

Instituto de Matemática e Estatística

Universidade de São Paulo, Brasil

# Segmentação de imagens utilizando passeios aleatórios em grafos

Jefferson Serafim Ascaneo

Orientador: Prof. Dr. Paulo A. V. de Miranda

São Paulo, Dezembro de 2012

# Resumo

Segmentação de imagens consiste em dividir uma imagem digital em múltiplos segmentos, que possuam características em comum, permitindo assim analisá-los separadamente. Atualmente, os algoritmos de processamento de imagens que utilizam grafos possuem grande importância [LG12], e podem ser utilizados para realizar esta segmentação.

No final de 2006, foi proposto o método de Passeios Aleatórios (Random Walks) [Gra06], que possui diversas características desejáveis em um algoritmo de segmentação. Assim como outros algoritmos, utiliza sementes para realizar a segmentação, que consistem em conjuntos de pixels previamente marcados, onde todos os pixels de um mesmo conjunto devem pertencer ao mesmo segmento. Devido a várias características interessantes do método Passeios Aleatórios, como a solução única dos potenciais, e a capacidade de detectar bordas fracas ou ruidosas, este método foi utilizado em artigos recentes que tratam de segmentação de imagens [YCZL10]. Também foram verificadas diversas relações teóricas com outros métodos de segmentação [CGNT11] [CGNT09] [SG07].

Neste trabalho, realizou-se um estudo do método e sua implementação na linguagem C, integrando-o aos programas CAOS (Computer-Aided Object Segmentation) e BIA (Brain Image Analyzer). O primeiro realiza a segmentação de imagens em 2D, enquanto que o último trabalha com imagens em 3D. O caso 3D, por envolver maiores restrições de memória, exigiu uma implementação cuidadosa. Para a interface gráfica, foi utilizado o arcabouço wxWidgets<sup>1</sup>, que permite a portabilidade do programa resultante para vários ambientes, como GNU/Linux, OS X e Windows, e utiliza elementos nativos destes sistemas para construir a interface [SHC06]. Além disso, foram realizados testes e comparações com outros métodos da literatura.

---

<sup>1</sup><http://www.wxwidgets.org>. Acesso em: 29 mai. 2012.

# Sumário

<b>I</b>	<b>Objetiva</b>	<b>1</b>
<b>1</b>	<b>Introdução</b>	<b>2</b>
1.1	Motivação . . . . .	2
1.2	Objetivos . . . . .	2
1.3	Organização do Trabalho . . . . .	3
<b>2</b>	<b>Conceitos</b>	<b>4</b>
2.1	Grafos . . . . .	4
2.2	Cadeia de Markov . . . . .	4
2.3	Redes Elétricas . . . . .	5
<b>3</b>	<b>Segmentação de Imagens com Passeios Aleatórios</b>	<b>6</b>
3.1	Passeios aleatórios em uma dimensão . . . . .	6
3.1.1	O passeio de um bêbado . . . . .	6
3.1.2	Voltagens em uma rede elétrica . . . . .	7
3.2	Passeios aleatórios em duas ou mais dimensões . . . . .	8
3.2.1	Funções harmônicas . . . . .	9
3.2.2	O problema de Dirichlet . . . . .	10
3.2.3	Cadeias de Markov . . . . .	12
3.3	Algoritmos . . . . .	13
3.3.1	Método de Monte Carlo . . . . .	13
3.3.2	Método das relaxações . . . . .	13
3.3.3	Algoritmo do <i>Random Walker</i> . . . . .	14
3.4	Segmentação de imagens . . . . .	15
3.4.1	Propriedades do método . . . . .	17
<b>4</b>	<b>Implementação no programa CAOS</b>	<b>20</b>
<b>5</b>	<b>Implementação no programa BIA</b>	<b>22</b>
<b>6</b>	<b>Testes e comparações</b>	<b>24</b>
<b>7</b>	<b>Conclusões</b>	<b>29</b>

<b>II Subjetiva</b>	<b>30</b>
<b>8 Desafios e frustrações encontrados</b>	<b>31</b>
<b>9 Disciplinas relevantes para o trabalho</b>	<b>32</b>
<b>10 Trabalhos futuros</b>	<b>33</b>
<b>Referências Bibliográficas</b>	<b>34</b>

Parte I  
Objetiva

# Capítulo 1

## Introdução

Segmentação de imagens consiste em dividir uma imagem digital em múltiplos segmentos, que possuam características em comum, permitindo isolar elementos de interesse e analisá-los separadamente. Ou, colocando de outra forma, segmentação é a atribuição de rótulos aos *pixels* de uma imagem, de forma que *pixels* que pertençam ao mesmo rótulo tenham características em comum. Na segmentação de imagens com sementes, alguns *pixels* já estão rotulados, e o algoritmo deve usar isto para determinar os rótulos dos *pixels* restantes.

No final de 2006, foi proposto o método de Passeios Aleatórios (*Random Walks*) [Gra06], que possui diversas características desejáveis em um algoritmo de segmentação, e que é o objeto de estudo deste trabalho.

### 1.1 Motivação

Existem diversos algoritmos para realizar a segmentação de imagens, porém este ainda é um problema desafiador. Não existe um algoritmo “ótimo”, e cada um apresenta resultados melhores em certos conjuntos de imagens. Portanto, é importante analisar as características de cada um, e realizar comparações entre eles, para que seja possível a escolha do método a ser aplicado em um conjunto de imagens.

Além disso, uma segmentação correta é necessária para vários processamentos posteriores, que realizarão análises mais aprofundadas nos segmentos produzidos. Se a segmentação deixar de pegar uma região importante, ou acabar pegando regiões da imagem que não fazem parte do objeto de interesse, isto afetará negativamente o desempenho de todas as etapas posteriores de processamento.

O método de Passeios Aleatórios possui algumas características que o tornam interessante, como a capacidade de segmentar imagens com ruído e detectar bordas fracas. Além disso, foram verificadas algumas relações teóricas com outros métodos de segmentação [CGNT11] [CGNT09] [SG07].

### 1.2 Objetivos

Os objetivos deste trabalho incluem um estudo da teoria necessária para a compreensão do funcionamento do método, a implementação do método de Passeios Aleatórios na linguagem de programação C, e a integração desta implementação nos programas CAOS (Computer-Aided Object Segmentation) e BIA (Brain Image Analyzer). O primeiro realiza a segmentação de imagens em 2D, enquanto que o último trabalha com volumes 3D.

Além disso, tem como objetivo realizar comparações com outros métodos da literatura, também baseados em grafos, verificando características como acurácia e tempo de execução.

## 1.3 Organização do Trabalho

No Capítulo 2, apresentamos os conceitos que serão utilizados ao longo da monografia, principalmente no Capítulo 3. No Capítulo 3 é feito um estudo da teoria por trás do método, e da aplicação desta teoria no contexto de segmentação de imagens. Exibimos o resultado da integração do método aos programas CAOS e BIA nos Capítulos 4 e 5. Então analisamos as vantagens e desvantagens do método, comparando-o a outros métodos da literatura, no Capítulo 6. Finalmente, no Capítulo 7 discutimos algumas conclusões obtidas neste trabalho. Nos capítulos seguintes é feita uma análise subjetiva deste trabalho e sua relação com o curso de Bacharelado em Ciência da Computação.

# Capítulo 2

## Conceitos

Neste capítulo apresentaremos algumas definições, que serão utilizadas ao longo desta monografia. Espera-se que o leitor já esteja familiarizado com os conceitos aqui apresentados, e serão indicadas referências que os tratam em maiores detalhes.

### 2.1 Grafos

Um *grafo*  $G$  é um par ordenado  $(V, E)$ , onde  $V$  é um conjunto de *vértices*, e  $E$  é um conjunto disjunto de  $V$ , de *arestas*. Cada elemento de  $E$  é um par não-ordenado de elementos distintos de  $V$ , ou seja, um conjunto com dois vértices. Note que esta definição corresponde à que em muitos textos é tratada como a de grafos simples. Podemos nos referir aos vértices de um grafo  $G$  por  $V(G)$ , e às suas arestas por  $E(G)$ , permitindo que estes conjuntos sejam referenciados mesmo que não tenham sido nomeados anteriormente. Se uma aresta  $e = \{v, w\}$  pertence a  $E$ , diremos que os vértices  $v$  e  $w$  são *vizinhos* ou *adjacentes*, e denotaremos  $e$  simplesmente por  $vw$  ou  $wv$ . O conjunto de todos os vizinhos de um vértice  $v$  será chamado de  $N(v)$ . Estamos interessados apenas em grafos finitos, ou seja, grafos onde o conjunto de vértices e o conjunto de arestas são finitos.

Um *passeio* (*walk*) em um grafo é uma sequência finita não vazia  $(v_0, v_1, \dots, v_k)$  de vértices tal que  $v_{i-1}v_i \in E$  para todo  $1 \leq i \leq k$ . Dizemos que o passeio *vai de*  $v_0$  *a (para)*  $v_k$ , e que os vértices  $v_0$  e  $v_k$  são a *origem* e o *término* do passeio, respectivamente. Os vértices  $v_1, \dots, v_{k-1}$  são chamados de *vértices internos* do passeio.

Um grafo  $G$  é *conexo* se, para cada par  $(u, v)$  de seus vértices,  $G$  contém um passeio com extremos  $u$  e  $v$ .

Um *grafo com pesos*  $G$  é uma tripla  $(V, E, \ell)$ , tal que  $(V, E)$  é um grafo, e  $\ell: E(G) \rightarrow \mathbb{R}$  é uma função que associa um peso  $\ell_e$  a cada aresta  $e$ .

Um *grafo dirigido*  $G$  é um par ordenado  $(V, A)$ , onde  $V$  é um conjunto de vértices, e  $A$  é um conjunto disjunto de  $V$ , de *arcos*. Cada elemento de  $A$  é um par ordenado de elementos distintos de  $V$ .

Para maiores detalhes, uma boa referência é o livro *Graph Theory*, de Bondy e Murty [BM08].

### 2.2 Cadeia de Markov

Uma *cadeia de Markov* é um processo estocástico  $\{X_n, n = 0, 1, \dots\}$ , em que as variáveis aleatórias  $X_n$  assumem valores de  $S$ , onde  $S = \{s_1, s_2, \dots, s_r\}$  é um conjunto de *estados*. Trabalharemos apenas com o caso em que  $S$  é um conjunto finito, embora no caso geral  $S$  possa ser qualquer conjunto enumerável. Se  $X_n = s_k$ , dizemos que o processo *está no estado*  $s_k$  *no tempo*  $n$ . O processo inicia-se em um destes estados e move-se sucessivamente de um estado para outro. Cada movimento (transição) é chamado de *passo*. A probabilidade da cadeia mover-se para o estado  $s_j$ , estando no estado  $s_i$ , é denotada por  $p_{ij}$ , e não depende dos estados pelos quais a cadeia passou anteriormente, dependendo apenas do estado atual. Ou seja:



$$P\{X_{n+1} = s_j | X_n = s_i, X_{n-1} = s_{i_{n-1}}, \dots, X_0 = s_{i_0}\} = p_{ij} \quad \forall s_j, s_i, s_{i_{n-1}}, \dots, s_{i_0} \in S, \forall n \geq 0$$

As probabilidades  $p_{ij}$  são chamadas de *probabilidades de transição*. Uma distribuição de probabilidade inicial, definida em  $S$ , especifica o estado inicial, normalmente especificando um único estado. Como os valores  $p_{ij}$  são probabilidades, e a cada passo o processo precisa mover-se para algum estado (podendo inclusive mover-se para o estado atual), temos que:

$$p_{ij} \geq 0, \quad 1 \leq i, j \leq r; \quad \sum_{j=1}^r p_{ij} = 1, \quad i = 1, \dots, r$$

Maiores detalhes sobre cadeias de Markov e uma introdução à probabilidade podem ser encontrados nos livros *Introduction to Probability Models*, de S. M. Ross [Ros06], e *Introduction to Probability*, de Grinstead e Snell [GS97].

## 2.3 Redes Elétricas

Não entraremos em detalhes sobre este assunto, que será utilizado apenas para ilustrar o funcionamento do método de forma mais didática, além de exibir as relações entre o funcionamento do método e o de uma rede elétrica.

A *resistência elétrica* é a oposição à passagem de corrente elétrica por um elemento. A *condutância elétrica* é o recíproco da resistência, sendo a facilidade com que uma corrente elétrica passa por um elemento. São definidas pelas seguintes fórmulas:

$$R = \frac{V}{I} \quad C = \frac{I}{V}$$

Onde  $R$ ,  $C$ ,  $V$  e  $I$  são a resistência, condutância, voltagem e corrente, respectivamente.

**Lei de Ohm** A diferença de potencial entre dois pontos de um condutor é proporcional à corrente elétrica através destes pontos. Ou seja, em um resistor que obedece à esta lei,  $R$  é um valor constante.

**1ª Lei de Kirchhoff (Lei das Correntes)** Em um nó qualquer de um circuito elétrico, a soma das correntes entrando neste nó é igual à soma das correntes que saem. Ou seja:

$$\sum_{k=1}^n I_k = 0$$

Onde  $n$  é a quantidade de ramos que entram ou saem do nó.

**2ª Lei de Kirchhoff (Lei das Tensões)** A soma algébrica das diferenças de potencial elétrico (voltagens) em um percurso fechado é nula. Ou seja:

$$\sum_{k=1}^n V_k = 0$$

Onde  $n$  é a quantidade total de voltagens medidas neste percurso.

## Capítulo 3

# Segmentação de Imagens com Passeios Aleatórios

Primeiramente, iremos considerar o caso unidimensional. Depois veremos o caso bidimensional, e o algoritmo para realizar a segmentação. Faremos um paralelo entre passeios aleatórios como cadeias de Markov e conceitos elétricos como resistência e voltagem, e iremos nos restringir ao caso de redes finitas. Para um estudo mais aprofundado das relações entre passeios aleatórios e redes elétricas, recomendamos o livro *Random Walks and Electric Networks*, de Doyle e Snell [DS84].

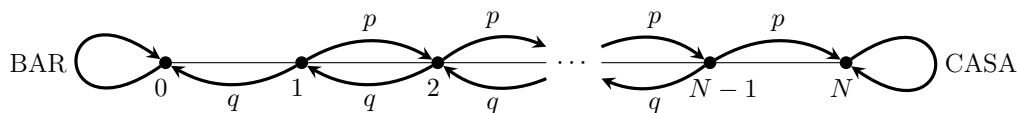
### 3.1 Passeios aleatórios em uma dimensão

#### 3.1.1 O passeio de um bêbado

Um bêbado está andando na rua, entre sua casa e o bar. A cada instante, ele dá um passo na direção do bar com probabilidade  $q > 0$ , e um passo na direção de casa com probabilidade  $p = 1 - q > 0$ . Queremos saber qual a probabilidade dele chegar em casa antes de chegar no bar, dependendo do local da rua em que começa a andar.

Este é um exemplo de um passeio aleatório em uma dimensão, e pode ser resolvido utilizando-se cadeias de Markov. Porém, estamos buscando uma visão mais aprofundada deste tipo específico de problema, com o objetivo de analisar o caso bidimensional e encontrar relações com redes elétricas, além de chegar em uma forma mais eficiente de resolvê-lo numericamente.

Para ilustrar o problema, adicionamos mais um detalhe: se o bêbado está em casa ou no bar, ele permanece no mesmo lugar. Isto é por estarmos interessados apenas em qual local ele chega primeiro, o que significa que devemos desconsiderar casos em que, por exemplo, ele vai para o bar e depois para casa.



**Figura 3.1:** Ilustração do passeio do bêbado, com uma distância de  $N$  passos entre a casa e o bar, probabilidade  $p$  de dar um passo em direção à casa, e probabilidade  $q$  de dar um passo em direção ao bar.

Vamos considerar que a distância entre a casa e o bar é de  $N$  passos, e nomearemos os pontos entre eles (inclusive) de 0 a  $N$ , a partir do bar. Seja  $p(x)$  a probabilidade de chegar em casa antes de chegar no bar partindo do ponto  $x$ . Primeiramente, observamos as seguintes propriedades de  $p(x)$ :

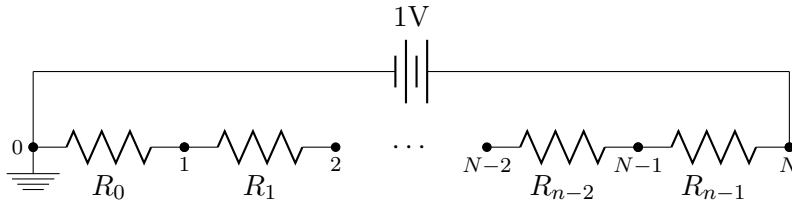
- (i)  $p(0) = 0$ ;
- (ii)  $p(N) = 1$ ;

(iii)  $p(x) = q \cdot p(x - 1) + p \cdot p(x + 1)$  para  $x = 1, \dots, N - 1$ .

O item (i) diz que, estando no bar, a probabilidade de chegar em casa antes de ir ao bar é 0. O item (ii) diz que, se ele está em casa, a probabilidade de chegar lá antes do bar é 1. O item (iii) é mais interessante: a probabilidade de chegar em casa, a partir de um dos pontos interiores do percurso, é igual à probabilidade  $q$ , de ir na direção do bar, vezes a probabilidade de chegar em casa a partir daquele ponto, mais a probabilidade  $p$ , de ir na direção de casa, vezes a probabilidade de chegar em casa a partir deste outro ponto. Podemos enxergar isto como sendo uma média ponderada das probabilidades de se chegar em casa a partir dos dois pontos aos quais o bêbado pode escolher ir.

### 3.1.2 Voltagens em uma rede elétrica

Agora analisaremos um problema aparentemente diferente, mas que possui as mesmas propriedades do anterior. Conectamos vários resistores em série e aplicamos uma voltagem unitária nas pontas, sendo que o ponto 0 está aterrado. Queremos saber qual é a voltagem  $v(x)$  nos pontos  $x$  entre os resistores.



**Figura 3.2:** Circuito com diferença de potencial de 1V entre os pontos 0 e N, e N resistores em série. Queremos saber qual é a diferença de potencial em cada ponto do circuito.

Iremos considerar que temos  $N$  resistores, e nomearemos os pontos entre eles (e nas pontas) de 0 a  $N$ , como na figura. Suponha que a resistência do  $n$ -ésimo resistor seja  $R_n$ , iniciando a contagem em 0. Queremos demonstrar as seguintes propriedades de  $v(x)$ :

(i)  $v(0) = 0$ ;

(ii)  $v(N) = 1$ ;

(iii)  $v(x) = \frac{\frac{1}{R_{x-1}}}{\frac{1}{R_{x-1}} + \frac{1}{R_x}} v(x - 1) + \frac{\frac{1}{R_x}}{\frac{1}{R_{x-1}} + \frac{1}{R_x}} v(x + 1)$  para  $x = 1, \dots, N - 1$ .

As propriedades (i) e (ii) são verdadeiras pela definição do problema. Iremos demonstrar a validade da propriedade (iii).

Para simplificar a notação, iremos usar a *condutância*  $C_x$  no lugar da resistência, onde  $C_x = 1/R_x$ . Queremos então provar a seguinte igualdade:

$$v(x) = \frac{C_{x-1}}{C_{x-1} + C_x} v(x - 1) + \frac{C_x}{C_{x-1} + C_x} v(x + 1) \quad \text{para } x = 1, \dots, N - 1$$

Pelas leis de Ohm e de Kirchhoff, como visto em 2.3, temos que a corrente que flui de um ponto  $x$  a um ponto  $y$  é:

$$i_{xy} = (v(x) - v(y)) \cdot C_{xy}$$

Além disso, para três pontos consecutivos  $x$ ,  $y$  e  $z$  :

$$i_{xy} = i_{yz}$$

Temos então, para  $x = 1, \dots, N - 1$ :

$$\begin{aligned} (v(x-1) - v(x)) \cdot C_{x-1} &= (v(x) - v(x+1)) \cdot C_x \\ v(x)C_x + v(x)C_{x-1} &= v(x-1)C_{x-1} + v(x+1)C_x \\ v(x) \cdot (C_x + C_{x-1}) &= v(x-1)C_{x-1} + v(x+1)C_x \\ v(x) &= v(x-1) \frac{C_{x-1}}{C_x + C_{x-1}} + v(x+1) \frac{C_x}{C_x + C_{x-1}} \end{aligned}$$

Como queríamos provar.

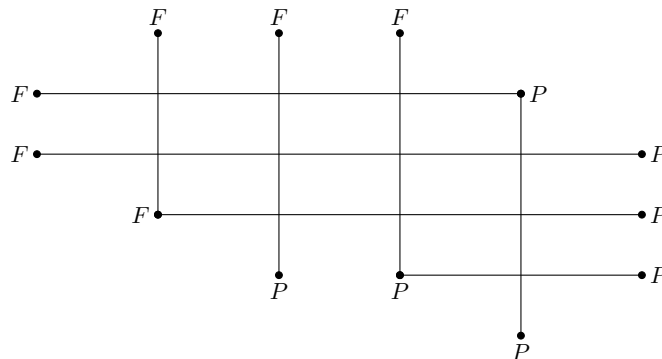
Podemos converter uma instância deste problema para o problema do passeio do bêbado, e vice-versa, da seguinte forma:

$$\begin{aligned} p &= \frac{C_{x-1}}{C_{x-1} + C_x} \\ p \cdot C_x &= (1 - p)C_{x-1} \\ \frac{p}{q} &= \frac{C_{x-1}}{C_x} && \text{pois } q = 1 - p \end{aligned}$$

Note que temos uma certa liberdade em escolher as resistências, pois podemos escolher um valor positivo qualquer para  $R_0$  (e conseqüentemente  $C_0$ ), por exemplo, e determinar  $R_i$  para  $i = 1, \dots, N - 1$  utilizando a equação acima. O que ocorre é o seguinte: como estamos mantendo a diferença de potencial constante entre as duas pontas 0 e  $N$ , diferentes valores para as resistências levarão a diferentes valores da corrente passando pelo circuito. Os valores para as voltagens nos diferentes pontos, entretanto, serão os mesmos se as proporções entre resistências adjacentes mantiverem-se constantes. Observe que estamos assumindo que  $p > 0$  e  $q > 0$  no caso do bêbado, e que não possuímos uma resistência infinita no problema do circuito (pois isso seria equivalente a cortar a ligação entre dois pontos do circuito).

### 3.2 Passeios aleatórios em duas ou mais dimensões

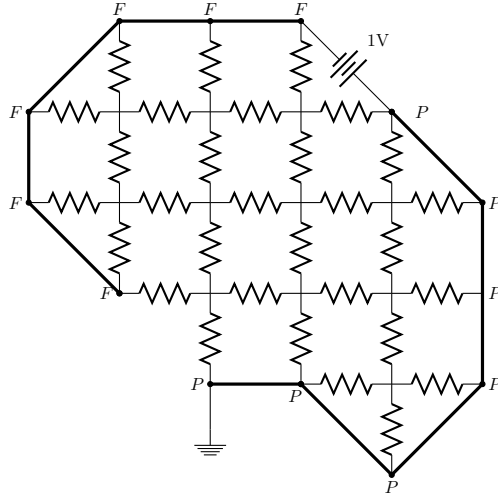
Um fugitivo está correndo pela cidade, conforme a figura 3.3, e queremos saber a probabilidade  $p_{\text{fuga}}(x)$  dele ir para uma rota de fuga antes de encontrar um policial, partindo de um ponto interior  $x$ . Os pontos marcados com  $F$  indicam as rotas de fuga, e os marcados com  $P$  indicam um policial. Em cada ponto  $i$ , ele escolhe ir para o ponto vizinho  $j$  com probabilidade  $p_{ij}$ , que não depende dos pontos pelos quais ele passou anteriormente. Se ele atingir uma rota de fuga ou um policial, ele permanece neste mesmo ponto.



**Figura 3.3:** Cidade por onde o fugitivo está correndo. Em cada cruzamento, ele decide a próxima direção com uma certa probabilidade, que não depende dos locais anteriores por onde ele passou. Os pontos rotulados com  $F$  representam os pontos de fuga, e os pontos rotulados com  $P$  representam os policiais.

Um problema similar em redes elétricas é ilustrado na figura 3.4, onde queremos saber a voltagem

$v(x)$  nos pontos interiores. Os pontos  $F$  estão conectados à uma bateria de um volt, e os pontos  $P$  estão aterrados.



**Figura 3.4:** Circuito similar ao problema do fugitivo. Os pontos com rótulo  $F$  são mantidos em  $1V$ , e os pontos com rótulo  $P$  estão aterrados.

### 3.2.1 Funções harmônicas

Para resolver estes problemas de forma mais genérica, introduziremos o conceito de uma função harmônica. Seja  $G = (V, E, \ell)$  um grafo com pesos finito e conexo, onde  $\forall e \in E, \ell_e > 0$ ;  $V = D \cup B$ ;  $D \cap B = \emptyset$ ; e  $D, B \neq \emptyset$ . Chamaremos os vértices em  $D$  de *pontos interiores* e os vértices em  $B$  de *pontos da fronteira*. Seja  $f: V \rightarrow \mathbb{R}$  uma função. O *operador laplaciano discreto*  $\Delta$  agindo em  $f$  é definido como:

$$(\Delta f)(v) = \sum_{w \in N(v)} \ell_{vw} [f(w) - f(v)]$$

Uma função  $f: V \rightarrow \mathbb{R}$  é chamada de *harmônica* em  $D$  se satisfaz a *equação de Laplace*

$$(\Delta f)(v) = 0 \quad \forall v \in D \quad (3.1)$$

Podemos reescrever a definição (3.1) da seguinte forma:

$$f(v) = \frac{\sum_{w \in N(v)} \ell_{vw} f(w)}{\sum_{w \in N(v)} \ell_{vw}} \quad \forall v \in D \quad (3.2)$$

Ou seja, o valor  $f(v)$  é uma média ponderada dos valores de  $f$  nos pontos vizinhos a  $v$ .

Vamos verificar que  $v(x)$  é uma função harmônica nos pontos interiores  $D$  do circuito, i.e., nos pontos que não estão diretamente ligados à bateria ou aterrados. Os pontos ligados à bateria ou aterrados formam o conjunto  $B$ . Podemos interpretar o circuito como um grafo com pesos, onde os vértices são os nós do circuito,  $\ell_{vw} := C_{vw}$ , e

$$\{v, w\} \in E \Leftrightarrow \text{os pontos } v \text{ e } w \text{ são ligados por um resistor (de condutância } C_{vw})$$

Pelas leis de Ohm e de Kirchhoff temos

$$\sum_{w \in N(u)} C_{uw} [v(w) - v(u)] = 0 \quad \forall u \in D \quad (3.3)$$

Ou seja,  $v(x)$  satisfaz 3.1, e portanto é uma função harmônica em  $D$ .

É possível converter uma instância deste problema da rede elétrica no problema do fugitivo. Primeiro definimos as probabilidades  $p_{ij}$  de escolher ir para o ponto  $j$  a partir de  $i$  como

$$p_{ij} = \begin{cases} \frac{C_{ij}}{\sum_{j \in N(i)} C_{ij}} & \text{se } i \in D \\ 1 & \text{se } i = j \text{ e } i \in B \\ 0 & \text{caso contrário} \end{cases}$$

Os pontos  $x \in B$  tais que  $v(x) = 1$  (os pontos ligados à bateria) representam os pontos de fuga, e os pontos  $x \in B$  tais que  $v(x) = 0$  (os pontos aterrados) representam os policiais.

Lembrando que  $p_{\text{fuga}}(x)$  é a probabilidade do fugitivo conseguir fugir a partir do ponto  $x$ , temos

$$p_{\text{fuga}}(x) = v(x) \quad \forall x \in B$$

Vamos verificar que  $p_{\text{fuga}}(x)$  é uma função harmônica nos pontos interiores da cidade, utilizando o seguinte fato:

**Fato.** Seja  $E$  um evento, e  $\{F_i \mid 1 \leq i \leq n\}$  um conjunto finito de  $n$  eventos tal que exatamente um dos eventos  $F_i$  ocorre. Então

$$\Pr(E) = \sum_{i=1}^n \Pr(F_i) \cdot \Pr(E \mid F_i)$$

Fazendo  $E_i$  ser o evento “o fugitivo escapa partindo de  $i$ ”, e  $F_{ij}$  o evento “a primeira direção escolhida é a  $j$ -ésima direção”, temos

$$\begin{aligned} \Pr(E_i) &:= p_{\text{fuga}}(i) \\ \Pr(E_i \mid F_{ij}) &:= p_{\text{fuga}}(j) \\ \Pr(F_{ij}) &:= p_{ij} \end{aligned}$$

E portanto

$$p_{\text{fuga}}(i) = \sum_{j \in N(i)} p_{ij} \cdot p_{\text{fuga}}(j) \quad \forall i \in D \quad (3.4)$$

Note que  $\sum_{j \in N(i)} p_{ij} = 1$ , o que completa a demonstração de que  $p_{\text{fuga}}$  é da forma especificada em (3.2), ou seja,  $p_{\text{fuga}}$  é harmônica em  $D$ .

Veremos adiante que, como  $p_{\text{fuga}}(x)$  e  $v(x)$  são iguais nos pontos  $x \in B$ , e as restrições (3.3) e (3.4) são equivalentes pela definição de  $p_{ij}$ , pelo Princípio da Unicidade ambos os problemas possuem a mesma solução, ou seja,  $p_{\text{fuga}}(x) = v(x) \quad \forall x \in V$ .

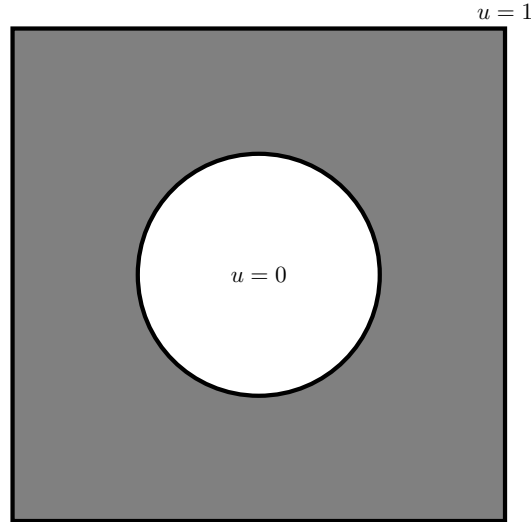
Para mostrar que  $v(x)$  é uma função harmônica, não utilizamos em nenhum momento a dimensão do problema ao qual esta função se refere. Portanto, os resultados são válidos mesmo em problemas mais complexos do que os que ilustram o início desta seção. Isso será importante quando tratarmos da segmentação de imagens em 3D.

### 3.2.2 O problema de Dirichlet

Imagine que temos uma fatia fina de metal, com um furo no centro, e queremos saber as temperaturas ao longo desta fatia, enquanto mantemos a borda externa na temperatura 1 e a borda interna na temperatura 0. Se  $u(x, y)$  é a temperatura no ponto  $(x, y)$ , a função  $u$  satisfaz a equação de Laplace

$$\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$$

Encontrar esta função  $u$  é um exemplo do problema original de Dirichlet, sendo que os problemas que vimos até agora são uma versão discreta deste problema. Mais precisamente, o problema de



**Figura 3.5:** Chapa de metal com um furo no centro. A borda externa é mantida na temperatura 1, e a borda interna é mantida na temperatura 0. Queremos saber a temperatura  $u(x, y)$  para cada ponto  $(x, y)$  da chapa.

Dirichlet consiste em, dado um domínio  $D$ , um contorno (ou fronteira)  $\partial D$ , e uma função  $g$  definida em  $\partial D$ , encontrar uma função  $f$  que satisfaça

$$\begin{cases} \Delta f = 0 & \text{se } x \in D \\ f = g & \text{se } x \in \partial D \end{cases}$$

Uma forma de encontrar uma solução aproximada para este problema é justamente considerar uma versão discreta, dividindo a fatia de metal em uma quantidade suficientemente grande de quadrados de mesmo tamanho, e resolver o problema na grade resultante.

Iremos provar agora algumas propriedades sobre soluções para a versão discreta do problema de Dirichlet.

**Princípio do Máximo.** Uma função  $f: V \rightarrow \mathbb{R}$  harmônica em  $D$  assume seu valor máximo em  $B = V \setminus D$ .

*Demonstração.* Seja  $M$  o valor máximo de  $f$ . Se  $f(v) = M$  para algum  $v \in B$ , não há nada a provar.

Suponha que existe  $v \in D$  tal que  $f(v) = M$ . Como

$$f(v) = \frac{\sum_{t \in N(v)} \ell_{vt} f(t)}{\sum_{t \in N(v)} \ell_{vt}}; \quad \text{e } f(v) \geq f(t), \forall t \in N(v)$$

segue que  $f(v) = f(t) = M, \forall t \in N(v)$ . Ou seja, a função  $f$  também assume este valor máximo  $M$  em todos os vizinhos de  $v$ .

Repetindo este raciocínio para os vizinhos de  $v$ , e assim por diante, e lembrando que  $G = (V, E)$  é conexo, eventualmente chegamos ao caso em que pelo menos um vizinho  $u$  do vértice sendo considerado pertence a  $B$ . Então concluímos que  $f(u) = M$ , ou seja,  $f$  assume seu valor máximo em  $B$ , como queríamos provar.  $\square$

**Princípio do Mínimo.** Uma função  $f: V \rightarrow \mathbb{R}$  harmônica em  $D$  assume seu valor mínimo em  $B = V \setminus D$ .

*Demonstração.* Análoga à anterior.  $\square$

**Princípio da Unicidade.** Se  $f(x)$  e  $g(x)$  são funções harmônicas em  $D$  tais que  $f(x) = g(x)$  em  $B$ , então  $f(x) = g(x)$  para todo  $x \in V$ .

*Demonstração.* Seja  $h(x) = f(x) - g(x)$ . Então

$$h(x) = \frac{\sum_{y \in N(x)} \ell_{xy} f(y)}{\sum_{y \in N(x)} \ell_{xy}} - \frac{\sum_{y \in N(x)} \ell_{xy} g(y)}{\sum_{y \in N(x)} \ell_{xy}} = \frac{\sum_{y \in N(x)} \ell_{xy} (f(y) - g(x))}{\sum_{y \in N(x)} \ell_{xy}} = \frac{\sum_{y \in N(x)} \ell_{xy} h(y)}{\sum_{y \in N(x)} \ell_{xy}}$$

e  $h(x)$  é harmônica em  $D$ . Como  $h(x) = 0$  para todo  $x \in B$ , segue pelos Princípios do Máximo e do Mínimo que os valores máximo e mínimo de  $h$  são 0. Portanto  $h(x) = 0$  para todo  $x \in V$ , e  $f(x) = g(x)$  para todo  $x \in V$ .  $\square$

### 3.2.3 Cadeias de Markov

Podemos utilizar a teoria de Cadeias de Markov para modelar um passeio aleatório e encontrar, por exemplo, as probabilidades do fugitivo ir para uma rota de fuga antes de encontrar um policial. Neste contexto os pontos da fronteira são chamados de *estados absorventes*, e os pontos interiores são chamados de *estados transientes*. Os estados absorventes são os estados dos quais não é possível sair uma vez que são atingidos. Como existe pelo menos um estado absorvente, e a partir de qualquer estado é possível atingir um estado absorvente (não necessariamente em um passo), temos uma *cadeia de Markov absorvente*.

Seja  $r$  a quantidade de estados absorventes, e  $t$  a quantidade de estados transientes. Construímos a matriz  $P$  com as probabilidades de transição  $p_{ij}$  na forma canônica, isto é, com os estados transientes antes dos estados absorventes:

$$P = \begin{bmatrix} Q & R \\ \mathbf{0} & I_r \end{bmatrix}$$

onde  $Q$  é uma matriz de tamanho  $t \times t$ ,  $R$  é uma matriz  $t \times r$ , e  $I_r$  é a matriz identidade  $r \times r$ .

Um objeto importante no estudo de cadeias de Markov absorventes é a matriz fundamental, definida como:

$$N = (I_t - Q)^{-1}$$

e é possível provar a existência desta matriz inversa, mas não faremos esta prova pois ela pode ser encontrada, por exemplo, em [GS97].

Seja  $b_{ij}$  a probabilidade da cadeia ser absorvida pelo estado absorvente  $s_j$ , ao iniciar no estado transiente  $s_i$ . Seja  $B$  a matriz  $t \times r$  com entradas  $b_{ij}$ . Então

$$B = NR$$

e com isso temos as probabilidades que queríamos, como a de o fugitivo encontrar um policial antes de fugir. Note que, na prática, encontrar a matriz  $B$  envolveria a resolução do sistema linear

$$(I_t - Q)B = R$$

mas infelizmente a matriz  $I_t - Q$  não possui algumas propriedades que seriam interessantes, como simetria, por exemplo, já que em geral  $p_{ij} \neq p_{ji}$ .

É importante salientar que estamos interessados no caso de passeios aleatórios em grafos *não dirigidos*, e estes podem ser modelados por uma cadeia de Markov absorvente. Porém, o contrário nem sempre é verdade, e algumas cadeias de Markov absorventes só podem ser transformadas em passeios aleatórios em grafos *dirigidos*. Dependendo das probabilidades que se queira utilizar no problema do fugitivo, ele nem sempre pode ser convertido em um problema elétrico equivalente que possui apenas resistores, pois os resistores permitem a passagem de corrente elétrica em ambas as direções. Se estívéssemos interessados no caso de grafos dirigidos, poderíamos estender nossa



definição de função harmônica:

**Definição.** Sejam  $\mathbf{P}$  uma cadeia de Markov absorvente,  $D$  os estados transientes de  $\mathbf{P}$ , e  $B$  os estados absorventes de  $\mathbf{P}$ . Uma função  $f$ , cujo domínio é o espaço de estados de  $\mathbf{P}$ , é chamada de *harmônica* em  $D$  se

$$f(i) = \sum_j p_{ij} f(j) \quad \forall i \in D$$

onde as probabilidades  $p_{ij}$  são as probabilidades de transição de  $\mathbf{P}$ .

Porém, encontrar os valores de  $f$  neste caso não é prático, e utilizaremos apenas a definição anterior, que permite soluções mais eficientes.

### 3.3 Algoritmos

Iremos agora discutir alguns algoritmos que resolvem a versão discreta do problema de Dirichlet. Para simplificar, iremos supor que os valores de  $f$  nos pontos da fronteira são 0 ou 1 (ou seja,  $g: B \rightarrow \{0, 1\}$ ), e que existem pontos  $u$  e  $v$  da fronteira tais que  $f(u) = 0$  e  $f(v) = 1$  (caso contrário  $f$  seria constante).

#### 3.3.1 Método de Monte Carlo

O primeiro algoritmo é conhecido como o *Método de Monte Carlo*. Para cada ponto interior  $x$ , simulamos  $n$  passeios aleatórios começando neste ponto, que terminam ao alcançar algum ponto da fronteira. Calculamos o valor  $f(x)$  como sendo a média dos valores de  $f$  nos pontos da fronteira que foram alcançados pelos passeios aleatórios.

Embora este algoritmo seja bem simples de implementar, e consuma pouca memória, para se conseguir uma aproximação de  $f$  com uma precisão razoável, a quantidade de passeios simulados deve ser extremamente grande. Na prática este algoritmo é inviável, devido à sua ineficiência.

#### 3.3.2 Método das relaxações

Um algoritmo mais eficiente do que o anterior é o seguinte:

---

**Algoritmo 2** Método das relaxações

---

**Entrada:** número  $n$  de vértices, conjunto de pontos da fronteira  $B$ , e função  $g: B \rightarrow \mathbb{R}$ ;

**Saída:** aproximação da função  $f$  harmônica em  $D = V \setminus B$ , tal que  $f(x) = g(x) \forall x \in B$ .

```

1: para  $x \leftarrow 1$  até  $n$  faça
2:   se  $x \in B$  então
3:      $f[x] \leftarrow g[x]$ 
4:   senão
5:      $f[x] \leftarrow 0$ 
6: para  $i \leftarrow 1$  até  $K$  faça
7:   para  $x \leftarrow 1$  até  $n$  faça
8:     se  $x \notin B$  então
9:        $f[x] \leftarrow \frac{\sum_{w \in N(v)} \ell_{vw} f[w]}{\sum_{w \in N(v)} \ell_{vw}}$ 
devolva  $f$ 

```

---

Este algoritmo é bem mais rápido do que o anterior, e também consome pouca memória. Porém, o número de iterações necessárias para se conseguir uma boa aproximação de  $f$  ainda é muito grande, ou seja, o valor de  $K$  no algoritmo deve ser bem alto para que a resposta seja satisfatória.

### 3.3.3 Algoritmo do *Random Walker*

Defina o grau  $d_i$  do vértice  $v_i$  como:

$$d_i = \sum_{\forall j \in N(i)} \ell_{i,j}$$

E seja a matriz Laplaciana combinatória  $L$  como definida a seguir:

$$L_{ij} = \begin{cases} d_i & \text{se } i = j \\ -\ell_{ij} & \text{se } v_i \text{ e } v_j \text{ são adjacentes} \\ 0 & \text{caso contrário} \end{cases}$$

onde  $L_{ij}$  está indexado pelos vértices  $v_i$  e  $v_j$ . É fácil ver que  $L_{ij}$  é uma matriz simétrica, basta observar que a ordenação dos vértices nas linhas e colunas é a mesma, e a definição é simétrica com relação a  $i$  e  $j$ .

Vamos analisar a multiplicação  $Lf$ . Para cada linha  $i$  de  $L$ , esta multiplicação é equivalente a

$$[Lf]_i = d_i f(i) + \sum_{j \in N(i)} -\ell_{ij} f(j) = \sum_{j \in N(i)} (\ell_{ij} f(i) - \ell_{ij} f(j)) = \sum_{j \in N(i)} \ell_{ij} (f(i) - f(j)) = -(\Delta f)(i)$$

Esta inversão do sinal do operador laplaciano discreto faz com que  $L$  tenha uma propriedade interessante: ela é uma matriz semidefinida positiva.

**Definição.** Sejam  $G = (V, E, \ell)$  um grafo, e  $e$  uma aresta de  $G$ . Defina  $L_e^G$  como a matriz Laplaciana do subgrafo  $H = (V, \{e\}, \ell)$ .

Isto nos permite escrever  $L = \sum_{e \in E} L_e^G$ .

**Fato.** A matriz Laplaciana  $L$  é semidefinida positiva.

*Demonstração.* Sejam  $x$  um vetor não nulo e  $L$  a matriz laplaciana de um grafo  $G$ . Então

$$x^T Lx = x^T \left( \sum_{e \in E} L_e^G \right) x = \sum_{e \in E} x^T L_e^G x = \sum_{\{i,j\} \in E} \ell_{ij} (x_i - x_j)^2$$

Como  $\ell > 0$ , temos que  $x^T Lx \geq 0$ , o que conclui a prova.  $\square$

Estamos interessados em achar  $f$  que satisfaça

$$\begin{cases} \Delta f = 0 & \text{se } x \in D \\ f = g & \text{se } x \in B \end{cases} \quad (3.5)$$

Podemos ordenar as linhas e colunas de  $L$  de forma que os pontos da fronteira  $B$  apareçam primeiro, e os pontos interiores  $D$  apareçam por último.

$$L = \begin{bmatrix} L_B & C \\ C^T & L_D \end{bmatrix}$$

Portanto, para encontrar  $f = \begin{bmatrix} f_B \\ f_D \end{bmatrix}$  que satisfaça 3.5, basta fazer  $f_B = g_B$ , e encontrar  $f_D$  que satisfaça

$$\begin{bmatrix} C^T & L_D \end{bmatrix} \begin{bmatrix} g_B \\ f_D \end{bmatrix} = \mathbf{0}$$

o que equivale a resolver o seguinte sistema linear

$$L_D f_D = -C^T g_B \quad (3.6)$$

**Fato.** A matriz  $L_D$  é definida positiva.

*Demonstração.* Seja  $x$  um vetor não nulo, e  $H$  o subgrafo de  $G$  induzido pelos vértices em  $D$ . Note que, como  $G$  é um grafo conexo, existem arestas  $\{i, j\}$  tais que  $i \in B$  e  $j \in D$ . Denote o conjunto destas arestas por  $\partial(D)$ . Defina  $E_k^j$  como a matriz de tamanho  $|D| \times |D|$  que possui o valor  $k$  na posição  $(j, j)$ , e zeros em todas as outras posições. Então

$$\begin{aligned} x^T L_D x &= x^T \left( \sum_{\{i,j\}|i,j \in D} L_{\{i,j\}}^H + \sum_{\substack{\{i,j\} \in \partial(D) \\ j \in D}} E_{\ell_{\{i,j\}}}^j \right) x \\ &= \sum_{\{i,j\}|i,j \in D} x^T L_e^G x + \sum_{\substack{\{i,j\} \in \partial(D) \\ j \in D}} x^T E_{\ell_{\{i,j\}}}^j x \\ &= \sum_{\{i,j\}|i,j \in D} \ell_{ij} (x_i - x_j)^2 + \sum_{\substack{\{i,j\} \in \partial(D) \\ j \in D}} \ell_{\{i,j\}} x_j^2 \\ &> 0 \end{aligned}$$

Note que a única forma do primeiro somatório ser igual a 0 é se  $x$  possuir o mesmo valor em todas as posições, já que  $H$  é um grafo conexo. Porém, neste caso o segundo somatório será estritamente maior do que 0.  $\square$

Como a matriz  $L_D$  é simétrica e definida positiva, o sistema linear (3.6) pode ser resolvido utilizando o método do gradiente conjugado<sup>1</sup>, que é um método iterativo para solucionar sistemas esparsos deste tipo. Note que este sistema linear possui  $|D|$  equações, o que pode ser um número muito grande, porém cada linha de  $L_D$  possui no máximo  $K + 1$  posições não nulas, onde  $K$  é o grau máximo de  $G$ . Este algoritmo é bem mais rápido do que o Método das relaxações, porém consome mais memória. Ainda assim, a utilização de um método iterativo permite que o sistema linear possa ser resolvido na prática, com consumo de memória bem inferior ao de um método direto, como uma decomposição de Cholesky.

Observe que, pelas propriedades de soluções do problema discreto de Dirichlet, que foram provadas anteriormente, o sistema possui solução, e ela é única. Isto também segue do fato da matriz  $L_D$  ser definida positiva.

### 3.4 Segmentação de imagens

Agora discutiremos a aplicação do algoritmo no contexto de segmentação de imagens com sementes.

Segmentação de imagens consiste em atribuir rótulos aos *pixels* de uma imagem, onde *pixels* de mesmo rótulo possuem características em comum. Isto permite que as regiões da imagem sejam isoladas e analisadas separadamente. Na segmentação de imagens com sementes, alguns *pixels* já estão rotulados (estes são chamados de *sementes*), e o algoritmo deve utilizar estes rótulos para determinar os rótulos dos *pixels* restantes.

A partir de uma imagem, podemos construir um grafo  $G = (V, E, \ell)$  da seguinte forma: para cada *pixel* da imagem, temos um vértice que o representa. As arestas podem ser definidas de diferentes formas, sendo as mais comuns a *vizinhança-4* e a *vizinhança-8*. Na vizinhança-4 os vizinhos de um

<sup>1</sup>[http://en.wikipedia.org/wiki/Conjugate\\_gradient\\_method](http://en.wikipedia.org/wiki/Conjugate_gradient_method). Acesso em: 25 nov. 2012.

*pixel* são os *pixels* imediatamente adjacentes na horizontal e na vertical. Na vizinhança-8 temos como vizinhos, além dos *pixels* da vizinhança-4, também os *pixels* nas diagonais.



**Figura 3.6:** Para cada *pixel*, seus vizinhos podem ser definidos de inúmeras formas. A figura mostra a vizinhança-4 e a vizinhança-8, os tipos mais comuns de vizinhança para imagens 2D, com os *pixels* em azul representando os vizinhos do *pixel* em vermelho.

Os pesos  $\ell_e$  podem ser definidos de inúmeras formas, mas iremos utilizar a seguinte definição:

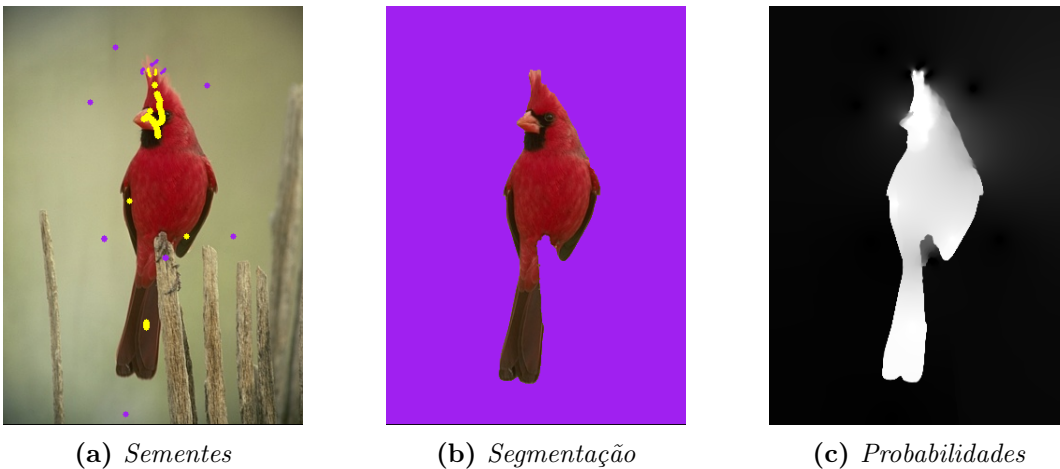
$$\ell_{ij} = \left( \frac{\text{MAX} + 1 - |h_i - h_j|}{\text{MAX} + 1} \right)^p \quad (3.7)$$

onde  $h_i$  representa a intensidade do *pixel*  $i$ , MAX é a maior intensidade que um *pixel* pode assumir, e  $p \geq 1$  é uma constante (tipicamente  $p \geq 7$ , para obter bons resultados). Utilizamos MAX + 1, e não MAX, para garantir que  $\ell_{ij} > 0$ .

A ideia do algoritmo é a seguinte: para determinarmos qual é o rótulo de um *pixel*  $u$ , consideramos passeios aleatórios que iniciam em  $u$  e terminam assim que chegam em um *pixel* rotulado. Calculamos então a probabilidade de cada rótulo ser o primeiro a ser atingido, a partir de um passeio aleatório iniciado em  $u$ . O rótulo que possui a maior probabilidade de ser o primeiro a ser atingido é atribuído a  $u$ . Nestes passeios aleatórios, a probabilidade  $\Pr(e_{u,v})$  de ir para um vértice  $v$ , estando em  $u$ , é proporcional ao peso  $\ell_{u,v}$  associado à aresta  $e_{u,v}$ .

$$\Pr(e_{u,v}) = \frac{\ell_{u,v}}{\sum_{w \in N(u)} \ell_{u,w}}$$

Como os pesos representam as diferenças entre os *pixels*, os passeios têm maior probabilidade de percorrerem regiões mais uniformes, evitando bordas.



**Figura 3.7:** Segmentação de um pássaro com rótulos de objeto e fundo, utilizando passeios aleatórios

Devido à discussão anterior sobre passeios aleatórios, sabemos que para encontrar as probabilidades que queremos, podemos utilizar os algoritmos da seção anterior. Além disso sabemos que,

para cada *pixel*, a soma de todas as probabilidades calculadas é 1. Portanto podemos evitar a resolução do sistema linear referente ao último rótulo, somando as probabilidades dos outros rótulos e subtraindo o resultado de 1. Como normalmente a quantidade  $K$  de rótulos é pequena, isto gera um ganho de velocidade significativo. Chegamos então ao seguinte algoritmo:

1. Encontre os pesos como definido em (3.7).
2. Receba o conjunto  $V_B$  de *pixels* rotulados, com  $K$  rótulos.
3. Para cada rótulo  $s_i$ , exceto o último, calcule a probabilidade  $f^{s_i}(v)$  associada a cada *pixel*  $v$  resolvendo (3.6), onde o valor de  $g_B^{s_i}$  é definido da seguinte forma:

$$g_B^{s_i}(v) = \begin{cases} 1 & \text{se } v \text{ pertence a } s_i \\ 0 & \text{se } v \text{ pertence a um rótulo diferente de } s_i \end{cases}$$

4. Para o último rótulo  $s_K$ , calcule a probabilidade  $f^{s_K}(v)$  associada a cada *pixel*  $v$  somando as probabilidades calculadas para os outros rótulos e subtraindo de 1:

$$f^{s_K}(v) = 1 - \sum_{i=1}^{K-1} f^{s_i}(v)$$

5. Atribua a cada *pixel*  $v$  o rótulo correspondente a  $\arg \max_i(f^{s_i}(v))$ .

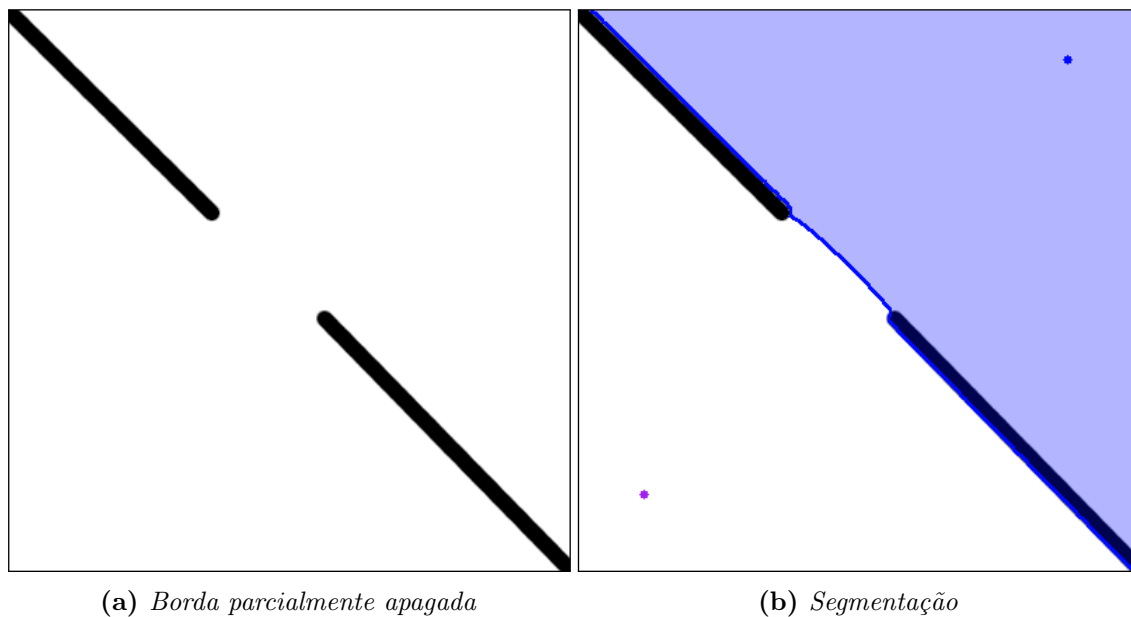
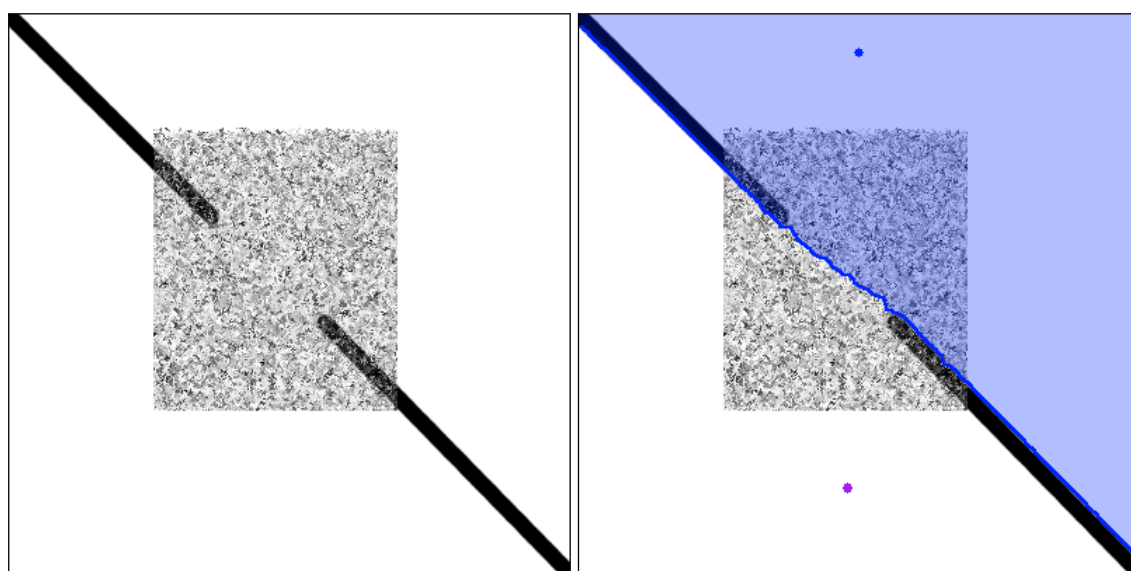
### 3.4.1 Propriedades do método

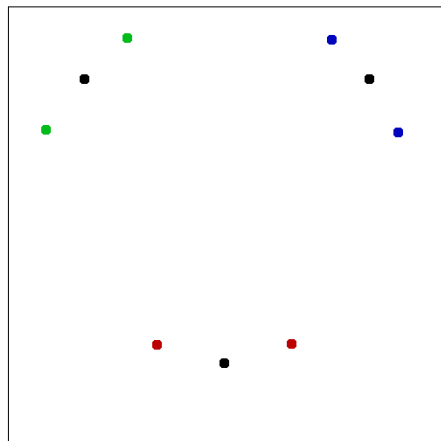
O método de segmentação de imagens com passeios aleatórios possui algumas propriedades interessantes, dentre as quais destacamos as seguintes:

1. A  $K$ -tupla de probabilidades para cada vértice é igual à média ponderada das tuplas de probabilidades dos vértices vizinhos;
2. A solução das probabilidades é única;
3. A segmentação esperada de uma imagem de ruído puro é igual à obtida em uma imagem uniforme.
4. O método é capaz de detectar algumas bordas parcialmente apagadas.

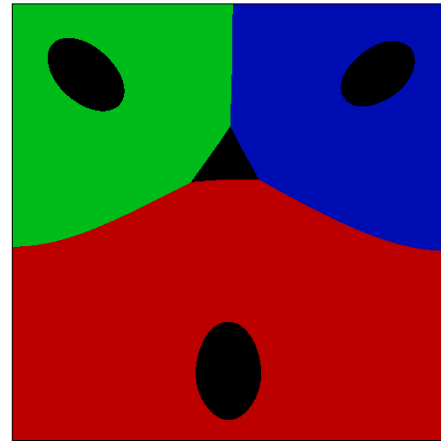
As primeiras duas propriedades já foram provadas, e as restantes serão apenas ilustradas, com alguns exemplos de segmentações com ruído e bordas parciais, que podem ser vistos nas figuras 3.8 e 3.9.

Uma propriedade que não é válida [CZ11], mas que aparece no artigo original do método de passeios aleatórios, é a de que todos os segmentos possuem pelo menos um *pixel* que é uma semente, ou seja, não ocorrem regiões isoladas que não estão conectadas a nenhuma semente. Isto é verdade para segmentações com dois rótulos, mas é possível criar contra-exemplos com três ou mais rótulos, e um destes contra-exemplos pode ser visto na figura 3.10. Entretanto, eles parecem não ocorrer na prática, e normalmente os segmentos estão conectados a alguma semente de mesmo rótulo.

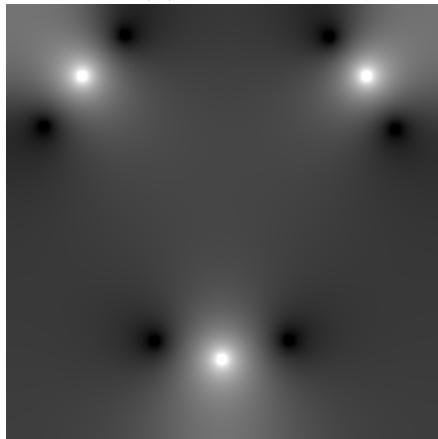
(a) *Borda parcialmente apagada*(b) *Segmentação***Figura 3.8:** *Segmentação de imagem que apresenta uma borda parcialmente apagada.*(a) *Borda parcialmente apagada e com ruído*(b) *Segmentação***Figura 3.9:** *Segmentação de imagem que apresenta uma borda parcialmente apagada e com ruído.*



(a) Sementes



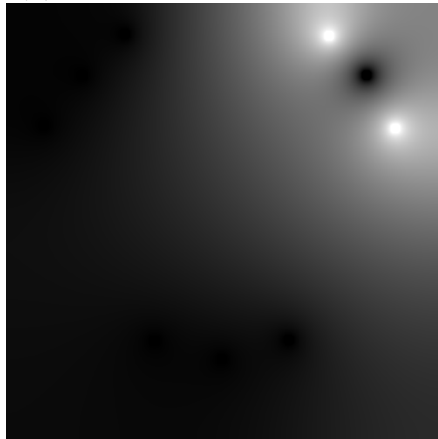
(b) Segmentação



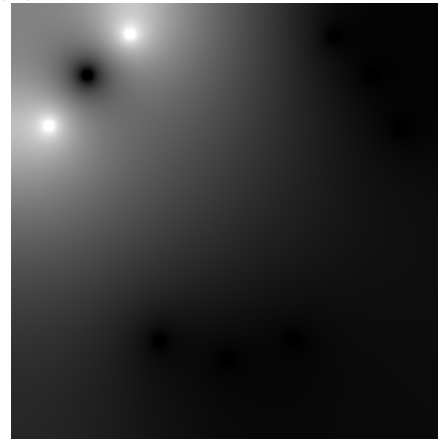
(c) Probabilidade do rótulo preto



(d) Probabilidade do rótulo vermelho



(e) Probabilidade do rótulo azul



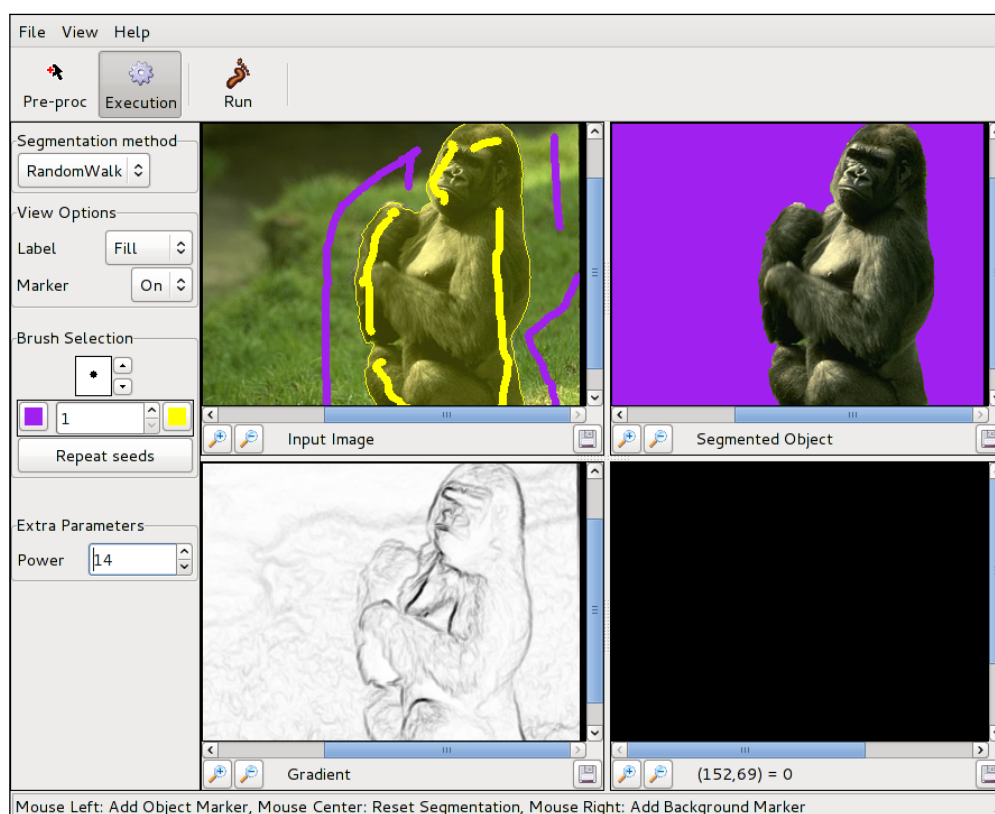
(f) Probabilidade do rótulo verde

**Figura 3.10:** Segmentação de imagem onde aparece uma região isolada sem sementes. A imagem é uniforme (branca), e são utilizadas sementes de 4 rótulos. O rótulo da cor preta, que possui 3 sementes, é atribuído a uma região triangular no centro da imagem, porém esta região não contém nenhuma semente deste rótulo. Podemos ver pelas probabilidades que, por estarem distribuídas pela imagem, as sementes do rótulo preto fazem com que a probabilidade de um pixel pertencer a ele seja uniforme no centro da imagem. Já os outros rótulos possuem grande probabilidade apenas na região próxima às suas sementes, e perdem para o rótulo preto na região central.

## Capítulo 4

# Implementação no programa CAOS

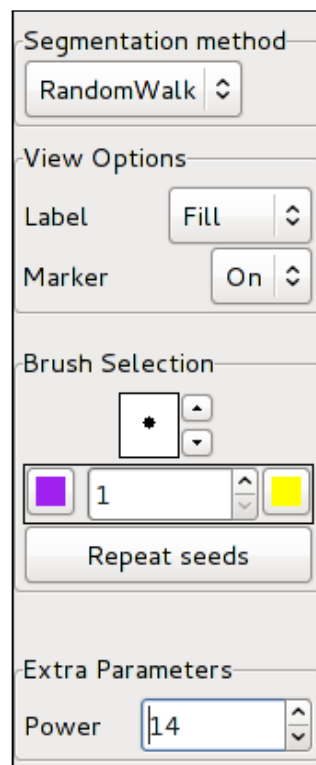
O método foi integrado ao programa CAOS (Computer-Aided Object Segmentation), que realiza segmentações de imagens em duas dimensões, e utiliza o arcabouço wxWidgets para a interface gráfica. A figura 4.1 mostra o programa sendo utilizado para realizar a segmentação de uma foto de um gorila. A tela é dividida em quatro partes: a primeira mostra a imagem original com os marcadores de objeto (em amarelo) e fundo (em roxo), já com o resultado da segmentação e sua borda sobrepostos; a segunda mostra apenas o objeto que foi segmentado e o fundo é exibido em roxo; a terceira mostra o gradiente da imagem original; e a quarta não é utilizada pelo método.



**Figura 4.1:** CAOS utilizando o método “Random Walks”

O programa possui um painel, como pode ser visto na figura 4.2, que permite selecionar o método “Random Walks”, entre outros. Em “Extra Parameters”, pode-se selecionar a potência utilizada nos pesos do grafo.



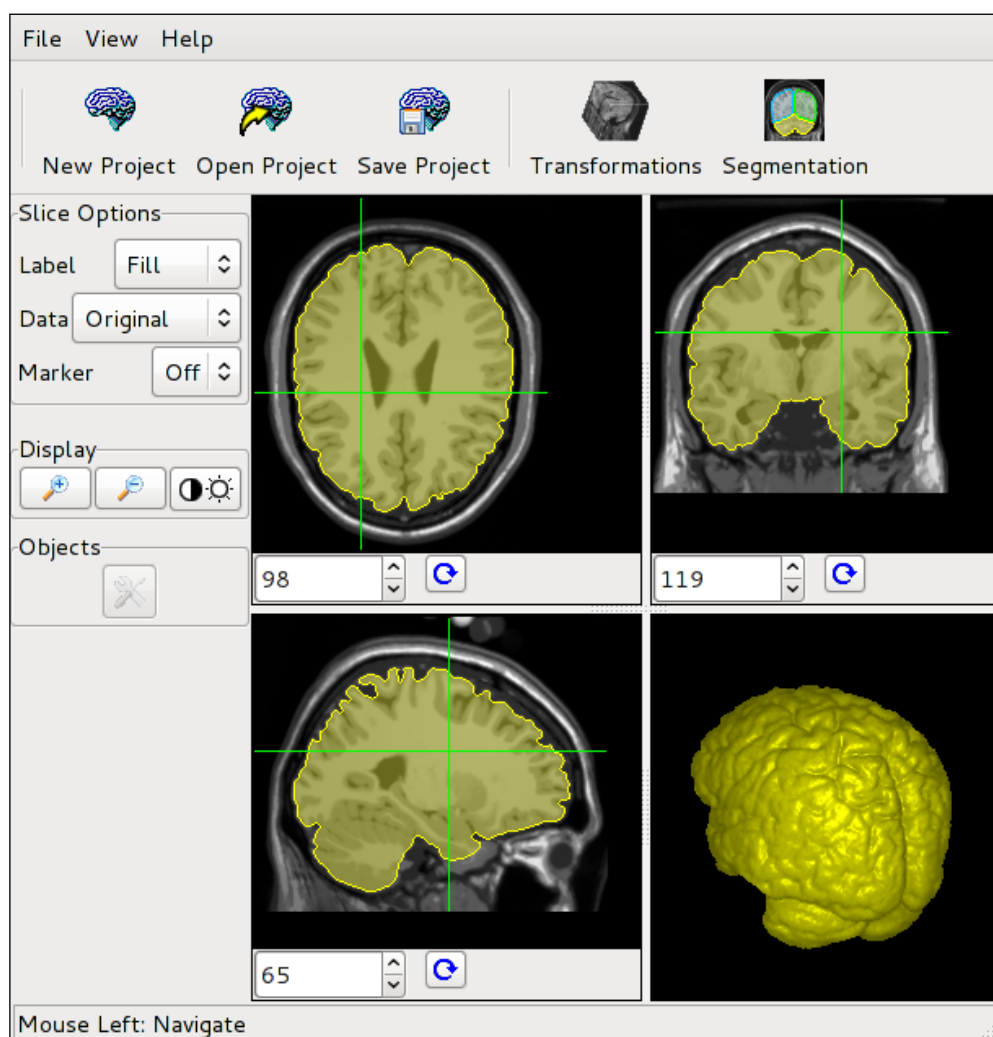


**Figura 4.2:** Painel do CAOS com o método “Random Walks”. O painel permite alterar algumas opções de visualização da segmentação, mudar o tamanho e cor do pincel que é utilizado para criar as sementes, e alterar a potência utilizada pelo método nos pesos do grafo.

## Capítulo 5

# Implementação no programa BIA

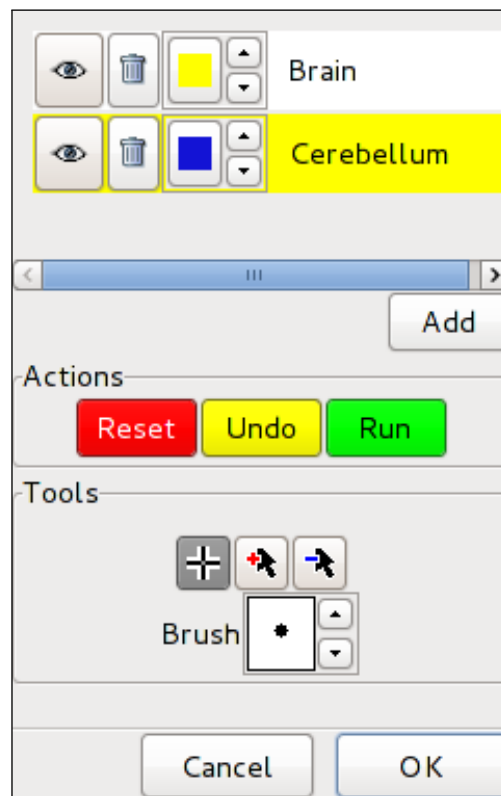
O método também foi integrado ao programa BIA (Brain Image Analyzer), que realiza segmentações de volumes em três dimensões, e utiliza o arcabouço wxWidgets para a interface gráfica. A figura 5.1 mostra o programa sendo utilizado para realizar a segmentação de uma ressonância magnética sintética do *BrainWeb - Simulated Brain Database*<sup>1</sup>. A tela é dividida em quatro partes: a três primeiras exibem diferentes eixos de visualização – axial, coronal e sagital – e a última mostra uma visualização 3D da superfície do objeto segmentado.



**Figura 5.1:** Resultado de uma segmentação 3D de uma ressonância magnética sintética (do BrainWeb - Simulated Brain Database) no programa BIA

<sup>1</sup><http://brainweb.bic.mni.mcgill.ca/brainweb/>

O programa possui um painel, como pode ser visto na figura 5.2. É possível adicionar rótulos com diferentes nomes e cores, além de adicionar e remover sementes, mudar o tamanho do pincel e executar o método de segmentação.



**Figura 5.2:** Painel do BIA com o método “Random Walks”. O painel permite adicionar e remover rótulos com diferentes nomes e cores. Nesta figura existem dois rótulos: o cérebro em amarelo, e o cerebelo em azul. Também permite alternar entre a adição e a remoção de sementes, mudar o tamanho do pincel que é utilizado para adicionar as sementes, e executar o método de segmentação.

## Capítulo 6

# Testes e comparações

O método de segmentação com Passeios Aleatórios (*Random Walk* – RW) foi comparado aos métodos *Relative Fuzzy Connectedness (RFC) + Graph Cut (GC)* [CMUF12] e *Iterative Relative Fuzzy Connectedness (IRFC)* [CUSZ07], ambos baseados em grafos. Para a execução do *Random Walk*, os pesos do grafo foram elevados à potência  $p = 20$ , pois este valor gerou os melhores resultados na segmentação de ossos do pé, e valores maiores deixam o método extremamente lento.

Para comparar a acurácia dos métodos, foi utilizada uma medida de similaridade entre o gabarito e as segmentações obtidas pelos métodos, para diferentes conjuntos de sementes. A medida de similaridade usada foi o coeficiente de Dice <sup>1</sup>, que mede a similaridade entre conjuntos. Quanto mais próximo de 1 no eixo  $y$  (*dice*), melhor o resultado. As sementes são obtidas pela erosão e dilatação do gabarito por um raio  $r$ , exibido no eixo  $x$  (*erosion radius*). Um exemplo de obtenção das sementes a partir do gabarito, e as respectivas segmentações dos métodos, pode ser visto nas figuras 6.1 e 6.2.

Foram testadas imagens de ressonância magnética dos ossos do pé, para realizar a segmentação do Calcâneo e do Tálus. São 40 fatias de  $256 \times 256$  *pixels* que foram selecionadas a partir de dados de cenas 3D de ressonância magnética, relativos ao pé esquerdo ou direito de 20 indivíduos diferentes. As fatias foram selecionadas aleatoriamente, em localizações anatômicas semelhantes de cada cena 3D. Nesta base de dados, o método de Passeios Aleatórios apresentou os melhores resultados, como pode ser visto nas figuras 6.3, 6.4 e 6.5.

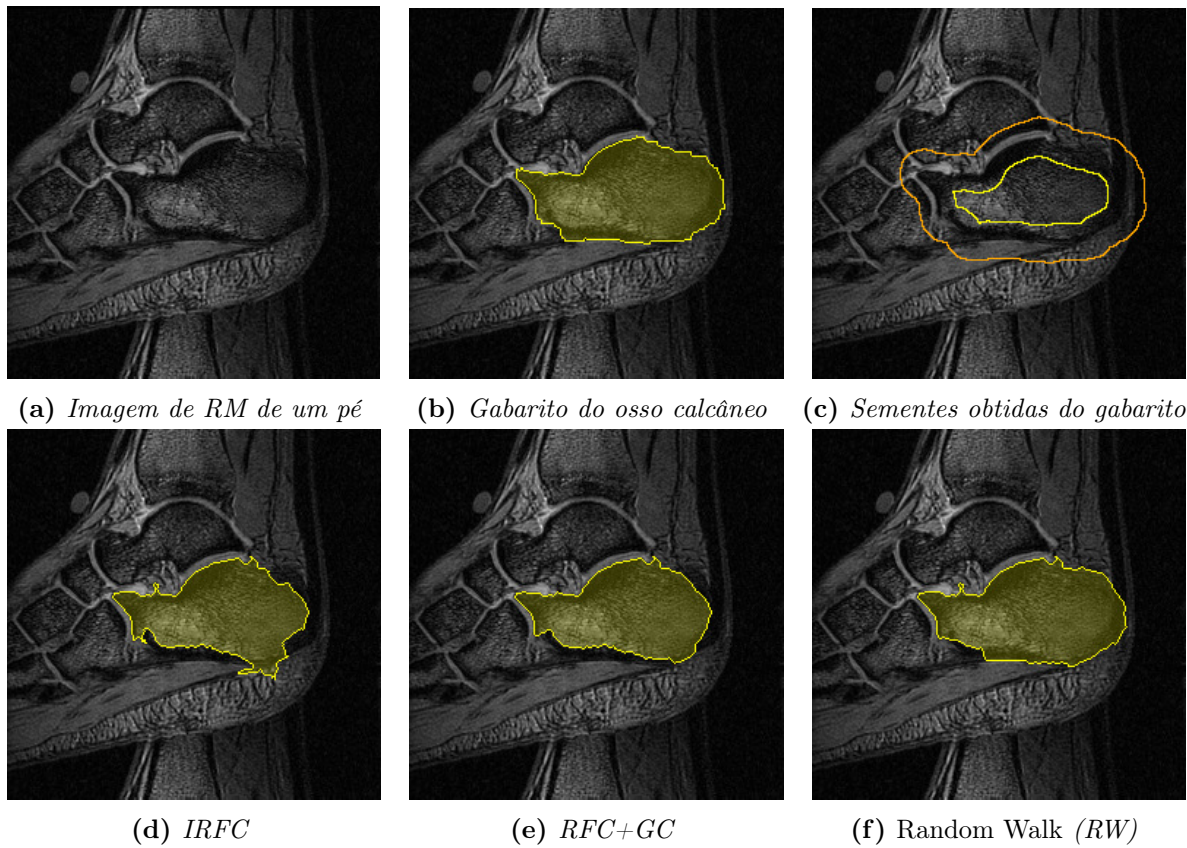
As imagens 6.6a e 6.6b mostram os resultados de experimentos de segmentação da anatomia óssea do pé, para diferentes valores da potência  $p$ , onde os objetos de interesse são o Tálus e o Calcâneo, respectivamente. Note a influência do parâmetro  $p$  conforme o raio  $r$  de dilatação e erosão. Para valores menores de  $r$ , resultados melhores são obtidos com um valor menor de  $p$ , mas a partir de um certo ponto valores maiores de  $p$  apresentam resultados superiores.

Também foram testadas fatias da coluna vertebral, em imagens de tomografia computadorizada, com 40 fatias de  $512 \times 512$  *pixels*. Como a estrutura apresenta partes finas, que são mais facilmente perdidas pelo RW e pelo RFC+GC, o método IRFC foi o método que realizou as melhores segmentações, como pode ser visto na figura 6.7.

Em imagens de tomografia computadorizada do fígado, com 40 fatias de  $512 \times 512$  *pixels*, os resultados dos diferentes métodos são similares para sementes equidistantes da borda, como pode ser visto na figura 6.8a. Porém, com padrões assimétricos de sementes, o método de Passeios Aleatórios demonstra maior sensibilidade à posição das sementes, tendo resultados inferiores, como pode ser visto na figura 6.8b.

Por último, foram realizadas comparações utilizando a base heterogênea de 50 imagens coloridas do grabcut [RKB04]. Os pesos das arestas  $w(x, y)$  foram calculados como o complemento da diferença máxima de intensidade dos canais de cores, i.e.,  $K - \max\{|f_R(x) - f_R(y)|, |f_G(x) - f_G(y)|, |f_B(x) - f_B(y)|\}$ , onde  $f_R$  é o canal do vermelho,  $f_G$  o canal do verde, e  $f_B$  o canal do azul. Como esta base de dados é heterogênea, cada imagem poderia requerer um valor diferente para o parâmetro de potência, logo o método de Passeios Aleatórios foi o que obteve os piores resultados

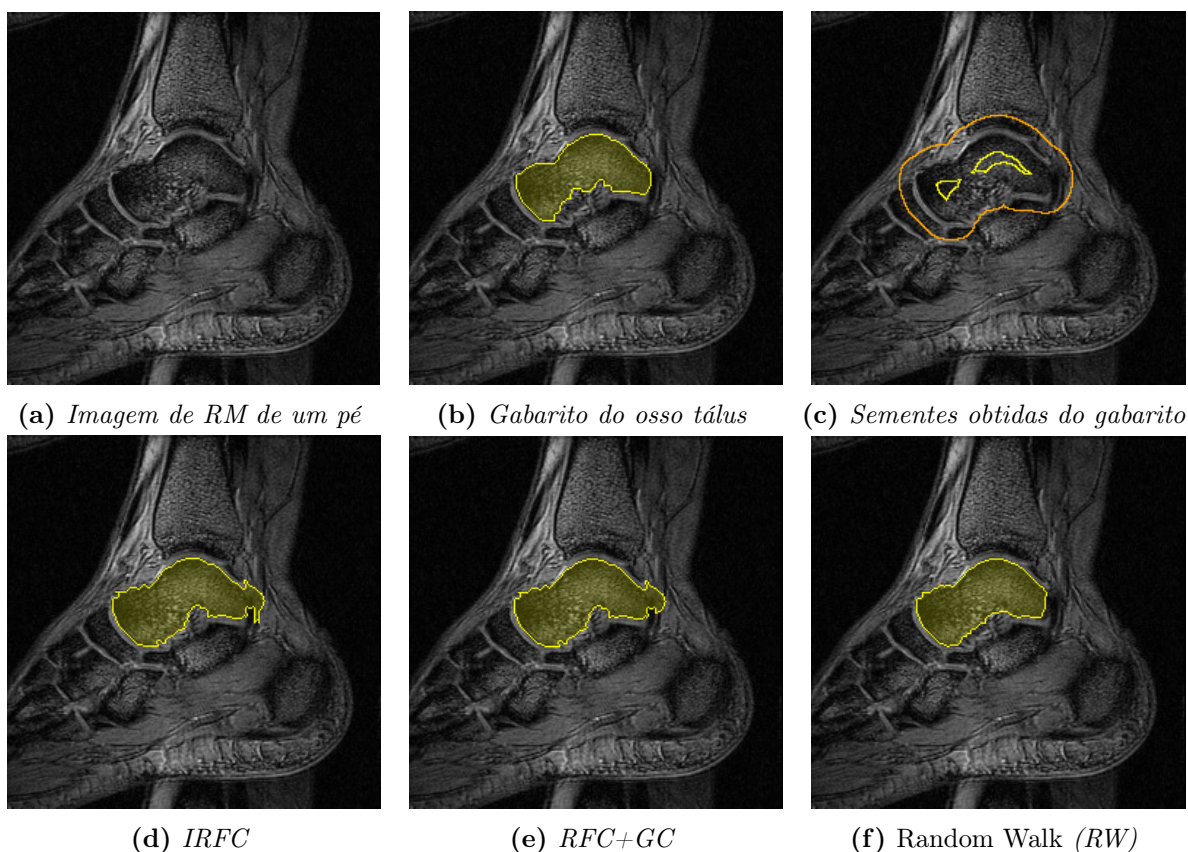
<sup>1</sup>[http://en.wikipedia.org/wiki/Dice%27s\\_coefficient](http://en.wikipedia.org/wiki/Dice%27s_coefficient). Acesso em: 17 set. 2012.



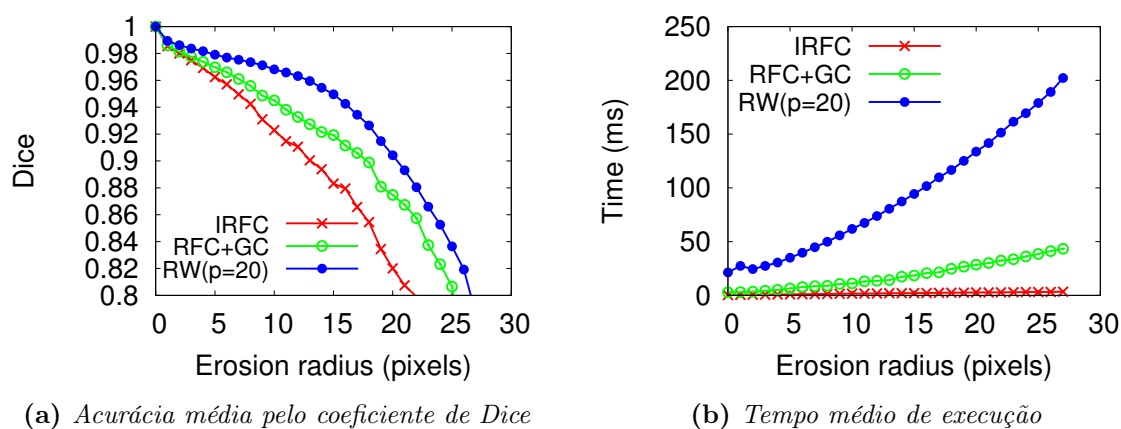
**Figura 6.1:** (a) Fatia de uma imagem 3D de Ressonância Magnética de um pé de nossos dados experimentais. (b) Gabarito de segmentação do osso calcâneo. (c) Exemplo de um conjunto de sementes obtido pela erosão e dilatação do gabarito de segmentação. Resultados de segmentação do calcâneo: (d) IRFC, (e) RFC+GC, e (f) Random Walk (RW).

nestas imagens. O melhor método para esta base foi o RFC+GC, como pode ser visto na figura 6.9.

Podemos ver também que o tempo de execução do método de Passeios Aleatórios está bem acima do tempo dos outros métodos. Porém, isto poderia ser melhorado através da utilização de diferentes técnicas, como o artigo original do método indica.

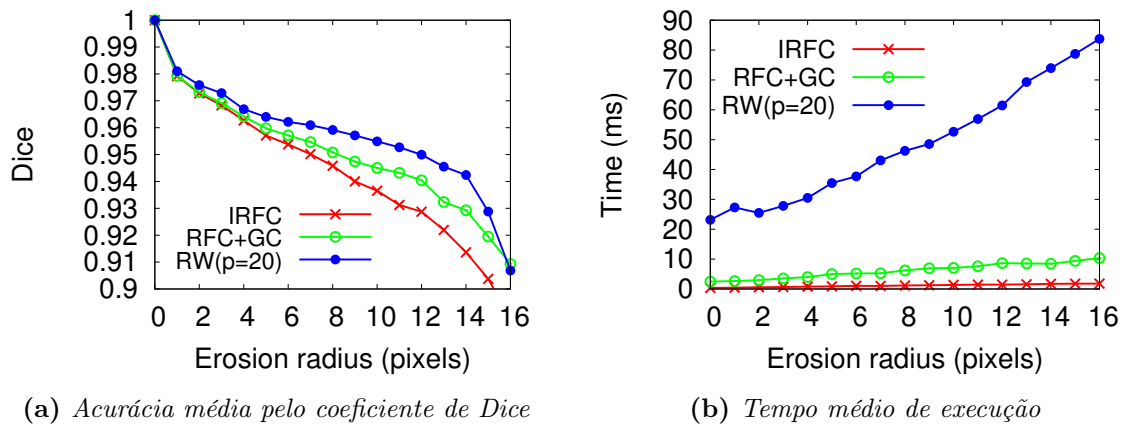


**Figura 6.2:** (a) Fatia de uma imagem 3D de Ressonância Magnética de um pé de nossos dados experimentais. (b) Gabarito de segmentação do osso tálus. (c) Exemplo de um conjunto de sementes obtido pela erosão e dilatação do gabarito de segmentação. Resultados de segmentação do tálus: (d) IRFC, (e) RFC+GC, e (f) Random Walk (RW).

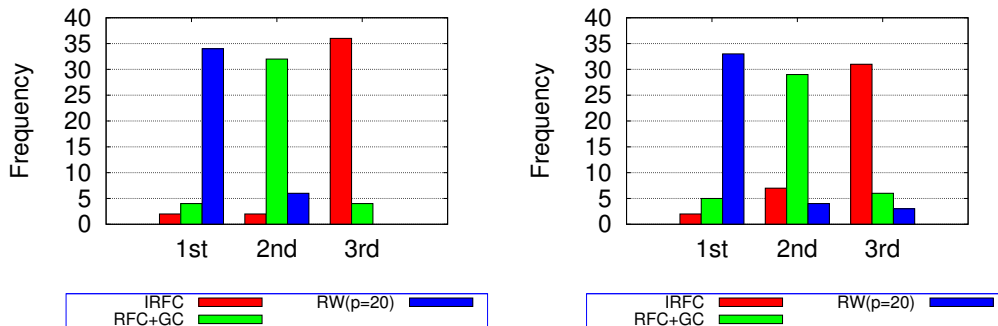


**Figura 6.3:** (a) Curva de acurácia média pelo coeficiente de Dice, e (b) tempo médio de execução para a segmentação 2D de fatias do osso calcâneo.

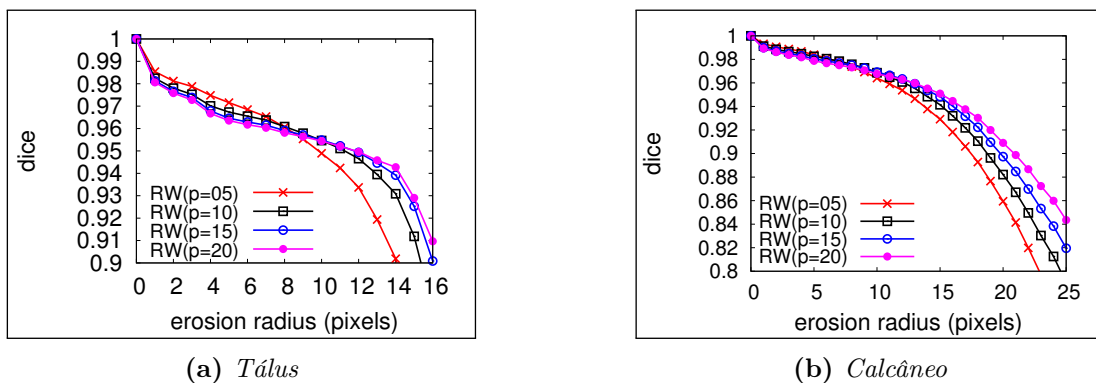




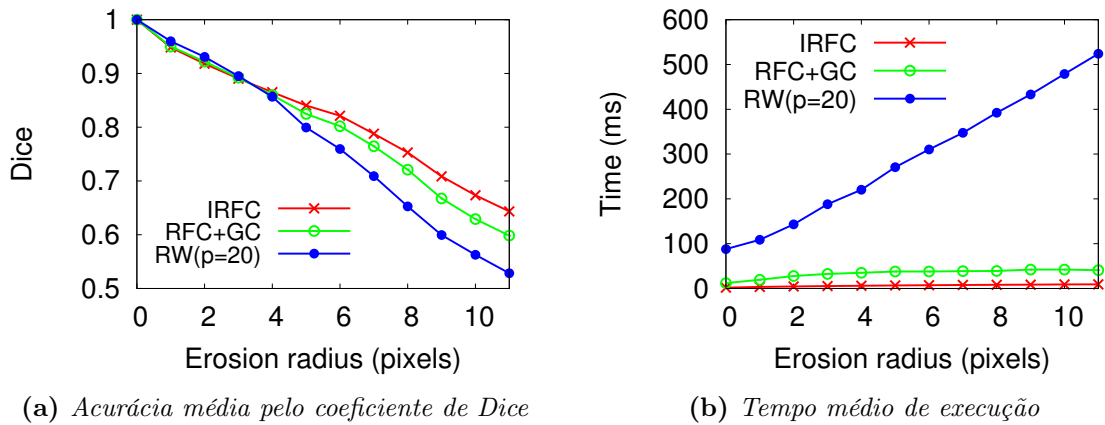
**Figura 6.4:** (a) Curva de acurácia média pelo coeficiente de Dice, e (b) tempo médio de execução para a segmentação 2D de fatias do osso tálus.



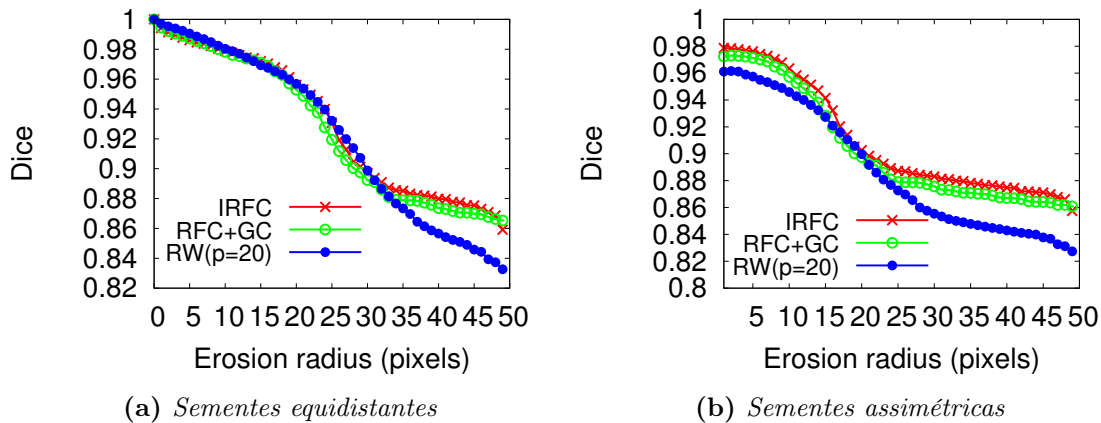
**Figura 6.5:** Para cada imagem individual, os métodos podem ser classificados de acordo com seus valores médios do coeficiente de Dice, como primeiro (melhor), segundo, ou terceiro (pior). Calculando a frequência para cada posição de classificação, temos a distribuição de classificação: (a) para a segmentação do calcâneo, e (b) para a segmentação do tálus.



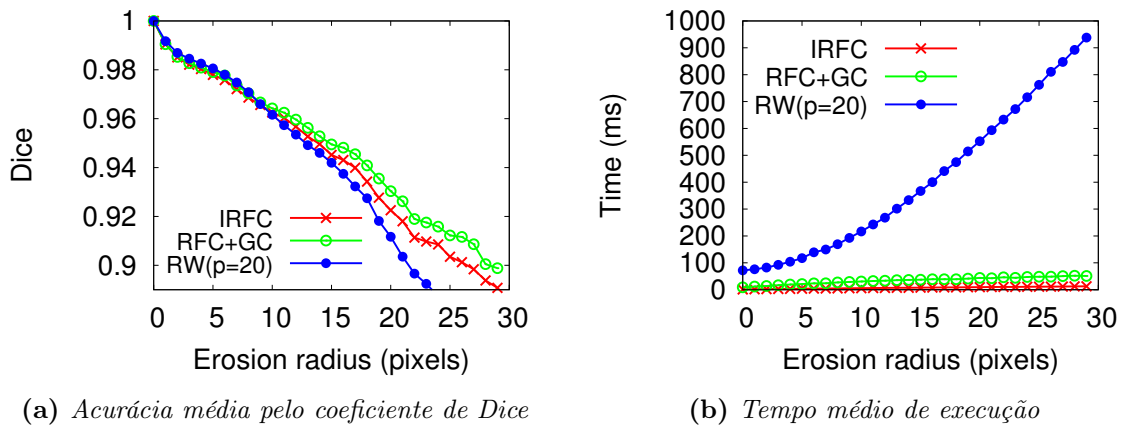
**Figura 6.6:** Comparando diferentes valores de  $p$ , a potência utilizada nos pesos das arestas, e sua influência na acurácia do método de Passeios Aleatórios (RW), segundo o coeficiente de Dice. Valores maiores de  $p$  geram resultados melhores para um maior raio de erosão, que é quando há uma maior distância entre as sementes. Porém, os resultados para raios menores de erosão pioram com valores altos desta potência.



**Figura 6.7:** Segmentação 2D de fatias da coluna vertebral, em imagens de tomografia computadorizada. Além de ter um desempenho inferior neste conjunto de imagens, o algoritmo de Passeios Aleatórios (RW) possui tempo de execução bem acima dos outros algoritmos.



**Figura 6.8:** Curva de acurácia média, pelo coeficiente de Dice, para a segmentação do fígado usando: (a) sementes equidistantes da borda, e (b) padrão assimétrico de sementes (raio de erosão das sementes de fundo sempre maior por 10 pixels). Observe a sensibilidade do método de Passeios Aleatórios à posição das sementes.



**Figura 6.9:** (a) Curva de acurácia média pelo coeficiente de Dice, e (b) tempo médio de execução para a segmentação 2D de imagens coloridas da base do grabcut.



## Capítulo 7

# Conclusões

Como pôde ser visto no capítulo 6, o método de Passeios Aleatórios apresenta resultados superiores em alguns conjuntos de imagens. Além disso, não existe um método claramente superior aos demais, e a escolha de qual será utilizado deve se basear no tipo de imagens a serem segmentadas.

Entretanto, o método de Passeios Aleatórios apresenta uma grande sensibilidade a certos parâmetros. Nos testes os pesos das arestas foram elevados a uma potência  $p$ , e como pôde ser visto pelos resultados, a qualidade das segmentações varia bastante com este valor. Além disso, há uma sensibilidade à localização das sementes, pois o método também utiliza esta localização para realizar a segmentação. Outros métodos possuem certa robustez em relação às sementes, e a mudança do posicionamento delas dentro de certas regiões não altera o resultado da segmentação, mas isto não ocorre com o método de Passeios Aleatórios.

Algo que pôde ser observado é que valores menores da potência  $p$  geram segmentações com bordas mais suaves, pois é dada maior importância à posição das sementes, e isto pode ser interessante em alguns conjuntos de imagens. O valor  $p = 20$  que foi utilizado ainda gera bordas relativamente suaves, porém dá uma importância maior ao conteúdo da imagem, realizando boas segmentações dos ossos do pé.

Um outro problema é o tempo de execução do algoritmo, que é bem elevado. A implementação conseguiu utilizar uma quantidade de memória que permitisse a aplicação do método mesmo em volumes 3D. Entretanto, a lentidão faz com que não seja possível sua utilização de modo interativo no caso 3D, demorando alguns minutos para realizar cada segmentação. Ainda assim, o artigo original aponta algumas técnicas que poderiam ser utilizadas para aumentar a velocidade de execução do método, e com isto poderia ser possível ter uma implementação eficiente.

O método de Passeios Aleatórios apresenta resultados interessantes, e tendo sido o melhor método testado para certas imagens, é uma alternativa a ser considerada ao escolher um método de segmentação.

Parte II  
Subjetiva

## Capítulo 8

# Desafios e frustrações encontrados

A primeira dificuldade foi simplesmente entender o conteúdo do artigo que trata do método de Passeios Aleatórios (*Random Walks*). É um conteúdo extremamente denso, e que faz referências a muitas coisas diferentes que eu nunca havia estudado.

Acabei me concentrando em entender apenas a teoria por trás do funcionamento do método, e não tentei entender as provas que foram feitas de suas propriedades. Isto já foi o suficiente para entender o método, implementá-lo, e escrever muitas páginas de teoria na monografia, que acabou ficando maior do que eu esperava.

Quando eu estava terminando de escrever a monografia, tentei colocar uma “prova” intuitiva dos motivos pelos quais o método não iria gerar segmentações com regiões isoladas, sem sementes. O artigo original faz uma prova desta propriedade, mas eu não consegui entendê-la e, mesmo que conseguisse, eu precisaria adicionar mais algumas páginas de teoria na monografia só para conseguir fazer esta prova.

Pensando em idéias mais intuitivas, escrevi um argumento de que estas regiões sem sementes não poderiam ocorrer. Porém, lendo o meu próprio argumento, eu mesmo não fiquei convencido de sua validade. Tentei argumentar de outra forma, mas ainda não estava convencido. Foi quando parei para pensar se esta propriedade realmente era válida. Pensando nos motivos que me levavam a duvidar de minhas argumentações, acabei encontrando um contra-exemplo que mostrava claramente que o método não possuía aquela propriedade, e que a prova exibida no artigo estava incorreta. Pesquisando um pouco, encontrei um outro artigo [CZ11], publicado ano passado (2011), justamente sobre isso. Neste artigo, um outro contra-exemplo desta propriedade é apresentado, e é feito um estudo de algumas características do método e de como construir contra-exemplos.

Isto foi um tanto frustrante, principalmente ao perceber que se passaram mais de 4 anos até que alguém percebesse que aquela prova estava errada. Por outro lado, me senti melhor ao perceber que não sou o único que não estava entendendo aquelas provas. E ao mesmo tempo, fiquei decepcionado ao ver que um artigo é publicado em um journal importante, e referenciado inúmeras vezes durante anos, sem que alguém realmente entenda todo o seu conteúdo e verifique que está correto.

Mesmo com estas dificuldades, e em parte devido a elas, aprendi bastante coisa e, a partir de agora, pensarei bem antes de simplesmente “acreditar” no conteúdo de algum artigo, mesmo que seja algo aparentemente intuitivo. Infelizmente, nem sempre há tempo de verificar todos os resultados.

## Capítulo 9

# Disciplinas relevantes para o trabalho

Diversas disciplinas foram importantes de uma forma mais indireta, proporcionando um maior domínio sobre as áreas de programação e provas matemáticas. Porém, irei listar as que influenciaram mais diretamente este trabalho:

1. **MAC0122 Princípios de Desenvolvimento de Algoritmos e MAC0323 Estruturas de Dados:** Ambas foram importantes para que eu pudesse fazer a implementação do método em C sem maiores problemas, conseguindo integrá-lo a dois projetos já existentes.
2. **MAC0328 Algoritmos em grafos:** Proporcionou uma introdução à área de grafos, permitindo que eu entendesse o algoritmo deste trabalho e as ideias por trás dele.
3. **MAT0139 Álgebra Linear para Computação e MAC0300 Métodos Numéricos da Álgebra Linear:** Aprendi toda a base de álgebra linear, propriedades de matrizes e métodos numéricos de solucionar sistemas lineares, o que permitiu que eu entendesse facilmente parte da teoria utilizada no algoritmo e as vantagens e desvantagens das possíveis formas de implementá-lo.
4. **MAC0417 Visão e Processamento de Imagens:** Tive um contato com a área de visão computacional e seus diversos problemas, dentre eles o de segmentar objetos. Além disso, aprendi sobre vários algoritmos que são utilizados na área, dando uma visão mais ampla e permitindo que eu enxergasse a importância de um bom algoritmo de segmentação.

## Capítulo 10

# Trabalhos futuros

Como ficou claro nos testes, seria interessante tentar realizar uma implementação do método que fosse mais rápida. Isto demandaria bastante tempo e esforço, pois o gasto de memória no caso de volumes 3D não poderia aumentar muito em relação à implementação atual. Além disso, embora algumas técnicas sejam mais diretas, como a implementação da resolução do sistema linear em uma GPU, outras demandam um grande conhecimento de teoria.

Outra possibilidade seria explorar passeios aleatórios em que as arestas são direcionadas, ou seja, com pesos possivelmente diferentes para cada direção. Isto poderia trazer resultados interessantes, que poderiam ser comparados com os de outros métodos que também utilizam arestas direcionadas. Porém, seria ainda mais difícil conseguir uma implementação eficiente, e talvez o método se torne completamente inviável no caso 3D com esta modificação.

# Referências Bibliográficas

- [BM08] A. Bondy e U.S.R. Murty. *Graph Theory*. Graduate Texts in Mathematics. Springer, 2008. 4
- [CGNT09] C. Couprie, L. Grady, L. Najman e H. Talbot. Power watersheds: A new image segmentation framework extending graph cuts, random walker and optimal spanning forest. Em *Computer Vision, 2009 IEEE 12th International Conference on*, páginas 731–738. IEEE, 2009. ii, 2
- [CGNT11] C. Couprie, L. Grady, L. Najman e H. Talbot. Power Watershed: A Unifying Graph-Based Optimization Framework. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(7):1384–1399, Julho 2011. ii, 2
- [CMUF12] K.C. Ciesielski, P.A.V. Miranda, J.K. Udupa e A.X. Falcão. Image segmentation by combining the strengths of relative fuzzy connectedness and graph cut. Em *Proc. of the International Conference on Image Processing (ICIP)*, Orlando, Florida, USA, 2012. IEEE. to appear. 24
- [CUSZ07] K.C. Ciesielski, J.K. Udupa, P.K. Saha e Y. Zhuge. Iterative relative fuzzy connectedness for multiple objects with multiple seeds. *Computer Vision and Image Understanding*, 107(3):160–182, 2007. 24
- [CZ11] Ming-Ming Cheng e Guo-Xin Zhang. Connectedness of random walk segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33:200–202, 2011. 17, 31
- [DS84] P.G. Doyle e J.L. Snell. *Random walks and electric networks*. Carus mathematical monographs. Mathematical Association of America, 1984. 6
- [Gra06] Leo Grady. Random walks for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(11):1768–1783, Novembro 2006. ii, 2
- [GS97] C.M. Grinstead e J.L. Snell. *Introduction to Probability*. 2nd ed. American Mathematical Society, 1997. 5, 12
- [LG12] O. Lezoray e L. Grady. *Image Processing and Analysis With Graphs: Theory and Practice*. Digital Imaging and Computer Vision Series. Taylor & Francis, 2012. ii
- [RKB04] C. Rother, V. Kolmogorov e A. Blake. “grabcut”: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, 2004. 24
- [Ros06] S.M. Ross. *Introduction to Probability Models*. Elsevier Science, 2006. 5
- [SG07] Ali K. Sinop e Leo Grady. A Seeded Image Segmentation Framework Unifying Graph Cuts And Random Walker Which Yields A New Algorithm. Em *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, páginas 1–8. IEEE, 2007. ii, 2

- [SHC06] J. Smart, K. Hock e S. Csomor. *Cross-Platform Gui Programming With wxWidgets*. Bruce Perens' Open Source Series. Prentice Hall Ptr, 2006. [ii](#)
- [YCZL10] Wenxian Yang, Jianfei Cai, Jianmin Zheng e Jiebo Luo. User-Friendly Interactive Image Segmentation Through Unified Combinatorial User Inputs. *IEEE Transactions on Image Processing*, 19(9):2470–2479, Setembro 2010. [ii](#)