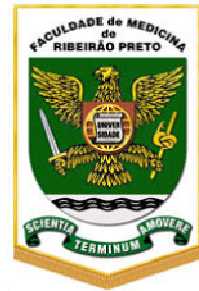




Universidade de São Paulo

**Faculdade de Filosofia, Ciências e Letras
de Ribeirão Preto
Faculdade de Medicina de Ribeirão Preto**



Informática Biomédica

**DESENVOLVIMENTO DE UM MÉTODO COMPUTACIONAL PARA
AVALIAÇÃO DE ESTENOSES EM VASOS CARDÍACOS ATRAVÉS
DE IMAGENS DE ANGIOGRAFIA DE CONTRASTE RADIOLÓGICO**

Carolina Paccagnella

Ribeirão Preto – SP

2008



Universidade de São Paulo

**Faculdade de Filosofia, Ciências e Letras
de Ribeirão Preto
Faculdade de Medicina de Ribeirão Preto**



Informática Biomédica

**DESENVOLVIMENTO DE UM MÉTODO COMPUTACIONAL PARA
AVALIAÇÃO DE ESTENOSES EM VASOS CARDÍACOS ATRAVÉS
DE IMAGENS DE ANGIOGRAFIA DE CONTRASTE RADIOLÓGICO**

Carolina Paccagnella

Orientador: Prof. Luiz Otavio Murta Jr.
Co-Orientador: Prof. José Antonio Marin Neto.

Monografia apresentada à Faculdade de Filosofia Ciências e Letras e à Faculdade de Medicina, ambas pertencentes à Universidade de São Paulo, como parte dos requisitos para obtenção do título de Bacharel em Informática Biomédica.

Ribeirão Preto – SP

2008

RESUMO

Doenças das artérias coronárias são muito prevalentes e associadas à alta taxa de morbidade e mortalidade. Tais desfechos clínicos podem ser parcialmente previstos por técnicas de angiografia coronária mediante cateterismo cardíaco, detectando-se precocemente estenoses significativas nesses vasos.

Neste trabalho foi desenvolvido um *software* para quantificação de estenose em artérias coronárias a partir de imagens originadas de angiografia cardíaca mediante material radiologicamente contrastante por cateterismo intravascular.

Utilizando métodos de processamento de imagens como contorno ativo ou *Snakes*, o objetivo do *software* desenvolvido é encontrar automaticamente as bordas de um vaso cardíaco determinado pelo usuário, e através de medidas de distância entre as bordas, estimar o diâmetro na região da estenose, comparando-o com um diâmetro de uma parte adjacente do vaso definido como sendo de calibre normal.

O método que determina a intensidade de obstrução de um vaso cardíaco é conhecido como Angiografia Coronária Quantitativa. O intuito é encontrar a porcentagem de redução do diâmetro luminal, além da extensão dessa obstrução, para que a melhor decisão de tratamento seja tomada. Nesse sentido, quando a melhor conduta envolve uma angioplastia coronária realizada por via transluminal percutânea, tais mensurações permitem melhor estimar o diâmetro e extensão dos balões e stents a serem empregados.

O software foi desenvolvido em linguagem Java, fazendo uso da biblioteca de classes para processamento de imagens presentes no ImageJ.

SUMÁRIO

RESUMO	1
SUMÁRIO	1
ÍNDICE DE FIGURAS.....	2
1. INTRODUÇÃO.....	1
1.1. CONCEITUAÇÃO	1
1.2. JUSTIFICATIVA E OBJETIVO	12
1.3. ORGANIZAÇÃO DO DOCUMENTO.....	16
2. DESENVOLVIMENTO.....	17
2.1. IMAGENS.....	17
2.2. IMPLEMENTAÇÃO	18
2.3. PROCEDIMENTO	21
2.3.1. PRIMEIRA SOLUÇÃO PROPOSTA – SEGMENTAÇÃO POR WATERSHED.....	21
2.3.2. SEGUNDA SOLUÇÃO PROPOSTA – SEGMENTAÇÃO POR <i>SNAKES</i>	22
3. RESULTADOS.....	26
4. INSTRUÇÕES DE OPERAÇÃO DO SOFTWARE.....	38
4.1. ABRINDO E ESCOLHENDO UMA IMAGEM.....	38
4.2. INICIANDO O PRÉ-PROCESSAMENTO	40
4.3. INICIANDO O PROCESSAMENTO	41
4.4. INTERPRETANDO O GRÁFICO.....	42
5. CONCLUSÕES	44
6. REFERÊNCIAS	45

ÍNDICE DE FIGURAS

Figura 1: Imagem original	3
Figura 2: Imagem com filtro gaussiano	3
Figura 3: Imagem com filtro Anisotrópico.....	3
Figura 4: Determinação das bacias de retenção e das linhas watershed.....	5
Figura 5: Esqueleto pela noção de disco	12
Figura 6: Esqueletonização: imagem original	12
Figura 7: Esqueletonização: imagem esqueletizada	12
Figura 8: Imagem de angiografia coronária	13
Figura 9: Stent.	15
Figura 10: Processo de cálculo do raio da artéria.....	25
Figura 11: Gradiente da imagem	26
Figura 12: Abertura do gradiente.....	26
Figura 13: Fechamento do gradiente	26
Figura 14: Watershed com sobre-segmentação	26
Figura 15: Visão do software em execução.....	27
Figura 16: Janela de seleção do melhor parâmetro q	28
Figura 17: Demonstração da rotulação dos objetos.....	28
Figura 18: Roi resultante da limiarização.....	28
Figura 19: Aproximação da Roi - vértices móveis	29
Figura 20: Interface do software exibindo o gráfico resultante do cálculo da estenose .	30
Figura 21: Imagem analisada no segundo caso	30
Figura 22: Aproximação do vaso.....	31
Figura 23: Gráfico do segundo caso	32
Figura 24: Roi reajustável	33
Figura 25: Vértices móveis.....	33
Figura 26: Esqueleto do vaso em preto e outline (paredes) em cinza	34
Figura 27: Terceiro caso analisado.....	35
Figura 28: Gráfico gerado com obstrução de 45%	36
Figura 29: Esqueleto do vaso em preto e paredes em cinza	37
Figura 30: Interface de abertura de uma imagem	39
Figura 31: Interface contando barra de navegação das imagens	40
Figura 32: Barra de navegação das imagens	40
Figura 33: Escolha do melhor parâmetro q para limiarização.....	41
Figura 34: Botões	42
Figura 35: Janela com a seleção do contorno.	42
Figura 36: A linha vermelha demonstra a curva de variação da área e a azul, a área esperada.	43

1. INTRODUÇÃO

1.1. CONCEITUAÇÃO

Durante esse capítulo serão abordados conceitos teóricos utilizados para o desenvolvimento do projeto.

Redução de ruídos: Filtro Anisotrópico

O filtro anisotrópico consiste em um filtro de difusão, que conserva as bordas da imagem. Normalmente é utilizado o filtro Gaussiano, onde a imagem é convoluída com uma função Gaussiana, porém é difícil obter precisão em áreas que contêm informações significativas. A razão para a distorção causada pelo filtro gaussiano é que ele não respeita as bordas naturais dos objetos (PERONA e MALIK 1990). O efeito visual do filtro anisotrópico é a suavização da imagem e a redução da variabilidade dos níveis de cinza, preservando as bordas da imagem (GONZALEZ e WINTZ 1992). Processos de difusão anisotrópica, ou não linear, têm sido usados recentemente para melhorar a tarefa de detecção de bordas e eliminar ruídos de imagens em várias áreas.

Witkin (WITKIN 1983), seguido por Koenderink (KOENDERINK 1984) e Hummel (HUMMEL 1986), propuseram o espaço de escala linear, que consiste em trabalhar com uma imagem em vários níveis de simplificação, lidando com o problema de multi-resolução. A teoria espaço-escala é uma base teórica para a representação de imagens em múltiplas escalas, desenvolvida pelas comunidades de processamento de imagens, permitindo manipular imagens em diferentes escalas, de forma que atributos

em escalas maiores podem ser sucessivamente suprimidos e um parâmetro de escala t pode ser associado a cada nível da representação espaço-escala (MIRANDA e CAMARGO NETO 2007). Na proposta do espaço de escala linear de Witkin, Koenderink e Hummel, foi utilizada a convolução com núcleo Gaussiano para simplificar a imagem. A convolução com a máscara Gaussiana de desvio-padrão σ pode ser vista como a solução da equação de condução de calor, onde o valor da imagem original Q num ponto (x, y) é a temperatura inicial nesse ponto, o tempo decorrido é $t = \sigma^2 / 2$, e a imagem Q na escala σ representa as temperaturas no instante t . Assim, a convolução Gaussiana é um processo de difusão isotrópica. Isotrópico significa que apresenta as mesmas propriedades físicas em todas as direções.

Perona e Malik (PERONA e MALIK 1990) formularam importantes modificações no espaço de escala resultando em difusão anisotrópica para substituir a isotrópica, com a propriedade de simplificar a imagem mantendo as bordas nítidas. Na proposta deles, existe uma função chamada parada-na-aresta (*edge stopping function*) g , que controla a intensidade da difusão de acordo com o gradiente do ponto que deve sofrer difusão. Essa função possui um parâmetro de escala que, em conjunto com o gradiente, indica se a difusão deve ser forte ou fraca. A solução apresentada pelos autores para a modificação do paradigma espaço-escala linear ocorreu na equação de difusão, onde o coeficiente de difusão c é assumido ser uma constante independente da localização no espaço. A idéia essencial é envolver a imagem original, I_0 , numa família de imagens derivadas, I_t , obtidas pela convolução da imagem original com um filtro Gaussiano com variância t . O “tempo” t é um parâmetro de escala: incrementos em t geram representações mais simples da imagem, ou com resoluções menores. Eles propuseram a seguinte equação:

$$I_t = \text{div}(c(x, y; t)\nabla I) = c(x, y; t)\Delta I + \nabla c \cdot \nabla I$$

Onde div é o operador divergente, ∇ e Δ representam os operadores gradiente e Laplaciano, respectivamente, com respeito à variável espacial e I é a imagem. A equação se reduz à equação isotrópica de difusão do calor $I_t = c \times \Delta I$ se $c(x, y; t)$ for uma constante. Modificando o coeficiente para zero nas bordas e um na região interna, é possível suavizar dentro das regiões, ao invés de suavizar percorrendo as bordas. Esse efeito pode ser visto nas figuras 1, 2 e 3. Na figura 1 é exibida a imagem original. Na figura 2 e 3 são exibidas a imagem com filtro gaussiano e com filtro anisotrópico respectivamente.

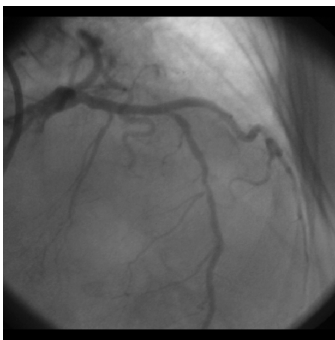


Figura 1: Imagem original

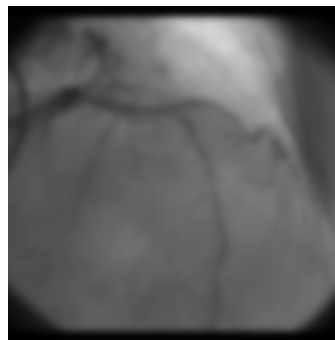


Figura 2: Imagem com filtro gaussiano

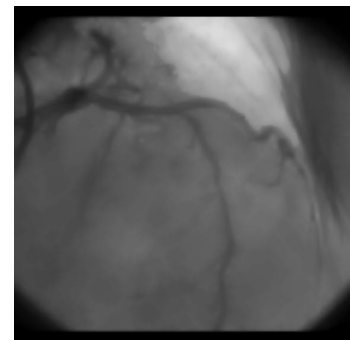


Figura 3: Imagem com filtro Anisotrópico

Gradiente Morfológico

O gradiente morfológico é a informação diferencial dos dados, e pode ser calculado a partir dos operadores lineares de convolução, tais como operadores de Sobel, Prewitt ou Kirsch, entre outros. Outra abordagem do gradiente é determinada com a utilização da Morfologia Matemática. Há de maneira geral, dois tipos de

abordagens em imagens realizadas por Morfologia Matemática: a morfologia binária e a morfologia em tons de cinza. A morfologia binária aplica-se em imagens binárias (aquelas que apresentam apenas pixels na cor preta ou branca). Em morfologia em tons de cinza é preciso conhecer o valor do pixel mais escuro e o mais claro em uma vizinhança e, com base nestes valores, encontra-se o valor do pixel em análise. As imagens em nível de cinza são consideradas como conjuntos tridimensionais nos quais os operadores morfológicos podem ser aplicados. Do ponto de vista topográfico, as regiões escuras são vistas como vales e as claras como picos.

Os operadores básicos da morfologia são a erosão e dilatação, e quando aplicados isoladamente, são transformações que nem sempre evidenciam características das imagens. De qualquer forma, permitem construir algumas funções muito interessantes, como por exemplo, o gradiente morfológico. A partir dos operadores básicos, Soille (Soille 1999), define o gradiente morfológico como a diferença aritmética entre a dilatação e erosão de uma imagem por um elemento estruturante. O operador gradiente é comumente utilizado para detecção de contornos, e pode contribuir na etapa da segmentação de imagens (GONZALEZ e WINTZ 1992).

Também foram utilizadas as operações de abertura e fechamento que consistem em combinações de erosão e dilatação. A abertura é dada pela erosão seguida de dilatação e o fechamento por dilatação seguida de erosão.

Segmentação por Watershed

A segmentação por watershed consiste em definir linhas de partição de uma imagem a partir do princípio de inundação de relevos topográficos (PECCINI 2004). Essa abordagem vem do princípio de que uma imagem em níveis de cinza pode ser vista

como um relevo topográfico, formado por vales, que representam as regiões mais escuras da imagem, e por montanhas, que representam as porções mais claras, como já foi citado. Conforme os vales são inundados, surgem linhas divisoras, as chamadas watersheds, como visto na figura 4.

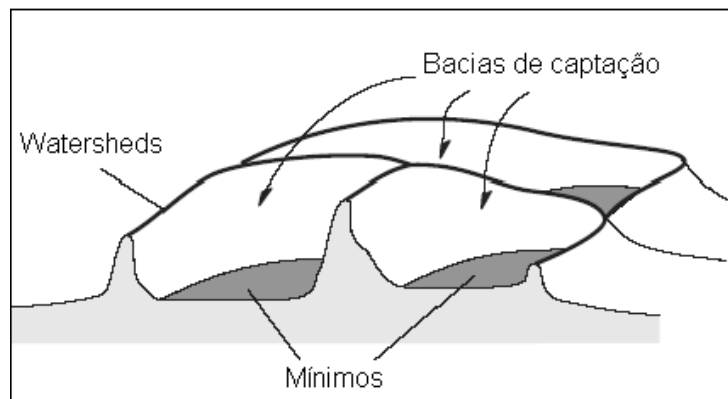


Figura 4: Determinação das bacias de retenção e das linhas watershed
(Fonte: Adaptado de Soille, 1999)

Essa técnica normalmente é aplicada no gradiente morfológico de uma imagem, e não na imagem original, já que ele realça as discontinuidades de uma imagem. Contudo, o gradiente morfológico pode ser muito sensível a ruídos, podendo formar uma grande quantidade de picos e vales. Isso resultar algumas vezes em uma super-segmentação. O watershed sempre gera contornos fechados e constitui uma abordagem geral de detecção de contornos.

Normalização

A normalização de histograma ou normalização da imagem consiste em distribuir estaticamente os tons de cinza da imagem digital de maneira que o contraste da imagem gerada permita melhor compreensão da mesma. Para isso, utiliza os valores mínimos e máximos da imagem como os valores limites para a normalização.

Limiarização

Esse passo consiste em gerar uma máscara binária, a partir do cálculo de entropia de Tsallis, para determinar um limiar que represente, de forma próxima, o vaso a ser segmentado. Testes foram feitos anteriormente para determinar um parâmetro q , parâmetro de não extensividade (que se relaciona a conseqüências de correlações estatísticas entre os pixels), de forma a obter uma segmentação satisfatória. Essa segmentação serve de base para a segmentação seguinte, já que é pré-requisito para a aplicação do método de contorno ativo que seja feita uma seleção da área de interesse.

A limiarização de imagens é uma ferramenta de segmentação muito simples e é indicada para aplicações que exijam a identificação e extração do objeto da imagem. A limiarização trata-se de encontrar em um conjunto de níveis de cinza, por exemplo de 0 a 255, um valor que separe os pixels em duas classes: objeto, com valor 0 e fundo, com valor 255.

A entropia é freqüentemente utilizada para indicar a quantidade de informação contida em uma determinada fonte, mas também é utilizada para graduar a desordem de um conjunto de dados. Um evento com probabilidade baixa, ou seja, um evento incomum terá maior contribuição no aumento da entropia (LOPES 2003). No caso de imagens digitais, o evento corresponde ao pixel, e a probabilidade corresponde à probabilidade p_i desse pixel assumir um valor de intensidade i . Essas probabilidades podem ser alcançadas com a transformação do histograma dos níveis de intensidade da imagem em uma função de densidade de probabilidade. A máxima entropia ocorre quando uma imagem contém a mesma quantidade de pixels para todas as intensidades,

isto é, todas as intensidades têm a mesma probabilidade de ocorrerem. A formulação clássica da teoria da informação propõe a chamada entropia de Shannon, descrita pela seguinte expressão:

$$S = - \sum_{i=1}^k p_i \ln p_i .$$

Este formalismo é o mais apropriado dentro do domínio de validade da estatística de Boltzmann-Gibbs-Shannon (BGS). Esta estatística aparentemente descreve melhor os fenômenos com interações de curto alcance, chamados geralmente de sistemas extensivos. Entretanto para algumas classes de sistemas físicos que apresentam comportamento fractal e memórias de longo alcance é necessária uma extensão deste formalismo. Um fractal é um padrão que se repete nas diversas escalas de um sistema complexo. Esta foi a proposta de Tsallis para generalização da estatística BGS. A estatística proposta por Tsallis é hoje útil para descrever propriedades de sistemas não extensivos e está baseada na seguinte expressão matemática para a Entropia:

$$S_q = \frac{1}{q-1} \left(1 - \sum_{i=1}^k p_i^q \right),$$

Equação 1: q-Entropia de Tsallis

onde k se refere ao número total de possibilidades do Sistema, o número real q é o índice entrópico que define o grau de não extensividade do sistema e p é a probabilidade.

Muitas pesquisas têm sido desenvolvidas baseadas nas características dos fractais, por exemplo, se órgãos como os pulmões, rins e veias são estruturas fractais. Na hipótese de que imagens complexas, como as imagens médicas, possam apresentar características não extensivas, a entropia utilizada para determinar o limiar foi a entropia de Tsallis (ALBUQUERQUE, et al. 2006) (SÁ e LUZ 2003).

O algoritmo utilizado tem como requisito a passagem de um parâmetro real q , que será utilizado no cálculo da q -Entropia. A partir dessa entropia encontrada, é realizada a limiarização.

Segmentação por *Snake*

Esse método consiste na deformação de contornos pré-definidos pela técnica anterior de entropia.

O modelo *Snake* surgiu da necessidade de detectar contornos em imagens com ruídos. A segmentação de contornos baseada em modelos deformáveis tem sido amplamente utilizada nas últimas décadas. O campo da imagem médica tem sido um dos campos em que a sua aplicação se revelou mais fértil. As *snakes* podem ser imaginadas como curvas definidas no domínio da imagem, que se podem deslocar sob a influência de forças internas (definidas na própria curva) e forças externas calculadas a partir da imagem ou de processos de alto nível. Os fundamentos matemáticos das *snakes* resultam da confluência da geometria, física e teoria da aproximação. A geometria permite a representação da forma do objeto, a física impõe restrições à variação da forma ao longo do espaço e do tempo e a teoria da aproximação permite formalizar os mecanismos que possibilitam o ajuste do modelo aos dados (SILVA, et al. 2004). A

energia que será minimizada é uma combinação das forças internas e externas, sendo uma *snake* definida como $v(s)$ e pode ser escrita como:

$$E_{ex} = - \int |\nabla I(v(s))|^2 ds$$

Equação 2: Energia Externa

$$E_{int} = \int (|v'(s)|^2 + |v''(s)|^2) ds$$

Equação 3: Energia interna

Nesse caso, a energia externa expressa a maximização do gradiente ao longo da curva e a energia interna expressa o grau de regularidade dessa curva, por minimização da primeira e segunda derivada da mesma. A minimização da energia é feita iterativamente, fazendo com que o contorno se ajuste a cada iteração, produzindo um movimento semelhante ao de uma serpente. A deformação de uma curva $C(s, t)$, onde s é o tamanho do arco que será deformado e t a variação temporal, é descrita por :

$$\frac{\partial C(s, t)}{\partial t} = X(s, t)\vec{T}(s, t) + Y(s, t)\vec{N}(s, t) = \vec{F}$$

F é o vetor de deformação, T é o vetor tangente e N o vetor normal. A componente YN modifica a forma da curva. Para evitar que a curva deformada se aproxime apenas pelo contorno mais próximo, não encontrando estruturas como cavidades, também entra no processo de busca do contorno o direcionamento da curva deformável através do vetor normal, fazendo com que a curva detecte as cavidades. O modelo utilizado nesse projeto envolve dois aspectos: a regularização das *snakes* e a deformação do contorno. Quando detecta um contorno, faz a regularização desse com a minimização da energia. Antes de deslocar o contorno, o método é re-parametrizado. Esse processo de detectar, deformar, regularizar e re-parametrizar é repetido até que a deformação seja desprezível.

Inicialmente, os pontos da curva inicial são colocados como um polígono, a tangente de cada ponto M_i é então aproximada pelo vetor unitário, o vetor normal é calculado pela perpendicular do vetor unitário e a curvatura é calculada pela inversa do raio da esfera que passa pelos 3 pontos M_{i-1} , M_i e M_{i+1} . Os pontos se movem em direção ao contorno ao longo da normal. O contorno normalmente é reconhecido pela mudança abrupta de tons de cinza. No algoritmo utilizado, são fixados dois limiares de tom de cinza: um alto e um baixo. O contorno então é detectado quando, no vetor normal, o nível médio de cinza dos pixels da parte superior for maior que o limiar alto e o nível médio de cinza dos pixels da parte inferior for menor que o limiar baixo. Para cada ponto, é detectado o contorno mais próximo ao longo do vetor normal em ambas as direções, dentro de uma distância fixa. Após a deformação, é realizada a regularização, o que permite impor certas restrições ao modelo. Seja então a energia interna $E_{int} = \int (\alpha(s)|v'(s)|^2 + \beta(s)|v''(s)|^2) ds$ onde v é a curva, α e β são parâmetros que controlam a tensão e a rigidez da curva, a minimização dessa energia permite obter uma curva uniforme. O parâmetro α controla o grau de alisamento do modelo. A cada iteração, caso os pontos se distanciem muito, são adicionados novos pontos, e caso aproximem-se muito, um ponto é eliminado. Esse espaço ideal deve ser calculado e irá variar ao longo da *snake*. A *snake* então é re-parametrizada e a distribuição dos pontos é otimizada. Nas *snakes* clássicas, o ponto de parada é quando a energia mínima é alcançada. Neste modelo, a *snake* pára quando o movimento realizado não foi significativo em relação ao anterior (BOUDIER 1997). Este modelo de *snake* foi desenvolvido por Thomas Boudier (ANDREY e BOUDIER 2006), como forma de um *plugin* do ImageJ e modificado para esse projeto para ser executado em um software *Stand Alone*, com o nome de *ABSsnake*. A utilização do método de segmentação por contornos ativos foi possível após testes e estudos sobre os melhores parâmetros para

regularização da *snake*. Inicialmente foram realizados testes com a imagem inteira e com uma pré-segmentação requerida pelo sistema com forma elíptica no interior de um vaso de interesse. Os valores para o limiar e número de interações modificaram o contorno final conforme foram alterados. Dos parâmetros avançados, o que pronunciou maior modificação foi o deslocamento máximo. O deslocamento mínimo interfere no tempo necessário para chegar a um contorno mais próximo do real. Algumas medidas foram realizadas durante o teste dos parâmetros. Nas imagens utilizadas, foi feita uma seleção de uma região de interesse, e esses parâmetros foram modificados e testados durante a execução do programa, e a conclusão foi de que o melhor limiar foi 5, o número de iterações necessárias foi 20 e os outros parâmetros continuaram sendo os definidos como padrão pelo *plugin* modificado.

Esqueletonização

O método de esqueletonização consiste em armazenar a informação mínima de uma imagem binária, comprimindo a imagem em um conjunto de pixels no seu interior, permitindo análises futuras de forma mais rápida. Esse conjunto de pontos é chamado de esqueleto. O esqueleto de uma imagem pode ser definido pela união dos pontos que definem o centro de círculos máximos inscritos na mesma, tocando a borda em pelo menos dois pontos distintos. A figura 5 ilustra a idéia de esqueleto pela união dos centros dos discos.

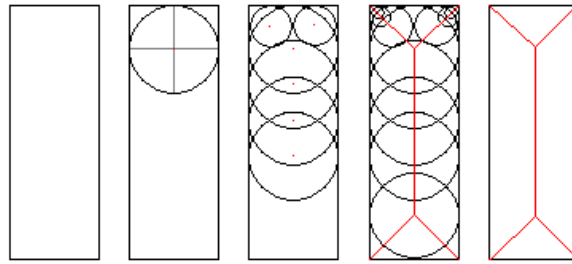


Figura 5: Esqueleto pela noção de disco

A intenção da eskeletonização nesse processo é de encontrar a luz do vaso sanguíneo a partir da distancia calculada de um ponto do esqueleto ate um ponto que esteja situado de forma perpendicular na parede do vaso. Essa distancia equivale ao raio da luz do vaso, e a área é facilmente calculada a partir dele.



Figura 6: Eskeletonização: imagem original

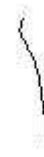


Figura 7: Eskeletonização: imagem esqueletizada

1.2.JUSTIFICATIVA E OBJETIVO

Os vasos sanguíneos estão sujeitos a anomalias, dentre elas doenças oclusivas e estenoses, impedindo que a circulação normal ocorra, ocasionando assim uma diminuição da perfusão do tecido, ou mesmo a morte do mesmo. O infarto agudo do miocárdio é um processo que pode levar à morte de parte do músculo cardíaco por falta de aporte adequado de nutrientes e oxigênio, causada por redução do fluxo sanguíneo coronariano de duração suficiente para que as reservas naturais não sejam capazes de

sustentar o tecido. É a principal causa de morte nos países ocidentais. Das mortes conseqüentes a um infarto, a maior parte é rápida, na primeira hora.

Um método muito utilizado ultimamente para detectar essas anomalias nos vasos sanguíneos é a angiografia digital com sondagem por cateterismo, que consiste em imagens de raios-X utilizando um contraste à base de iodo, injetado via cateter para tornar possível a visualização dos vasos. O iodo atenua a passagem dos raios-X, portanto, permitindo visualizar e armazenar a imagem desses vasos. A introdução do cateter pode ser dada via femoral ou braquial. Este cateter é posicionado nas artérias coronárias e no ventrículo esquerdo, para a realização de injeções do contraste, o que permitirá observar a presença de placas de ateroma nas artérias ou outras anormalidades que estas possam apresentar. Após a injeção de contraste nestas artérias, são obtidas imagens de raios-X em diversas posições.

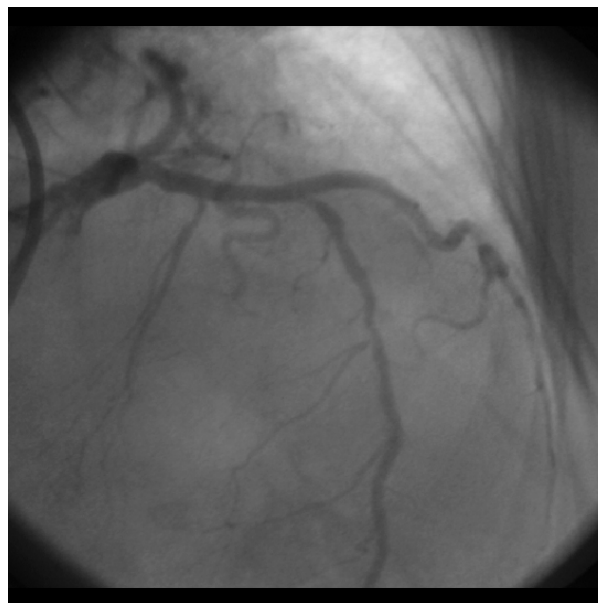


Figura 8: Imagem de angiografia coronária

Na figura 8, as áreas mais escuras representam o contraste introduzido através do cateter na coronária impedindo a passagem dos raios-x, conseqüentemente possibilitando a sua visualização.

Com base nas imagens obtidas através desse método e utilizando métodos modernos de processamento de imagens, é possível calcular o índice de estenose nas artérias, que se dá pela relação entre a obstrução do vaso e a área total do vaso. Através desses índices, é possível decidir que tipo de tratamento ou intervenção será adotado. As obstruções superiores a 50% da luz do vaso geralmente são consideradas significativas, embora as obstruções maiores de 70% sejam as que provocam as manifestações isquêmicas. Essas informações são de valor prognóstico.

O estudo realizado para avaliar o fluxo de sangue nas artérias coronárias é conhecido como Arteriografia Coronária Quantitativa (QCA), iniciais em inglês de *Quantitative Coronary Arteriography*. Como intervenção, existe a técnica de colocação de um *stent*, uma endoprótese expansível que permite ampliar o fluxo em um vaso. O *stent* é mais utilizado como tratamento em artérias coronárias, mas também é amplamente utilizado em outras estruturas do organismo com a mesma finalidade de sustentar a ampliação do vaso, impedindo um colapso por retração elástica do mesmo.

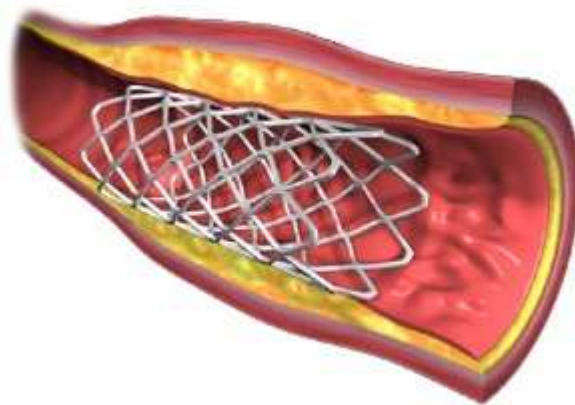


Figura 9: Stent. Disponível em www.endovasc.com/images/graphics/stent.jpg

A figura 9 ilustra como a endoprótese se expande dentro do vaso pela insuflação de um balão ao qual a endoprótese está inicialmente acoplada, fazendo com que a luz do vaso aumente, e permitindo fluxo normal de sangue pelo local.

Apenas a interpretação visual dos exames pode não ser precisa o suficiente, a ponto de causar decisões errôneas a respeito do tratamento ideal para determinado paciente. Com o intuito de ajudar na decisão de qual o tratamento melhor aplicado, além de documentar o diagnóstico dado pelos exames de angiografia, o proposto por este trabalho é um software desenvolvido em linguagem Java que, através de imagens geradas por angiografia, calcule computacionalmente o índice de estenose da artéria e gere gráficos de comprimento da artéria por graus de estenose, tornando possível a avaliação e obtenção de novas variáveis para análise dos exames dos pacientes em situações de anomalias mais sérias do fluxo cardíaco.

O software deve ser capaz de ler imagens armazenadas no formato DICOM e calcular o mais intenso índice de estenose em um determinado intervalo de tempo determinado pelo usuário.

1.3. ORGANIZAÇÃO DO DOCUMENTO

Este documento está dividido da seguinte forma: na secção 2 estão informações sobre o desenvolvimento do projeto, subdivididas entre informações sobre as imagens, sobre o código a linguagem utilizada na implementação e sobre o procedimento de desenvolvimento; na secção 3 estão alguns resultados do projeto; na secção 4 estão as instruções de como utilizar o software; na secção 5 estão as conclusões obtidas e por último na secção 6 estão as referências utilizadas.

2. DESENVOLVIMENTO

2.1. IMAGENS

Foram utilizadas imagens obtidas por angiografia cardíaca, geradas na sala de cateterismo na Divisão de Cardiologia do Hospital das Clínicas de Ribeirão Preto. As imagens utilizadas estão codificadas em 256 tons de cinza e estão em arquivos formatados em padrão DICOM. O objetivo do padrão DICOM (abreviação de *Digital Imaging Communications in Medicine*) é prover compatibilidade e aumentar a eficiência da transmissão de informações entre diversos sistemas de imagens e de informação no meio médico permitindo trocas e visualização de imagens e demais informações de exames em qualquer centro de imagens que adote este protocolo. Dados como nome do paciente, tipo de exame, número de imagens, data do procedimento, formato das imagens gravadas, tipo de compressão utilizada e características que garantem a segurança da imagem gravada são algumas das inúmeras informações do padrão DICOM. O conjunto de documentos de especificação do protocolo DICOM é constituído por 18 partes, cada uma concentrada em um ponto de vista do padrão. Como exemplo, na parte 5 estão informações sobre estrutura dos dados e semântica, especificando suporte a técnicas de compressão de imagem. Compressões são efetuadas para reduzir o tamanho dos arquivos para facilitar transporte e armazenamento. Algoritmos como os de JPEG, permitidos pelo DICOM, podem ser de compressão com perda de informação (*lossy*) ou sem perda de informação (*lossless*).

2.2.IMPLEMENTAÇÃO

Para a implementação da solução proposta nesse projeto, foi utilizada a linguagem Java. O software ImageJ serviu como apoio para testes sobre as imagens antes que fosse implementada a aplicação *Stand Alone* proposta no projeto. O ImageJ é um software livre que permite a visualização de imagens em diferentes formatos, inclusive no padrão de imagens médicas, o DICOM. Permite também o processamento dessas imagens, através de *plugins*, uma forma de extensão do código do ImageJ. Uma pesquisa aprofundada sobre as classes da API ImageJ foi realizada com a finalidade de utilizar as formas de representação de imagem e alguns métodos do software ImageJ. Uma API, de *Application Programming Interface* (ou Interface de Programação de Aplicativos) é um conjunto de rotinas e padrões estabelecidos por um software para a utilização das suas funcionalidades por programas aplicativos, ou seja, programas que não querem envolver-se em detalhes da implementação do software, mas apenas usar seus serviços.

A API ImageJ possui os seguintes pacotes: *IJ*, *ij.gui*, *ij.io*, *ij.macro*, *ij.measure*, *ij.plugin*, *ij.plugin.filter*, *ij.plugin.frame*, *ij.process*, *ij.text*, *ij.util*. Uma classe que não consta nessa API, mas que foi utilizada por possuir métodos que facilitam o processamento da imagem através de seus pixels é a classe *ImageAccess*. Serão descritos os pacotes mais utilizados durante o desenvolvimento do projeto.

O pacote *IJ* possui classes que participam diretamente na execução do ImageJ. Classes com a representação de uma imagem, com rotinas para exibir essas imagens, entre outras. Do pacote *ij.IJ*, foram utilizadas as classes *ImagePlus* e *ImageStack*. A representação de uma imagem pelo ImageJ é feita através de um objeto do tipo *ImagePlus*, baseado na classe *ImageProcessor*. Através dele é possível obter o conteúdo

de uma imagem e suas principais propriedades. Ela permite também a visualização da imagem em uma janela, dentre varias outras ações que se pode realizar com uma *ImagePlus*. Muitos processos já predefinidos no ImageJ utilizam, como parâmetro de entrada ou de saída, um objeto do tipo *ImagePlus*. A classe *ImagePlus* possui vários construtores, que recebem ou não argumentos, podendo ser uma String especificando o caminho da imagem, um objeto representando uma pilha de imagens, *ImageProcessor* ou um objeto do tipo *Image*.

Também foi utilizada a classe *ImageStack*, para representar imagens armazenadas de forma análoga a uma pilha. Uma *ImageStack* pode representar imagens sequenciais em uma escala de tempo ou mesmo imagens em 3D. Um objeto do tipo *ImageStack* permite realizar ações interessantes como acessar uma determinada imagem da pilha, dada a posição em que ela se encontra.

O pacote *ij.gui* possui classes que permitem utilizar unidades de interface gráfica do ImageJ, como uma seleção de regiões de interesse, uma ROI (Region of Interest). Essa é uma classe que representa uma região de interesse de uma imagem, permitindo processar apenas a região de interesse, e não toda a imagem, o que às vezes pode ser computacionalmente custoso e desnecessário.

Existem varias subclasses para tipos específicos de regiões de interesse. Elas podem ser retangulares, ovais, com formas livres, cada uma para um determinado interesse ou aplicação.

Outra classe desse pacote que foi utilizada foi a *Wand*, que reproduz uma ferramenta de seleção de região de interesse com a qual é possível selecionar pixels com intensidades delimitadas pelos parâmetros passados.

Uma terceira classe é a *ImageCanvas*, derivada da *java.awt.Canvas*. Cada imagem é exibida por um objeto do tipo *ImageWindow*, que deriva de uma

java.awt.Frame. Esse frame contém um *ImageCanvas*, no qual a imagem é desenhada. Um objeto *ImageCanvas* também possui métodos que permitem manipular a imagem.

No pacote *ij.io* estão contidas classes para ler e escrever arquivos que representam imagens. Varias classes desse pacote foram de grande utilidade durante o desenvolvimento, como a classe *Opener*, que permite abrir arquivos de vários formatos, como DICOM, e retorna um objeto de fácil manipulação, uma *ImagePlus*.

A classe *ImageProcessor* do pacote *ij.process* possui métodos para que eles sejam aplicados sobre a imagem que representa. É uma superclasse abstrata de processadores de imagem para certos tipos específicos de imagem. Uma classe que herda *ImageProcessor* e que também foi de grande utilidade é a *ByteProcessor* (Bailer 2006)

A classe Java *ImageAccess* foi criada para facilitar a manipulação dos pixels das imagens no ImageJ. Métodos da *ImageAccess* provém facilidade e robustez para o acesso aos pixels. As informações são armazenadas em matrizes do tipo *double* e essa classe já tem tratamento adequado para as fronteiras das imagens.

Para a análise final da estenose, foi utilizada a biblioteca *JFreeChart*, uma biblioteca que contém classes para a criação de gráficos de alta qualidade para aplicações Java. É uma API consistente e bem documentada que suporta uma grande quantidade de tipos de gráficos.

2.3.PROCEDIMENTO

O presente trabalho consta do desenvolvimento de um método, utilizando técnicas de processamento de imagens, como Morfologia Matemática, para calcular o índice de estenose em uma determinada região de uma imagem estática pertencente ao arquivo do filme de angiografia cardíaca utilizado como base para o projeto

Duas abordagens foram avaliadas durante o desenvolvimento: uma utilizando Watershed e outra utilizando *Snakes*. Inicialmente, a solução proposta para esse projeto era a utilização das técnicas de esqueletonização e segmentação por Watershed sobre o gradiente da imagem original, utilizando como filtro para amenizar os ruídos o filtro anisotrópico de difusão. A seguir será analisado cada passo seguido durante a abordagem inicial.

2.3.1. Primeira solução proposta – Segmentação por Watershed

A solução proposta inicialmente para segmentação utilizando o método de watershed segue um roteiro onde inicialmente é feita a redução dos ruídos da imagem utilizando o filtro anisotrópico de difusão, seguido pelo cálculo do gradiente da imagem, o qual passa por abertura e fechamento. Em seguida é feita a segmentação por Watershed e finalmente é realizada a esqueletonização da imagem,

Desenvolvimento da primeira proposta

Para desenvolver os métodos presentes na proposta inicial, foram utilizados operações morfológicas e outros métodos de processamento de imagens com a linguagem Java. Os testes foram realizados no software ImageJ, no qual esses métodos estão implementados sob a forma de *Plugins*. A imagem então foi submetida a um método computacional utilizando técnicas de Morfologia Matemática o qual filtra a imagem retirando ruídos e melhorando o aspecto visual, facilitando a atuação das funções de detecção dos objetos, foi aplicado o gradiente e a abertura seguida de fechamento desse gradiente. Em seguida foi utilizada a técnica de watershed para delimitar os objetos e então seria possível encontrar o esqueleto, e assim estimar um diâmetro normal para o vaso obstruído.

2.3.2. Segunda solução proposta – Segmentação por *Snakes*

A solução proposta nessa abordagem consiste na utilização do filtro anisotrópico descrito para diminuir os ruídos, seguido de um método de limiarização para determinar uma região de interesse contendo o vaso a ser encontrado, e por final passando pelo método de contorno ativo.

Desenvolvimento da segunda proposta

Inicialmente a imagem foi aberta a partir de um arquivo com a classe *Opener* do pacote *ij.io*. O método *openImage(java.lang.String path)* permite abrir imagens do tipo DICOM a partir do caminho do arquivo e retorna um objeto do tipo *ImagePlus*, também amplamente utilizado no desenvolvimento. A imagem então passa por um filtro anisotrópico de difusão, anteriormente implementado sob forma de um *plugin* de ImageJ, então adaptado para ser utilizado em uma aplicação *Stand Alone*. O filtro anisotrópico é uma classe, *AnisotropicDiffusion*, e tem como auxiliares outras duas classes, *AnisotropicDiffusionPixel* e *AnisotropicDiffusionStructure*, que auxiliam o armazenamento de informações dos pixels e estrutura de dados do contorno. O método *applyAnisotropicDiffusion* é o que invoca os construtores das outras classes e tem como parâmetro de entrada um objeto do tipo *ImageProcessor*, e após o processamento da imagem também retorna um objeto do tipo *ImageAccess*, apesar de que todas as modificações ficaram armazenadas no próprio parâmetro de entrada.

Após a aplicação do filtro anisotrópico a imagem é normalizada, utilizando um método implementado na classe Java *ImageAccess*, o *normalizeContrast*. Ele utiliza o valor máximo e o mínimo dos níveis de cinza da imagem para recalcular os valores limites. Esse processo faz com que o contraste seja modificado e com isso a observação dos objetos contidos na imagem se torna mais fácil.

Seguinte a normalização, a imagem passará então pelo processo de limiarização, realizada pela q-entropia, que está implementado em forma de uma classe chamada Q-Entropy. Essa classe inicialmente era estruturada para executar em forma de plugin do ImageJ. Foram feitas modificações para que ela pudesse ser aplicada durante a execução do software desenvolvido. A partir de um valor *double* q passado como parâmetro para

o objeto do tipo Q-Entropy, é calculado um valor inteiro que expresse a maximização da q-entropia do histograma pertencente a imagem. Esse valor é utilizado no método já implementado da classe *ImageProcessor*, *threshold(int t)*, que limiariza a imagem no nível de cinza t. Esse processo servira como base para encontrar uma segmentação inicial, que privilegiará a visualização dos vasos. Porém essa limiarização pode resultar em vários objetos desconexos, o que atrapalha a pré-segmentação. Foi então necessária a utilização de uma classe com métodos de morfologia matemática capazes de rotular esses objetos e determinar o maior objeto, *BinaryLabel8*, adaptada de um *plugin* para o ImageJ. A partir da imagem gerada pela limiarização, entra em ação uma ferramenta do ImageJ, *Wand*, já citada anteriormente, que seleciona os pixels que estão conectados e possuem o mesmo valor correspondente ao preto. Isso traz uma *ROI*, região de interesse ou seleção que é parâmetro de entrada para o processo de *Snakes*. A deformação desse contorno ocorrera devida a utilização do objeto do tipo *ABSnakeApplication* que coloca em prática o conceito de contornos ativos. Ao final dessa deformação, o contorno encontrado será o mais próximo do contorno do vaso sanguíneo em foco.

Para realizar o cálculo do índice de estenose, serão utilizados conceitos de esqueletonização e geometria. A distância entre as bordas desse vaso deve ser calculada por toda a extensão do mesmo na região de interesse, determinando o diâmetro observado.

Inicialmente faz-se necessário a determinação do esqueleto e das paredes do vaso. Isso foi alcançado a partir de métodos já existentes na classe *ByteProcessor* da API ImageJ. Um objeto desse tipo pode determinar o esqueleto e a *outline* de uma imagem. O esqueleto é percorrido e calcula-se para cada pixel o ponto da *outline* que corresponde à perpendicular desse pixel. O ponto perpendicular foi encontrado com a verificação do ângulo formado entre o segmento do ponto no esqueleto e três pontos de distancia com

um ponto do contorno. Caso esse ângulo formasse um ângulo reto, esse ponto encontrado seria armazenado. Como imagens possuem valores inteiros de coordenadas, foi necessário encontrar um desvio padrão do ângulo de 90 graus, que representa a ortogonalidade, para que cada ponto do esqueleto tivesse realmente um representante na parede do vaso. Com isso, foi encontrado mais de um ponto perpendicular para cada pixel do esqueleto. Os pontos que representavam menor distância foram escolhidos.

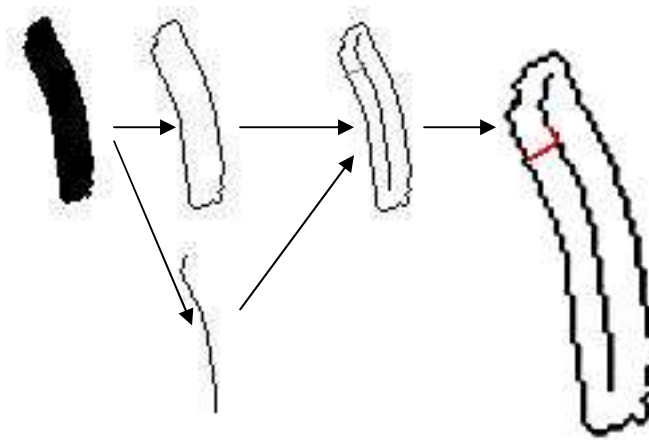


Figura 10: Processo de cálculo do raio da artéria

No cálculo do índice de estenose, é necessário ter um valor de área normal da luz do vaso. Esse valor será baseado em um diâmetro esperado, calculado como sendo a média entre duas outras medidas de diâmetros próximos à localização do menor diâmetro observado. Com isso é possível estimar o índice de estenose como sendo $\frac{A_{obs}}{A_{esp}}$, onde A_{obs} é a área calculada através do diâmetro observado na imagem, e A_{esp} é a área calculada através do diâmetro estimado como sendo um diâmetro normal da artéria.

3. RESULTADOS

Os resultados da segmentação por Watershed se mostraram ineficazes, já que houve super-segmentação das imagens, exibida na figura 14. As figuras 11, 12 e 13 exibem o processo anterior à segmentação constituído por gradiente da imagem, abertura e fechamento respectivamente.

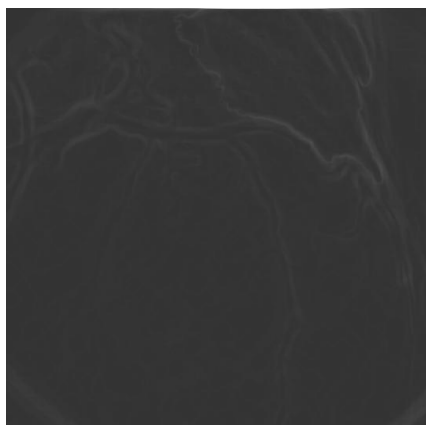


Figura 11: Gradiente da imagem

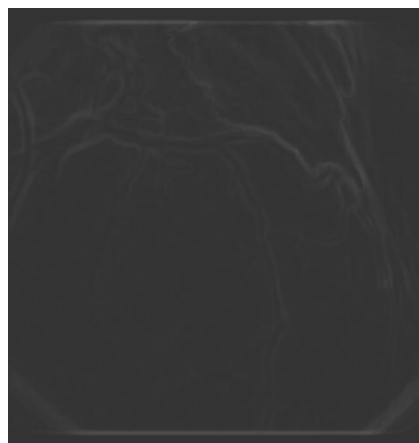


Figura 13: Fechamento do gradiente

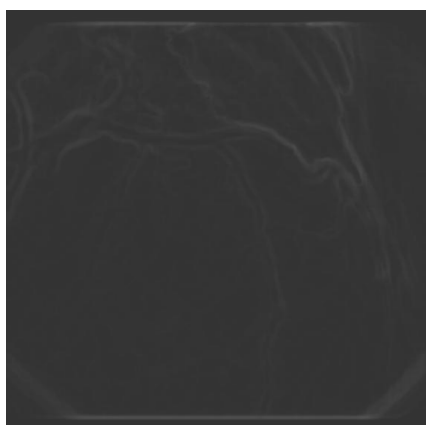


Figura 12: Abertura do gradiente

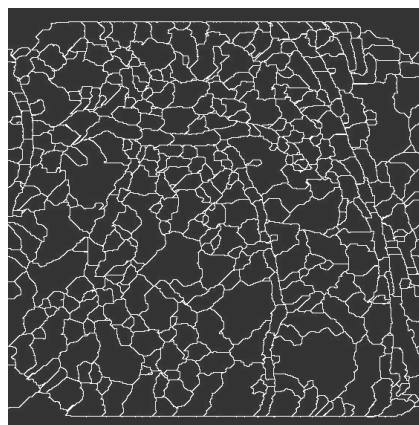


Figura 14: Watershed com sobre-segmentação

Devido ao método de segmentação por Watershed ter sido ineficiente, foi utilizado o método de segmentação por *snakes*.

Os resultados da segmentação da coronária no ponto designado pelo usuário são melhores interpretados quando exibidos por um gráfico, utilizando a biblioteca JFreeChart descrita em Implementação. A curva descreve a luz do vaso pelo comprimento, e permite avaliar a porcentagem de obstrução no ponto de menor fluxo. Com essa porcentagem, é possível avaliar se a obstrução está dentro do limite ou acima do parâmetro significativo de 70% de impedimento do fluxo.

Abaixo estão alguns exemplos da segmentação realizada pelo software desenvolvido. A figura 15 exibe uma interface do software.

Inicialmente é necessário que o usuário indique o ponto inicial e final do vaso que deseja avaliar.

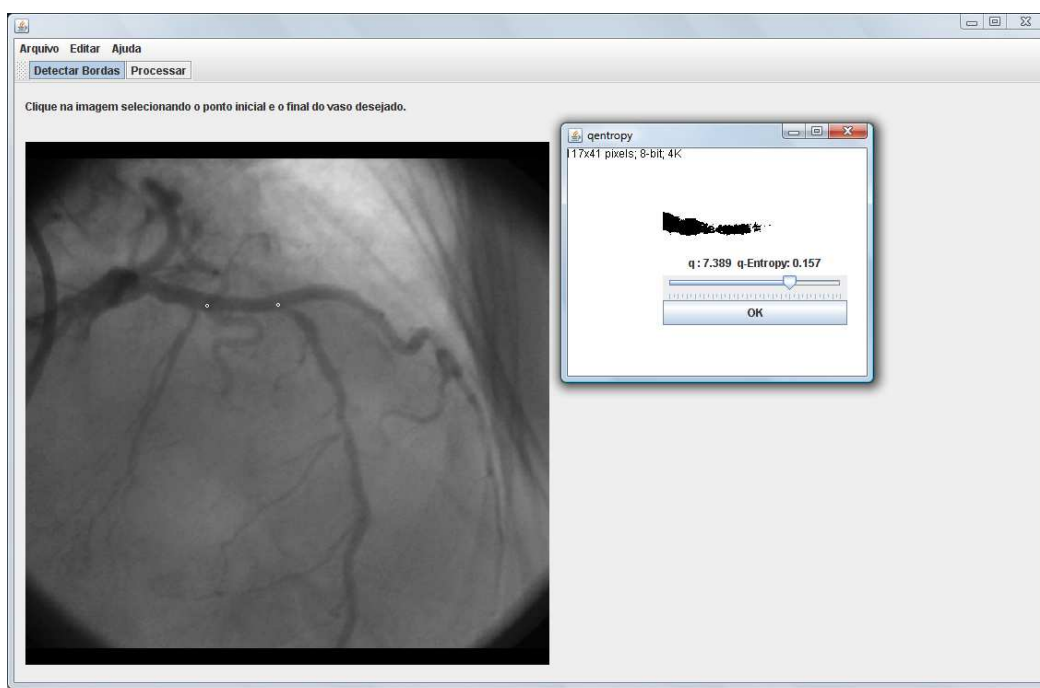


Figura 15: Visão do software em execução

A partir desses pontos, o software cria uma região de interesse (Roi). É então solicitado ao usuário que ele escolha em uma janela auxiliar, um valor q que segmenta o

vaso previamente, demonstrado na figura 16. Esse valor é armazenado e utilizado como parâmetro da Q-Entropia, responsável pela limiarização da imagem. Esse passo foi necessário pelo fato de que nem todas as imagens foram corretamente segmentadas com o valor testado para q de 7,77, que apesar de não ter sido o valor generalizado, foi eficiente na maioria dos casos.

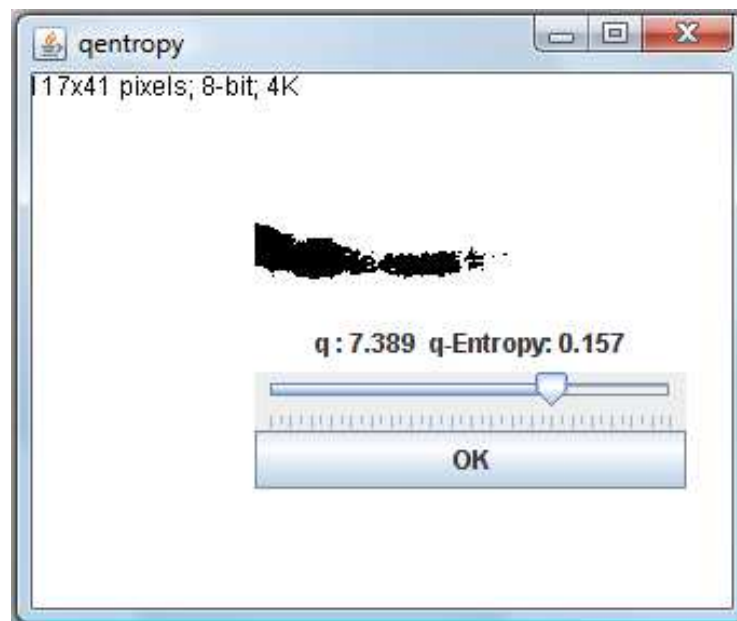


Figura 16: Janela de seleção do melhor parâmetro q

A máscara gerada pela limiarização permite a seleção de uma Roi que será o parâmetro para a *Snakes*. As figuras 17 e 18 demonstram respectivamente a etapa do processamento em que a imagem é rotulada e a etapa em que a Roi do maior objeto é determinada.



Figura 17: Demonstração da rotulação dos objetos



Figura 18: Roi resultante da limiarização

Ao final da *Snakes*, o resultado é uma Roi ajustada ao vaso sanguíneo inicial, sendo que é possível fazer alterações nessa Roi, através dos vértices móveis, exemplificado na figura 19, e reaplicar o cálculo da estenose, gerando um novo gráfico. A figura 20 exibe a interface do software contendo o gráfico resultante do cálculo de estenose. O gráfico em vermelho ilustra a curva de área pelo comprimento da artéria, e em azul está o valor de uma área esperada para essa artéria.

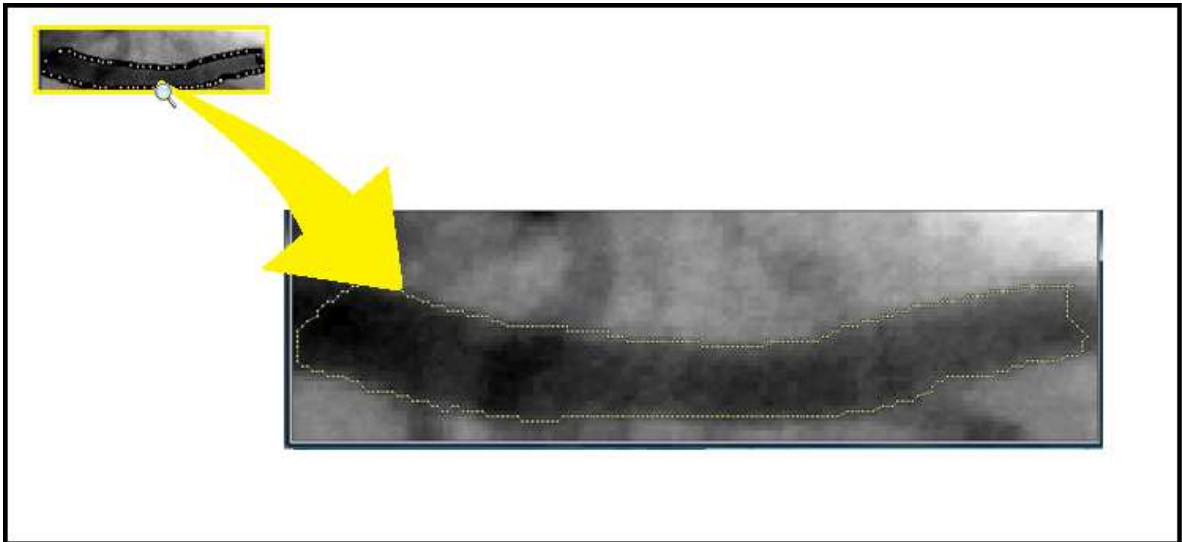


Figura 19: Aproximação da Roi - vértices móveis

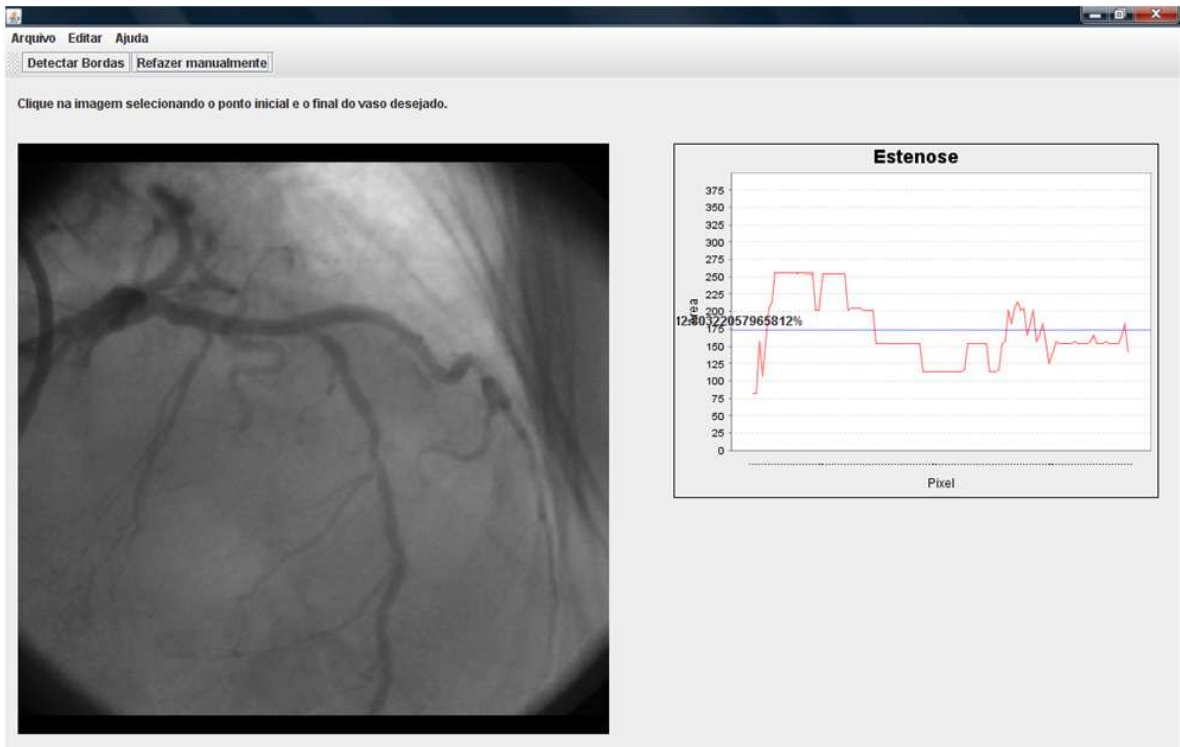


Figura 20: Interface do software exibindo o gráfico resultante do cálculo da estenose

Durante essa análise foi possível verificar que o vaso segmentado não possui obstrução significativa, com obstrução observada de aproximadamente 12%.

Em um segundo caso, foi estudada a imagem ilustrada na figura 21.



Figura 21: Imagem analisada no segundo caso

Aproximando do vaso sanguíneo de interesse é possível verificar uma obstrução, demonstrada na figura 22, no interior do retângulo em destaque.

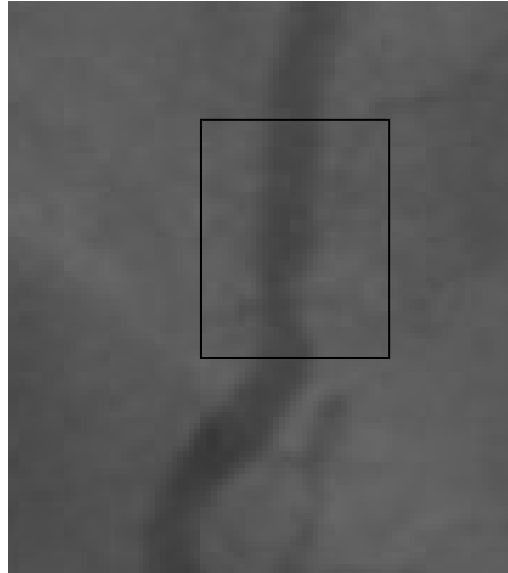


Figura 22: Aproximação do vaso

O gráfico na figura 23 mostra obstrução de quase 67%, uma obstrução possivelmente significativa. Lembrando que se diz hemodinamicamente significativa uma estenose maior ou igual a 70%.

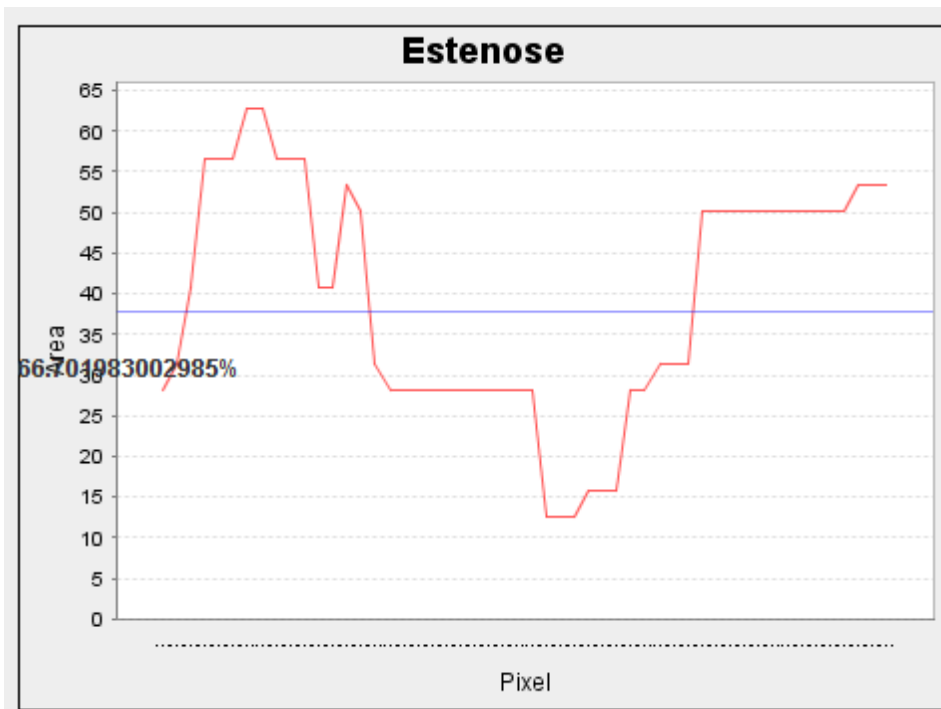


Figura 23: Gráfico do segundo caso

A Roi resultante do processamento pode ser reajustada através dos pontos, de forma definida, vistos nas figuras 24 e 25. A figura 25 exhibe detalhadamente os vértices moveis que permitem a modificação da Roi para um novo cálculo.

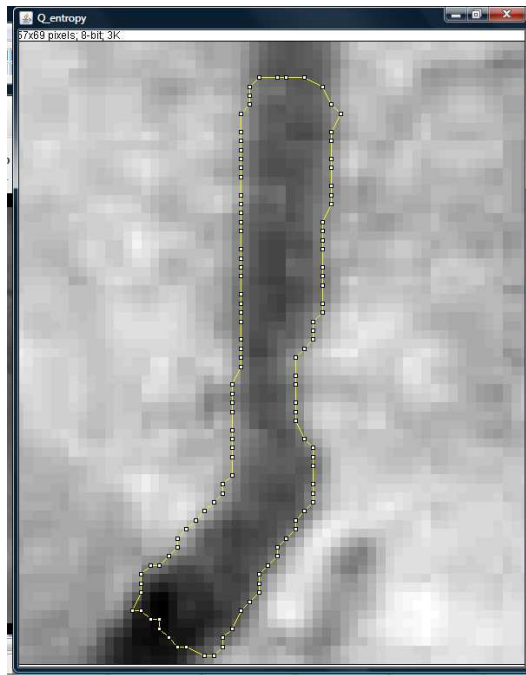


Figura 24: Roi reajustável

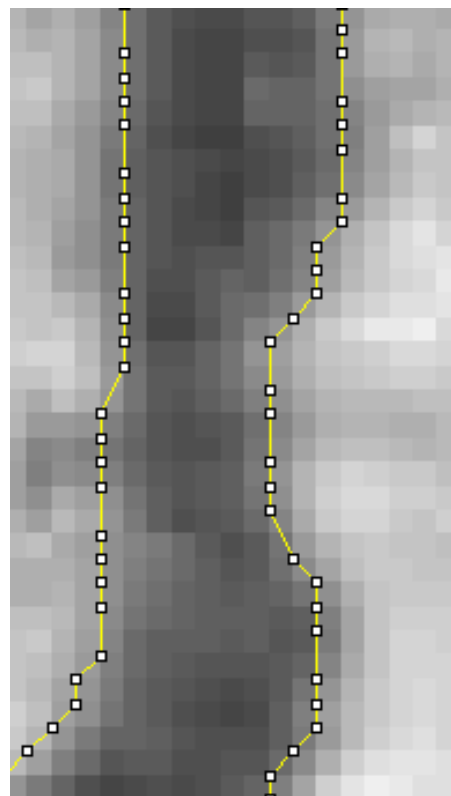


Figura 25: Vértices móveis

O processo que define a luz do vaso pode ser bem entendido pela figura 26: em preto está o esqueleto do vaso e em cinza, a *outline*, que representa a parede da coronária.

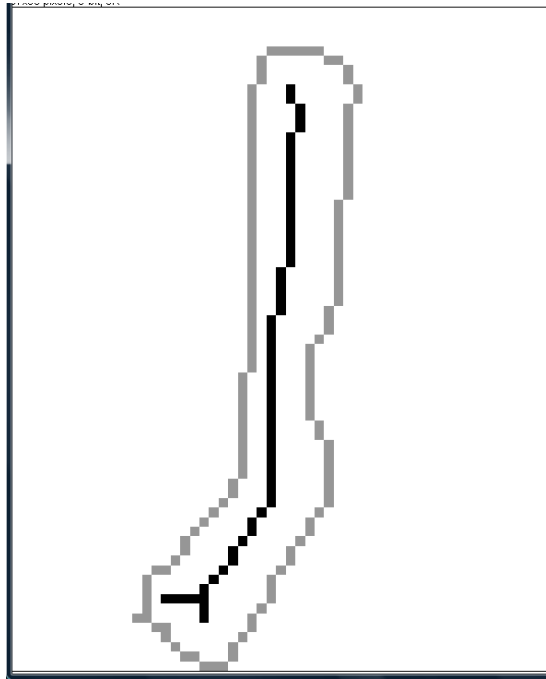


Figura 26: Esqueleto do vaso em preto e outline (paredes) em cinza

Em um terceiro caso, foram obtidas as imagens a seguir. A figura 27 ilustra mais uma vez que é possível modificar uma Roi determinada pelo sistema a fim de obter os resultados mais próximos do desejado. Na figura 28 está o gráfico de luz do vaso pelo comprimento desse caso e na figura 29 o esquema de esqueleto e paredes do vaso em cores diferentes.

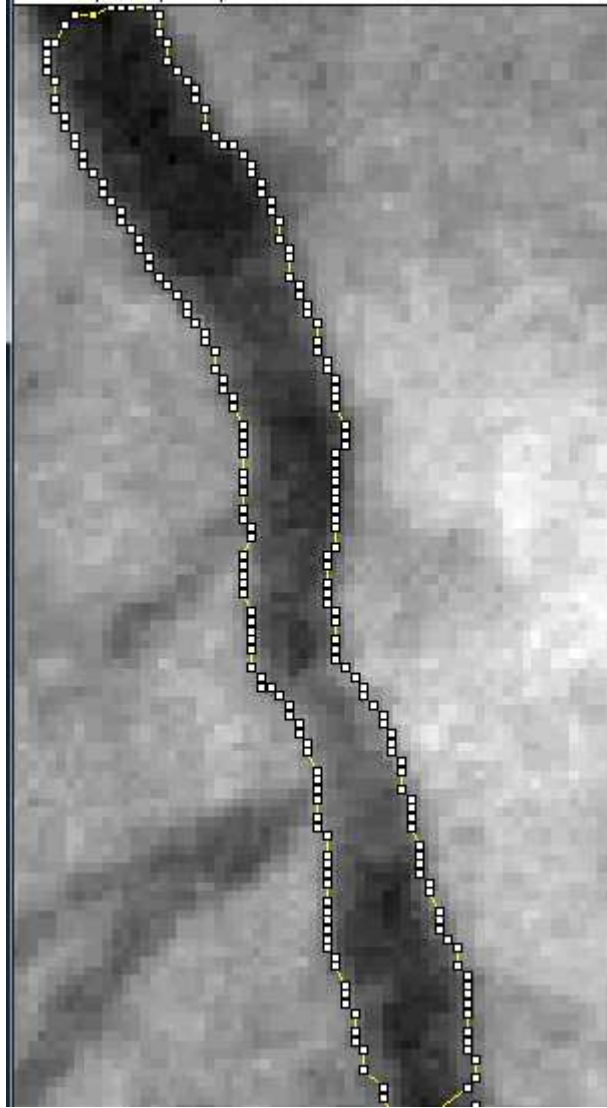


Figura 27: Terceiro caso analisado

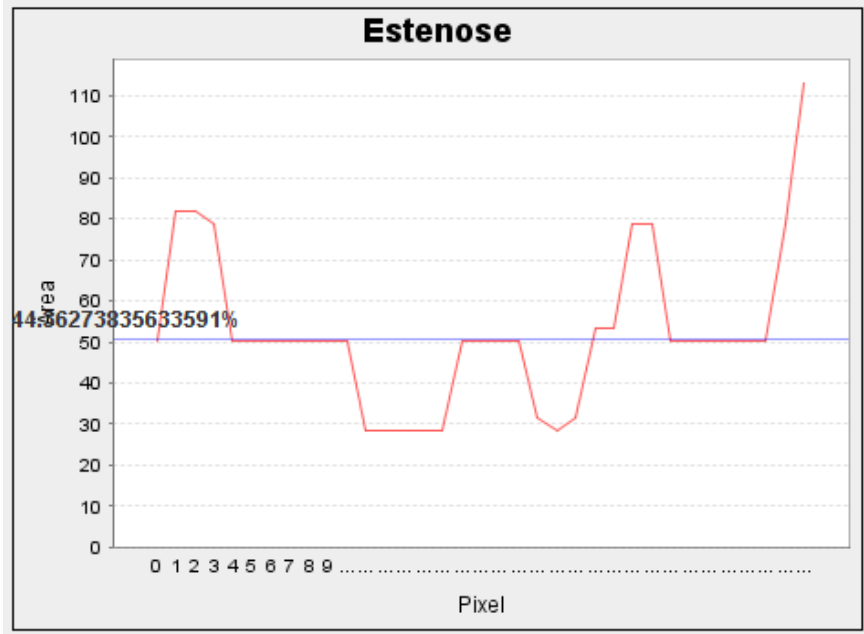


Figura 28: Gráfico gerado com obstrução de 45%

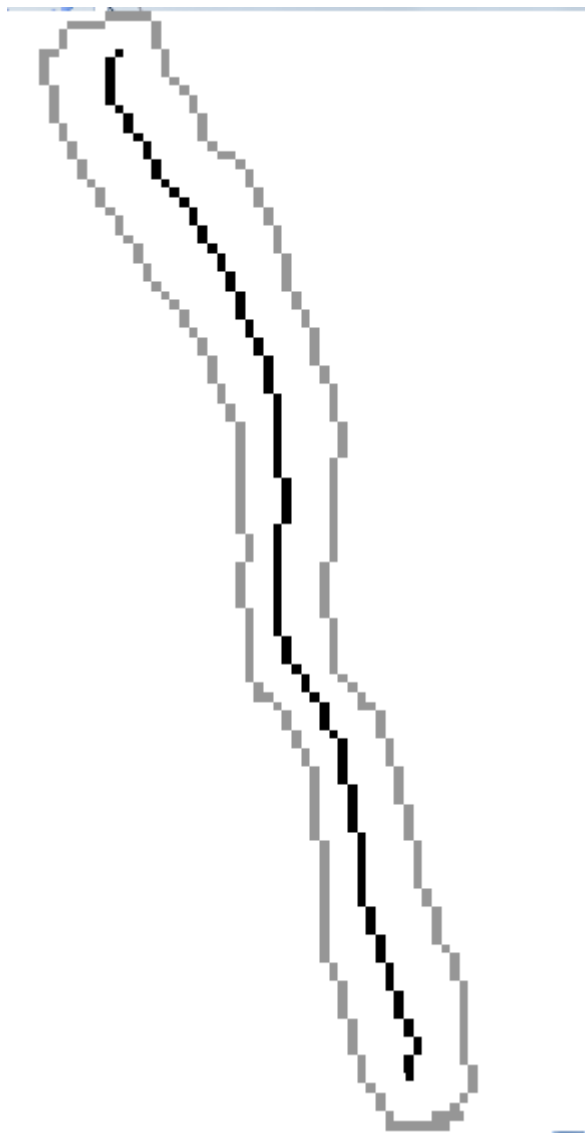


Figura 29: Esqueleto do vaso em preto e paredes em cinza

4. INSTRUÇÕES DE OPERAÇÃO DO SOFTWARE

A qualidade de um software deve ser medida do ponto de vista do usuário. Utiliza-se a palavra qualidade para descrever o grau de excelência de um produto ou serviço. De acordo com a norma ISO (ISO/CD8402 1990), "qualidade é a totalidade das características de um produto ou serviço que lhe confere a capacidade de satisfazer as necessidades implícitas de seus usuários". Portanto, a qualidade está diretamente relacionada à satisfação do usuário ou cliente e é percebida de formas diferentes. Pensando nisso, foi desenvolvido um software semi-automático, visando à facilidade do usuário durante a interação com o sistema. Para que essa interação seja mais fácil e proveitosa, seguirá abaixo algumas instruções de uso desse software.

4.1.ABRINDO E ESCOLHENDO UMA IMAGEM

Para iniciar, o usuário deve escolher uma imagem, ou uma seqüência de imagens que poderá ser nos formatos TIFF, DICOM, FITS, PGM, JPEG, BMP ou GIF. A figura ilustra a interface encontrada pelo usuário para abrir uma imagem. Caso ele deseje abrir mais que uma imagem, ele deve selecionar todas as imagens que deseja visualizar.

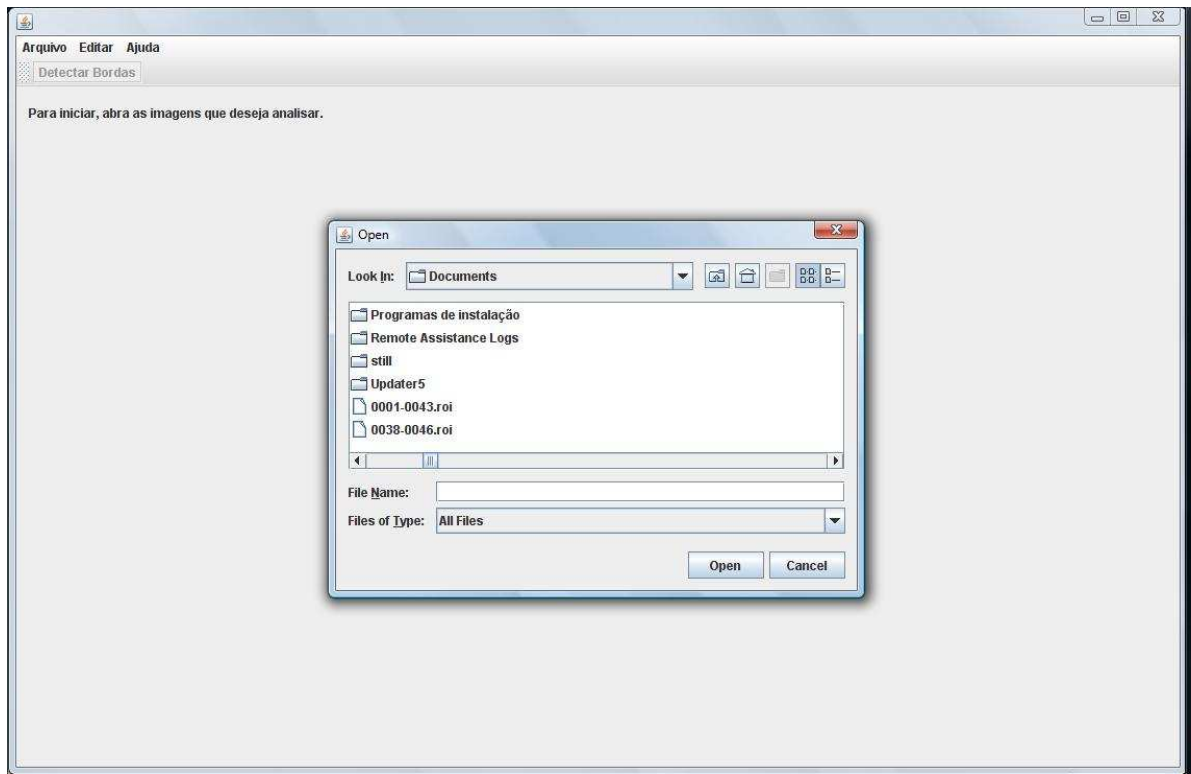


Figura 30: Interface de abertura de uma imagem

Em seguida, caso o usuário tenha escolhido um grupo de imagens ele poderá alterná-las em busca de uma imagem de interesse, como é visto na figura 31. Isso é possível posicionando-se sobre o ponto da barra o qual permita visualizar a imagem desejada, ilustrado na figura 32.

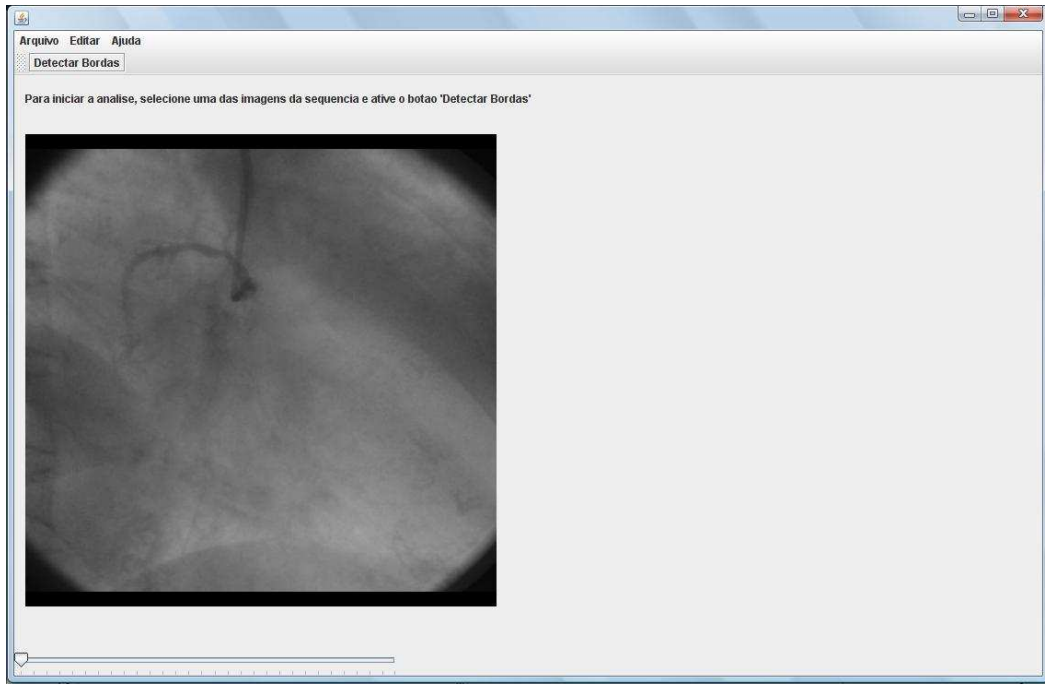


Figura 31: Interface contando barra de navegação das imagens

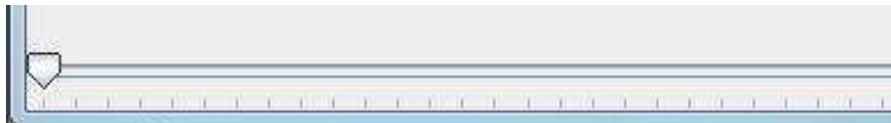


Figura 32: Barra de navegação das imagens

4.2.INICIANDO O PRÉ-PROCESSAMENTO

Após a imagem ter sido determinada, o pré-processamento irá iniciar quando o usuário acionar o botão “Detectar Bordas”. Esse botão abre uma janela que permite ao usuário escolher o melhor valor para q , parâmetro da Q-Entropia. Para isso, também existe uma barra de navegação que desliza sobre valores de q , exibindo assim a limiarização da imagem. Esse passo é demonstrado na figura 33.



Figura 33: Escolha do melhor parâmetro q para limiarização

Conforme a barra de navegação é movida, uma representação da limiarização utilizando o valor de q indicado pelo cursor é exibida. O botão “OK” deverá ser pressionado assim que o usuário decidir qual o melhor valor.

4.3.INICIANDO O PROCESSAMENTO

A partir da determinação do parâmetro q , um botão estará acessível na interface, indicado como “Processar”, encontrado na barra de ferramentas, na parte superior da interface. O resultado dele será um gráfico e uma janela com a seleção do contorno encontrado, o que possibilitaria a detecção manual pelo usuário pressionando o botão “Refazer manualmente” após ter modificado a seleção inicial movendo os vértices da Roi. Os botões estão visíveis na figura 34.

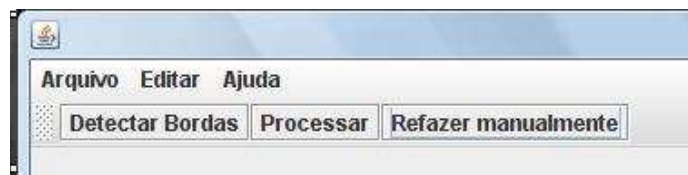


Figura 34: Botões

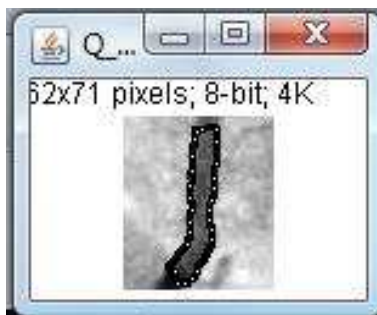


Figura 35: Janela com a seleção do contorno.

4.4.INTERPRETANDO O GRÁFICO

O gráfico gerado tem como eixo X os passos em pixels do comprimento, e eixo Y, a área relativa calculada naquele ponto. A curva vermelha especifica a variação dessa área ao longo do vaso, e a linha azul determina um valor estimado para a área do vaso definido a partir da media entre as áreas de todo o vaso. Sobre o gráfico é indicado o valor calculado de porcentagem de obstrução, visto na figura 35.

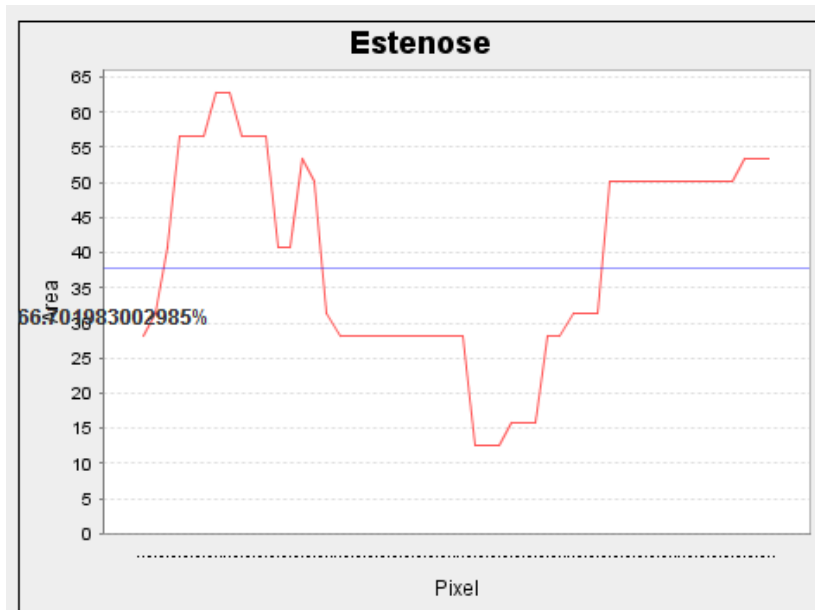


Figura 36: A linha vermelha demonstra a curva de variação da área e a azul, a área esperada.

5. CONCLUSÕES

O objetivo inicial do projeto visava o desenvolvimento de um software totalmente automático para o cálculo de estenose. Porém foi possível perceber que essa automatização completa é difícil, custosa e demorada, não sendo possível realizá-la durante o período de desenvolvimento do projeto de um ano. Foi possível desenvolver um software semi-automatizado para o auxílio a esse cálculo de maneira satisfatória, o que permitirá uma etapa de validação ampla dos métodos propostos aqui, o que não foi possível dentro do tempo disponível para a conclusão da disciplina.

Foi possível também verificar que o método de segmentação por Snakes é muito eficaz dada uma boa segmentação inicial e ajustando os parâmetros de regularização, quando comparado ao método de Watershed, o qual não mostrou resultados viáveis para esta aplicação.

Como etapa posterior a este projeto, pode-se sugerir a melhoria na etapa de pré-segmentação, no sentido de aumentar o nível de automatização da ferramenta.

6. REFERÊNCIAS

- ALBUQUERQUE, M. P., M P. ALBUQUERQUE, D. OLIVEIRA, M. M. G. MACEDO, e I. A. ESQUEF. “Técnica de segmentação de imagens por entropia relativa não extensiva.” 2006.
- ANDREY, P., e T. BOUDIER. “Adaptive active contours (snakes) for the segmentation of complex structures in biological images.” *ImageJ Conference*. 2006.
- BAILER, W. “Writing ImageJ Plugins—A Tutorial.” 2006.
- BENOIT, TREMBLAIS, AUGEREAU BERTRAND, e LEARD MICHEL. “A P.D.E based approach for medial axis detection in X-rays angiographies .” *Proc. SPIE*, 3 de Julho de 2001.
- BOUDIER, T. “Developing a Deformation Model for Complex-Shaped Contours.” *Innov. Tech. Biol. Med.*, 1997: 1-14.
- EICHEL, P. H., E. J. DELP, K. KORAL, e A. J. BUDA. “A method for a fully automatic definition of coronary arterial edges from cine angiograms.” *IEEE TRANSACTIONS ON MEDICAL IMAGING*, 4 de Dezembro de 1988: 313 -320.
- EZQUERRA, N., S. CAPELL, L. KLEIN, e P DUIJVES. “Model-guided labeling of coronary structure Medical Imaging.” *Medical Imaging IEEE Transactions*, Junho de 1998: 429 – 441.
- GIACOMANTONE, J.O. “Ressonância magnética funcional com filtragem pela difusão anisotrópica robusta.” *Dissertação de Mestrado*. 07 de Outubro de 2005.
- GONZALEZ, R.C., e C. WINTZ. *Digital Image Processing*. Adison Wesley, 1992.

- HARIS, K., S.N. EFSTRATIADIS, N. MAGLAVERAS, C. PAPPAS, J. GOURASSAS, e G LOURIDAS. “Model-based morphological segmentation and labeling of coronary angiograms .” *Medical Imaging, IEEE Transactions* , outubro de 1999: 1003-1015.
- HUMMEL, A. “Representations Based on Zero-crossings in Scale Space.” *IEEE COMPUTER VISION AND PATTERN RECOGNITION.*, 1986: 204–209.
- ISO/CD8402. “Quality Concepts and Terminology - Part One: Generic Terms and Definition.” International Standards Organization, 1990.
- KLEIN, A.K., F. LEE, e A.A AMINI. “Quantitative Coronary Angiography with Deformable Spline Models.” *IEEE TRANSACTIONS ON MEDICAL IMAGING*, 5 de outubro de 1997: 468-482.
- KOENDERINK, J. “The Structures of Images.” *Biol. Cybern.*, 1984: 363–370.
- LOPES, F. M. “Um Modelo Perceptivo de Limiarização de Imagens Digitais. .” 2003.
- MEDEIROS, N. G., E. A. SIVA, e J. R. NOGUEIRA. “Segmentação morfológica de imagens utilizando o gradiente morfológico multi escala.” *Revista Brasileira de Cartografia* 54 (Dezembro 2002).
- MIRANDA, J. I, e J. CAMARGO NETO. “DETECÇÃO DE BORDAS COM O MODELO DE DIFUSÃO ANISOTRÓPICA.” *Anais XIII Simpósio Brasileiro de Sensoriamento Remoto*, Abril de 2007: 5957-5964.
- NAJMAN, L, e M. COUPRIE. “Watershed algorithms and contrast preservation.” *Springer*, 2003: 62-71 .

“National Hearth Lung and Blood Institute .”

http://www.nhlbi.nih.gov/health/dci/images/stent_lowres.gif.

PECCINI, G. “Segmentação de Imagens por Watersheds: Uma Implementação Utilizando a Linguagem Java.” 2004.

PERONA, P, e J MALIK. “Scale - Space and Edge Detection Using Anisotropic Diffusion.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Julho de 1990: 629.

SÁ, P. F., e J. L. S. LUZ. “Geometria Fractal ? !” *Traços (UNAMA)* 6 (2003): 103-114.

SILVA, J. S., B. SANTOS, A. SILVA, e J. MADEIRA. “Modelos Deformáveis na Segmentação de Imagens Médicas: uma introdução.” *Revista do Detua* 4 (2004): 360-367.

SOILLE, P. “Morphological Image Analysis - Principles and Applications.” *Springer-Verlag Berlin Heidelberg*, 1999.

WITKIN, A. P. “Scale Space Filtering.” *INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE*, 1983: 1019-1022.