre a separação de um vaso em dois ou mais ramos, a partir do esqueleto da imagem do vaso sanguíneo.

Para encontrar os pontos de bifurcação, o algoritmo analisa a quantidade de vizinhos de cada ponto da imagem, após a esqueletização. Se o número de vizinhos na cor branca for maior que dois, o ponto está em uma bifurcação, se for igual a um o ponto é um ponto terminal e se o ponto tiver exatamente dois ele está em uma linha, não sendo necessário marcá-lo na imagem original. A Figura 25 apresenta um exemplo com duas imagens com o respectivo número de bifurcações. As imagens apresentadas pertencem a categorias diferentes de patologias de vasos sanguíneos: aneurisma e estenose.

Figura 25 – Exemplo de imagens com o respectivo número de bifurcações.

Vaso sanguíneo com
aneurisma
bifurcações = 3

Vaso sanguíneo com
estenose
bifurcações = 7

Fonte: Elaborada pelo autor.

Pontos terminais: este extrator identifica a quantidade de pontos terminais de uma

imagem. Os pontos terminais ou de extremidade, bem como os pontos de bifurcação, são determinados pelo número de vizinhos dos pontos do vaso. Assim, se o número de vizinhos na cor branca for igual a um o ponto é terminal. A Figura 26 apresenta um exemplo com duas imagens com o respectivo número de pontos terminais. As imagens apresentadas pertencem a categorias diferentes de patologias de vasos sanguíneos: aneurisma e estenose.

Figura 26 – Exemplo de imagens com o respectivo número de pontos terminais.

Vaso sanguíneo com
aneurisma
pontos terminais = 5

Vaso sanguíneo com
estenose
pontos terminais = 9

Fonte: Elaborada pelo autor.

Quantidade de segmentos: este extrator identifica a quantidade de segmentos do vaso sanguíneo. Segmentos são partes ou divisões do vaso. A estratégia adotada para identificação e contagem dos segmentos do vaso sanguíneo foi separar o vaso a partir do número de bifurcações e pontos terminais. Desse modo, um segmento é uma parte do vaso que está entre um ponto terminal e um ponto de bifurcação ou vice-versa, e entre pontos de bifurcação. A Figura 27 apresenta um exemplo com duas imagens com o respectivo número de segmentos. As imagens apresentadas pertencem a categorias diferentes de patologias de vasos sanguíneos: aneurisma e estenose.

Figura 27 – Exemplo de imagens com o respectivo número de segmentos.

Vaso sanguíneo com
aneurisma
segmentos = 7

Vaso sanguíneo com
estenose
segmentos = 15

Fonte: Elaborada pelo autor.

Tamanho médio de segmentos: este extrator mede uma proporção do tamanho médio dos segmentos. Para tanto, é realizado o mesmo processo descrito no extrator de segmentos, levando em consideração o tamanho de cada segmento. A Figura 28 apresenta um exemplo com duas imagens com o respectivo tamanho médio dos segmentos. As imagens apresentadas pertencem a categorias diferentes de patologias de vasos sanguíneos: aneurisma e estenose.

Figura 28 – Exemplo de imagens com o respectivo valor do tamanho médio dos segmentos.

Vaso sanguíneo com
aneurisma
tam. médio segmentos = 17.28

Vaso sanguíneo com
estenose
tam. médio segmentos = 40.93

Fonte: Elaborada pelo autor.

4.7.1 Correlação entre os extratores de características

Como uma forma de avaliar e evidenciar a correlação entre os extratores de características desenvolvidos no escopo do presente trabalho de mestrado, foi gerada uma matriz de correlação, com base nos coeficientes de Kendall's (KOHONEN, 1972), entre todos os pares de características. A matriz de correlação foi gerada por meio do pacote *corrplot* que contém um algoritmo para gerar e reordenar uma matriz de correlação. A matriz de correlação gerada pode ser apreciada no Apêndice B. Importante ressaltar que o gráfico foi gerado a partir dos dados discutidos no Experimento 5.3.

A matriz de correlação ilustrada no Apêndice B apresenta a correlação e os coeficientes de correlação de Kendall's entre todos os pares de características. A correlação representa a dependência ou associação entre duas características. Por outro lado, o coeficiente de correlação de Kendall's, destacados em vermelho, indicam quais pares de características possuem correlação significativamente diferentes de zero. Assim, o gráfico de correlação indica, em relação as séries temporais, que todos os pares de características têm correlações significativamente diferentes de zero.

Importante ressaltar que a matriz de correlação representa uma dependência ou associação entre duas variáveis, mas não evidencia se há uma implicação em se utilizar todas ou um número reduzido de variáveis em atividades de classificação ou recuperação de objetos. No contexto de sistemas CBIR, é comum utilizar um conjunto maior de características durante o processo de

recuperação de objetos. Desse modo, a matriz de correlação foi utilizada no presente trabalho de mestrado como uma forma de evidenciar a dependência ou associação entre os extratores de características, sendo base para a implementação de características que complementam uma outra característica como, por exemplo, extrator de número de bifurcações e de pontos terminais, e do extrator de número de segmentos e de tamanho médio dos segmentos.

4.8 Considerações finais

Este capítulo apresentou a abordagem de oráculos de teste baseados em características como uma estratégia promissora para apoiar a automatização de oráculos de teste para sistemas que processem imagens médicas 3D de vasos sanguíneos. Para tanto, o capítulo apresenta as extensões e configurações realizadas no *framework* O-FIm/CO como suporte adequado aos oráculos de teste baseados em características na prática. Além disso, um catálogo de *plug-ins* que representam extratores de características para imagens médicas 3D de vasos sanguíneos são apresentados.

Durante o processo de implementação dos extratores, algumas dificuldades puderam ser identificadas. Dentre tais dificuldades, destacam-se: (i) necessidade de conhecimentos específicos acerca do domínio das imagens 3D; (ii) disponibilidade de poucas APIs e *frameworks* que funcionem com diferentes tipos de objetos 3D; (iii) o problema do oráculo ocorre com os extratores, haja vista que sua correção deve ser averiguada manualmente; (iv) necessidade de *benchmarks* feitos por especialistas para validar os resultados dos oráculos; e (v) a aplicação de alguns extratores em alguns domínios pode ser complexa e o número de falsos positivos/negativos pode ser alto.

A partir da definição da estrutura do *framework* O-FIm/CO para implementar a abordagem de oráculos de teste baseados em características, espera-se ilustrar a aplicabilidade da abordagem, do ponto de vista dos testadores, para sistemas que processem imagens médicas 3D de sanguíneos. O próximo capítulo apresenta a condução e discussão de três estudos experimentais que objetivam demonstrar a eficiência da abordagem de oráculos de teste baseados em características para sistemas que processam imagens sintéticas 3D de vasos sanguíneos.

APLICAÇÃO DA ABORDAGEM EM IMAGENS MÉDICAS 3D DE VASOS SANGUÍNEOS

5.1 Considerações iniciais

Este capítulo apresenta três estudos que associam a abordagem de oráculos de teste baseados em características e a validação de sistemas que geram ou processam imagens sintéticas 3D de vasos sanguíneos. Para tanto, é estabelecido um conjunto de critérios para o uso de extratores de características para o domínio explorado nessa dissertação.

O estudo apresentado na Seção 5.2 avalia a sensibilidade e a precisão de oráculos de teste baseados em características na categorização de imagens sintéticas 3D e 2D de vasos sanguíneos. Esse experimento foi essencial para estabelecer um escopo da utilização da abordagem proposta na avaliação de imagens médicas 3D, gerando subsídios fundamentais para condução de estudos futuros. O estudo apresentado na Seção 5.3 investiga a precisão de oráculos de teste baseados em características na avaliação de imagens sintéticas 3D de vasos sanguíneos frente a aplicação de diferentes níveis de ruído gaussiano. Por fim, o estudo apresentado na Seção 5.4 visa complementar o estudo apresentado na Seção 5.3 com uma comparação da precisão de oráculos humanos em relação aos oráculos automatizados. Importante ressaltar que a condução dos três estudos experimentais tiveram a supervisão de dois especialistas na área de processamento de imagens médicas. A seguir, tais estudos são apresentados e os resultados obtidos são discutidos.

5.2 Categorização de imagens sintéticas 3D e 2D de vasos sanguíneos

O presente experimento consiste na condução de um estudo que objetiva avaliar a precisão dos oráculos de teste baseados em características na categorização de imagens sintéticas 3D

e 2D de vasos sanguíneos. Para tanto, especialistas em processamento de imagens médicas categorizaram as imagens do conjunto de dados, manualmente, em quatro categorias distintas com base na similaridade do número de características das imagens. As características utilizadas para realizar a categorização manual foram: número de pontos de bifurcação, número de pontos terminais e número de segmentos. Tais características possibilitam uma contagem manual por meio do aspecto visual do ser humano.

A partir da definição da categorização realizada por especialistas, foi utilizada a estrutura do *framework* O-FIm/CO para implementar a abordagem de oráculos de teste baseados em características para automatizar o processo de categorização das imagens do conjunto de dados. Para aplicar a abordagem no contexto do presente experimento, uma imagem do conjunto de dados é considerada como modelo, enquanto as outras imagens são consideradas como imagens a serem avaliadas ou comparadas. Assim, às imagens que o oráculo julga como similares à imagem modelo, são atribuídas a mesma categoria da imagem modelo. Imagens julgadas pelo oráculo como não similares não são atribuídas à mesma categoria da imagem modelo. A categorização realizada pelo oráculo é comparada com a categorização realizada pelos especialistas para avaliar o desempenho da categorização do oráculo.

O experimento foi inicialmente conduzido em imagens sintéticas 3D de vasos sanguíneos e, em seguida, em uma versão 2D das imagens. Tal variação entre os modelos tem como objetivo avaliar a precisão do oráculo em julgar a similaridade das imagens 3D em relação a imagens 2D e identificar particularidades do oráculo em julgar corretamente a similaridade das imagens. Ainda, foi realizada uma comparação da precisão do oráculo em categorizar corretamente os dois modelos de imagens em relação a classificadores randômicos.

Em seguida, são apresentados os detalhes técnicos dos materiais e métodos utilizados na condução do experimento (Seção 5.2.1), as questões de pesquisa e as métricas (Seção 5.2.2), formulação das hipóteses (Seção 5.2.3), a composição do conjunto de dados (Seção 5.2.4), as etapas de condução do experimento (Seção 5.2.5), os resultados obtidos (Seção 5.2.6), as ameaças à validade (Seção 5.2.7) e as conclusões (Seção 5.2.8).

5.2.1 Materiais e métodos

Nessa seção são apresentados detalhes técnicos dos materiais e métodos utilizados para condução do estudo.

5.2.1.1 Extratores de características

Seguindo o fluxo de trabalho necessário para implementação da abordagem de oráculos de teste baseados em características, foram identificadas características relevantes para extração das imagens. Durante a definição dos extratores de características a serem desenvolvidos, foi necessária a supervisão de dois especialistas na área de processamento de imagens médicas.

Os especialistas indicaram as características a serem utilizadas como fonte de informação do oráculo e supervisionaram o processo de validação dos extratores.

Os extratores desenvolvidos foram disponibilizados como código aberto em um repositório online para imagens médicas 3D e 2D¹. A Figura 29 apresenta as características extraídas de um vaso sanguíneo sintético com aneurisma. Sistemas baseados em um único atributo de imagem podem não atingir níveis de recuperação adequados, razão pela qual os sistemas usam vários atributos de imagem para indexação e recuperação de interesse. Desse modo, foram desenvolvidos extratores de características que se complementam e aumentam o nível de significância entre as características, aumentando a precisão na comparação das imagens.

Figura 29 – Exemplo de características extraídas por meio dos extratores desenvolvidos – Estudo 1.

Características extraídas
Dimensões da imagem: 120 120 120
#####Extrator de densidade#####
Densidade: 1,03x10-4
#####Extrator de bifurcação#####
Número de bifurcações: 3
#####Extrator de pontos terminais#####
Número de pontos terminais: 5
#####Extrator de segmentos#####
Número de segmentos: 7

Fonte: Elaborada pelo autor.

Após a extração das características, para cada uma das imagens, os dados foram normalizados na escala linear. Também referida como normalização de interpolação linear, a normalização linear consiste em considerar os valores mínimo e máximo para cada atributo no ajuste da escala. Esta operação mapeia os valores de um atributo para um intervalo fechado de 0 a 1 (LINDBERG; ROMENY, 1994).

5.2.1.2 Função de similaridade

Após a normalização das características extraídas, seus valores são armazenados em um vetor de características e a semelhança entre uma imagem e outra é medida calculando-se a distância entre os vetores de características. Para medir a distância entre os vetores de características, é utilizada a DE. DE é uma medida de distância amplamente utilizada, inclusive em estudos anteriores (OLIVEIRA, 2014; DELAMARO; NUNES; OLIVEIRA, 2013). Considerando dois vetores de características x e y , de tamanho n , onde os elementos são chamados x_i e y_i , respectivamente, a DE pode ser definida como:

$$DE(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (5.1)$$

¹ <https://goo.gl/gqqzIM>

5.2.1.3 Classificador randômico

Uma das etapas conduzidas no estudo está relacionada em comparar a precisão obtida pelo oráculo de teste baseado em características na categorização correta das imagens 3D e 2D em relação a precisão obtida por meio de um classificador randômico. Classificadores randômicos são métodos de aprendizagem de conjuntos para classificar aleatoriamente os objetos nas classes. Para realizar esta atividade, foi utilizada a classe *Random* da linguagem de programação Python². A classe *Random* foi executada duas vezes para cada imagem, avaliando a categorização correta das imagens, por meio das métricas de precisão, revocação e especificidade.

Para aplicação do classificador randômico no experimento, a imagem considerada como modelo é retirada do conjunto de dados e as outras imagens são obtidas de maneira randômica. Para avaliar a corretude da categorização, é esperado que as imagens pertencentes a mesma categoria que a imagem modelo sejam retornadas a frente das outras imagens do conjunto de dados.

5.2.1.4 Configuração do limiar

A qualidade do processo de similaridade entre os objetos não depende apenas da capacidade da função de similaridade para separar os valores relevantes de irrelevantes, mas também da definição do valor do limiar selecionado na condução do estudo. O limiar (do inglês, *threshold*) representa a máxima distância admissível para classificar um objeto numa determinada classe. A configuração do valor do limiar pode ser realizada de acordo com a finalidade da aplicação.

Para este estudo, foi realizado uma análise empírica com o conjunto de dados de imagens 2D e 3D, avaliando as métricas de precisão e revocação de acordo com o valor do limiar. Ao realizar estudos iniciais, estudos empíricos são frequentemente usados para avaliar a influência do valor do limiar na base de dados em estudo. Assim, o valor limiar é variado, com valor inicial 0,1, com incrementos de 0,05 até 0,7. A Figura 30 e a Figura 31 mostram o valor da média (AVG, do inglês *average*) do conjunto de dados para cada valor do limiar selecionado. É possível notar que existe um *trade-off* entre a precisão e a revocação. Desse modo, quanto maior o valor do limiar, menor a precisão e maior a revocação. Por outro lado, quanto menor for o valor do limiar, maior a precisão e menor a revocação.

Durante a análise dos resultados, foi observado que, para as imagens com um maior número de características, valores de limiar altos funcionam bem. No entanto, para imagens com menores números de características, valores baixos de limiar funcionam melhor. Dado que o objetivo do estudo é obter um número maior de imagens categorizadas corretamente, mas, ao mesmo tempo, para garantir que as mesmas imagens de categoria sejam retornadas, a melhor escolha seria um limiar igual 0,35 para os dois modelos de imagens. No entanto, se o estudo pretende obter um maior número de imagens da mesma categoria sem garantir que o oráculo

² <https://goo.gl/01HnAW>

Figura 30 – Imagens 2D - Comparação da média da precisão e revocação para o conjunto de valores de limiares.

Fonte: Elaborada pelo autor.

Figura 31 – Imagens 3D - Comparação da média da precisão e revocação para o conjunto de valores de limiares.

Fonte: Elaborada pelo autor.

tenha sido preciso em categorizar as imagens corretamente, a melhor escolha seria 0,5. Como este estudo pretende verificar se o oráculo foi necessário para categorizar as imagens nos dois modelos, foi selecionado o valor de limiar de 0,35.

5.2.2 Questões de pesquisa e métricas

Este estudo foi definido para responder duas Questões de Pesquisa (QPs). As QPs e as métricas utilizadas são descritas a seguir:

- **QP1:** Oráculos de teste baseados em características são capazes de categorizar corretamente imagens sintéticas de redes vasculares em modelos 3D e 2D?
- **QP2:** Oráculos de teste baseados em características são capazes de categorizar correta-

mente imagens sintéticas de redes vasculares em modelos 3D e 2D em comparação com classificadores randômicos?

A QP1 aborda a eficácia dos oráculos de teste baseados em características na categorização correta das imagens sintéticas 3D e 2D de vasos sanguíneos. A QP2 aborda a eficácia dos oráculos de teste baseados em características na categorização correta das imagens sintéticas 3D e 2D de vasos sanguíneos em relação a classificadores randômicos. O desempenho do oráculo de teste baseado em características na categorização correta das imagens indica a utilidade dos oráculos nas atividades de teste. Nesse cenário, as métricas de precisão, revocação e especificidade são avaliadas para verificar o desempenho da categorização das imagens 3D e 2D de vasos sanguíneos. De maneira similar, as mesmas métricas foram avaliadas nos classificadores randômicos.

No contexto do presente estudo, a precisão indica se as imagens recuperadas pelo oráculo pertencem à mesma categoria da imagem modelo (mas não indica se todos os elementos da categoria foram recuperados). Por outro lado, a revocação indica se todas as imagens que pertencem à mesma categoria da imagem modelo foram recuperadas (mas não indica de quantos elementos irrelevantes foram recuperados). Por fim, a especificidade indica se as imagens que são irrelevantes (não pertencem à mesma categoria da imagem modelo) não foram retornadas pelo oráculo. A seguir, tais métricas são especificadas:

$$\text{Precisão} = \frac{VP}{VP + FP} \quad (5.2)$$

$$\text{Revocação} = \frac{VP}{VP + FN} \quad (5.3)$$

$$\text{Especificidade} = \frac{VN}{VN + FP} \quad (5.4)$$

onde:

- Verdadeiro Positivo (VP): Imagem categorizada como correta no grupo esperado;
- Falso Positivo (FP): Imagem categorizada como correta no grupo não esperado;
- Verdadeiro Negativo (VN): Imagem categorizada como incorreta no grupo não esperado;
- Falso Negativo (FN): Imagem categorizada como incorreta no grupo esperado.

Para colher as métricas especificadas anteriormente, o oráculo de teste baseado em características é definido com base na distância entre o conjunto de imagens definidas como modelo $R^1 = \{ImR_1^1, ImR_2^1, \dots, ImR_m^1\}$ com o conjunto de imagens a serem avaliadas ou comparadas $R^2 = \{ImR_1^2, ImR_2^2, \dots, ImR_n^2\}$. Com base na categorização realizadas pelos especialistas, o oráculo de

teste baseado em características verifica se a distância obtida na categorização entre a ImR_1^1 e uma ImR_i^2 está acima do limiar.

Oráculo de teste: se a imagem modelo tiver uma distância acima do limiar para a imagem a ser avaliada, elas serão classificadas em categorias diferentes.

se $DE(ImR_i^1, ImR_i^2) > \text{limiar}$
então ImR_i^2 não é categorizada na mesma categoria de ImR_i^1

As métricas são colhidas de acordo com as imagens que o oráculo de teste julga como similar à imagem modelo. Ao definir o limiar, o oráculo de teste baseado em características julga como imagens similares as que obtêm um valor de similaridade menor ou igual ao valor do limiar.

5.2.3 Formulação de hipóteses

Em seguida, as hipóteses relacionadas a cada QP são descritas e os testes estatísticos para as hipóteses são apresentados.

Para a QP1, as seguintes hipóteses foram definidas:

Hipótese nula H_0 :

“Oráculos de teste baseados em características não são capazes de tomar a decisão correta entre a similaridade das imagens a serem avaliadas a partir de uma imagem modelo”.

$$H_0 : \theta_o = 0,05$$

Hipótese alternativa H_1 :

“Oráculos de teste baseados em características são capazes de tomar a decisão correta entre a similaridade das imagens a serem avaliadas a partir de uma imagem modelo”.

$$H_1 : \theta_o < 0,05$$

A evidência estatística demonstra a precisão no uso de oráculos de teste baseados em características para testar sistemas que produzem imagens sintéticas de vasos sanguíneos. Assim, a H_0 afirma que não há diferença entre o número de verdadeiros veredictos (VP/N) e falsos veredictos (VP/N) do oráculo de teste baseado em características. Para testar H_0 , foi realizado o teste binomial. O teste binomial é recomendado para verificar a evidência estatística de dados em resultados binários (casos de sucesso) (PECK; OLSEN; DEVORE, 2015). No caso dos oráculos de teste, considerando o número de verdadeiros veredictos, os testes binomiais podem mostrar a significância estatística dos desvios de uma distribuição esperada de observações em duas

categorias (falha e sucesso). Para esse estudo, foi utilizado o teste binomial com uma cauda com nível de significância igual a $\alpha = 0,05$, estabelecendo uma necessidade de 95% de confiança.

O mesmo procedimento foi adotado para a QP2. Assim, as seguintes hipóteses foram definidas:

Hipótese nula (H_0):

“Não há diferença na precisão para categorizar as imagens sintéticas de vasos sanguíneos entre o oráculo de teste baseado em características e os classificadores randômicos.”

$$H_0 : \theta_o = \theta_r$$

Hipótese alternativa (H_1):

“O oráculo de teste baseado em características obtém um valor de precisão diferente na categorização de imagens sintéticas de redes vasculares em comparação com classificadores randômicos.”

$$H_1 : \theta_o \neq \theta_r$$

O teste estatístico é realizado por meio *t-test*, que é um teste estatístico em duas amostras para avaliar a diferença na precisão das categorizações realizadas pelo oráculo de teste baseado em características e o classificador randômico. O *t-test* de duas amostras mede um intervalo de confiança e realiza um teste de hipóteses para a diferença entre duas médias da população quando os desvios padrão são desconhecidos e as amostras são geradas independentemente uma da outra (PECK; OLSEN; DEVORE, 2015). Para testar a H_0 , foi realizado o *t-test* bicaudal:

- Oráculo de teste baseado em características versus classificador randômico com um nível de significância de $\alpha = 0,05$.

5.2.4 Conjunto de dados

O conjunto de dados explorado no estudo é composto por 13 imagens sintéticas 3D de vasos sanguíneos e 13 imagens sintéticas 2D de vasos sanguíneos. As imagens 3D são geradas pelo método proposto por Galarreta-Valverde *et al.* (2013), e as imagens 2D são obtidas por meio de *screenshots* das imagens 3D usando a ferramenta BioImage Suite. O uso de imagens sintéticas 3D e 2D de vasos sanguíneos visa avaliar se o modelo da imagem influencia no julgamento correto do oráculo de teste baseado em características. Além disso, as imagens 3D permitem uma análise mais detalhada da estrutura da imagem, enquanto as imagens 2D são limitadas.

As imagens são geradas a partir de quatro categorias de vasos sanguíneos: (i) aneurisma; (ii) não-regulares; (iii) normal; e (iv) estenose. Vasos sanguíneos com aneurisma são caracterizados pela dilatação anormal causada por paredes debilitadas do vaso, trauma ou doença vascular.

Vasos sanguíneos não-regulares são caracterizados pelo comportamento “anormal”, levando em consideração o aspecto visual. Vasos sanguíneos normais apresentam uma aparência visual sem a presença de dilatação ou estreitamento. Por fim, vasos sanguíneos com estenose apresentam um estreitamento anormal no vaso.

5.2.5 Condução do experimento

Buscando evidências experimentais para responder as QPs, o estudo foi realizado em seis etapas, ilustradas na Figura 32 e detalhadas da seguinte maneira: inicialmente, as imagens 3D foram geradas por meio do método proposto por Galarreta-Valverde *et al.* (2013). As imagens 2D foram obtidas a partir de *screenshots* das imagens 3D. Na segunda etapa, as imagens foram manualmente categorizadas em quatro categorias distintas, com base no número de características de cada imagem. A categorização foi padronizada para ambos os modelos de imagem (3D e 2D), portanto, a característica de densidade não foi considerada na categorização. A categorização das imagens foi realizada por especialistas em processamento de imagens médicas.

Figura 32 – Etapas para condução do estudo 1.

Fonte: Elaborada pelo autor.

A Tabela 12 apresenta as quatro categorias definidas pelos especialistas. O nome definido a cada imagem representa uma patologia que o vaso sanguíneo possui. Por exemplo, a imagem “aneurism (1 e 2)” representa um vaso sanguíneo com aneurisma, a imagem “nonRegular (1, 2, 3 e 4)” representa um vaso sanguíneo não regular, a imagem “normal (1, 2, 3, 4 e 5)” representa um vaso sanguíneo normal e a imagem “stenose (1 e 2)” representa um vaso sanguíneo com estenose. De acordo com a Tabela 12, a categoria A possui cinco imagens, a categoria B possui quatro imagens, e as categorias C e D possuem duas imagens que se assemelham em relação ao número de características que possuem. Nota-se que a patologia que a imagem representa não foi utilizada como critério para categorizar as imagens. As imagens foram categorizadas, manualmente, com base no número de bifurcações, pontos terminais e segmentos. Tais características possibilitam uma contagem manual por meio do aspecto visual do ser humano.

Tabela 12 – Categorias de imagens definidas a partir da similaridade entre o número de características.

Categoria	Imagens
A	aneurism1, nonRegular1, nonRegular2, normal3 e stenose1
B	aneurism2, nonRegular3, normal4 e stenose2
C	nonRegular4 e normal5
D	normal1 e normal2

Fonte: Elaborada pelo autor.

Na terceira etapa, foram realizadas as atividades relacionadas ao pré-processamento das imagens. Nessa etapa, os processos de segmentação e esqueletização das imagens foram realizados por meio da ferramenta Fiji. Na quarta etapa, foi realizada a identificação de características relevantes para extração nas imagens. Assim, foram utilizados os extratores de densidade, bifurcação, pontos terminais e de segmentos. Importante ressaltar que as características foram selecionadas com a supervisão de especialistas em processamento de imagens médicas. Na quinta etapa, foram adaptados os extratores de características, a função de similaridade e realizadas outras atividades relacionadas à adaptação da estrutura do *framework* O-FIm/CO ao contexto do estudo. Além disso, nessa etapa, foi definida a configuração do limiar e a composição do oráculo de teste. O *framework* O-FIm/CO oferece uma interface de implementação que permite a adaptação de extratores e funções de similaridade para realizar as atividades de teste. Para compor o oráculo automatizado, o *framework* O-FIm/CO permite que o testador descreva, por meio de um arquivo texto, os extratores de características, a função de similaridade e o valor do limiar que será considerado na atividade de teste. A Figura 33 ilustra o descritor do oráculo utilizado neste estudo.

Figura 33 – Descritor de oráculo do estudo 1.

```

similarity Euclidean
extractor Density {}
extractor Bifurcation {}
extractor Endpoints {}
extractor Segments {}
precision = 0.35

```

Fonte: Elaborada pelo autor.

Na sexta etapa, o experimento foi realizado com as imagens 3D e os resultados avaliados. De maneira similar, o estudo foi realizado com as imagens 2D e os resultados analisados. Por fim, o classificador randômico foi executado e os resultados comparados com os resultados do oráculo.

A Figura 34 ilustra o fluxo de atividades para categorização automática das imagens. Inicialmente uma imagem do conjunto de dados é definida como modelo e as demais imagens do

conjunto de dados como imagens a serem avaliadas ou comparadas. Em seguida, as características são extraídas da imagem modelo e das demais imagens do conjunto de dados. A partir dos vetores de características, a métrica de similaridade é aplicada para medir o grau de similaridade entre a imagem modelo e cada imagem do conjunto de dados. O oráculo, por meio de um limiar, julga quais das imagens do conjunto de dados são similares à imagem modelo e, consequentemente, estão na mesma categoria da imagem modelo. O estudo é realizado definindo cada imagem do conjunto de dados como modelo objetivando avaliar o desempenho do oráculo em julgar a similaridade de cada imagem com as demais imagens do conjunto de dados.

Figura 34 – Fluxo de atividades para categorização das imagens.

Fonte: Elaborada pelo autor.

5.2.6 Resultados e discussão

Inicialmente, os extratores de características foram executados para obter as características das imagens do conjunto de dados. A Tabela 13 apresenta o número de características de cada imagem do conjunto de dados. Os valores das características de bifurcação, pontos terminais e segmentos são semelhantes para as imagens em modelos 3D e 2D. No entanto, o valor da característica de densidade é definida para imagens 3D, de modo que a densidade não é usada para a categorização padrão das imagens 2D.

A seguir, são apresentados os resultados obtidos e as respostas para as QPs. Além disso, estudos estatísticos foram realizados para refutar ou aceitar as hipóteses.

5.2.6.1 QP1

A QP1 aborda a eficácia dos oráculos de teste baseados em características na categorização correta das imagens sintéticas 3D e 2D de vasos sanguíneos. Assim, foram analisados o desempenho da categorização realizada pelo oráculo para as imagens sintéticas 3D e 2D de vasos sanguíneos por meio das métricas de precisão, revocação e especificidade. Além disso, foi

Tabela 13 – Imagens com seus respectivos números de características – Estudo 1.

Imagem	Densidade	Bifurcação	Pontos Terminais	Segmentos
aneurism1	$1,03 \times 10^{-4}$	3	5	7
aneurism2	$1,22 \times 10^{-4}$	7	9	15
nonRegular1	$4,91 \times 10^{-5}$	2	4	5
nonRegular2	$7,46 \times 10^{-5}$	2	4	5
nonRegular3	$1,2 \times 10^{-4}$	6	8	13
nonRegular4	$2,19 \times 10^{-4}$	20	18	37
normal1	$1,9 \times 10^{-5}$	0	2	1
normal2	$3,47 \times 10^{-5}$	1	3	3
normal3	$5,96 \times 10^{-5}$	3	4	7
normal4	$1,16 \times 10^{-4}$	7	9	15
normal5	$1,9 \times 10^{-4}$	15	17	31
stenose1	$6,46 \times 10^{-5}$	3	5	7
stenose2	$7,87 \times 10^{-5}$	7	9	15

Fonte: Elaborada pelo autor.

realizado o teste binomial com o objetivo de avaliar a eficiência do oráculo de teste baseado em características.

A Tabela 14 apresenta os resultados obtidos pelo oráculo de teste baseado em características na categorização de imagens sintéticas 3D de vasos sanguíneos. Os valores de precisão, revocação e especificidade demonstram o desempenho do oráculo de teste baseado em características na categorização das imagens 3D com base nas categorias pré-definidas na Tabela 12. O valor da média representa a média do valor de cada métrica. Desse modo, o oráculo de teste baseado em características apresentou uma média de 77% de precisão, 100% de revocação e 88% de especificidade.

Cada linha da Tabela 14 representa o resultado ao se utilizar uma determinada imagem como modelo. Por exemplo, na primeira linha, a imagem “aneurism1” foi utilizada como modelo. Assim, das quatro imagens que deveriam ser julgadas como da mesma classe da imagem “aneurism1”, as quatro foram julgadas corretamente pelo oráculo. No entanto, o oráculo classificou, incorretamente, uma imagem como da mesma classe da imagem “aneurism1”. Nesse cenário, o oráculo apresentou uma precisão, revocação e especificidade iguais a, respectivamente, 0,80, 1,00 e 0,87. Essa mesma metodologia foi aplicada para julgar o desempenho do oráculo na classificação das demais imagens.

O oráculo de teste baseado em características apresentou resultados com alta precisão para a maior parte das imagens 3D. A média da precisão igual a 77% para imagens 3D significa que o oráculo categorizou a maioria das imagens como corretas nas categorias esperadas. A média de revocação igual a 100% em todas as imagens significa que o oráculo categorizou como corretas todas as imagens que são corretas. Por outro lado, a média da especificidade de 88% significa que, para a maior parte das imagens, o oráculo não categorizou imagens que não pertencem à categoria da imagem que está sendo analisada.

Tabela 14 – Resultados retornados pelo oráculo por meio das métricas de precisão, revocação e especificidade para imagens 3D – limiar = 0,350.

X Imagem	VP/VN e FP/FN				Métricas		
	VP	VN	FP	FN	Precisão	Revocação	Especificidade
aneurism1	4	7	1	0	0,80	1,00	0,87
aneurism2	3	9	0	0	1,00	1,00	1,00
nonRegular1	4	6	2	0	0,67	1,00	0,75
nonRegular2	4	6	2	0	0,67	1,00	0,75
nonRegular3	3	8	1	0	0,75	1,00	0,90
nonRegular4	1	11	0	0	1,00	1,00	1,00
normal1	1	9	2	0	0,33	1,00	0,81
normal2	1	7	4	0	0,20	1,00	0,63
normal3	4	7	1	0	0,80	1,00	0,87
normal4	3	9	0	0	1,00	1,00	1,00
normal5	1	11	0	0	1,00	1,00	1,00
stenose1	4	7	1	0	0,80	1,00	0,87
stenose2	3	9	0	0	1,00	1,00	1,00
Média	—	—	—	—	0,77	1,00	0,88

Fonte: Elaborada pelo autor.

Em relação às imagens *normal1* e *normal2*, o oráculo apresentou uma taxa de baixa precisão. Tais imagens, como apresentado na Tabela 13, são as imagens que possuem o menor número de características. Assim, para o valor do limiar definido, o oráculo categorizou essas imagens com pouca precisão.

Na Tabela 15 são apresentadas as médias de precisão, revocação e especificidade obtidos pelo oráculo de teste baseado em características na categorização de imagens sintéticas 2D de vasos sanguíneos. Nota-se que a precisão apresentou uma média igual a 72%, demonstrando que o oráculo automatizado categorizou corretamente a maior parte das imagens. A revocação apresentou uma média igual a 81%, demonstrando que, para a maior parte das imagens, o oráculo categorizou como imagens positivas as imagens que são positivas. Por outro lado, a especificidade apresentou uma média de 93%, demonstrando que o oráculo categorizou as imagens negativas como negativas.

Assim como a Tabela 14, cada linha da Tabela 15 representa o resultado ao se utilizar uma determinada imagem como modelo. Por exemplo, na segunda linha, a imagem “aneurism2” foi utilizada como modelo. Assim, das três imagens que deveriam ser julgadas como da mesma classe da imagem “aneurism2”, as três foram julgadas corretamente pelo oráculo. No caso dessa imagem, o oráculo retornou apenas as imagens esperadas como da mesma classe da imagem “aneurism2”. Nesse cenário, o oráculo apresentou uma precisão, revocação e especificidade iguais a, respectivamente, 1,00, 1,00 e 1,00. Essa mesma metodologia foi aplicada para julgar o desempenho do oráculo na classificação das demais imagens.

Diferentemente do desempenho do oráculo na categorização das imagens 3D, para categorizar as imagens 2D o oráculo obteve uma média de revocação igual a 81%. Isso pode ter ocorrido devido ao valor do limiar utilizado. Além disso, para imagens com um maior número de

Tabela 15 – Resultados retornados pelo oráculo por meio das métricas de precisão, revocação e especificidade para imagens 2D – limiar = 0,350.

X Imagem	VP/VN e FP/FN				Métricas		
	VP	VN	FP	FN	Precisão	Revocação	Especificidade
aneurism1	2	7	1	2	0,67	0,50	0,87
aneurism2	3	9	0	0	1,00	1,00	1,00
nonRegular1	4	7	1	0	0,80	1,00	0,87
nonRegular2	3	7	1	1	0,75	0,75	0,87
nonRegular3	3	9	0	0	1,00	1,00	1,00
nonRegular4	0	11	0	1	0,00	0,00	1,00
normal1	1	11	0	0	1,00	1,00	1,00
normal2	1	7	4	0	0,20	1,00	0,63
normal3	4	7	1	0	0,80	1,00	0,87
normal4	3	9	0	0	1,00	1,00	1,00
normal5	0	11	0	1	0,00	0,00	1,00
stenose1	3	8	0	1	1,00	0,75	1,00
stenose2	3	9	0	0	1,00	1,00	1,00
Média	—	—	—	—	0,72	0,81	0,93

Fonte: Elaborada pelo autor.

características, o oráculo apresentou baixos números de precisão e revocação. Por exemplo, para as imagens *nonRegular4* e *normal5*, o oráculo apresentou uma média de precisão e revocação igual a 0, embora apresentasse uma média de 100% de especificidade. De modo similar, para a imagem *normal2* o oráculo apresentou um baixo número de precisão em consequência do valor do limiar.

Complementando a avaliação da QP1, foi realizada uma análise estatística com o objetivo de avaliar a eficiência do oráculo de teste baseado em características. Assim, foi realizado o teste binomial para verificar a evidência estatística de dados. O teste binomial demonstra a significância estatística de desvios de uma distribuição esperada de observações em duas categorias (categorização correta ou não) (PECK; OLSEN; DEVORE, 2015). Nesse contexto, a Tabela 16 apresenta a análise estatística realizada com base nos números obtidos neste estudo. A Tabela 16 apresenta uma análise global, incluindo o número de falsos veredictos (FP/FN), o número de verdadeiros veredictos (VP/VN) e o teste estatístico de H_0 com base no número de verdadeiro veredictos.

Na coluna de alcances da Tabela 16, é apresentada uma interpretação estatística que os números de verdadeiros veredictos revelam sobre os resultados. Essas interpretações mostram a probabilidade de ocorrência dos principais eventos das combinações do oráculo. As interpretações estatísticas são baseadas em conceitos do teste binomial unicaudal, que descreve a chance de ocorrência de eventos em três ocasiões diferentes: (i) a probabilidade de VP/N precisos em 156 vezes; (ii) a probabilidade de menos de VP/N em 156 vezes; e (iii) a probabilidade de mais de VP/N em 156 vezes.

Em relação aos testes de hipóteses apresentados, a Tabela 16 revela que os oráculos de teste para imagens 3D e 2D atingiram uma alta significância estatística, fazendo com que a H_0

Tabela 16 – Teste binomial a partir das métricas coletadas no estudo 1.

Abordagem	Métrica	Oráculo	<i>p-value</i>	$H_0?$	alcances
Imagens 3D	FP	14	1×10^{-12}	Rej.	A prob. de exatamente 142 de 156 é $p = 1 \times 10^{-12}$
	FN	0			A prob. de exatamente, ou menos do que, 142 de 156 é $p = 0,999999$
	VP	36			A prob. de exatamente, ou mais de, 142 de 156 é $p = 1 \times 10^{-12}$
	VN	106			A (unicaudal) prob. de exatamente, ou maior que, 142 de 156 é $p = 1 \times 10^{-12}$
Imagens 2D	FP	8	1×10^{-12}	Rej.	A prob. de exatamente 142 de 156 é $p = 1 \times 10^{-12}$
	FN	6			A prob. de exatamente, ou menos do que, 142 de 156 é $p = 0,999999$
	VP	30			A prob. de exatamente, ou mais de, 142 de 156 é $p = 1 \times 10^{-12}$
	VN	112			A (unicaudal) prob. de exatamente, ou maior que, 142 de 156 é $p = 1 \times 10^{-12}$
Random 1	FP	24	1×10^{-12}	Rej.	A prob. de exatamente 108 de 156 é $p = 1 \times 10^{-12}$
	FN	24			A prob. de exatamente, ou menos de, 108 de 156 é $p = 0,999999$
	VP	12			A prob. de exatamente, ou mais de, 108 de 156 é $p = 1 \times 10^{-12}$
	VN	96			A (unicaudal) prob. de exatamente, ou maior que, 108 de 156 é $p = 1 \times 10^{-12}$
Random 2	FP	28	1×10^{-12}	Rej.	A prob. de exatamente 100 de 156 é $p = 1 \times 10^{-12}$
	FN	28			A prob. de exatamente, ou menos de, 100 de 156 é $p = 0,999999$
	VP	10			A prob. de exatamente, ou mais do que, 100 de 156 é $p = 1 \times 10^{-12}$
	VN	90			A (unicaudal) prob. de exatamente, ou maior que, 100 de 156 é $p = 1 \times 10^{-12}$

Fonte: Elaborada pelo autor.

fosse rejeitada nos dois cenários. Os oráculos tiveram muitos verdadeiros veredictos atingindo 142 para as imagens 3D e 142 para as imagens 2D. Para a categorização aleatória, o número de verdadeiros veredictos variou de 100 a 108. Nesse cenário, os oráculos tiveram um desempenho que lhes permite ser explorados em ambientes de teste. Além dos números fornecidos por este estudo, as configurações do oráculo de teste baseado em características mostraram que o número de verdadeiros veredictos revelou que os oráculos poderiam ser efetivamente explorados em ambientes de teste para avaliar aplicações que geram ou processam imagens médicas de redes vasculares.

5.2.6.2 QP2

A QP2 aborda a eficácia dos oráculos de teste baseados em características na categorização correta das imagens sintéticas 3D e 2D de vasos sanguíneos em relação a classificadores randômicos. Para demonstrar se há existência estatística entre o oráculo e o classificador randômico foi realizado o *t-test* bicaudal.

O *t-test* é um teste de hipótese que usa conceitos estatísticos para avaliar se há diferença significativa entre as médias de duas amostras (PECK; OLSEN; DEVORE, 2015). Importante ressaltar que o *t-test* foi realizado com base no número de verdadeiros veredictos (VP/VN) dos oráculos e do classificador randômico na categorização das imagens sintéticas 3D e 2D de vasos sanguíneos. Assim, os testes realizados mostraram que, por meio do critério convencional, o oráculo de teste baseado em características versus categorização randômica (*p-value* = 0,0109) atingiu uma diferença considerada estatisticamente significativa. Em outras palavras, as evidências demonstraram que a média da categorização das imagens sintéticas 3D e 2D de vasos sanguíneos do oráculo e do classificador randômico são diferentes.

5.2.7 Ameaças à validade

Ameaças à validade interna: a principal ameaça associada à relação causa-efeito entre o oráculo de teste baseado em característica e a qualidade de categorização de imagens sintéticas de redes vasculares em modelos 3D e 2D está relacionada ao processo usado para categorizar as imagens. A categorização das imagens foi realizada por especialistas em processamento de imagens médicas objetivando mitigar essa ameaça, mas pode ser imperfeito.

Da mesma forma, a estratégia de definição do limiar é uma ameaça de validade interna. Essa ameaça está intimamente associada à abordagem de oráculo de teste baseado em característica. Para mitigar essa ameaça, este estudo definiu um limiar empiricamente, avaliando diferentes limiares e os valores da precisão e revocação para cada um.

Ameaças à validade externa: a principal ameaça à validade externa está relacionada à seleção de características extraídas das imagens. Para limitar essa ameaça, especialistas em processamento de imagens médicas apoiaram o estudo, auxiliando na identificação de características relevantes nas imagens.

Ameaças à validade de construção: a principal ameaça à validade de construção deste estudo está associada à possibilidade de implementação incorreta dos oráculos de teste ou dos extratores de características. Durante a implementação dos extratores de características, foram utilizadas APIs de processamento de imagens médicas para o formato NIfTI. Com o objetivo de mitigar essa ameaça, os extratores de características foram validados individualmente por um especialista em processamento de imagens médicas. Desse modo, é possível reivindicar uma alta validade para a relação entre a teoria dos oráculos de teste baseados em características e os resultados do experimento.

Da mesma forma, a geração das imagens 2D é uma ameaça à validade de construção extra. As imagens em modelos 2D são obtidas a partir de *screenshots* das imagens 3D usando a ferramenta BioImage Suite e com supervisão de um especialista em processamento de imagens médicas.

Ameaças à validade de conclusão: é possível afirmar que este estudo possui uma alta validade de conclusão devido a três razões: (i) a condução do experimento foi apresentada em detalhes, destacando e justificando todas as decisões; (ii) os resultados foram disponibilizados por meio de tabelas, e eles estão disponíveis *on-line* para futuros estudos; e (iii) alta relevância estatística dos testes ($\alpha = 0,05$) que apoiam as conclusões.

5.2.8 Considerações sobre o experimento 1

O estudo apresentado nesta seção objetivou avaliar a precisão dos oráculos de teste em verificar a similaridade de imagens sintéticas de vasos sanguíneos por meio da abordagem de oráculos de teste baseados em características. Essa avaliação foi realizada por meio de um estudo experimental que visa categorizar imagens a partir da similaridade de características. Como

ferramenta de suporte à automatização de oráculos de teste baseados em características, foi utilizada a estrutura do *framework* O-FIm/CO para automatizar o processo de categorização.

Com base no processo aplicado, foi observado que as imagens sintéticas 3D de vasos sanguíneos permitem uma análise mais detalhada da imagem. Assim, a extração de características em imagens 3D com alto número de características é menos complexa que em imagens 2D. Com base nas evidências, a abordagem de oráculos de teste baseados em características pode ser usada como uma estratégia promissora para automatizar a atividade de validação de objetos complexos, especificamente de imagens sintéticas de vasos sanguíneos.

Objetivando complementar o estudo, foi conduzida a categorização das imagens 3D e 2D por classificadores randômicos e os resultados comparados com a categorização realizada pelo oráculo. A evidência demonstrou que, em relação ao número de verdadeiros veredictos (VP/VN), o oráculo tem uma diferença estatisticamente significativa na categorização das imagens 3D e 2D, em comparação com o classificador randômico.

Espera-se que o estudo descrito instigue outros pesquisadores em dedicar esforços para aproximarem conceitos da ES com sistemas com saídas complexas. Além disso, este estudo forneceu alternativas e subsídios essenciais para a condução de estudos que objetivam a avaliação de sistemas que gerem ou processem imagens médicas 3D por meio de uma nova abordagem de oráculos de teste.

5.3 Avaliação de imagens sintéticas 3D de vasos sanguíneos após aplicação de ruído gaussiano

O presente experimento consiste na condução de um estudo que avalia a precisão dos oráculos de teste baseados em características na avaliação de imagens sintéticas 3D de vasos sanguíneos após aplicação de diferentes níveis de ruído gaussiano. Assim, é avaliado o desempenho do oráculo em julgar se uma imagem definida como modelo, sem ruído, é similar à respectiva imagem após aplicação de diferentes níveis de ruído gaussiano. Para tanto, especialistas em processamento de imagens médicas aplicaram seis níveis de ruído gaussiano determinados pelo valor do sigma usado em uma Distribuição Gaussiana (DG). As principais fontes de ruído gaussiano em imagens digitais são problemas de iluminação ou alta temperatura durante aquisição ou transmissão (ETTEN, 2006). O conjunto de dados do estudo é composto por oito imagens, divididas em quatro categorias distintas de vasos sanguíneos. Para cada imagem gerada são aplicados seis níveis de ruído gaussiano, variando o valor do sigma em 1, 2, 3, 4, 5 e 10.

Após aplicação dos níveis de ruído nas imagens, foi utilizada a estrutura do *framework* O-FIm/CO para implementar a abordagem de oráculos de teste baseados em características para automatizar o processo de avaliação das imagens sintéticas 3D de vasos sanguíneos. Cada imagem do conjunto de dados, sem ruído, é considerada como modelo e todas as outras imagens,

com ruído, são consideradas como imagens a serem avaliadas. Assim, é avaliada a capacidade e a precisão dos oráculos de teste baseados em características em julgar corretamente as imagens que são similares à imagem modelo.

Em seguida, são apresentados os detalhes técnicos dos materiais e métodos utilizados na condução do experimento (Seção 5.3.1), as questões de pesquisa e as métricas (Seção 5.3.2), formulação de hipóteses (Seção 5.3.3), a composição do conjunto de dados do estudo (Seção 5.3.4), as etapas de condução do experimento (Seção 5.3.5), os resultados obtidos (Seção 5.3.6), as ameaças à validade (Seção 5.3.7) e as conclusões (Seção 5.3.8).

5.3.1 Materiais e métodos

Nessa seção, são apresentados detalhes técnicos dos materiais e métodos utilizados para condução do estudo.

5.3.1.1 Extratores de características e Função de similaridade

Para seleção dos extratores de características do presente estudo, o estudo foi conduzido, a princípio, com o conjunto de extratores do estudo apresentado na Seção 5.2. Assim, observou-se a necessidade de inserir mais características para implementação de um oráculo mais preciso. Os seguintes extratores foram utilizados no presente estudo: (i) densidade; (ii) bifurcação; (iii) pontos terminais; (iv) segmentos; e (v) tamanho médio dos segmentos. A Figura 35 ilustra as características extraídas de um vaso sanguíneo sintético 3D, incluindo a característica de tamanho médio dos segmentos.

Figura 35 – Exemplo de características obtidas por meio dos extratores desenvolvidos – Estudo 2.

Características extraídas
Dimensões da imagem: 120 120 120
#####Extrator de densidade#####
Densidade: 1,03x10-4
#####Extrator de bifurcação#####
Números de bifurcações: 3
#####Extrator de pontos terminais#####
Número de pontos terminais: 5
#####Extrator de segmentos#####
Número de segmentos: 7
#####Extrator de tamanho médio de segmentos####
Tamanho médio de segmentos: 17,28

Fonte: Elaborada pelo autor.

Após extração das características, os dados foram normalizados na escala linear. Em seguida, é medida a distância entre os vetores de características por meio da DE, especificada na Equação 5.1.

5.3.1.2 Aplicação do ruído gaussiano

Para cada imagem modelo foram aplicados seis níveis de ruído, variando de 1, 2, 3, 4, 5 e 10. Os níveis de ruído são determinados pelo valor do sigma usado em uma DG. A Equação 5.5 apresenta como o nível de ruído foi aplicado em cada imagem modelo:

$$\text{sigma} = x * (\text{max}/100) \quad (5.5)$$

onde:

- *max*: intensidade máxima na imagem modelo (por exemplo, 255);
- *x*: valor que pertence ao alcance de 1, 2, 3, 4, 5 a 10.

A Figura 36 apresenta um exemplo de uma imagem de vaso sanguíneo com aneurisma após aplicação de seis diferentes níveis de ruído. Nota-se que a intensidade do ruído influencia diretamente na aparência da imagem como consequência da variação dos valores cinza dos pontos na imagem causados por erros na transmissão de dados ou possíveis distorções introduzidas na fase de aquisição. Para reduzir o nível de ruído nas imagens, foram utilizados alguns filtros de suavização como, por exemplo, o Filtro Gaussiano (FG). O FG é bastante eficiente na redução do ruído gaussiano (ruído aleatório).

Figura 36 – Exemplo de uma imagem de vaso sanguíneo com aneurisma após aplicação de seis diferentes níveis de ruído gaussiano.

Fonte: Elaborada pelo autor.

5.3.1.3 Configuração do limiar

A estratégia de definição do limiar aplicada no presente estudo é baseada no desvio padrão dos valores das características. Para evitar vieses, foram considerados três valores de limiar: (i) th1 para o valor do desvio padrão; ii) th2 para metade de th1; e (iii) th3 para o dobro de th1. A Equação 5.7 apresenta o cálculo do valor do desvio padrão, mensurado a partir das características das imagens do conjunto de dados. Considerando o conjunto de valores de características de N números com uma lista de características $S = \{x_0, x_1, x_2, \dots, x_{N-1}\}$ cuja média é $\bar{x}$. Para este conjunto de dados, o desvio padrão σ pode ser mensurado como segue abaixo.

$$\bar{x} = \frac{\sum_{i=0}^{N-1} x_i}{N} \quad (5.6)$$

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (x_i - \bar{x})^2} \quad (5.7)$$

Assim, os valores definidos para os limiares são apresentados a seguir:

- **th1: 0,313273919**
- **th2: 0,15663696**
- **th3: 0,626547839**

A Figura 37 apresenta o valor da média da precisão e revocação a partir da definição de um conjunto de limiares. O conjunto de limiares foi definido variando em 0,05 iniciando em 0,1 a 0,6, totalizando um conjunto de dez limiares. A variação da média da precisão e revocação apresentada na Figura 37 representa os resultados obtidos no estudo a partir da execução do experimento com o respectivo limiar. É possível notar que existe um *trade-off* entre a precisão e a revocação. Desse modo, quanto maior o valor do limiar, menor a precisão e maior a revocação. Por outro lado, quanto menor for o valor do limiar, maior a precisão e menor a revocação.

Durante a análise dos resultados, foi observado que valores de limiar altos funcionaram bem para a métrica de revocação. No entanto, a precisão ficou muito baixa no mesmo cenário. Dado que o objetivo do estudo é avaliar, principalmente, a precisão dos oráculos em julgar corretamente a similaridade das imagens modelo com a respectiva imagem após aplicação de diferentes níveis de ruído, a melhor escolha seria um limiar abaixo de 0.3. Com base no valor de th1, th2 e th3, preferiu-se utilizar um limiar entre 0.15 e 0.2 em consequência da variação das médias de precisão e revocação.

5.3.2 Questões de pesquisa e métricas

Este estudo foi definido para responder uma única Questão de Pesquisa (QP). A QP e as métricas utilizadas são descritas a seguir:

Figura 37 – Média da precisão e revocação com variação do limiar – Estudo 2.

Fonte: Elaborada pelo autor.

- **QP1:** Oráculos de teste baseados em características são capazes de avaliar corretamente a similaridade entre imagens modelo e as respectivas imagens após aplicação de diferentes níveis de ruído gaussiano?

O QP1 aborda a eficácia e precisão dos oráculos de teste baseados em características na avaliação de imagens sintéticas de vasos sanguíneos 3D após aplicação de diferentes níveis de ruído gaussiano. A precisão do oráculo de teste baseado em características no julgamento correto das imagens indica a utilidade dos oráculos em atividades de teste. Assim, as métricas de precisão, revocação e especificidade são avaliadas para verificar o desempenho do oráculo no julgamento correto da similaridade das imagens.

No contexto do presente estudo, a precisão indica se as imagens recuperadas pelo oráculo são similares à imagem modelo (mas não indica se todos os elementos similares foram recuperados). Por outro lado, a revocação indica se todas as imagens que são similares à imagem modelo foram recuperadas (mas não indica de quantos elementos irrelevantes foram recuperados). Por fim, a especificidade indica se as imagens que são irrelevantes (não similares à imagem modelo) não foram retornadas pelo oráculo. As métricas de precisão (Equação 5.2), revocação (Equação 5.3) e especificidade (Equação 5.4) dos oráculos são avaliadas para verificar o desempenho do oráculo no julgamento correto das imagens.

Complementando a avaliação da QP1, uma das formas mais utilizadas para avaliar sistemas CBIR é utilizando o gráfico de Precisão *versus* Revocação (PxR). Santos (2006) exemplifica o processo de elaboração do gráfico de PxR da seguinte forma:

"Suponha que exista um conjunto A possua as imagens relevantes e um conjunto B as imagens recuperadas, sendo $A = \{i_2, i_4, i_5, i_7, i_9\}$ e $B = \{i_5, i_6, i_7, i_1, i_3, i_8, i_9, i_{10}\}$. Considere a

primeira imagem recuperada i5. Esta imagem corresponde a 20% de todas as imagens relevantes, portanto tem uma revocação de 20%. Além disso, ela tem uma precisão de 100% (uma entre uma imagem recuperada é relevante), podendo-se dizer que esta imagem tem uma precisão de 100% em 20% de revocação. Considere agora a segunda imagem recuperada que é relevante, i7. Esta imagem tem uma precisão de 66,67% (duas entre três imagens recuperadas são relevantes) em 40% de revocação (foram recuperadas duas imagens relevantes de um total de cinco imagens relevantes). Continuando com a análise descrita, têm-se todos os pontos necessários para se a curva de PxR. Caso ao final do conjunto B não tenham sido recuperadas todas as imagens consideradas relevantes, o próximo nível de revocação será igual à zero."

Para elaboração do gráfico PxR e avaliação de um sistema CBIR é comum realizar diversas consultas e avaliar a média da Precisão para cada valor de Revocação calculado a partir do resultado da busca. A Figura 38 apresenta um exemplo de um gráfico PxR, o eixo x armazena os valores da Revocação e o eixo y os valores de Precisão.

Figura 38 – Exemplo de um gráfico de Precisão *versus* Revocação

Fonte: Elaborada pelo autor.

No gráfico PxR da Figura 38 é possível observar que houve um *tradeoff* entre a precisão e revocação. Em outras palavras, quando o valor de precisão foi alto (100%), o valor de revocação foi baixo (20%). Ou seja, o índice de precisão foi alto para valores pequenos de revocação, indicando que os primeiros objetos retornados estavam corretos, embora, conforme o número de objetos recuperados aumentava, o número de objetos não esperados começou a aumentar.

5.3.3 Formulação de hipóteses

Em seguida, as hipóteses relacionadas a QP1 são descritas e os testes estatísticos para as hipóteses são apresentados.

Hipótese nula H_0 :

“Oráculos de teste baseados em características não são capazes de tomar a decisão correta entre a similaridade das imagens a serem avaliadas com ruído gaussiano a partir de uma imagem modelo”.

$$H_0 : \theta_o = 0,05$$

Hipótese alternativa H_1 :

“Oráculos de teste baseados em características são capazes de tomar a decisão correta entre a similaridade das imagens a serem avaliadas com ruído gaussiano a partir de uma imagem modelo com alta precisão”.

$$H_1 : \theta_o < 0,05$$

A evidência estatística demonstra a precisão no uso de oráculos de teste baseados em características para testar sistemas que produzem imagens médicas de vasos sanguíneos. Para testar H_0 , foi realizado o teste binomial. Para esse estudo, foi utilizado o teste binomial com uma cauda com nível de significância igual a $\alpha = 0,05$, estabelecendo uma necessidade de 95% de confiança.

5.3.4 Conjunto de dados

O conjunto de dados utilizado no estudo é composto por imagens sintéticas 3D de vasos sanguíneos utilizadas no estudo apresentado na Seção 5.2. No contexto do presente estudo, foram utilizadas duas imagens de cada categoria de patologia de vaso sanguíneo: (i) aneurisma; (ii) não regulares; (iii) normal; e (iv) estenose. Assim, o conjunto de dados é composto por um conjunto de oito imagens modelo. Para composição do conjunto de imagens a serem avaliadas, seis níveis de ruído gaussiano foram aplicados nas imagens modelo, formando um conjunto de dados com 48 imagens a serem avaliadas.

5.3.5 Condução do experimento

Para levantar evidências experimentais para responder a QP, o estudo foi realizado seguindo seis etapas, ilustradas na Figura 39 e detalhadas da seguinte forma: inicialmente, as imagens 3D foram geradas por meio do método proposto por Galarreta-Valverde *et al.* (2013). Na segunda etapa, foram aplicados seis níveis de ruído em cada imagem gerada, totalizando 48 imagens a serem avaliadas. Na terceira etapa, foram realizadas as atividades relacionadas ao pré-processamento das imagens. Assim, foram realizados os processos de segmentação e esqueletização das imagens. As atividades relacionadas aos passos um, dois e três foram realizadas e supervisionadas por especialistas em processamento de imagens médicas.

Figura 39 – Etapas para condução do estudo 2.

Fonte: Elaborada pelo autor.

Após o pré-processamento das imagens, na quarta etapa, foram identificadas características relevantes nas imagens. Assim, foram selecionados os extratores de densidade, quantidade de bifurcações, quantidade de pontos terminais, quantidade de segmentos e tamanho médio dos segmentos. Na quinta etapa, os extratores de características e a função de similaridade foram adaptados ao contexto do *framework* O-FIm/CO. Além disso, foram realizadas as atividades relacionadas à adaptação da estrutura do *framework* O-FIm/CO para o contexto do estudo a ser realizado. Para composição do oráculo, o *framework* O-FIm/CO permite que o testador descreva, por meio de um arquivo de texto, os extratores de características, a função de similaridade e o valor do limiar que será considerado na atividade de teste. Os extratores desenvolvidos foram disponibilizados em um repositório online³. A Figura 40 ilustra o descritor do oráculo usado na condução do presente estudo.

Figura 40 – Descritor de oráculo do estudo 2.

```

similarity Euclidean
extractor Density {}
extractor Bifurcation {}
extractor Endpoints {}
extractor Segments {}
extractor AVG_Size_Segments {}
precision = 0.156636
  
```

Fonte: Elaborada pelo autor.

Por fim, na sexta etapa, o estudo é conduzido e os resultados são observados e avaliados. A Figura 41 ilustra o fluxo de atividades para avaliação das imagens. Inicialmente, cada imagem 3D de vasos sanguíneos gerada é definida como modelo. Por outro lado, as imagens com ruído

³ <https://goo.gl/g49NHM>

são definidas como imagens a serem avaliadas. As características são extraídas da imagem modelo e das demais imagens com ruído. A partir dos vetores de características das imagens, a métrica de similaridade é aplicada para medir o grau de similaridade entre a imagem modelo e cada imagem com ruído. O oráculo por meio de um limiar, julga quais das imagens com ruído são similares à imagem modelo.

Figura 41 – Fluxo de atividades para avaliação das imagens.

Fonte: Elaborada pelo autor.

5.3.6 Resultados e discussão

Esta seção descreve os resultados obtidos a partir do julgamento do oráculo de teste baseado em características em relação à similaridade entre as imagens a serem avaliadas a partir da imagem modelo. Desse modo, o desempenho do oráculo é avaliado por meio das métricas de precisão, revocação e sensibilidade. Além disso, para rejeitar ou não a hipótese nula, o resultado do teste estatístico é apresentado e discutido.

Inicialmente, as características de cada imagem modelo são extraídas como fonte de conhecimento para o oráculo. O número de características extraídas foi validado por especialistas em processamento de imagens médicas. A Tabela 17 apresenta o número de características extraídas de cada imagem modelo.

Durante a condução do estudo, os resultados obtidos por meio do oráculo de teste baseado em características foram observados por meio da definição do limiar. Os resultados demonstraram que o limiar igual a th_2 resulta em uma maior média de precisão para o oráculo. Desse modo, os resultados do estudo foram discutidos e apresentados com th_2 . A Tabela 18 apresenta a média de cada métrica calculada para avaliar o desempenho do oráculo a partir do conjunto de características do estudo apresentado na Seção 5.2 em relação a th_2 .

Tabela 17 – Imagens com os respectivos número de características – Estudo 2.

Imagem	Densidade	Bifurcações	Pontos terminais	Segmentos	Tam. médio de segmentos
aneurism1	$1,03 \times 10^{-4}$	3	5	7	17,28
aneurism2	$1,22 \times 10^{-4}$	7	9	15	31,8
nonRegular1	$4,91 \times 10^{-5}$	2	4	5	19,4
nonRegular2	$7,46 \times 10^{-5}$	2	4	5	36
normal1	$1,9 \times 10^{-5}$	0	2	1	1
normal2	$3,47 \times 10^{-5}$	1	3	3	17
stenose1	$6,46 \times 10^{-5}$	3	5	7	28
stenose2	$7,87 \times 10^{-5}$	7	9	15	40,93

Fonte: Elaborada pelo autor.

Cada linha da Tabela 18 representa o resultado ao se utilizar uma determinada imagem como modelo. Por exemplo, na primeira linha à imagem “aneurism1” foi utilizada como modelo. Assim, das seis imagens que deveriam ser julgadas como similares a imagem “aneurism1”, as seis foram julgadas corretamente pelo oráculo. Nesse cenário, o oráculo apresentou uma precisão, revocação e especificidade iguais a, respectivamente, 1,00, 1,00 e 1,00. Essa mesma metodologia foi aplicada para julgar o desempenho do oráculo em julgar a similaridade das demais imagens.

Tabela 18 – Desempenho do oráculo por meio das métricas de precisão, revocação e especificidade, com o primeiro conjunto de características – Limiar = th_2 .

X	VP/VN e FP/FN				Métricas		
Imagem	VP	VN	FP	FN	Precisão	Revocação	Especificidade
aneurism1	6	42	0	0	1,00	1,00	1,00
aneurism2	6	42	0	0	1,00	1,00	1,00
nonRegular1	6	39	3	0	0,66	1,00	0,92
nonRegular2	3	39	3	3	0,5	0,50	0,92
normal1	4	42	0	2	1,00	0,66	1,00
normal2	4	42	0	2	1,00	0,66	1,00
stenose1	6	42	0	0	1,00	1,00	1,00
stenose2	6	42	0	0	1,00	1,00	1,00
Média	—	—	—	—	0,89	0,85	0,98

Fonte: Elaborada pelo autor.

Com base nos resultados apresentados na Tabela 18, a característica de tamanho médio de segmentos foi incluída, gerando resultados melhores e, conseqüentemente, um oráculo mais preciso. Os resultados apresentados e discutidos nesse estudo incluem a característica de tamanho médio de segmentos além das características já definidas no estudo apresentado na Seção 5.2.

5.3.6.1 QP1

A Tabela 19 apresenta a precisão do oráculo de teste baseado em características em julgar a similaridade das imagens a serem avaliadas a partir da imagem modelo. Os resultados apresentados na Tabela 19 incluem as características definidas no estudo da Seção 5.2 com a característica de tamanho médio dos segmentos. O desempenho do oráculo é avaliado por meio das métricas de precisão, revocação e especificidade. O valor da média representa uma

média de cada métrica. Uma média igual ou próximo de 1 representa que o oráculo julgou a semelhança das imagens corretamente e com alta precisão. Por outro lado, uma média perto de 0 representa que o oráculo de teste baseado em características avaliou a similaridade das imagens incorretamente e com baixa precisão.

Assim como a Tabela 18, cada linha da Tabela 19 representa o resultado ao se utilizar uma determinada imagem como modelo. Por exemplo, na segunda linha, a imagem “aneurism2” foi utilizada como modelo. Assim, das seis imagens que deveriam ser julgadas similares à imagem “aneurism2”, as seis foram julgadas corretamente pelo oráculo. Nesse cenário, o oráculo apresentou uma precisão, revocação e especificidade iguais a, respectivamente, 1,00, 1,00 e 1,00. Essa mesma metodologia foi aplicada para julgar o desempenho do oráculo na classificação das demais imagens.

Tabela 19 – Desempenho do oráculo por meio das métricas de precisão, revocação e especificidade, com o segundo conjunto de características – Limiar = $th2$.

X Imagem	VP/VN e FP/FN				Métricas		
	VP	VN	FP	FN	Precisão	Revocação	Especificidade
aneurism1	6	42	0	0	1,00	1,00	1,00
aneurism2	6	42	0	0	1,00	1,00	1,00
nonRegular1	6	42	0	0	1,00	1,00	1,00
nonRegular2	3	42	0	3	1,00	0,5	1,00
normal1	4	42	0	2	1,00	0,66	1,00
normal2	4	42	0	2	1,00	0,66	1,00
stenose1	6	42	0	0	1,00	1,00	1,00
stenose2	6	42	0	0	1,00	1,00	1,00
Média	---	---	---	---	1,00	0,85	1,00

Fonte: Elaborada pelo autor.

De acordo com os resultados apresentados na Tabela 19, o oráculo apresentou uma média de precisão igual a 1 (100%), revocação igual a 0,85 (85%) e especificidade igual a 1 (100%). Uma média de precisão igual a 1 (100%) significa que o oráculo julgou corretamente as imagens a partir da imagem modelo. Uma média de revocação igual a 0,85 (85%) significa que o oráculo, apesar de ter julgado corretamente a maioria das imagens, não retornou todas as imagens esperadas. Por fim, uma média de especificidade igual a 1 (100%) significa que o oráculo julgou corretamente as imagens que não são similares à imagem modelo.

Em relação à média de revocação igual a 0,85 (85%), as imagens nonRegular2, normal1 e normal2 apresentaram baixa precisão em comparação com as outras imagens do conjunto de dados. Uma média baixa de revocação para essas imagens está relacionado à implicação da presença de ruído nas imagens. O nível de ruído, mesmo com a aplicação do filtro de suavização, não foi completamente removido da imagem e isso pode ter influenciado os resultados, como a densidade da imagem e o tamanho médio do segmento.

A Figura 19 apresenta o gráfico PxR a partir do desempenho no retorno das imagens. É possível observar por meio do gráfico PxR que os valores de precisão iguais a 1 foram os

que mais predominaram durante a busca. Em outras palavras, foram recuperados 100% das imagens corretas ou esperadas. Além disso, observa-se que houve um *tradeoff* entre a precisão e revocação. Ou seja, o valor de precisão foi alto para valores pequenos (20% a 50%) e médios de revocação (60%). No entanto, assim que o número de imagens esperadas aumentava, o valor de precisão consequentemente caiu rapidamente para 85%.

Figura 42 – Gráfico PxR com o segundo conjunto de características – Limiar = $th2$.

Fonte: Elaborada pelo autor.

Completando a QP1, foi realizado uma análise estatística para avaliar a precisão do oráculo. Foi realizado o teste binomial para verificar a evidência estatística dos dados em relação ao número de verdadeiros veredictos (VP/VN). A Tabela 20 apresenta o número de falsos veredictos (FP/FN), o número de verdadeiros veredictos (VP/VN) e o *p-value* do oráculo em relação a $th1$, $th2$ e $th3$. A partir desses números, a tabela apresenta o teste de H_0 em relação a cada valor de limiar. O valor de 1×10^{-12} é um valor padrão para testes estatísticos que têm um alto número de tentativas em testes binomiais. Uma vez que o conjunto de dados é composto por 48 imagens, definiu-se que esta avaliação poderia ser ajustada a esse nível de significância.

Na coluna de alcances da Tabela 20, é apresentada uma interpretação estatística dos números de veredictos em relação ao significado dos resultados. Essas interpretações apresentam a probabilidade de ocorrência dos principais eventos das combinações do oráculo. As interpretações estatísticas baseiam-se em conceitos do teste binomial unicaudal, esses testes descrevem a chance de ocorrência de eventos em três ocasiões diferentes: (i) a probabilidade de vezes de VP/N precisos em 384 vezes; (ii) a probabilidade de menos de VP/N em 384 vezes; e (iii) a probabilidade de mais de VP/N em 384 vezes.

Em relação aos testes de hipóteses apresentados, a Tabela 20 revela que os oráculos

Tabela 20 – Teste binomial a partir das métricas coletadas no estudo 2.

Limiar	Métrica	Oráculo	<i>p-value</i>	$H_0?$	alcance
th1	FP	34	1×10^{-12}	Rej.	A prob. de exatamente 346 de 384 é $p = 1 \times 10^{-12}$
	FN	4			A prob. de exatamente, ou menos que, 346 de 384 é $p = 0,999999$
	VP	44			A prob. de exatamente, ou mais que, 346 de 384 é $p = 1 \times 10^{-12}$
	VN	302			A (unicaudal) prob. de exatamente, ou maior que, 346 de 384 é $p = 1 \times 10^{-12}$
th2	FP	0	1×10^{-12}	Rej.	A prob. de exatamente 377 de 384 é $p = 1 \times 10^{-12}$
	FN	41			A prob. de exatamente, ou menor, 377 de 384 é $p = 0,999999$
	VP	30			A prob. de exatamente, ou mais de, 142 de 156 é $p = 1 \times 10^{-12}$
	VN	112			A prob. de exatamente, ou maior que, 142 de 156 é $p = 1 \times 10^{-12}$
th3	FP	133	1×10^{-12}	Rej.	A prob. de exatamente 251 de 384 é $p = 1 \times 10^{-12}$
	FN	0			A prob. de exatamente, ou menor que, 251 de 384 é $p = 0,999999$
	VP	48			A prob. de exatamente, ou mais que, 251 de 384 é $p = 1 \times 10^{-12}$
	VN	203			A (unicaudal) prob. de exatamente, ou maior que, 251 de 384 é $p = 1 \times 10^{-12}$

Fonte: Elaborada pelo autor.

atingiram uma alta significância estatística, fazendo com que a H_0 fosse rejeitada nos três cenários. Os oráculos tiveram muitos verdadeiros veredictos atingindo 346 em th1, 377 em th2 e 251 em th3. Assim, os oráculos tiveram um desempenho que lhes permitem ser explorados em ambientes de teste. Além dos números fornecidos por este estudo, as configurações do oráculo de teste baseado em características demonstraram que o número de verdadeiros veredictos revelou que os oráculos poderiam ser efetivamente explorados em ambientes de teste para avaliar sistemas que gerem ou processem imagens médicas 3D de vasos sanguíneos.

5.3.7 Ameaças à validade

Ameaças à validade interna: a principal ameaça à validade interna do estudo está relacionada à aplicação de ruído nas imagens. Para mitigar essa ameaça, a aplicação do ruído gaussiano foi definida por um especialista em processamento de imagens médicas, aplicando diferentes níveis de ruído gaussiano nas imagens.

Da mesma forma, a estratégia de definição do limiar é uma ameaça de validade interna. Essa ameaça está intimamente associada à abordagem oráculo de teste baseado em característica. Para mitigar essa ameaça, este estudo considerou três valores de limiares diferentes, demonstrando a precisão do julgamento do oráculo nos três cenários.

Ameaças à validade externa: a principal ameaça à validade externa está relacionada à seleção de características extraídas das imagens. Para limitar essa ameaça, especialistas em processamento de imagens médicas apoiaram o estudo na identificação de características relevantes nas imagens.

Ameaças à validade de construção: a principal ameaça à validade de construção deste estudo está associada à possibilidade de implementação incorreta dos oráculos de teste ou dos extratores de características. Durante a implementação dos extratores de características, foram utilizadas APIs de processamento de imagens médicas para o formato NIfTI. Com o objetivo de mitigar essa ameaça, os extratores de características foram validados individualmente por um especialista em processamento de imagens médicas. Desse modo, é possível reivindicar uma

alta validade para a relação entre a teoria dos oráculos de teste baseados em características e os resultados do experimento.

Da mesma forma, o processo de esqueletização aplica níveis de ruído na imagem. Dessa forma, mesmo com a aplicação dos filtros de suavização, a esqueletização pode prejudicar, suavemente, a qualidade da imagem. Para minimizar essa ameaça, um especialista em processamento de imagens médicas auxiliou na supervisão do processo de esqueletização das imagens.

Ameaças à validade de conclusão: é possível afirmar que este estudo possui uma alta validade de conclusão devido a três razões: (i) a condução do experimento foi apresentada em detalhes, destacando e justificando todas as decisões; (ii) os resultados foram disponibilizados por meio de tabelas, e eles estão disponíveis *on-line* para futuros estudos; e (iii) alta relevância estatística dos testes que apoia as conclusões.

5.3.8 Considerações sobre o experimento 2

O estudo apresentado nessa seção avaliou a precisão de oráculos de teste na avaliação de imagens sintéticas 3D de vasos sanguíneos após aplicação de diferentes níveis de ruído gaussiano por meio da abordagem oráculos de teste baseados em características. Para condução do estudo, foi utilizada a estrutura do *framework* O-FIm/CO para automatizar o processo de avaliação. Além disso, o estudo utilizou subsídios produzidos por meio do estudo apresentado na Seção 5.2 como estratégia para implementar oráculos mais precisos no julgamento correto da similaridade de imagens sintéticas 3D de vasos sanguíneos.

Com base nos resultados obtidos, foi observado que o oráculo de teste apresentou alta precisão no julgamento correto das imagens sintéticas 3D de vasos sanguíneos após aplicação de diferentes níveis de ruído gaussiano. A alta precisão do oráculo está relacionada a alguns fatores observados na condução do estudo como, por exemplo: (i) o número de características utilizadas refletem diretamente na precisão do oráculo; (ii) características que se complementam tornam o oráculo mais preciso; e (iii) abordagens de definição de limiar baseadas no desvio padrão produzem melhores resultados. As evidências demonstraram que oráculos de teste baseados em características podem ser utilizados como uma estratégia promissora para automatizar a atividade de validação de objetos complexos, especificamente de imagens sintéticas 3D de vasos sanguíneos.

A seguir é apresentado um estudo realizado com o objetivo de avaliar a precisão de oráculos humanos em julgar corretamente a similaridade de imagens sintéticas 3D de vasos sanguíneos, após aplicação de níveis de ruído. Tal estudo foi conduzido objetivando avaliar se há equivalência entre o julgamento dos oráculos de teste baseados em características e dos oráculos humanos.

5.4 Oráculos humanos para avaliação da qualidade de sistemas que geram imagens sintéticas 3D de vasos sanguíneos

O presente experimento apresenta a condução de um estudo que se destina a avaliar a precisão dos oráculos humanos durante a avaliação da similaridade de imagens sintéticas 3D de vasos sanguíneos após aplicação de níveis de ruído gaussiano. O experimento visa replicar o estudo apresentado na Seção 5.3, avaliando a eficiência dos oráculos humanos no julgamento correto da similaridade entre as imagens modelo com as imagens a serem avaliadas em comparação a oráculos automatizados. Para tanto, as imagens definidas como modelo e as imagens definidas como imagens a serem avaliadas são as mesmas imagens geradas no estudo da Seção 5.3.

Para condução do estudo, as imagens foram inseridas em um formulário de pesquisa implementado por meio da ferramenta *Google Forms*, e o link de hospedagem foi distribuído por email a todos os possíveis participantes. Para inserção das imagens no formulário de pesquisa, foi gerada uma versão das imagens na extensão *gif*. Tal processo foi realizado com o objetivo de possibilitar que o oráculo humano pudesse ter uma visão em 360° da estrutura da imagem, possibilitando o julgamento sobre a similaridade da imagem modelo com as imagens a serem avaliadas, já que se tratam de imagens 3D.

Durante o preenchimento do formulário de pesquisa, os participantes foram orientados a selecionar as imagens que foram mais similares à imagem modelo, levando em consideração a similaridade entre as características por meio do aspecto visual. O desempenho dos oráculos humanos é avaliado por meio do julgamento correto das imagens a partir das métricas de precisão, revocação e especificidade. Por fim, avaliações estatísticas foram realizadas objetivando comparar o desempenho do julgamento do oráculo humano em relação aos oráculos de teste baseados em características (Seção 5.3).

Em seguida, são apresentados os detalhes técnicos dos materiais e métodos utilizados na condução do experimento (Seção 5.4.1), seleção dos sujeitos (Seção 5.4.2), questões de pesquisa e métricas (Seção 5.4.3), formulação de hipóteses (Seção 5.4.4), as etapas para condução do estudo (Seção 5.4.5), os resultados obtidos (Seção 5.4.6), as ameaças à validade (Seção 5.4.7) e as conclusões (Seção 5.4.8).

5.4.1 Materiais e métodos

Nessa seção, são apresentados detalhes técnicos dos materiais e métodos utilizados para condução do estudo.

5.4.1.1 Imagens sintéticas 3D de vasos sanguíneos

As imagens utilizadas para condução do estudo são as mesmas imagens sintéticas 3D de vasos sanguíneos, geradas a partir do método proposto por Galarreta-Valverde *et al.* (2013). Para inserir as imagens em um formulário web, utilizado na condução do experimento, foi gerada uma imagem na extensão *gif* para cada imagem do conjunto de dados. Tal processo foi realizado com o objetivo de possibilitar que o oráculo humano pudesse ter uma visão em 360° da estrutura da imagem, possibilitando o julgamento sobre a similaridade da imagem modelo com as imagens a serem testadas.

Para geração de uma versão *gif* das imagens do conjunto de dados, foi utilizado o *plugin 3D Viewer* da ferramenta Fiji. O *plugin 3D Viewer* oferece funcionalidades de visualização aceleradas por hardware para pilhas de imagens, usando a biblioteca Java 3D. As pilhas podem ser exibidas como renderizações de volume, superfícies ou ortográficos baseados em textura. Além disso, o *plugin* disponibiliza uma funcionalidade de rotação em 360° dos objetos para geração de vídeos e *gifs*.

5.4.1.2 Formulário de pesquisa

Com o objetivo de recrutar pessoas para avaliar a similaridade entre as imagens, foi implementado um formulário de pesquisa *web* para a condução do estudo. O formulário de pesquisa foi implementado por meio da ferramenta *Google Forms*, e o *link* de hospedagem foi distribuído por email a todos os possíveis participantes. Todos os dados colhidos a partir do estudo podem ser apreciados a partir de uma planilha *online* disponibilizada na *web*⁴. Para alcançar uma quantidade razoável de participantes, o estudo avaliou dados de, no mínimo, 30 participantes.

O formulário de pesquisa é formado por um conjunto de arquivos de imagem em formato *gif*, com uma imagem definida como modelo ou referência, sem ruído, e outras 48 imagens com seis diferentes níveis de ruído gaussiano, definidas como imagens a serem testadas. Os participantes foram aconselhados a visualizar as imagens quantas vezes fosse necessário. Considerando os arquivos de imagem, levou-se em conta que a imagem sem ruído, posicionada de forma centralizada e em maior tamanho na página, definida como modelo ou referência e as demais imagens, menores e posicionadas em formato *grid*, foram definidas como as imagens a serem avaliadas. Após a visualização da imagem modelo, os participantes marcaram as imagens no formulário mais similares à imagem modelo, levando em consideração o aspecto visual.

A Figura 43 apresenta as quatro etapas para preenchimento completo do formulário de pesquisa. Na primeira etapa, os oráculos humanos eram informados sobre o contexto do experimento, as orientações fundamentais para o preenchimento correto e informações úteis como, por exemplo: (i) nome; (ii) email; (iii) área de atuação; (iv) tempo em que trabalha na

⁴ <https://goo.gl/fXLqaJ>

área de atuação; (v) e o horário de início de preenchimento do formulário. Em seguida, na segunda etapa, disponibilizou-se um exemplo com duas imagens similares, uma imagem modelo, sem ruído, e outra imagem com ruído. A terceira etapa consiste na execução do julgamento da similaridade entre as imagens. Esta etapa foi composta por oito páginas, cada página com uma imagem modelo e 48 imagens a serem avaliadas. Por fim, o oráculo humano fornecia informações relacionadas ao horário de término do preenchimento do formulário e as dificuldades identificadas na realização do experimento.

Figura 43 – Screenshots das etapas de preenchimento do formulário.

Fonte: Elaborada pelo autor.

Para evitar qualquer tipo de viés no julgamento das imagens, o oráculo humano não foi informado sobre quantidade de imagens com ruído para cada imagem modelo. Além disso, para evitar viés no posicionamento das imagens a serem julgadas, elas foram aleatoriamente posicionadas no formulário.

5.4.2 Seleção de sujeitos

O link para acesso ao formulário de pesquisa foi distribuído por email a um grupo de pessoas para participarem do experimento. O email foi enviado a listas de email em programas de pós-graduação em diferentes áreas de pesquisa. O trabalho está inserido na área de teste de software e processamento de imagens médicas. No entanto, o julgamento sobre a similaridade das imagens leva em consideração apenas o aspecto visual do ser humano em identificar aspectos comuns nas imagens que possam facilitar o julgamento sobre a similaridade entre elas. Desse modo, não houve restrição de área de pesquisa ou perfil dos possíveis participantes no experimento.

A identidade dos participantes foi mantida em sigilo. Todos os participantes foram previamente informados da natureza do experimento. Por sua vez, uma descrição dos objetivos e considerações do experimento foram incluídos no formulário com a finalidade de oferecer ao participante uma fundamentação teórica sobre o experimento. Além disso, foi incluído um exemplo com duas imagens similares no formulário, apresentando ao participante aspectos

semelhantes nas imagens para auxiliá-lo a identificar semelhanças entre as demais imagens do estudo. A realização do experimento não exigiu um conhecimento prévio do participante, assim, não foi incluído nenhum critério de restrição para validação do estudo a não ser a capacidade de visualização e o julgamento do participante.

A Figura 44 ilustra as diferentes áreas de pesquisa identificadas durante a análise dos oráculos humanos. As áreas de pesquisa foram divididas em seis grupos: o grupo A corresponde à área de processamento de imagens (5/45); o grupo B corresponde à área de aprendizado de máquina (4/45); o grupo C corresponde à área de inteligência artificial (5/45); o grupo D corresponde à área de visualização (2/45); o grupo E corresponde à área de ES e, mais precisamente, teste de software (17/45); e o grupo F corresponde a áreas de pesquisa em geral como, por exemplo, ciência de dados, retorno de informações, sistemas de informação, banco de dados, otimização, redes de computadores, informática na educação, teoria da computação, internet das coisas, física computacional e combinatória (12/45).

Figura 44 – Gráfico de pizza com áreas de pesquisa dos oráculos humanos.

Fonte: Elaborada pelo autor.

Em relação ao tempo de atuação na área de pesquisa indicada, os oráculos humanos tinham que marcar uma das seguintes opções: de 1 a 2 anos (21/45); de 2 a 3 anos (5/45); de 3 a 4 anos (9/45); e mais de 4 anos (10/45). A Figura 45 ilustra a distribuição, em porcentagem, do tempo de atuação na área de pesquisa indicada pelos oráculos humanos. Por fim, os oráculos humanos preencheram o formulário de pesquisa com uma média de 21 minutos.

5.4.3 Questões de pesquisa e métricas

Este estudo foi definido para responder uma única Questão de Pesquisa (QP). A QP e as métricas utilizadas são descritas a seguir:

- **QP1:** Oráculos humanos apresentam alta precisão no julgamento correto da similaridade entre imagens modelo e as respectivas imagens após aplicação de ruído em relação a oráculos de teste baseados em características?

Figura 45 – Gráfico de pizza com o tempo de atuação na área de pesquisa indicada.

Fonte: Elaborada pelo autor.

A QP1 avalia a efetividade e a precisão dos oráculos humanos em julgar corretamente a similaridade das imagens em relação aos oráculos de teste baseados em características (Seção 5.3). Para evidenciar a precisão dos oráculos humanos em comparação aos oráculos de teste baseados em características, foram avaliados, manualmente, o número de verdadeiros veredictos (VP/VN) e os falsos veredictos (FP/FN) para cada um dos participantes.

No contexto do presente estudo, a precisão indica se as imagens julgadas pelo oráculo humano são similares à imagem modelo (mas não indica se todos os elementos similares foram recuperados). Por outro lado, a revocação indica se todas as imagens que são similares à imagem modelo foram julgadas corretamente pelo oráculo humano (mas não indica de quantos elementos irrelevantes foram recuperados). Por fim, a especificidade indica se as imagens que são irrelevantes (não similares à imagem modelo) não foram julgadas como similares pelo oráculo humano.

Para verificar o número de verdadeiros veredictos (VP/VN) e falsos veredictos (FP/FN), foi avaliado, individualmente, o julgamento dos oráculos humanos por meio do formulário de pesquisa. Como o posicionamento das imagens a serem avaliadas foi realizado por meio de um algoritmo randômico, foi elaborado um documento com a ordem das imagens inseridas no formulário de pesquisa, permitindo identificar o desempenho do julgamento dos oráculos humanos.

5.4.4 Formulação de hipóteses

Em seguida, as hipóteses relacionadas a QP1 são descritas e os testes estatísticos para as hipóteses são apresentados.

Hipótese nula H_0 :

“Oráculos de teste baseados em características (θ_a) e oráculos humanos (θ_h) apresentam equivalência no julgamento correto da similaridade das imagens modelo com as respectivas

imagens após aplicação de ruído gaussiano.”

$$H_0 : \theta_a = \theta_h$$

Hipótese alternativa H_1 :

“Oráculos de teste baseados em característica (θ_a) e oráculos humanos (θ_h) não apresentam equivalência no julgamento correto da similaridade das imagens modelo com as respectivas imagens após aplicação de ruído gaussiano.”

$$H_1 : \theta_a \neq \theta_h$$

A evidência estatística demonstra a precisão de oráculos de teste baseados em características em avaliar sistemas que gerem ou processem imagens sintéticas 3D de vasos sanguíneos em relação a oráculos humanos. Desse modo, o objetivo é avaliar se o uso da abordagem oráculos de teste baseados em características apresenta uma precisão equivalente aos oráculos humanos no julgamento correto entre imagens sintéticas 3D de vasos sanguíneos. O teste estatístico é realizado com base no desempenho do oráculos de teste baseados em características em relação a th_1 , th_2 e th_3 . Por outro lado, o desempenho dos oráculos humanos é realizado por meio do AVG das métricas de precisão, revocação e especificidade.

Para testar a H_0 , foi realizado o *t-test* pareado com nível de significância igual a $\alpha = 0,05$. O *t-test* de duas amostras mede um intervalo de confiança e realiza um teste de hipóteses para a diferença entre duas médias da população quando os desvios padrão são desconhecidos e as amostras são geradas independentemente uma da outra (PECK; OLSEN; DEVORE, 2015).

5.4.5 Condução do experimento

Para levantar evidências experimentais para responder a QP1, o estudo foi realizado seguindo seis etapas, ilustradas na Figura 46 e detalhadas da seguinte forma: inicialmente foi realizado o processo de geração de uma versão *gif* das imagens do conjunto de dados. Para geração das imagens em *gif* foi utilizado o *plugin 3D Viewer* da ferramenta Fiji. Na segunda etapa, o formulário de pesquisa foi implementado por meio da ferramenta *Google Forms*. Nesta etapa, as imagens foram inseridas no formulário de pesquisa e as informações necessárias para entendimento e realização do experimento também foram inseridas. Na terceira etapa, o *link* para acesso ao formulário de pesquisa foi distribuído por email a um grupo de pessoas para participarem do experimento. Desse modo, na quarta etapa, foi iniciada a etapa de execução do experimento.

Na sexta etapa, é realizado o registro do julgamento dos oráculos humanos. Além disso, nessa etapa são realizadas as atividades de análise dos dados, incluindo as análises qualitativas e quantitativas dos resultados. A Figura 47 apresenta a planilha de registro dos dados e verificação

Figura 46 – Etapas para condução do estudo 3.

Fonte: Elaborada pelo autor.

da validade do experimento. A planilha é composta por uma página com a sumarização dos resultados obtidos, assim como também os quatro oráculos humanos excluídos por meio dos critérios de exclusão anteriormente citados. Além disso, a planilha é composta por outras 45 páginas referentes aos 45 participantes do experimento.

Figura 47 – Planilha de análise e registro dos dados do estudo.

The figure displays two screenshots of Microsoft Excel. The left screenshot shows a worksheet titled "Página Inicial" with a large orange cell containing the text "Sumarização dos dados" and a yellow cell below it with the text "Critérios de exclusão". The right screenshot shows a worksheet titled "Páginas Iniciais" with a table of data. The table has columns for "Imagem modelo", "Precisão", "Recall", "Specificity", and "TP". The data rows show results for various models and images, with a final row for "AVG total".

Imagem modelo	Precisão	Recall	Specificity	TP
anurium1	1	0.666666667	1	4
anurium2	0.5	0.666666667	0.904761905	4
nonReguar1	1	0.75	0.952380952	6
nonReguar2	1	0.833333333	1	6
normal1	0.857142857	1	0.974350476	5
normal2	1	0.833333333	1	5
steneos1	1	1	1	6
steneos2	0.74285714	0.833333333	0.952380952	5
AVG total	0.853789187	0.854166667	0.979124886	41

Fonte: Elaborada pelo autor.

5.4.6 Resultados e discussão

Essa seção descreve os resultados obtidos a partir do julgamento dos oráculos humanos sobre a similaridade entre as imagens a serem avaliadas a partir da imagem modelo. Desse modo, o número de verdadeiros veredictos (VP/VN) e falsos veredictos (FP/FN) de cada oráculo humano foram avaliados manualmente. Além disso, para rejeitar ou não a hipótese nula, os

resultados dos testes estatísticos são apresentados. Após validação dos resultados dos oráculos humanos e aplicação dos critérios de exclusão, 45 oráculos humanos foram considerados como válidos para análise dos resultados.

A seguir, são apresentados os resultados obtidos e a resposta para a QP1. Além disso, estudos estatísticos foram realizados para refutar ou aceitar as hipóteses.

5.4.6.1 QP1

A QP1 avalia a efetividade e a precisão dos oráculos humanos em julgar corretamente a similaridade das imagens em relação aos oráculos de teste baseados em características. Para testar H_0 e responder a QP1, foi realizado o *t-test* pareado com nível de significância igual a $\alpha = 0,05$. Com relação à precisão entre o oráculo de teste baseado em característica (th2) com os oráculos humanos (ho1, ho2, ho3,..., ho45), a Tabela 21 representa um cenário comparativo entre o oráculo de teste baseado em característica (th2) e os oráculos humanos para cada imagem do conjunto de dados. Cada linha da Tabela 21 representa o resultado ao se utilizar uma certa imagem como modelo. Por exemplo, a primeira linha da imagem “aneurism1” foi utilizada como modelo. Os 45 oráculos humanos apresentaram uma média de precisão, revocação e especificidade iguais a, respectivamente, 0,73201, 0,72380 e 0,94648. Por outro lado, o oráculo automatizado (th2) apresentou uma precisão, revocação e especificidade iguais a, respectivamente, 1, 1 e 1.

Tabela 21 – Desempenho dos oráculos humanos em relação ao oráculo automatizado (th2).

X Imagem	Oráculos humanos			Oráculo automatizado		
	Precisão	Revocação	Especificidade	Precisão	Revocação	Especificidade
aneurism1	0,68961	0,72222	0,93015	1,00	1,00	1,00
aneurism2	0,60749	0,67777	0,92010	1,00	1,00	1,00
nonRegular1	0,47701	0,70740	0,88941	1,00	1,00	1,00
nonRegular2	0,78791	0,73703	0,96031	1,00	0,50	1,00
normal1	0,97804	0,87936	0,99629	1,00	0,66	1,00
normal2	0,87388	0,72962	0,98465	1,00	0,66	1,00
stenose1	0,76248	0,65925	0,95608	1,00	1,00	1,00
stenose2	0,67966	0,67777	0,93492	1,00	1,00	1,00
Média	0,73201	0,72380	0,94648	1,00	0,85	1,00

Fonte: Elaborada pelo autor.

A partir dos dados apresentado na Tabela 21, foi realizado o *t-test* pareado a partir da média dos oráculos humanos, de cada métrica, em relação ao respectivo desempenho do oráculo automatizado (th2). Por exemplo, para evidenciar se há uma diferença estatística em relação à métrica de precisão, foi considerado as oito médias do julgamento dos oráculos humanos e os oito valores de precisão do oráculo automatizado. O *t-test* pareado foi conduzido a partir de cada métrica, avaliando se há diferença estatística ou não entre o oráculo automatizado e os oráculos humanos.

A Tabela 22 ilustra que em um cenário (revocação), o julgamento do oráculo de teste baseado em características e dos oráculos humanos são equivalentes. Esse cenário demonstra que

a média da revocação do oráculo de teste baseado em características e dos oráculos humanos são equivalentes. Por outro lado, em dois cenários (precisão e especificidade) não houve uma equivalência entre os julgamentos, demonstrando que para esses casos o oráculo de teste baseado em características e os oráculos humanos apresentaram uma diferença estatisticamente significativa.

Tabela 22 – *T-test* entre o oráculo de teste baseado em características e os oráculos humanos.

Métrica	<i>p-value</i>	<i>t-value</i>	H ₀
Precisão	0.0018	4.8671	Rej.
Revocação	0.2000	1.4151	Ac.
Especificidade	0.0035	4.3245	Rej.

Fonte: Elaborada pelo autor.

Em outras palavras, no contexto do presente estudo, o oráculo de teste baseado em características apresentou alto desempenho em julgar imagens como corretas/similares para as imagens modelos esperadas. Os resultados apresentados na Tabela 22 demonstram que o oráculo de teste baseado em características tomou decisões equivalentes e melhores que os oráculos humanos, evidenciando a eficiência no uso da abordagem oráculos de teste baseados em característica na avaliação de sistemas que gerem ou processem imagens sintéticas 3D de vasos sanguíneos.

5.4.7 Ameaças à validade

Ameaças à validade interna: a principal ameaça à validade interna do experimento está relacionada à seleção dos participantes. Os participantes do estudo, em sua maioria, são ex-colegas de trabalho e membros do mesmo grupo de pesquisa do pesquisador. Para minimizar esse risco, tanto nos instrumentos utilizados no estudo como na elaboração realizada no início do estudo, buscou-se deixar claro que os participantes devem ser imparciais na sua análise.

Uma ameaça à validade interna extra é o caso do estudo não ser executado em um único dia por todos os participantes. Desse modo, o resultado dos participantes que iriam realizar o estudo poderia sofrer influência por meio da troca de informações entre os participantes que já haviam realizado o estudo. Para mitigar essa ameaça, não foram disponibilizadas informações mais técnicas sobre a execução do estudo ou aplicação da quantidade de níveis de ruído nas imagens. Além disso, foram estabelecidos critérios de inclusão e exclusão objetivando mitigar esse tipo de ameaça.

Ameaças à validade externa: a principal ameaça à validade externa está relacionada ao conhecimento dos participantes nas imagens ou tema de pesquisa do estudo. Para mitigar essa ameaça, não houve restrição de participantes de uma determinada área de pesquisa no estudo e foram incluídas informações sobre o contexto do trabalho no formulário de pesquisa para informar todos os participantes sobre a natureza do estudo.

Uma ameaça à validade externa extra está relacionada à geração das imagens em *gif*. As imagens em *gif* foram geradas a partir da projeção em *texture2D* das imagens modelo e das imagens com ruído. Tal projeção foi aplicada para evitar que o nível de ruído prejudicasse, drasticamente, a qualidade das imagens e inviabilizasse o julgamento dos oráculos humanos. Desse modo, não foi utilizada a projeção MIP (Projeção de Intensidade Máxima) que é recomendada para destacar partes importantes da imagem e, no contexto desse estudo, visualizar com maior nitidez o ruído nas imagens. Para mitigar essa ameaça, o processo de aplicação da projeção nas imagens e a geração das imagens em *gif* foi supervisionado por especialistas em Processamento de Imagens Médicas.

Ameaças à validade de construção: a principal ameaça à validade de construção deste estudo está associada com um número alto de imagens, não sendo o ideal, pois pode impactar diretamente na rigorosidade dos participantes, tornando o processo de julgamento tedioso. Uma forma de diminuir a falha relacionada à rigorosidade dos participantes, é estabelecer critérios de inclusão e exclusão.

Ameaças à validade de conclusão: uma possível ameaça à validade de conclusão do estudo está relacionada ao tamanho da amostra ou quantidade de oráculos humanos. Para mitigar esse tipo de ameaça, foi estabelecido um número mínimo, 30, de participantes para estudo. É possível afirmar que este estudo possui uma alta validade de conclusão devido a três razões: (i) a condução do experimento foi apresentada em detalhes, destacando e justificando todas as decisões; (ii) os resultados foram disponibilizados por meio de tabelas, e eles estão disponíveis *on-line* para futuros estudos; e (iii) alta relevância estatística dos testes que apoia as conclusões.

5.4.8 Considerações sobre o experimento 3

O presente estudo objetivou avaliar a precisão e eficiência de oráculos humanos na avaliação de imagens sintéticas 3D de vasos sanguíneos após aplicação de diferentes níveis de ruído gaussiano em relação a oráculos de teste baseados em características. O estudo utilizou subsídios gerados por meio dos resultados do estudo apresentado na Seção 5.3, objetivando avaliar se o oráculo de teste baseado em características apresenta um desempenho equivalente ao dos oráculos humanos no julgamento correto da similaridade das imagens.

A precisão dos oráculos humanos foi avaliada por meio das métricas de precisão, revocação e especificidade, a partir do número de verdadeiros veredictos (VP/VN) e falsos veredictos (FP/FN). Além disso, para evidenciar se há equivalência entre oráculos automatizados e oráculos humanos, foi realizado o teste estatístico *t-test* pareado.

Com base nos resultados obtidos, foi observado que o oráculo de teste baseado em características apresentou equivalência aos oráculos humanos em um cenário (revocação) no julgamento correto da similaridade das imagens sintéticas 3D de vasos sanguíneos após aplicação de diferentes níveis de ruído. O estudo evidenciou uma diferença estatística na métrica de

precisão e especificidade, demonstrando que as abordagens apresentaram diferença estatística na capacidade em prever corretamente as imagens similares à imagem modelo e ausência da condição para casos que realmente não a tem. Desse modo, os resultados evidenciam que os oráculos de teste baseados em características podem ser usados como uma estratégia promissora para automatizar a atividade de validação de objetos complexos, especificamente de imagens sintéticas 3D de vasos sanguíneos.

5.5 Considerações finais

Este capítulo apresentou três avaliações experimentais sobre a aplicabilidade da abordagem oráculos de teste baseados em características de sistemas que gerem ou processem imagens sintéticas 3D de vasos sanguíneos. O estudo apresentado na Seção 5.2 avaliou o uso da abordagem em categorizar automaticamente imagens sintéticas 3D e 2D de vasos sanguíneos com base na similaridade entre características. O estudo possibilitou avaliar a precisão da abordagem em julgar a similaridade das imagens 3D em relação às imagens 2D de vasos sanguíneos e identificar particularidades no julgamento das imagens. Além disso, avaliou a precisão da abordagem em relação a classificadores randômicos, demonstrando que a abordagem possui uma diferença considerada estatisticamente significativa em relação ao número de verdadeiros veredictos (VP/VN).

O estudo apresentado na Seção 5.3 objetivou avaliar a capacidade e precisão da abordagem na avaliação de imagens sintéticas 3D de vasos sanguíneos após aplicação de diferentes níveis de ruído gaussiano. O estudo utilizou subsídios gerados por meio do estudo apresentado na Seção 5.2 na implementação de oráculos mais precisos em julgar a similaridade das imagens. Assim, foi avaliado o desempenho do oráculo a partir da definição de três configurações de limiares, demonstrando, estatisticamente, a eficácia da abordagem na avaliação das imagens.

Por fim, o estudo apresentado na Seção 5.4 avaliou os esforços e a precisão de oráculos humanos em julgar corretamente a similaridade de imagens sintéticas 3D de vasos sanguíneos após aplicação de diferentes níveis de ruído gaussiano em relação a oráculos de teste baseados em características. Com base nos resultados, oráculos de teste baseados em características e oráculos humanos apresentaram diferença estatística no julgamento da similaridade das imagens em relação às métricas de precisão e especificidade. Com relação à métrica de revocação, o oráculo de teste baseados em característica e os oráculos humanos apresentaram uma equivalência estatística. O estudo evidenciou a eficácia da abordagem em automatizar tarefas de validação em sistemas que gerem ou processem imagens sintéticas 3D de vasos sanguíneos.

Com base nos resultados obtidos neste capítulo, por meio dos estudos experimentais, é demonstrada a aplicabilidade e a eficiência da abordagem oráculos de teste baseados em características em ambientes de teste. Além disso, a abordagem mostrou-se ser uma estratégia válida e flexível. Os resultados apresentados também podem contribuir como referência para

estudos experimentais futuros que objetivam estabelecer critérios que evidenciam vantagens e desvantagens de novos métodos ou abordagens na ES e, mais precisamente, na área de teste de software.

CONCLUSÕES E TRABALHOS FUTUROS

Domínios de saída complexa limitam a aplicabilidade das atividades de teste de software automatizado (CHAN; TSE, 2013). Devido à complexidade associada com os dados processados pelo SUT, técnicas de teste tradicionais são ineficientes e, muitas vezes, impraticáveis (OLIVEIRA *et al.*, 2014). Assim, estratégias para a automação de teste em sistemas com saídas não-triviais são um desafio. A complexidade associada à saída do SUT leva ao emprego de estratégias subjetivas e *ad-hoc* de teste (HINTERLEITNER *et al.*, 2011).

Neste trabalho de mestrado buscou-se relatar as experiências do uso da abordagem de oráculos de teste baseados em características em avaliar sistemas que gerem ou processem imagens médicas sintéticas 3D de vasos sanguíneos. Em geral, sistemas que manipulem imagens médicas sintéticas 3D de vasos sanguíneos são fundamentais no auxílio à área médica contribuindo na antecipação do diagnóstico de determinadas patologias (GALARRETA-VALVERDE *et al.*, 2013). No entanto, sistemas que geram ou manipulam imagens médicas 3D têm problemas recorrentes de qualidade devido à falta de estratégias de teste automatizado. Assim, é essencial o uso de ferramentas automatizadas que auxiliem o profissional de saúde no diagnósticos de possíveis anomalias ou patologias no sistema circulatório, fator cada vez mais comum em países desenvolvidos e em desenvolvimento (CHAGAS *et al.*, 2009).

A seguir são apresentadas e discutidas as contribuições do presente trabalho (Seção 6.1), discussão das limitações (Seção 6.2), trabalhos futuros que podem ser e estão sendo conduzidos a partir dos resultados obtidos (Seção 6.3) e as considerações finais (Seção 6.4).

6.1 Contribuições

Por meio da condução do presente trabalho, a abordagem de oráculos de teste baseados em características é proposta como uma estratégia para apoiar a automatização de oráculos de teste para sistemas que processem imagens médicas 3D de vaso sanguíneos. Para tanto, extensões

e configurações foram realizadas na arquitetura do *framework* O-Flm/CO como suporte adequado aos oráculos de teste baseados em características. Além disso, um catálogo de *plug-ins* que representam extratores de características para imagens médicas 3D de vasos sanguíneos foram desenvolvidos e disponibilizados como contribuição do presente trabalho.

Um MS foi conduzido objetivando identificar abordagens e estratégias para o teste de software e análise de qualidade em sistemas biomédicos, sintetizando a evidência disponível e identificando os principais problemas que envolvem avaliação de qualidade nesse domínio de pesquisa. Como resultado principal, o MS demonstrou que o desenvolvimento de estratégias de validação para sistemas biomédicos ocorreu com mais frequência recentemente.

Com base nos estudos experimentais apresentados nesta dissertação, é demonstrada a aplicabilidade e eficiência da abordagem de oráculos de teste baseados em características para sistemas em ambientes de teste, mostrando ser uma abordagem flexível. Os resultados apresentados podem contribuir como referência para a condução de estudos experimentais futuros que evidenciem vantagens e desvantagens da abordagem.

A partir dos resultados gerados no presente trabalho, é possível destacar a efetividade da abordagem oráculos de teste baseados em características como uma estratégia promissora para sistemas com saídas complexas. Sistemas que manipulem imagens médicas 3D carecem de qualidade e as atividades de teste automatizado são essenciais. Desse modo, o presente trabalho contribui na aplicabilidade de estratégias de teste de sistemas que processem imagens médicas 3D.

6.2 Discussão das limitações

Em seguida, são apresentadas e discutidas uma revisão das limitações identificadas na condução do presente trabalho:

- **Imagens no formato NIfTI:** as imagens médicas 3D utilizadas neste trabalho são no formato NIfTI. O formato de imagens NIfTI foi projetado há cerca de uma década com o objetivo de substituir o então amplamente difundido, mas problemático, formato de arquivo 7.5. A escolha por esse formato está relacionada à experiência dos especialistas na área de processamento de imagens médicas envolvidos na supervisão dos estudos experimentais conduzidos. Desse modo, os materiais resultantes como contribuição no presente trabalho foram utilizados em imagens no formato NIfTI;
- **Extratores de características:** a escolha das características relevantes nas imagens e, consequentemente, da implementação dos extratores de características foram realizadas com a supervisão de especialistas na área de processamento de imagens médicas. Nesse cenário, os estudos experimentais realizados priorizaram a opinião dos especialistas em

indicar e sugerir características relevantes nas imagens e metodologias de implementação dos extratores de características;

- **Imagens sintéticas 3D de vasos sanguíneos:** um dos desafios enfrentados na condução do presente trabalho foi a dificuldade de obtenção de conjuntos de dados de imagens médicas 3D reais. Para alcançar este objetivo, foi iniciada uma parceria com o IME (Instituto de Matemática e Estatística) da Universidade de São Paulo (USP) e com o Hospital Albert Einstein. No entanto, por motivos burocráticos, tal parceria não foi efetivada até o fim da presente pesquisa. Desse modo, os estudos experimentais foram conduzidos para avaliar apenas imagens sintéticas 3D de vasos sanguíneos; e
- **Estudos experimentais:** devido à limitação na obtenção de conjuntos de dados de imagens médicas 3D reais, os estudos empíricos conduzidos neste trabalho foram realizados com o objetivo de avaliar sistemas que gerem ou processem imagens sintéticas 3D de vasos sanguíneos. A utilização de imagens reais poderia enriquecer a aplicabilidade da abordagem e identificar vantagens e desvantagens na aplicação da abordagem em sistemas reais.

6.3 Trabalhos futuros

A flexibilidade e aplicabilidade da abordagem oráculos de teste baseados em características possibilita a sua aplicação em outros domínios de sistemas de saída complexa. Desse modo, a condução do presente trabalho de mestrado permitiu a identificação de algumas direções futuras para a abordagem baseada em características. Tais trabalhos futuros podem ser descritos como segue:

- **Aplicação da abordagem em sistemas que gerem ou processem imagens médicas reais:** os estudos experimentais conduzidos no presente trabalho foram realizados exclusivamente com imagens médicas sintéticas 3D. Durante o período de mestrado foi iniciada uma parceria entre o IME/USP e o Hospital Israelita Albert Einstein objetivando a coleta de conjunto de dados de imagens médicas reais de vasos sanguíneos. A obtenção do conjunto de dados iria possibilitar a condução de mais estudos experimentais para reforçar a aplicabilidade da abordagem em ambientes reais. No entanto, o conjunto de dados não foi obtido até o final da pesquisa e assim os estudos experimentais foram realizados, exclusivamente, para avaliar as imagens sintéticas 3D de vasos sanguíneos;
- **Avaliação da aplicabilidade de métodos de aprendizagem de máquina na abordagem:** um dos trabalhos que está sendo conduzido a partir da abordagem oráculos de teste baseados em características é a avaliação de técnicas de classificação de imagens. Assim, um trabalho está sendo conduzido objetivando avaliar métodos de aprendizagem de máquina que possam ser comparados com a eficiência da abordagem oráculos de teste baseados em características; e

- **Avaliação de técnicas de geração de escoamento de fluido:** a arquitetura do *framework* O-FIm/CO foi adaptada para suporte de teste para imagens médicas 3D. No entanto, a arquitetura também suporta a condução de atividades de teste para sistemas que geram ou processam objetos 3D. Assim, um trabalho vem sendo conduzido com objetivo de avaliar técnicas de geração de escoamento de fluido em objetos 3D.

Além dos pontos destacados, novos domínios de pesquisa podem ser identificados por meio da condução de um mapeamento sistemático para aplicação da abordagem oráculos de teste baseados em características.

6.4 Considerações finais

Esta dissertação apresentou uma estratégia que visa automatizar oráculos de teste para imagens médicas em modelos 3D por meio da abordagem oráculos de teste baseados em características. Sistemas que processam imagens médicas em modelos 3D apresentam problemas recorrentes de qualidade devido à falta de estratégias de teste automatizado, no entanto, são importantes na evolução da área médica, contribuindo para diagnósticos precoces e mais precisos em diversos segmentos da área médica. Como ferramenta de suporte à automação de oráculos de teste baseados em características, foi utilizada a estrutura de *framework* O-FIm/CO para automatizar o processo acima mencionado. Além disso, por meio da estratégia aplicada, é possível reduzir esforços de testadores que têm a função de observar e detectar erros em sistemas que processem imagens médicas 3D.

Espera-se que os resultados apresentados nesta dissertação instigue outros pesquisadores a dedicar esforços em aproximar conceitos e estratégias de validação da área de ES que possam ser aplicados em sistemas com saída complexa. Além disso, a abordagem apresentada pode instigar outros pesquisadores a propor melhoras na aplicação em sistemas do domínio, aplicando a abordagem na avaliação da qualidade das saídas de tais sistemas.

REFERÊNCIAS

- AMMANN, P.; OFFUTT, J. **Introduction to software testing**. [S.l.]: Cambridge University Press, 2008.
- ASRAFI, M.; LIU, H.; KUO, F. On testing effectiveness of metamorphic relations: A case study. In: **Proceedings of the 5th International Conference on Secure Software Integration and Reliability Improvement (SSIRI)**. [S.l.: s.n.], 2011. p. 147–156.
- BANERJEE B. NGUYEN, V. G. I.; MEMON, A. Advances in gui testing. In: MEMON, A. (Ed.). [S.l.]: Elsevier, 2003, (Advances in Computers, v. 58). p. 149–201.
- BANERJEE, I.; NGUYEN, B.; GAROUSI, V.; MEMON, A. Graphical user interface (gui) testing: Systematic mapping and repository. **Information and Software Technology**, p. 1679–1694, 2013.
- BARBOSA, E. F.; CHAIM, M. L.; VINCENZI, A. M. R.; DELAMARO, M. E.; JINO, M.; MALDONADO, J. C. Capítulo 4 - Teste Estrutural. In: DELAMARO, M. E.; MALDONADO, J. C.; JINO, M. (Ed.). **Introdução ao Teste de Software**. 2. ed. Rio de Janeiro: Campus, 2016. p. 59–87.
- BARR, E. T.; HARMAN, M.; MCMINN, P.; SHAHBAZ, M.; YOO, S. The oracle problem in software testing: A survey. **IEEE Transactions on Software Engineering**, p. 507–525, 2015.
- BERGAMASCO, L. C.; OLIVEIRA, R. A.; WECHSLER, H.; DAJUDA, C.; DELAMARO, M.; NUNES, F. L. Content-based image retrieval of 3d cardiac models to aid the diagnosis of congestive heart failure by using spectral clustering. In: **Proceedings of the 28th International Symposium on Computer-Based Medical Systems (CBMS)**. [S.l.: s.n.], 2015. p. 183–186.
- BERTOLINO, A. Software testing research and practice. In: **Proceedings of the Abstract State Machines**. [S.l.: s.n.], 2003. p. 1–21.
- BERTOLINO, A. Software testing research: Achievements, challenges, dreams. In: **Proceedings of the Future of Software Engineering**. [S.l.: s.n.], 2007. p. 85–103.
- BRIAND, L.; LABICHE, Y. A uml-based approach to system testing. **Software and Systems Modeling**, p. 10–42, 2002.
- CAO, Y.; ZHOU, Z. Q.; CHEN, T. Y. On the correlation between the effectiveness of metamorphic relations and dissimilarities of test case executions. In: **Proceedings of the 13th International Conference on Quality Software (QSIC)**. [S.l.: s.n.], 2013. p. 153–162.
- CHAGAS, A. C. P.; ZILLI, E. C.; FERREIRA, J. F. M.; MORETTI, M. A.; RAMOS, R. F. Saúde cardiovascular do homem brasileiro—visão da sociedade brasileira de cardiologia. **Arq Bras Cardiol**, p. 584–587, 2009.
- CHAN, W. K.; TSE, T. H. Oracles are hardly attain'd, and hardly understood: confessions of software testing researchers. In: **Proceedings of the 13th International Conference on Quality Software (QSIC)**. [S.l.: s.n.], 2013. p. 245–252.

- CHEON, Y. Abstraction in assertion-based test oracles. In: **Proceedings of the 7th International Conference on Quality Software**. [S.l.: s.n.], 2007. p. 410–414.
- COHEN, K. B.; HUNTER, L. E.; PALMER, M. Assessment of software testing and quality assurance in natural language processing applications and a linguistically inspired approach to improving it. In: **Trustworthy Eternal Systems via Evolving Software, Data and Knowledge**. [S.l.: s.n.], 2013. p. 77–90.
- DATTA, R.; JOSHI, D.; LI, J.; WANG, J. Z. Image Retrieval: Ideas, Influences, and Trends of the New Age. **ACM Computing Surveys (CSUR)**, p. 1–60, 2008.
- DELAMARO, M. E.; MALDONADO, J. C.; JINO, M. Capítulo 1 – Conceitos Básicos. In: DELAMARO, M. E.; MALDONADO, J. C.; JINO, M. (Ed.). **"Introdução ao Teste de Software"**. 2. ed. Rio de Janeiro: Campus, 2016. p. 1–7.
- DELAMARO, M. E.; NUNES, F. L. S.; OLIVEIRA, R. A. P. Using concepts of content-based image retrieval to implement graphical testing oracles. **Software Testing, Verification and Reliability**, p. 171–198, 2013.
- DELAMARO, M. E.; OLIVEIRA, R. A. P.; BARBOSA, E. F.; MALDONADO, J. C. Capítulo 5 – Teste de Mutação. In: DELAMARO, M. E.; MALDONADO, J. C.; JINO, M. (Ed.). **"Introdução ao Teste de Software"**. 2. ed. Rio de Janeiro: Campus, 2016. p. 93–136.
- DEMILLO, R. A. Completely validated software: test adequacy and program mutation (panel session). In: **Proceedings of the 11th International Conference on Software Engineering**. [S.l.: s.n.], 1989. p. 355–356.
- DEMILLO, R. A.; LIPTON, R. J.; SAYWARD, F. G. Hints on test data selection: Help for the practicing programmer. **IEEE Computer Society Press**, p. 34–41, 1978.
- D'ERRICO, V.; POTENA, L.; FIORE, D.; FABBRI, F.; GRIGIONI, F.; MAGNANI, G.; ORTOLANI, P.; BIANCHI, I.; CORAZZA, I.; ZANNOLI, R. Reproducibility of ivus measurements in heart transplant recipients: increased quality of data by using dedicated software for image analysis. In: **Computers in Cardiology**. [S.l.: s.n.], 2008. p. 537–540.
- DOUGHERTY, G. **Medical image processing: techniques and applications**. [S.l.]: Springer Science & Business Media, 2011.
- DUSTIN, E.; GARRETT, T.; GAUF, B. **Implementing automated software testing: How to save time and lower costs while raising quality**. [S.l.]: Addison-Wesley Professional, 2009.
- ETTEN, W. C. V. **Introduction to Random signals and Noise**. [S.l.]: John Wiley & Sons, 2006.
- FABBRI, S. C. P. F.; VICENZI, A. M. R.; MALDONADO, J. C. Capítulo 2 – Teste Funcional. In: DELAMARO, M. E.; MALDONADO, J. C.; JINO, M. (Ed.). **"Introdução ao Teste de Software"**. 2. ed. Rio de Janeiro: Campus, 2016. p. 9–38.
- FILHO, A. C. S.; RODRIGUES, E. P.; JUNIOR, J. E.; CARNEIRO, A. A. O. A computational tool as support in b-mode ultrasound diagnostic quality control. **Revista Brasileira de Engenharia Biomédica**, p. 402–405, 2014.
- FROUNCHI, K.; BRIAND, L. C.; GRADY, L.; LABICHE, Y.; SUBRAMANYAN, R. Automating image segmentation verification and validation by learning test oracles. **Information and Software Technology**, p. 1337–1348, 2011.

GALARRETA-VALVERDE, M. A. **Three-dimensional synthetic blood vessels generation using stochastic Lindenmayer systems**. Dissertação (Mestrado) — Mathematics and statistics Institute, University of Sao Paulo, São Paulo, Brasil, 2012.

GALARRETA-VALVERDE, M. A.; MACEDO, M. M. G.; MEKKAAOI, C.; JACKOWSKI, M. P. Three-dimensional synthetic blood vessel generation using stochastic L-systems. In: **Proceedings of the Medical Imaging: Image Processing**. [S.l.: s.n.], 2013. p. 86691I–86691I–6.

GATO, H. E. R.; NUNES, F. L. S.; SCHIABEL, H. Uma proposta de recuperação de imagens mamográficas baseada em conteúdo. In: **Proceedings of the IX Congresso Brasileiro de Informática em Saúde**. [S.l.: s.n.], 2004. p. 1–5.

GIBSON, N. M.; DUDDLEY, N. J.; GRIFFITH, K. A computerised quality control testing system for b-mode ultrasound. **Ultrasound in medicine & biology**, p. 1697–1711, 2001.

GLENFORD, J. M.; SANDLER, C.; BADGETT, T. **The Art of Software Testing**. [S.l.]: John Wiley & Sons, Ltd, 2004.

GONZALEZ, R. C.; WOODS, R. C. **Processamento Digital de Imagens**. [S.l.]: Ed Edgard Blücher LTDA. São Paulo, 2000.

HINTERLEITNER, F.; NEITZEL, G.; MÖLLER, S.; NORRENBROCK, C. An evaluation protocol for the subjective assessment of text-to-speech in audiobook reading tasks. In: **Proceedings of the Blizzard Challenge Workshop**. [S.l.: s.n.], 2011. p. 1.

HOFFMAN, D. Using oracles in test automation. **Proceedings of Pacific Northwest Software Quality Conference**, p. 90–117, 2001.

HOFFMAN, D. **Using Oracles in Testing and Test Automation**. 2006. LogiGear – Software Testing. Disponível em: <<http://goo.gl/d3Fb6>>.

HUAI, L.; FEI-CHING, K.; TOWEY, D.; TSONG, Y. C. How effectively does metamorphic testing alleviate the oracle problem? **IEEE Transactions on Software Engineering**, p. 4–22, 2014.

HUNTER, C.; STROOPER, P. Systematically deriving partial oracles for testing concurrent programs. In: **Proceedings of the Australian Computer Science Communications**. [S.l.: s.n.], 2001. p. 83–91.

IEEE. IEEE standard glossary of software engineering terminology. p. 1–84, 1990.

JAVED, A. Z.; STROOPER, P. A.; WATSON, G. N. Automated generation of test cases using model-driven architecture. In: **Proceedings of the 2nd International Workshop on Automation of Software Test (AST)**. [S.l.: s.n.], 2007. p. 3.

JETLEY, R. P.; JONES, P. L.; ANDERSON, P. Static analysis of medical device software using codesonar. In: **Proceedings of the Workshop on Static Analysis**. [S.l.: s.n.], 2008. p. 22–29.

JIA, Y.; HARMAN, M. An analysis and survey of the development of mutation testing. **IEEE Transactions on Software Engineering**, p. 649–678, 2011.

JIANG, F.; SHI, D.; LIU, D. C. Fast adaptive ultrasound speckle reduction with bilateral filter on cuda. In: **Proceedings of the International Conference on Bioinformatics and Biomedical Engineerin (iCBBE)**. [S.l.: s.n.], 2011. p. 1–4.

- KAWAMURA, T.; KIMURA, T.; TSUMOTO, S. Data mining-based service quality estimation in hospital information system. In: **Proceedings of the International Conference on Data Mining Workshop (ICDMW)**. [S.l.: s.n.], 2014. p. 289–295.
- KEELE, S. Guidelines for performing systematic literature reviews in software engineering. In: **Technical report, Ver. 2.3 EBSE Technical Report. EBSE**. [S.l.: s.n.], 2007.
- KITCHENHAM, B. Procedures for performing systematic reviews. **Keele, UK**, p. 1–26, 2004.
- KOHONEN, T. Correlation matrix memories. **IEEE Transactions on Computers**, p. 353–359, 1972.
- LEE, T. C.; JASHYAP, R. L.; CHU, C. N. Building skeleton models via 3-d medial surface axis thinning algorithms. **CVGIP: Graphical Models and Image Processing**, p. 462–478, 1994.
- LIANG, C.; MALATPURE, A.; SHAFIEI, M.; VAGO, M.; ZHENG, T. Customer scenario focused end-to-end testing. In: **Proceedings of the 23rd International Symposium on Software Reliability Engineering Workshops (ISSREW)**. [S.l.: s.n.], 2012. p. 80–81.
- LINDEBERG, T.; ROMENY, B. M. T. H. Linear scale-space i: Basic theory. In: **Geometry-Driven Diffusion in Computer Vision**. [S.l.]: Springer, 1994. p. 1–38.
- MCCABE, T. J. A complexity measure. **IEEE Transactions on software Engineering**, p. 308–320, 1976.
- MCMINN, P.; STEVENSON, M.; HARMAN, M. Reducing qualitative human oracle costs associated with automatically generated test data. In: **Proceedings of the 1th International Workshop on Software Test Output Validation**. [S.l.: s.n.], 2010. p. 1–4.
- MYERS, G. J.; SANDLER, C.; BADGETT, T. **The Art of Software Testing**. 3rd. ed. [S.l.]: Wiley Publishing, 2011. ISBN 1118031962.
- OLIVEIRA, R. A. P. **Apoio à automatização de oráculos de teste para programas com interfaces gráficas**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação (ICMC) – Universidade de São Paulo (USP), São Carlos, SP, 2012.
- OLIVEIRA, R. A. P. Automating test oracles for systems with complex outputs. In: **Proceedings of the 7th International Conference on Software Testing, Verification and Validation Workshops (ICSTW)**. [S.l.: s.n.], 2014. p. 239–240.
- OLIVEIRA, R. A. P. **Test oracles for systems with complex outputs: the case of TTS systems**. Tese (Ph.D. Thesis) — Instituto de Ciências Matemáticas e de Computação (ICMC) – Universidade de São Paulo (USP), São Carlos, SP, 2017.
- OLIVEIRA, R. A. P.; DELAMARO, M. E.; NUNES, F. L. S. O-FIm – oracle for images. In: **Proceedings of the 23th Brazilian Symposium on Software Engineering (SBES)**. [S.l.: s.n.], 2009. p. 1–6.
- OLIVEIRA, R. A. P.; DELAMARO, M. E.; NUNES, F. L. S. Exploring the O-FIm framework to support the test of programs with GUIs. In: **Proceedings of the 4th Brazilian Workshop on Systematic and Automated Software Testing**. [S.l.: s.n.], 2010. p. 31–40.

- OLIVEIRA, R. A. P.; GIL, V. N.; NUNES, F. L. S.; DELAMARO, M. E. An extensible framework to implement test oracle for “non-testable programs”. In: **Proceedings of the 26th International Conference on Software Engineering & Knowledge Engineering**. [S.l.: s.n.], 2014. p. 199–204.
- OLIVEIRA, R. A. P.; KANEWALA, U.; NARDI, P. A. Automated test oracles: State of the art, taxonomies, and trends. In: MEMON, A. (Ed.). [S.l.]: Elsevier, 2014, (Advances in Computers, v. 95). p. 113–199.
- OLIVEIRA, R. A. P.; MEMON, A. M.; GIL, V. N.; NUNES, F. L. S.; DELAMARO, M. E. An extensible framework to implement test oracles for non-testable programs. In: **Proceedings of the 26th International Conference on Software Engineering and Knowledge Engineering (SEKE)**. [S.l.: s.n.], 2014. p. 199–204.
- PAECH, B.; WETTER, T. Rational quality requirements for medical software. In: **Proceedings of the 30th international Conference on Software Engineering**. [S.l.: s.n.], 2008. p. 633–638.
- PECK, R.; OLSEN, C.; DEVORE, J. L. **Introduction to statistics and data analysis**. [S.l.]: Cengage Learning, 2015.
- PETERSON, K.; VAKKALANKA, S.; KUZNIARZ, L. Guidelines for conducting systematic mapping studies in software engineering: An update. **Information and Software Technology**, p. 1–18, 2015.
- PEZZE, M.; ZHANG, C. Automated test oracles: A survey. **Advances in Computers**, p. 1–48, 2014.
- PRACHYABRUED, M.; BORST, C. W. Visual feedback for virtual grasping. In: **Proceeding of the 3D User Interfaces (3DUI)**. [S.l.: s.n.], 2014. p. 19–26.
- PRESSMAN, R. S. **Engenharia de software**. [S.l.]: McGraw Hill Brasil, 2011.
- RAFI, D. M.; MOSES, K. R. K.; PETERSON, K.; MANTYLA, M. V. Benefits and limitations of automated software testing: Systematic literature review and practitioner survey. In: **Proceedings of the 7th International Workshop on Automation of Software Test**. [S.l.: s.n.], 2012. p. 36–42.
- RAPPS, S.; WAYUKER, E. J. Selecting software test data using data flow information. **IEEE Transactions on Software Engineering**, p. 367–375, 1985.
- RORISSA, A. Image retrieval: Benchmarking visual information indexing and retrieval systems. **Bulletin of the American Society for Information Science and Technology**, p. 15–17, 2007.
- SANTOS, A. P. O. **Recuperação de imagens mamográficas baseadas em conteúdo**. Dissertação (Mestrado) — Centro Universitário Eurípides de Marília – UNIVEM, Marília, SP, 2006.
- WANG, Y.; HELMINEN, E.; JIANG, J. Building a virtual breast elastography phantom lab using open source software. In: **Proceedings of the International Ultrasonics Symposium (IUS)**. [S.l.: s.n.], 2014. p. 1841–1844.
- WARD, M. A.; OFORI, E. K.; SCUTT, D.; MOORES, B. M. Experiences of in-field and remote monitoring of diagnostic radiological quality in ghana using an equipment and patient dosimetry database. In: **Proceedings of the World Congress on Medical Physics and Biomedical Engineering**. [S.l.: s.n.], 2009. p. 36–39.

- WEYUKER, E. J. On testing non-testable programs. **The Computer Journal**, p. 465–470, 1982.
- YIN, Y.; LIU, B.; NI, H. Real-time embedded software testing method based on extended finite state machine. **Systems Engineering and Electronics**, p. 276–285, 2012.
- YU, T.; QU, X.; ACHARYA, M.; ROTHERMEL, G. Oracle-based regression test selection. In: **Proceedings of the 6th International Conference on Software Testing, Verification and Validation (ICST)**. [S.l.: s.n.], 2013. p. 292–301.
- ZHENG, K.; VYDISWARAN, V. G. V.; LIU, Y.; WANG, Y.; STUBBS, A.; UZUNER, O.; GURURAJ, A. E.; BAYER, S.; ABERDEEN, J.; RUMSHISKY, A. *et al.* Ease of adoption of clinical natural language processing software: An evaluation of five systems. **Journal of biomedical informatics**, p. S189–S196, 2015.

CÓDIGO-FONTE DA FUNÇÃO PRINCIPAL DOS EXTRATORES DE CARACTERÍSTICAS PARA IMAGENS MÉDICAS 3D

A seguir são apresentados os código-fonte da função principal de cada extrator de característica desenvolvido no presente trabalho de mestrado.

Código-fonte da função principal do extrator de pontos de bifurcação:

Código-fonte 4 – Código-fonte do extrator de quantidade de bifurcações

```

1: package extractor.bifurcation.structure;
2: import java.io.FileNotFoundException;
3: import java.util.ArrayList;
4: import extractor.bifurcation.utils.Neighborhood;
5:
6: /**
7:  * This class the extraction and counting of the amount of
8:  * bifurcation of the image is performed.
9:  * The count is performed based on the amount of neighbors of
10:  * a voxel of the image. The neighbor check function is
11:  * responsible for identifying the neighbors of a given voxel.
12:  *
13:  * @author Misael Jr
14:  */
15: public class Bifurcation {
16:

```

```

17:  ArrayList<Double[]> bifurc_points = new ArrayList<Double
    []>();
18:  ArrayList<Double[]> neighbors;
19:  ArrayList<Double[]> neighborTemp;
20:  Neighborhood neighbor;
21:
22:  /**
23:   * Function that measures the number of bifurcations
24:   * @param data -- Matrix with image points.
25:   * @param nx -- Numbers of points on the x-axis
26:   * @param ny -- Numbers of points on the y-axis
27:   * @param nz -- Numbers of points on the z-axis
28:   * @param dim -- Image size
29:   * @return Number of bifurcations (normalized)
30:   * @throws FileNotFoundException
31:   */
32:
33:  public double extractorCountBifurc(double[][][] data, int
    nx, int ny, int nz, int dim) throws FileNotFoundException
    {
34:    int quantBifurc = 0;
35:    neighbor = new Neighborhood();
36:    double bifurc_normalized = 0;
37:
38:    for (int d = 0; d < dim; d++){
39:      for (int i = 0; i < nx; i++) {
40:        for (int j = 0; j < ny; j++){
41:          for (int k = 0; k < nz; k++){
42:            if (data[i][j][k][d] != 0){
43:              Double[] points = {(double) i, (double) j, (
double) k};
44:              neighbors = neighbor.checkNeighbors26(data,
bifurc_points, i, j, k, d, nx, ny, nz, dim);
45:              if(neighbors.size() > 2){
46:                quantBifurc++;
47:                bifurc_points.add(points); // if it's
bifurcation point, add in the array of bifurcation points
48:              }else{
49:                continue;
50:              }
51:            }else
52:              continue;

```

```

53:         }
54:     }
55: }
56: }
57:     bifurc_normalized = getNormalized(quantBifurc);
58:     return bifurc_normalized;
59: }
60:
61: /**
62:  * set the normalization according to the base of max and
63:  * min - linear scale
64:  * @param bifurcation
65:  * @param min
66:  * @param max
67:  * @return Normalized value
68:  */
69: public double getNormalized(double bifurcation, min, max){
70:     double normal_value = 0;
71:     normal_value = (bifurcation - min)/(max - min);
72:     return normal_value;
73: }

```

Fonte: Elaborada pelo autor.

Código-fonte da função principal do extrator de pontos de terminais:

Código-fonte 5 – Código-fonte do extrator de quantidade de pontos terminais

```

1: package extractor.endpoints.structure;
2: import java.io.FileNotFoundException;
3: import java.util.ArrayList;
4: import extractor.endpoints.utils.Neighborhood;
5:
6: /**
7:  * This class, the extraction and counting of the number of
8:  * end points of the image is performed.
9:  * The count is performed based on the amount of neighbors of
10:  * a voxel of the image. The neighbor check
11:  * function is responsible for identifying the neighbors of a
12:  * given voxel.
13:  *
14:  * @author Misael Jr

```

```

12:  *
13:  */
14:
15: public class EndPoints {
16:
17:     ArrayList<Double[]> endpoints = new ArrayList<Double[]>();
18:     ArrayList<Double[]> neighbors;
19:     ArrayList<Double[]> visitedTemp;
20:     Neighborhood neighbor;
21:
22:     /**
23:      * Function that measures the number of endpoints
24:      * @param data -- Matrix with image points.
25:      * @param nx -- Numbers of points on the x-axis
26:      * @param ny -- Numbers of points on the y-axis
27:      * @param nz -- Numbers of points on the z-axis
28:      * @param dim -- Image size
29:      * @return Number of image endpoints
30:      * @throws FileNotFoundException
31:      */
32:
33:     public double extratorCountBifurc(double[][][] data, int
        nx, int ny, int nz, int dim) throws FileNotFoundException{
34:         int quantTerm = 0;
35:         neighbor = new Neighborhood();
36:         double endpoints_normalized = 0;
37:
38:         for (int d = 0; d < dim; d++){
39:             for (int i = 0; i < nx; i++) {
40:                 for (int j = 0; j < ny; j++){
41:                     for (int k = 0; k < nz; k++){
42:                         if (data[i][j][k][d] != 0){
43:                             Double[] points = {(double) i, (double) j, (
                                double) k};
44:                             neighbors = neighbor.checkNeighbors26(data,
                                endpoints, i, j, k, d, nx, ny, nz, dim);
45:                             if (neighbors.size() == 1){
46:                                 quantTerm++;
47:                                 endpoints.add(points); // if it's an
                                endpoints, add in the array of endpoints
48:                             }else{
49:                                 continue;

```

```

50:         }
51:     }else
52:         continue;
53:     }
54: }
55: }
56: }
57: endpoints_normalized = getNormalized(quantBifurc);
58: return endpoints_normalized;
59: }
60:
61: /**
62:  * set the normalization according to the base of max and
63:  * min - linear scale
64:  * @param endpoints
65:  * @param min
66:  * @param max
67:  * @return Normalized value
68:  */
69: public double getNormalized(double endpoints, min, max){
70:     double value_normal = 0;
71:     value_normal = (endpoints - min)/(max - min);
72:     return value_normal;
73: }
74: }

```

Fonte: Elaborada pelo autor.

Código-fonte da função principal do extrator de quantidade de segmentos:

Código-fonte 6 – Código-fonte do extrator de quantidade de segmentos

```

1: package extractor.segments.structure;
2: import java.io.FileNotFoundException;
3: import java.util.ArrayList;
4: import java.util.Arrays;
5: import java.util.Collections;
6: import java.util.Comparator;
7:
8: /**
9:  * This class, the number of segments of the image is
10:  * measured. The segment extractor uses the bifurcation and

```

```
    termination points as references.
10:  * Thus, a segment of the image is classified as a
    bifurcation point to a terminal point or a point of
    bifurcation to another point of bifurcation.
11:  *
12:  * @author Misael Jr
13:  *
14:  */
15:
16: public class Count_Segmentos {
17:
18:     ArrayList<Double[]> bifurc_points = new ArrayList<Double
        []>();
19:     ArrayList<Double[]> end_points = new ArrayList<Double[]>();
20:     ArrayList<Double[]> A = new ArrayList<Double[]>();
21:     double nrSeg=0;
22:     double size_seg = 0;
23:
24:     /**
25:      * Function that measures the number of segments in an
        image according to the points of bifurcation and
        identified finals.
26:      *
27:      * @param data -- Matrix with image points.
28:      * @param bifurc_points
29:      * @param end_points
30:      * @param nx -- Numbers of points on the x-axis
31:      * @param ny -- Numbers of points on the y-axis
32:      * @param nz -- Numbers of points on the z-axis
33:      * @param dim -- Image size
34:      * @return normalized number of segments.
35:      * @throws FileNotFoundException
36:      */
37:
38:     public double extratorCountSeg(double[][][] data,
        ArrayList<Double[]> bifurc_points, ArrayList<Double[]>
        end_points,
39:         int nx, int ny, int nz, int dim) throws
        FileNotFoundException{
40:
41:         this.bifurc_points = bifurc_points;
42:         this.end_points = end_points;
```

```

43:    // Add bifurcation points and endpoints to an array and
    then sort them.
44:    A.addAll(end_points);
45:    A.addAll(bifurc_points);
46:    sortArray(A);
47:
48:    while(A.size() > 1){
49:        Double[] ponto1 = A.get(0);
50:        ArrayList<Double[]> caminhos;
51:        ArrayList<Double[]> pontoTemp = null;
52:        caminhos = new ArrayList<Double[]>();
53:        int x = ponto1[0].intValue();
54:        int y = ponto1[1].intValue();
55:        int z = ponto1[2].intValue();
56:        A.remove(0); //Extract points from the first coordinate
    and delete from the array
57:
58:        // traverse all points between the first and second
    points of A, classifying as segment
59:        for (int d = 0; d < dim; d++){
60:            for (int i = x; i < nx; i++) {
61:                for (int j = y; j < ny; j++){
62:                    for (int k = z; k < nz; k++){
63:                        if (data[i][j][k][d] != 0){
64:                            pontoTemp = new ArrayList<Double[]>();
65:                            Double[] pontos = {(double) i, (double) j, (
double) k};
66:                            pontoTemp.add(pontos);
67:                            if(check(pontoTemp, A)){ //check if the point
    is present in A. That is, if it is a terminal or
    bifurcation.
68:                                caminhos.add(pontos);
69:                            }else{
70:                                continue;
71:                            }
72:                        }else
73:                            continue;
74:                    }
75:                }
76:            }
77:        }
78:        nrSeg++;

```

```
79:     }
80:     double nrSeg_normalized = getNormalized(nrSeg);
81:     return nrSeg_normalized;
82: }
83:
84: /**
85:  * set the normalization according to the base of max and
86:  * min
87:  * @param nrSeg
88:  * @return Normalized value
89:  */
90: public double getNormalized(double nrSeg, min, max){
91:     double value_normal = 0;
92:     value_normal = (nrSeg - min)/(max - min);
93:     return value_normal;
94: }
95:
96: /**
97:  * Function that checks whether a point is already present
98:  * in another array
99:  * @param array1 -- Array 1
100:  * @param array2 -- Array 2
101:  * @return If the point is present, returns false.
102:  * Otherwise, it returns true.
103:  */
104: public boolean check(ArrayList<Double[]> array1, ArrayList<
105:     Double[]> array2){
106:     for(int i = 0; i < array1.size();i++){
107:         Double[] i1 = array1.get(i);
108:         for(int j = 0; j < array2.size();j++){
109:             Double[] i2 = array2.get(j);
110:             if(Arrays.equals(i1, i2)){
111:                 return false;
112:             }
113:         }
114:     }
115:     return true;
116: }
```

```

117:    * Function that sorts the points of an array of type
        ArrayList <Double []>
118:    * @param array -- Array to be sorted
119:    */
120:
121:    public void sortArray(ArrayList<Double []> array){
122:        Collections.sort(array,new Comparator<Double []>() {
123:            public int compare(Double[] strings, Double[]
                otherStrings) {
124:                return strings[0].compareTo(otherStrings[0]);
125:            }
126:        });
127:    }
128: }

```

Fonte: Elaborada pelo autor.

Código-fonte da função principal do extrator de tamanho médio de segmentos:

Código-fonte 7 – Código-fonte do extrator de tamanho médio de segmentos

```

1: package extractor.avgseg.structure;
2:
3: import java.io.FileNotFoundException;
4: import java.util.ArrayList;
5: import java.util.Arrays;
6: import java.util.Collections;
7: import java.util.Comparator;
8:
9: /**
10:  * This class the number of segments of the image is measured
        . The segment extractor uses the bifurcation and
        termination points as references.
11:  * Thus, a segment of the image is classified as a
        bifurcation point to a terminal point or a point of
        bifurcation to another point of bifurcation.
12:  *
13:  * @author Misael Jr
14:  *
15:  */
16:
17: public class Count_Segmentos_AVG {
18:

```

```

19:  ArrayList<Double[]> bifurc_points = new ArrayList<Double
    []>();
20:  ArrayList<Double[]> end_points = new ArrayList<Double[]>();
21:  ArrayList<Double[]> A = new ArrayList<Double[]>();
22:  double nrSeg=0;
23:  double size_seg = 0;
24:
25:  /**
26:   * Function that measures the amount of segments in an
    image according to the points of bifurcation and
    identified finals.
27:   *
28:   * @param data -- Matrix with image points.
29:   * @param bifurc_points
30:   * @param end_points
31:   * @param nx -- Numbers of points on the x-axis
32:   * @param ny -- Numbers of points on the y-axis
33:   * @param nz -- Numbers of points on the z-axis
34:   * @param dim -- Image size
35:   * @return Average segment size, normalized.
36:   * @throws FileNotFoundException
37:   */
38:
39:  public double extractorAVGseg(double[][][] data,
    ArrayList<Double[]> bifurc_points, ArrayList<Double[]>
    end_points,
40:    int nx, int ny, int nz, int dim) throws
    FileNotFoundException{
41:
42:    this.bifurc_points = bifurc_points;
43:    this.end_points = end_points;
44:    double avg = 0;
45:    // Add bifurcation points and endpoints to an array and
    then sort them.
46:    A.addAll(end_points);
47:    A.addAll(bifurc_points);
48:    sortArray(A);
49:
50:    while(A.size() > 1){
51:      Double[] point1 = A.get(0);
52:      ArrayList<Double[]> ways;
53:      ArrayList<Double[]> pointTemp = null;

```

```

54:     ways = new ArrayList<Double[]>();
55:     int x = point1[0].intValue();
56:     int y = point1[1].intValue();
57:     int z = point1[2].intValue();
58:     A.remove(0); //Extract points from the first coordinate
        and delete from the array
59:
60:     // traverse all points between the first and second
        points of A, classifying as segment
61:     for (int d = 0; d < dim; d++){
62:         for (int i = x; i < nx; i++) {
63:             for (int j = y; j < ny; j++){
64:                 for (int k = z; k < nz; k++){
65:                     if (data[i][j][k][d] != 0){
66:                         pointTemp = new ArrayList<Double[]>();
67:                         Double[] pontos = {(double) i, (double) j, (
double) k};
68:                         pointTemp.add(pontos);
69:                         if(check(pointTemp, A)){ //check if the point
                            is present in A. That is, if it is a terminal or
                            bifurcation.
70:                             ways.add(pontos);
71:                         }else{
72:                             continue;
73:                         }
74:                     }else
75:                         continue;
76:                 }
77:             }
78:         }
79:     }
80:     nrSeg++;
81:     size_seg += ways.size();
82: }
83: //Average Segment Size
84: avg = (size_seg/nrSeg);
85: double avg_normalized = getNormalized(avg);
86: return avg_normalized;
87: }
88:
89: /**
90:  * set the normalization according to the base of max and

```

```
        min - linear scale
91:    * @param avg value
92:    * @param min
93:    * @param max
94:    * @return Normalized value
95:    */
96:
97:    public double getNormalized(double avg, min, max){
98:        double value_normal = 0;
99:        value_normal = (avg - min)/(max - min);
100:        return value_normal;
101:    }
102:
103:    /**
104:     * Function that checks whether a point is already present
105:     * in another array
106:     * @param array1 -- Array 1
107:     * @param array2 -- Array 2
108:     * @return If the point is present, returns false.
109:     * Otherwise, it returns true.
110:     */
111:    public boolean check(ArrayList<Double[]> array1, ArrayList<
112:        Double[]> array2){
113:        for(int i = 0; i < array1.size();i++){
114:            Double[] i1 = array1.get(i);
115:            for(int j = 0; j < array2.size();j++){
116:                Double[] i2 = array2.get(j);
117:                if(Arrays.equals(i1, i2)){
118:                    return false;
119:                }
120:            }
121:        }
122:        return true;
123:    }
124:
125:    /**
126:     * Function that sorts the points of an array of type
127:     * ArrayList <Double []>
128:     * @param array -- Array to be sorted
129:     */
```



```
128:
129:  public void sortArray(ArrayList<Double[]> array){
130:      Collections.sort(array,new Comparator<Double[]>() {
131:          public int compare(Double[] strings, Double[]
              otherStrings) {
132:              return strings[0].compareTo(otherStrings[0]);
133:          }
134:      });
135:  }
136: }
```

Fonte: Elaborada pelo autor.

MATRIZ DE CORRELAÇÃO ENTRE TODOS OS PARES DE CARACTERÍSTICAS

A seguir é apresentada uma matriz de correlação, com base nos coeficientes de Kendall's, entre todos os pares de características utilizados no presente trabalho de mestrado.