



UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Campus de São José do Rio Preto

SEGMENTAÇÃO INTERATIVA DE IMAGENS USANDO REDES COMPLEXAS E COMPETIÇÃO E COOPERAÇÃO ENTRE PARTÍCULAS

Jefferson Antonio Ribeiro Passerini

São José do Rio Preto

2019

Jefferson Antonio Ribeiro Passerini

**SEGMENTAÇÃO INTERATIVA DE IMAGENS USANDO
REDES COMPLEXAS E COMPETIÇÃO E
COOPERAÇÃO ENTRE PARTÍCULAS**

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Câmpus de São José do Rio Preto.

Universidade Estadual Paulista – UNESP

"Julio de Mesquita Filho"

Programa de Pós-Graduação em Ciência da Computação

Orientador: Prof. Dr. Fabricio Aparecido Breve

São José do Rio Preto

2019

Passerini, Jefferson Antonio Ribeiro.

Segmentação interativa de imagens usando redes complexas e competição e cooperação entre partículas / Jefferson Antonio Ribeiro
Passerini. -- São José do Rio Preto, 2019

126 f. : il., tabs.

Orientador: Fabrício Aparecido Breve

Dissertação (mestrado) – Universidade Estadual Paulista (Unesp),
Instituto de Biociências, Letras e Ciências Exatas, São José do Rio Preto

1. Computação - Matemática. 2. Inteligência artificial.
3. Processamento de imagens – Técnicas digitais. 4. Algoritmos de
computador. 5. Aprendizado do computador. 6. Correlações de partículas.
7. Teoria dos grafos. I. Título.

CDU – 518.72

Ficha catalográfica elaborada pela Biblioteca do IBILCE
UNESP - Câmpus de São José do Rio Preto

Jefferson Antonio Ribeiro Passerini

SEGMENTAÇÃO INTERATIVA DE IMAGENS USANDO REDES COMPLEXAS E COMPETIÇÃO E COOPERAÇÃO ENTRE PARTÍCULAS

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Câmpus de São José do Rio Preto.

Comissão Examinadora:

Prof. Dr. Fabricio Aparecido Breve
UNESP - Câmpus de Rio Claro
Orientador

Prof. Dr. Eraldo Pereira Marinho
UNESP - Câmpus de Rio Claro

Prof. Dr. Fernando Vernal Salina
Instituto Federal de São Paulo - Câmpus São Carlos

Rio Claro

30 de janeiro de 2019

À minha esposa
À minha família
À ciência

Agradecimentos

Agradeço a Deus pela sabedoria e paciência concedidas em momentos críticos dessa caminhada.

Agradeço à minha esposa pelo companheirismo e por me apoiar em todos os momentos, além de abdicar da minha atenção para que pudesse dedicar ao trabalho.

Agradeço aos meus pais, sogro e sogra, e toda minha família por terem me apoiado e ensinado valores que permitiram tornar essa conquista uma realidade.

Agradeço ao meu orientador, Prof. Dr. Fabrício Aparecido Breve, por todos os conselhos e ensinamentos, pela paciência e ajuda nesse período.

Aos amigos que compartilharam dessa caminhada durante esses anos de batalha diária.

*“A tarefa não é tanto ver aquilo que ninguém viu,
mas pensar o que ninguém pensou sobre aquilo que todo mundo vê.”
(Arthur Schopenhauer)*

Resumo

A segmentação de imagens é o processo de identificar e separar estruturas e objetos relevantes em uma imagem, não é uma tarefa trivial para um algoritmo computacional devido à complexidade dos elementos envolvidos no processo. A variedade de imagens e sua grande quantidade de características impossibilitam a utilização de apenas um modelo para a tarefa de segmentação. Deste modo, a aplicação de algoritmos de aprendizado de máquina, ganha importância, pois este ramo da inteligência artificial tenta reproduzir a forma de “aprender” humano. A aplicação do modelo de competição e cooperação entre partículas na segmentação de imagens mostra-se uma abordagem interessante pois possui baixo custo computacional, além de reduzir a necessidade de especialistas para a geração de dados rotulados. Este trabalho propõe melhorias na construção da rede complexa que o modelo de competição e cooperação entre partículas utiliza-se, através da alteração das características extraídas dos píxeis, mudança na forma de ligação dos vértices do modelo utilizando a similaridade das características e a vizinhança espacial dos píxeis da imagem e a influência gerada por uma partícula sob outros vértices vizinhos dentro da rede complexa. A metodologia proposta resultou em uma melhoria do nível de automação do processo, eliminando a necessidade de incrementar ou decrementar a importância de uma característica sob as outras. Comparativamente, durante os testes realizados com 151 imagens e 4530 amostras de execução para cada imagem em cada modelo, a abordagem proposta obteve uma taxa de erro de 0,49% na classificação dos píxeis contra 3,14% para o método testemunha, sem a utilização de ajustes, além de mostrar-se mais estável em relação a este desempenho perante as diferentes imagens testadas, e ainda obteve melhora no tempo final de processamento do modelo.

Palavras-chave: Segmentação de imagens. Aprendizado de máquina. Aprendizado semi-supervisionado. Competição e cooperação entre partículas.

Abstract

Image segmentation is the process of identifying and separating relevant structures and objects in an image, it is not a trivial task for a computational algorithm due to the complexity of the elements involved in the process. The variety of images and its vast number of characteristics make it impossible to use only a model for the segmentation task. In this way, the application of machine learning algorithms is highlighted, because this branch of artificial intelligence tries to reproduce the way of human learning. The application of the competition and cooperation model among particles in the image segmentations is considered an interesting approach since it has low computational cost, besides reducing the need of specialists for generating labeled data. This work proposes improvements in the making of the complex network that the competition and cooperation model among particles uses, through the alteration of the characteristics extracted from pixels, changes in the shape of the model's connection of the vertices using the similarity of the characteristics and the image pixel's spatial neighboring and the influence generated by a particle under the other neighbor vertices within the complex network. The proposed methodology resulted in an improvement of the process's automation level, eliminating the need to increase or decrease the importance of a characteristic under others. Comparatively, during the tests made with 151 images and 4530 execution samples for each image in each model, the proposed approach obtained an error rate of 0,49% in the pixel classification versus 3,14% for the control method, without the use of adjustments, in addition to proving itself more stable compared to this performance regarding the different images that were tested and still it obtained some improvement in the model's final time of processing.

Keywords: Image segmentation. Machine learning. Semi-supervised learning. Particle competition and cooperation.

Lista de ilustrações

Figura 2.1 – Imagem Monocromática e a convenção de eixos x e y	17
Figura 2.2 – Modelo RGB : (a) Representação do modelo de cores RGB (b) Cubo de cores RGB	19
Figura 2.3 – Modelo de Cores HSV (Hexágono)	20
Figura 2.4 – Componentes de Cores ExR, ExG e ExB	21
Figura 2.5 – Resultado da Interpolação pelo vizinho mais próximo - Ampliação da região do rosto.	24
Figura 2.6 – Representação de como ocorre a média ponderada na imagem utilizando a interpolação bilinear. A intensidade do pixel interpolado possui a interferência dos valores de seus vizinhos, causando a suavidade nas bordas.	25
Figura 2.7 – Resultado da Interpolação Bilinear - Ampliação da região do rosto. . .	25
Figura 2.8 – Esquema da interpolação bicúbica.	26
Figura 2.9 – Ampliação da região do rosto pela interpolação bicúbica.	27
Figura 2.10 – Classificação dos Métodos de Segmentação.	29
Figura 2.11 – Processo de Convolução sobre uma imagem.	30
Figura 2.12 – Máscara de Convolução para detecção de Pontos Isolados	30
Figura 2.13 – Máscara de Convolução para detecção de Linhas	31
Figura 2.14 – Estilos de Borda: a) Borda em degrau; b) Borda em Rampa; c) Borda em forma de telhado	32
Figura 2.15 – Detecção de Bordas: a) Apresenta a imagem com cinco tipos de feijões; b) Resultado da segmentação utilizando uma escala pequena c) Resultado de segmentação utilizando uma escala maior	33
Figura 2.16 – Limiarização: a) Imagem Original; b) Histograma (os picos elevados foram cortados para realçar os detalhes nos valores mais baixos c) Resultado da segmentação utilizando o algoritmo global básico d) Resultado obtido pelo método Otsu	35
Figura 2.17 – Representação Gráfica do Método de Divisão e Conquista	36
Figura 2.18 – Aplicação de técnica de Bacias Hidrográficas: a) Imagem Original; b) Imagem em escala de cinza c) Imagem após a aplicação do detector de bordas d) Segmentação resultante do algoritmo de bacias hidrográficas	37
Figura 2.19 – Aplicação de técnica de Bacias Hidrográficas proposta (NAVON; MILLER; AVERBUCH, 2005) : a) Imagem Original; b) Resultado final da imagem após a segmentação	38
Figura 2.20 – Segmentação Imagem utilizando método proposto por Boykov e Jolly (2001)	39

Figura 2.21–Exemplo de aplicação do <i>GrabCut</i> . O usuário desenha um retângulo ao redor de um objeto, o qual é extraído da imagem inicial	40
Figura 2.22–Exemplo da Interação do Usuário com o algoritmo	41
Figura 3.1 – Matriz de Adjacências para grafo não ordenado	44
Figura 3.2 – Matriz de Incidência: (a) Grafo não Orientado (b) Grafo Orientado . .	44
Figura 4.1 – Classificação no conjunto de dados com dois objetos. (a) Dados de entrada; (b) resultado da classificação utilizando o Suport Vector Machines (SVM); (c) Classificação utilizando k-vizinhos; (d) classificação ideal.	51
Figura 5.1 – Ilustração da dinâmica de partículas e nós.	61
Figura 5.2 – Representação da geração da rede complexa pelo método de segmentação proposto. (a) Imagem original, (b) Imagem rotulada (c) Rede complexa com as partículas.	66
Figura 5.3 – Reserva Jaguari: a) Imagem Original b) Imagem com pixels rotulados c) Resultado da Segmentação	67
Figura 5.4 – Flores: a) Imagem Original b) Imagem com pixels rotulados c) Resultado da Segmentação	67
Figura 6.1 – Informações de Entrada: (a) Imagem Original, (b) Marcação de pixels rotulados, (c) Polígono de corte e (d) Sobreposição de informações para visualização	72
Figura 6.2 – Informações de Entrada: (a) Imagem com utilização de corte, (b) Imagem sem utilização de corte	73
Figura 6.3 – Processo de Limiarização - Otsu: a) Imagem Original b) Imagem em escala de cinza c) Imagem binária	75
Figura 6.4 – Comparação: a) Imagem limiarizada b) Gabarito Base <i>GrabCut</i>	76
Figura 6.5 – Dados de Entrada para cálculo da vizinhança	77
Figura 6.6 – Modelo para influência de Partícula sobre a vizinhança	78
Figura 6.7 – Metodologia Proposta para geração do Grafo a partir de uma imagem	80
Figura 7.1 – Imagem Base Grabcut : (a) Imagem Original (b) Gabarito (c) Detalhe de pixels cinza onde houve divergência de classificação	81
Figura 7.2 – Exemplos de Rótulos Gerados pelo Usuário: (a) Rótulos de Frente e Fundo (b) Marcação da área de corte na imagem (c) Montagem dos rótulos sobre a imagem original para visualização	83
Figura 7.3 – Imagens sem utilização de marcação de corte	84
Figura 7.4 – Resultado da segmentação das imagens sem utilização de marcação de corte	86
Figura 7.5 – Resultado da segmentação das cinco imagens com melhores taxas de erro do método proposto	88
Figura 7.6 – Imagem <i>person2</i> com melhor resultado no método testemunha	90

Figura 7.7 – Resultado da segmentação das cinco imagens com melhores taxas de erro do método testemunha	92
Figura 7.8 – Resultado da segmentação das cinco imagens com melhores taxas de erro do método testemunha	95
Figura 7.9 – Análise da relação % Taxa de Erro e Tempo de Processamento da base de imagens processadas - Método Proposto e Método Testemunha . . .	97

Lista de tabelas

Tabela 2.1 – Operadores de Gradiente 3×3	32
Tabela 3.1 – Exemplos de Função de Similaridade	46
Tabela 6.1 – Resultados Teste de Algoritmos de Interpolação	74
Tabela 7.1 – Características das redes geradas a partir de imagens sem o recurso de Corte	83
Tabela 7.2 – Taxa de erro dos modelos a partir de imagens sem o recurso de Corte .	84
Tabela 7.3 – Tempo de execução dos modelos a partir de imagens sem o recurso de Corte	85
Tabela 7.4 – Características das redes geradas a partir das imagens com cinco melhores taxas de erro no método proposto	87
Tabela 7.5 – Taxa de erro dos modelos a partir dos cinco melhores resultados do método proposto	87
Tabela 7.6 – Tempo de execução dos modelos a partir dos cinco melhores resultados do método proposto	89
Tabela 7.7 – Características das redes geradas a partir das imagens com cinco melhores taxas de erro no método testemunha	89
Tabela 7.8 – Taxa de erro dos modelos a partir dos cinco melhores resultados do método testemunha	90
Tabela 7.9 – Tempo de execução dos modelos a partir dos cinco melhores resultados do método testemunha	91
Tabela 7.10–Características das redes geradas a partir das imagens com cinco piores taxas de erro no método proposto	93
Tabela 7.11–Taxa de erro dos modelos a partir dos cinco piores resultados do método proposto	93
Tabela 7.12–Tempo de execução dos modelos a partir dos cinco piores resultados do método proposto	94
Tabela 7.13–Resultado da execução dos Métodos Proposto e Testemunha na Base Estudada	96
Tabela 7.14–Características das redes geradas a partir da imagens com pior tempo de processamento pelo método proposto	98
Tabela 7.15–Características das redes geradas a partir da imagens com pior tempo de processamento pelo método proposto	98

Sumário

1	INTRODUÇÃO	14
1.1	Objetivos	16
1.2	Estrutura do documento	16
2	PROCESSAMENTO DE IMAGENS	17
2.1	Imagem Digital	17
2.1.1	Sistema de cores	19
2.2	Transformações Geométricas	22
2.2.1	Métodos de Interpolação	22
2.3	Segmentação de Imagens	27
2.3.1	Técnicas Baseadas em Descontinuidade	29
2.3.2	Técnicas Baseadas em Similaridades	33
2.3.2.1	Técnicas de Limiarização	34
2.3.2.2	Crescimento de Regiões	35
2.3.2.3	Técnica de Divisão e Conquista	36
2.3.2.4	Técnica de Bacias Hidrográficas	36
2.3.3	Técnicas Baseadas em Grafos	39
3	REDES COMPLEXAS	42
3.1	Conceitos de Grafos	43
3.1.1	Medidas de Dissimilaridade e Similaridade	45
3.1.2	Qualidade de agrupamento do grafo	47
3.2	Deteccção de Comunidades em Redes	47
4	APRENDIZADO DE MÁQUINA	49
4.1	Aprendizado Semi-Supervisionado	50
4.1.1	Auto Treinamento	52
4.1.2	Modelos Generativos	52
4.1.3	Agrupar e Rotular	53
4.1.4	Co-Treinamento	53
4.1.5	Separação de Baixa Densidade	54
4.1.6	Métodos Baseados em Grafos	54
5	MODELO DE COMPETIÇÃO E COOPERAÇÃO DE PARTÍCULAS	57
5.1	Movimento guloso ou aleatório	61

5.2	Aprendizado Ativo aplicado modelo de competição e cooperação de partículas	64
5.3	Segmentação de Imagens através do modelo de Competição e Co- operação de Partículas	64
6	METODOLOGIA	71
6.1	Método Testemunha	71
6.2	Estruturação do Método Proposto e Justificativas	72
7	RESULTADOS	81
8	CONCLUSÃO	99
	REFERÊNCIAS	101
	APÊNDICE A – CARACTERÍSTICAS DAS REDES GERADAS A PARTIR DAS IMAGENS	109
	APÊNDICE B – RESULTADOS: TAXA DE ERRO E TEMPOS DE EXECUÇÃO	117

1 Introdução

A segmentação de imagens é uma das áreas de estudo do processamento de imagens digitais e exerce a função de separar estruturas relevantes em uma imagem, com o objetivo de facilitar a compreensão e a análise de um objeto de interesse sendo de ampla utilização pela sociedade (GONZALEZ; WOODS, 2010).

O processo de segmentar imagens possui diversas técnicas documentadas na literatura (ZAITOUN; AQEL, 2015), mas não existe um único método que possa ser considerado adequado a todas as imagens em suas mais diversas áreas de estudo e aplicação, não se obtendo o mesmo nível de acurácia em todas as situações.

Gonzalez e Woods (2010) afirmam que a segmentação de imagens não é uma tarefa trivial, mas sim, um dos processos mais difíceis dentro do processamento de imagens, pois este processo depende do domínio específico ao qual a imagem pertence e seu interesse de estudo, além da quantidade de características que porventura possam ser combinadas para realizar a extração da informação desejada.

Deste modo o estudo de modelos para a segmentação de imagens ainda é um campo de estudo que oferece desafios e este problema está relacionado ao campo de aprendizado de máquina, onde o interesse é construir algoritmos que possam “aprender”, visando melhorar sua adaptação e desempenho através da experiência (MITCHELL, 1997).

Dentre as principais categorias de aprendizado de máquina, pode-se citar o aprendizado supervisionado e o aprendizado não supervisionado. A abordagem de aprendizado não supervisionada é utilizada para diversas aplicações como tomada de decisão, mineração de dados, recuperação de documentos, reconhecimento de padrões e ainda na segmentação de imagens. O agrupamento de dados é uma das técnicas de aprendizado não supervisionado que é aplicado a segmentação de imagens, onde os píxeis de uma imagem são agrupados pela relação de similaridade entre eles (SCHAEFFER, 2007; DUDA; HART; STORK, 2000; JAIN; MURTY; FLYNN, 1999; KAUFMAN; ROUSSEEUW, 1990; JAIN; DUBES, 1988).

Os modelos de aprendizado supervisionado demonstram-se onerosos em relação ao trabalho da geração de amostras pré-rotuladas para que algoritmos desta técnica possam ser treinados e desta forma poderem classificar outras imagens. Enquanto no aprendizado não supervisionado devido as características destes modelos o conjunto de informações a serem analisadas podem possuir elementos tem características mescladas entre subgrupos que pode ocasionar imprecisões de classificação.

Assim Breve, Quiles e Zhao (2015b) apresentam a aplicação de técnicas de apren-

dizado semi-supervisionado na segmentação de imagens, através do modelo de competição e cooperação de partículas.

Os algoritmos de aprendizado semi-supervisionado utilizam-se de uma pequena quantidade de dados rotulados e uma grande quantidade de dados não rotulados combinando características das duas abordagens tradicionais (CHAPELLE; SCHÖLKOPF; ZIEN, 2006). No caso específico de segmentação de imagens esta abordagem reduz a necessidade do trabalho de especialistas em rotular grandes quantidade de píxeis em imagens.

O algoritmo de competição e cooperação de partículas utiliza-se de múltiplas partículas, organizadas em times, para representar as classes de dados, onde partículas do mesmo time cooperam entre si enquanto competem com partículas de outro time pelos dados não rotulados (BREVE et al., 2012).

A aplicação desta abordagem em segmentação de imagens possibilitou a segmentação de imagens a partir de uma pequena quantidade de dados rotulados e das características extraídas dos píxeis da imagem (BREVE, 2017; BREVE, 2015; BREVE; QUILES; ZHAO, 2015b; BREVE; QUILES; ZHAO, 2015a).

Nestes trabalhos, a execução do modelo para a segmentação de imagens envolve a extração de características dos píxeis das imagens e a construção de um grafo a partir destas informações.

O conjunto de características extraídas da imagem definem as relações entre os vértices do grafo construído. E para cada imagem é necessário a verificação da importância de cada uma das características dentro do conjunto para que o modelo obtenha uma boa acurácia na segmentação da imagem.

Para determinar a configuração de importância de cada característica dentro do conjunto é utilizado o vetor de pesos λ , através do qual o usuário do modelo pode incrementar ou diminuir a influência de uma característica para o processo de construção da rede complexa. Isto pode ocorrer na forma de ajustes manuais ou da aplicação de modelos automatizados como análise de histograma ou algoritmos genéticos.

Esse processo de definir os valores de incremento ou decremento do vetor de pesos λ é determinante para o resultado da segmentação utilizando o modelo de Competição e Cooperação entre partículas, e esta configuração não é uma tarefa simples de realizar e aumenta a dependência da interferência do especialista que utiliza este método para realizar a segmentação das imagens (BREVE, 2017; BREVE, 2015; BREVE; QUILES; ZHAO, 2015b; BREVE; QUILES; ZHAO, 2015a).

1.1 Objetivos

O objetivo deste trabalho é promover melhorias no modelo de aprendizado semi-supervisionado de Competição e Cooperação entre Partículas aplicado à segmentação de imagens, permitindo uma maior automação do processo e reduzindo a dependência da interferência de especialistas para ajuste da execução do modelo. Desta forma o trabalho propõe alterar a construção da rede complexa a partir da imagem, permitindo eliminar do modelo original a necessidade de ajustes dos pesos das características. Para isso, os objetivos específicos foram:

- Realizar mudanças no conjunto de características utilizados nos trabalhos publicados anteriormente;
- Utilizar métodos de interpolação para redução de tamanho das imagens possibilitando a segmentação de imagens maiores;
- Alterar a construção da rede complexa, modificando a forma como as arestas são construídas, representar as relações espaciais dos píxeis na imagem dentro do grafo, assim como, dos píxeis rotulados que geram as partículas no modelo;
- Eliminar a necessidade da utilização vetor de pesos λ para determinar a importância das características dentro do modelo.

1.2 Estrutura do documento

Neste trabalho é apresentada no capítulo 2 a área de conhecimento de processamento de imagens enfatizando a segmentação e seus principais algoritmos. No capítulo 3 é apresentada a base de conceitos referentes a redes complexas. O capítulo 4 explana sobre o aprendizado de máquina e suas subdivisões, enfatizando o estudo dos paradigmas de aprendizado semi-supervisionado.

No capítulo 5 é realizado o estudo referente ao modelo de Competição e Cooperação de Partículas e sua aplicação em segmentação de imagens demonstrando sua evolução até a presente data. No capítulo 6 é apresentada a abordagem proposta neste trabalho e no capítulo 7 são demonstrados os resultados obtidos com a execução do modelo proposto neste trabalho.

2 Processamento de Imagens

Este capítulo tem por objetivo o estudo do processamento de imagens, seus sistemas de cores, mudança de escala de imagens e processos de segmentação de imagens digitais.

2.1 Imagem Digital

Podemos trabalhar basicamente com dois tipos de imagens digitais, as imagens em tons de cinza e as imagens coloridas. Onde as imagens coloridas apresentam uma maior complexidade em sua representação computacional.

Gonzalez e Woods (2010) definem que uma imagem digital é uma função bidimensional $f(x, y)$, onde x e y representam as coordenadas em um plano. A amplitude da função f em uma determinada coordenada é chamada de intensidade ou nível de cinza naquele ponto, como pode-se observar na Figura 2.1.

Figura 2.1 – Imagem Monocromática e a convenção de eixos x e y



Fonte: Marques e Vieira (1999, pag. 19)

Os valores das intensidades possíveis em uma determinada coordenada são finitas e discretas e cada par de coordenadas que identificam um ponto na imagem é chamado de pixel. A representação da matriz formada pelas intensidades dos pixels no plano determina

uma imagem monocromática é dada por:

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \dots & f(0, N - 1) \\ f(1, 0) & f(1, 1) & \dots & f(1, N - 1) \\ \dots & \dots & \dots & \dots \\ f(M - 1, 0) & f(M - 1, 1) & \dots & f(M - 1, N - 1) \end{bmatrix} \quad (2.1)$$

A intensidade de energia capturada durante o processo de geração da imagem digital é coletada por sensores e é armazenada no pixel em uma determinada coordenada x e y da imagem. Este valor atribuído ao pixel provem da relação da iluminância (medida em lux ou lúmens/ m^2) e da refletância. A iluminância é a quantidade de iluminação que incide no objeto visualizado e a refletância é quantidade de iluminação refletida por esse objeto. Deste modo, a função $f(x, y)$ é determinada pela relação da iluminância e refletância, onde: $f(x, y) = i(x, y) * r(x, y)$ (MARQUES; VIEIRA, 1999).

Durante o processo de aquisição da imagem, de acordo com o sensor utilizado, é definida a sua resolução espacial e de intensidade. A resolução espacial é uma medida do menor detalhe perceptível em um pixel da imagem e é normalmente descrita por pontos por unidade de distância (dpi¹) (GONZALEZ; WOODS, 2010).

A resolução espacial determina o número de píxeis das imagens geradas pelos dispositivos de aquisição, por exemplo, em uma imagem com 1024x1024 píxeis, é possível determinar mais detalhes do que em uma de menor resolução espacial.

Em relação a resolução de intensidade, refere-se à menor variação do nível de intensidade na imagem, o número de intensidade mais comum é o de 8 bits, o que determina a faixa de 0 a 255 na escala de intensidade de cada pixel. Existem trabalhos específicos onde utiliza-se de imagens de 16 bits enquanto as imagens de 32 bits são raramente utilizadas (GONZALEZ; WOODS, 2010).

Nas imagens digitais coloridas, cada pixel pode representar uma cor entre os diversos tons e intensidades de cores diferentes perceptíveis ao olho humano.

Assim, as cores são determinadas pela reflectância de diferentes frequências da luz visível e, através das estruturas do olho humano, percebemos o vermelho (*red*), o verde (*green*) e o azul (*blue*), e suas combinações.

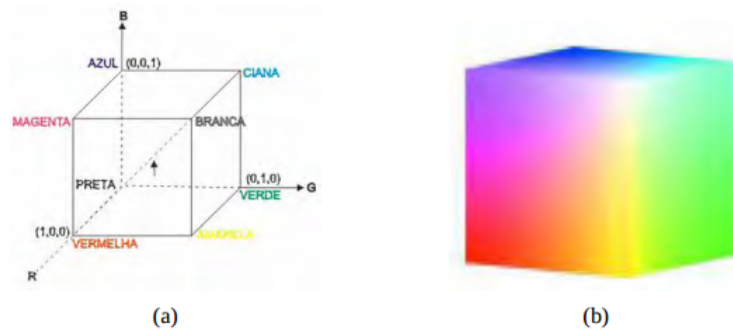
Com o objetivo de descrever as cores, padronizar e facilitar o trabalho foram descritos sistemas de cores. Dentre os modelos existentes, o RGB é, provavelmente, o mais utilizado e através deste é possível realizar a derivação para qualquer outro sistema de cores.

¹ Do Inglês: *Dot por Inch* - Pontos por polegada

2.1.1 Sistema de cores

No modelo RGB, seus componentes espectrais são representados por um sistema de coordenadas cartesianas 3D em formato de cubo, como pode-se observar na Figura 2.2.

Figura 2.2 – Modelo RGB : (a) Representação do modelo de cores RGB (b) Cubo de cores RGB



Fonte: Gonzalez e Woods (2010, pag. 265 e 266)

Na Figura 2.2(a) verifica-se que as cores primárias estão em três vértices e as secundárias (ciano, magenta e amarelo) estão em outros três vértices. A escala de cinza percorre o modelo do ponto preto para o branco, caracterizando onde os valores de R, G e B são iguais.

Assim, uma imagem colorida representada pelo modelo de cores RGB, constitui-se de três matrizes $M \times N$, onde M e N referem-se às linhas e às colunas da imagem. Cada uma das matrizes representa um dos canais do modelo, vermelho, verde e azul, onde a combinação destas coordenadas no cubo RGB nos proporciona uma definição de cores para a imagem.

O sistema de cores HSV descreve a formação de cores através de componentes que representam o seu nome: H, que representa a matiz² ou a cor; S a saturação³, sendo também conhecida como pureza, e V a intensidade ou o brilho⁴ daquela cor na imagem, conforme pode-se visualizar na Figura 2.3.

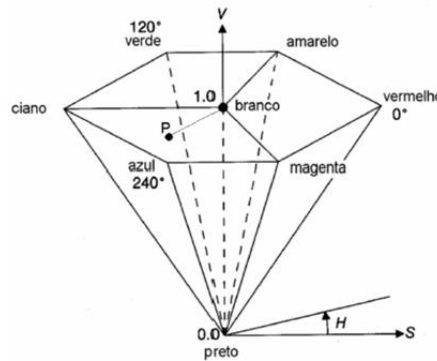
Este modelo de cores é representado por uma pirâmide de base hexagonal. A matiz está representada na base da pirâmide onde temos uma escala de 0° e 360° que, quando normalizada, está representada por valores entre 0 e 1. As cores nesta base são definidas a partir de ângulos específicos: em 0° temos a cor vermelha; de 120° , a cor verde; e 240° , a cor azul.

² Do Inglês - *Hue*

³ Do Inglês - *Saturation*

⁴ Do Inglês Value

Figura 2.3 – Modelo de Cores HSV (Hexágono)



Fonte: (BRYS, 2008)

A saturação é definida pelo elemento S, representada de 0% a 100% e diminui à medida que se aproxima do centro radial. O eixo central do hexágono define o elemento V de intensidade ou brilho, que também é representada por 0% até 100% e normalizada entre 0 e 1. O eixo central determina as cores acromáticas do espaço, sendo 0 a cor preta, 1 a cor branca e os valores entre estes correspondem às tonalidades de cinza.

A conversão do modelo de cores RGB para o modelo HSV é dada pelas Equações 2.2, 2.3, 2.4, 2.5, 2.6, onde seja I definido pelo espaço de cor RGB, e os componentes R, G e B estão entre 0 e 1.

$$mx_{(x,y)} = \max(I_{R(x,y)}, I_{G(x,y)}, I_{B(x,y)}); \quad (2.2)$$

$$mn_{(x,y)} = \min(I_{R(x,y)}, I_{G(x,y)}, I_{B(x,y)}); \quad (2.3)$$

$$H(i, j) = \begin{cases} \text{Se } I_{R(x,y)} > \text{maior}(I_{G(x,y)}, I_{B(x,y)}) & \text{calcular } \frac{60 \times (I_{G(x,y)} - I_{B(x,y)})}{mx - mn}; \\ \text{Se } I_{G(x,y)} > \text{maior}(I_{R(x,y)}, I_{B(x,y)}) & \text{calcular } \frac{60 \times (I_{B(x,y)} - I_{R(x,y)})}{mx - mn}; \\ \text{Se } I_{B(x,y)} > \text{maior}(I_{R(x,y)}, I_{G(x,y)}) & \text{calcular } \frac{60 \times (I_{R(x,y)} - I_{G(x,y)})}{mx - mn} \end{cases} \quad (2.4)$$

$$S(x, y) = \left(\frac{mx - mn}{mx} \right); \quad (2.5)$$

$$V(x, y) = (mx), \quad (2.6)$$

Onde: H refere-se ao canal matiz, S ao canal da saturação, V ao canal valor, mx ao maior nível de intensidade entre os canais R, G e B, e mn ao menor nível de intensidade entre os canais R, G e B.

A partir do sistema RGB podemos determinar outros componentes que possuem utilidades em aplicações específicas como manejo ambiental ou de colheitas.

Os componentes ExR, ExG e ExB, por exemplo, foram criados para a esta finalidade pois permitem a identificação de vegetação em imagens digitais. Podemos encontrar as definições destes componentes em Woebbecke et al. (1995), Meyer, Hindman e Laksmi (1999) e sua aplicação para segmentação de imagens na agricultura em Guijarro et al. (2011).

Os componentes ExR, ExG e ExB são calculados como descrito em Lichman (2013), conforme descreve as Equações 2.7, 2.8 e 2.9.

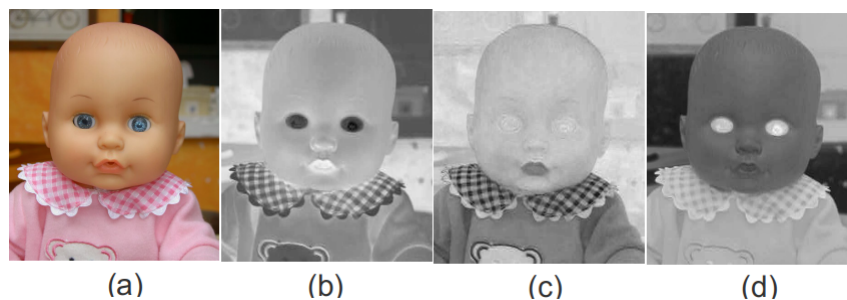
$$ExR = (2R - (G + B)) \quad (2.7)$$

$$ExG = (2G - (R + B)) \quad (2.8)$$

$$ExB = (2B - (G + R)) \quad (2.9)$$

Na Figura 2.4 pode-se observar as imagens derivadas destes componentes a partir de seus dados normalizados entre 0 e 255, onde (a) a figura original no sistema de cores RGB, (b) a imagem derivada em ExR, (c) a imagem derivada em ExG e (d) a imagem derivada em ExB.

Figura 2.4 – Componentes de Cores ExR, ExG e ExB



Fonte: Elaborado pelo autor

A partir do sistema RGB pode-se realizar a transformação para escala de cinza de uma imagem, que é determinada pela união das matrizes das três bandas de cores do sistema RGB, gerando uma nova banda monocromática.

Existem várias fórmulas para realizar essa transformação, assim será apresentada uma fórmula que possui maior aproximação de cor e luminância no resultado, de acordo

com a definição dada pela Comissão Internacional de Iluminação - CIE (*Commission Internationale de l'Eclairage*) (SCARANTI; BERNARDI; PLOTZE, 2010; CIE, 1931).

Essa transformação pode ser descrita pela Equação 2.10.

$$I_{cinza(x,y)} = 0,2989I_{R(x,y)} + 0,5870I_{G(x,y)} + 0,1140I_{B(x,y)} \quad (2.10)$$

Onde as variáveis i e j representam as coordenadas dos eixos cartesianos, o I_R , I_G e I_B indicam os canais de cores e I_{cinza} a imagem formada pelos níveis de cinza.

2.2 Transformações Geométricas

As transformações geométricas de imagens são operações de processamento que resultam no efeito da alteração espacial dos píxeis que a compõem, sendo utilizadas para redimensionamento, distorções ou ainda efeitos artísticos nas imagens. Pode-se citar como transformações o processo de ampliação e redução da imagem (*zoom*) ou ainda operações que resultam na alteração das dimensões da imagem (*scaling*, *resizing*).

Marques e Vieira (1999) demonstram que a maneira mais simples de ampliar uma imagem é duplicar o valor dos seus píxeis na direção de X e Y no plano cartesiano. Neste processo para expandir uma imagem com um fator de 2, cada pixel é copiado 4 vezes na imagem e, para reduzir, deve-se utilizar o processo inverso.

Afirmam ainda que o processo denominado *scaling* é caracterizado pela alteração das dimensões da imagem a partir de um fator. Esta alteração pode ser igual para os eixos X e Y, alterando o aspecto original da imagem ou não. No processo de *sizing*, o usuário deve informar uma nova dimensão para a imagem.

A seguir será explanado alguns modelos de algoritmos de redimensionamento de imagens.

2.2.1 Métodos de Interpolação

Muitas das técnicas de interpolação, como vizinho mais próximo, bicúbica e bilinear, estão disponíveis em muitas ferramentas de imagem como o *Photoshop*. As técnicas de interpolação são utilizadas para várias aplicações como redimensionamento de imagem, zoom de imagem, aprimoramento de imagem, redução de imagem, decomposição de imagem e distorções espaciais entre outras aplicações (KAUR; KAUR; KAUR, 2016; PATEL, 2013; PRAJAPATI; NAIK; MEHTA, 2012; GETREUER, 2011).

Uma imagem é um amostra de dados representada em uma matriz bidimensional de amostras espaçadas linearmente em um eixo X (horizontal) e um eixo Y (vertical) e,

se quisermos, por exemplo, redimensionar essa imagem para um tamanho maior, deve-se descobrir um novo pixel baseado em seus vizinhos mais próximos.

A interpolação é uma técnica para construir novos pontos a partir de pontos de dados conhecidos. Deste modo, esse novo pixel será caracterizado por assumir um valor entre os seus píxeis vizinhos, porém, o problema é encontrar valores adequados para as frações de valores de cada vizinho para esse novo elemento da imagem.

Prajapati, Naik e Mehta (2012) descrevem em seu trabalho a existência de duas categorias de técnicas para interpolação, a adaptativa e a não adaptativa.

O algoritmo adaptativo considera características da imagem como por exemplo, intensidade e textura e melhoram o resultado da interpolação mas possuem um custo computacional elevado.

Os algoritmos de interpolação não adaptativos não consideram qualquer recurso ou conteúdo da imagem sendo baseadas na manipulação direta dos píxeis, assim essa categoria caracteriza-se pela facilidade de execução e baixo custo computacional. Essa categoria de técnicas de interpolação é utilizada em produtos comerciais como o Adobe Photoshop.

As técnicas de interpolação não adaptativas por suas características serão abordadas neste trabalho e podemos citar métodos de interpolação dessa categoria os algoritmos: vizinho mais próximo, bilinear e bicúbica.

Kaur, Kaur e Kaur (2016) descrevem o modelo de interpolação por vizinho mais próximo⁵ como uma técnica de interpolação mais simples e que requer o menor tempo de processamento.

Getreuer (2011) descreve que este método de interpolação é chamado de duplicação de píxeis pois o ponto (x, y) desejado é definido pelo valor mais próximo, duplicando este pixel.

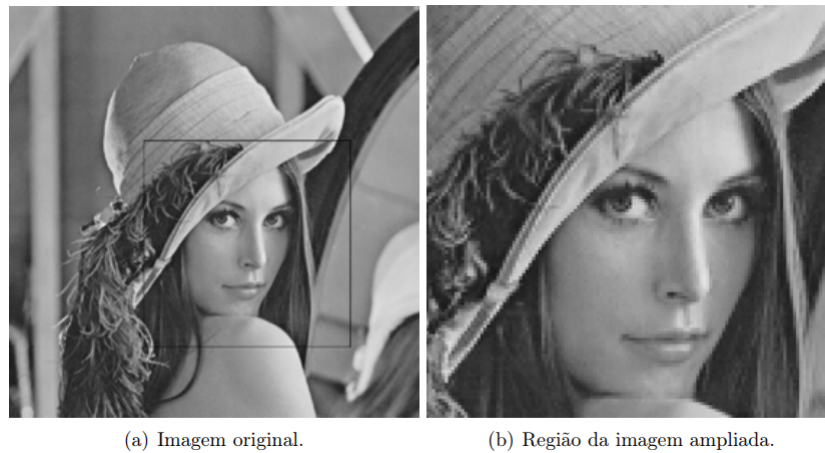
Patel (2013) afirma que é um método de simples implementação e que obtêm-se um bom resultado quando as imagens possuem uma boa resolução e que ainda assim algumas informações são perdidas durante o processo de interpolação.

Prajapati, Naik e Mehta (2012) descrevem que esse método não realiza uma interpolação real pois não calcula novos valores para as posições desejadas apenas realiza cópias dos valores mais próximos.

Dourado (2014) demonstra através da Figura 2.5, que a interpolação por vizinho mais próximo gera o efeito de serrilhamento na imagem resultante. Na borda do chapéu e na curva das bochechas do rosto pode-se observar no detalhe o efeito do serrilhamento resultante da interpolação por vizinho mais próximo.

⁵ Do Inglês: *Nearest Neighbor Interpolation*

Figura 2.5 – Resultado da Interpolação pelo vizinho mais próximo - Ampliação da região do rosto.



Fonte: Dourado (2014)

A interpolação bilinear ⁶ é usada para reduzir a distorção na imagem resultante do processo. A interpolação bilinear é feita em duas direções em linha e coluna. O valor do pixel é medido usando os quatro vizinhos mais próximos (KAUR; KAUR; KAUR, 2016).

O valor da posição desejada é medido usando a média ponderada dos quatro píxeis mais próximos, para cada célula $[x, x + 1] [y, y + 1]$. Os vizinhos próximos são determinados por $V_{x,y}$, $V_{x+1,y}$, $V_{x,y+1}$ e $V_{x+1,y+1}$, estando localizados nos cantos do pixel interpolado. A interpolação bilinear não gera píxeis com valores excedentes pois é limitada pelos seus vértices vizinhos, em seu valor máximo de V_{max} e mínimo de V_{min} (KAUR; KAUR; KAUR, 2016; PATEL, 2013; PRAJAPATI; NAIK; MEHTA, 2012; GETREUER, 2011).

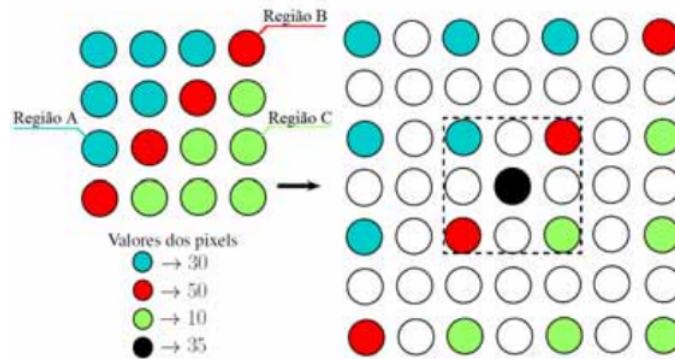
De acordo com Pedrini e Schwartz (2007), a interpolação bilinear é dada pela Equação 2.11.

$$\begin{aligned}
 f(x', y') &= (1 - dx) \cdot (1 - dy) \cdot f(x, y) + dx \cdot (1 - dy) \cdot f(x + 1, y) \\
 &\quad + (1 - dx) \cdot dy \cdot f(x, y + 1) + dx \cdot dy \cdot f(x + 1, y + 1)
 \end{aligned}
 \tag{2.11}$$

Dourado (2014) utilizando a Figura 2.6 demonstra como ocorre a média ponderada da interpolação bilinear. Na imagem o ponto preto é um novo pixel que foi interpolado. Este pixel está na borda das regiões A e C e pertencendo a região B que, de acordo com a legenda na imagem, recebe uma valor igual a 50, mas, por influência dos vizinhos da região A e da Região C que possuem valores de 30 e 10 respectivamente, o novo pixel terá valor 35, apesar de sua localização. Este processo de média ponderada proporciona transições mais suaves em locais de alta frequência, gerando uma suavização da imagem.

⁶ Do Inglês: *Bilinear Interpolation*

Figura 2.6 – Representação de como ocorre a média ponderada na imagem utilizando a interpolação bilinear. A intensidade do pixel interpolado possui a interferência dos valores de seus vizinhos, causando a suavidade nas bordas.



Fonte: Dourado (2014)

Devido a essa suavização, o efeito serrilhado observado na interpolação bilinear é diminuído neste modelo, como pode-se verificar na Figura 2.7.

Figura 2.7 – Resultado da Interpolação Bilinear - Ampliação da região do rosto.



(a) Imagem original

(b) Região da imagem ampliada

Fonte: Dourado (2014)

O algoritmo de interpolação bicúbica⁷ é o melhor entre todos os métodos não-adaptativos. Este modelo utiliza uma média ponderada dos 16 píxeis vizinhos para calcular seu valor interpolado final. Os píxeis vizinhos selecionados estão a diferentes distâncias do pixel a ser interpolado e os píxeis mais próximos recebem uma maior ponderação no cálculo resultando em imagens mais nítidas do que os métodos anteriores (KAUR; KAUR; KAUR, 2016; PATEL, 2013; PRAJAPATI; NAIK; MEHTA, 2012; GETREUER, 2011).

Segundo Dourado (2014), a cada um dos 16 vizinhos é atribuído um peso, e através da *spline* cúbica, é calculado o valor do novo pixel interpolado. A *B-spline* é um segmento

⁷ Do Inglês: *Bicubic Interpolation*

de curva definido matematicamente por dois ou mais pontos de controle e que possui o menor subconjunto fechado do domínio em relação a um determinado grau B. A função que descreve a regra geral da interpolação bicúbica é dada por:

$$f(x', y') = \sum_{m=-1}^2 \sum_{n=-1}^2 f(x+m, y+n) R(m-dx) R(dy-n) \quad (2.12)$$

em que,

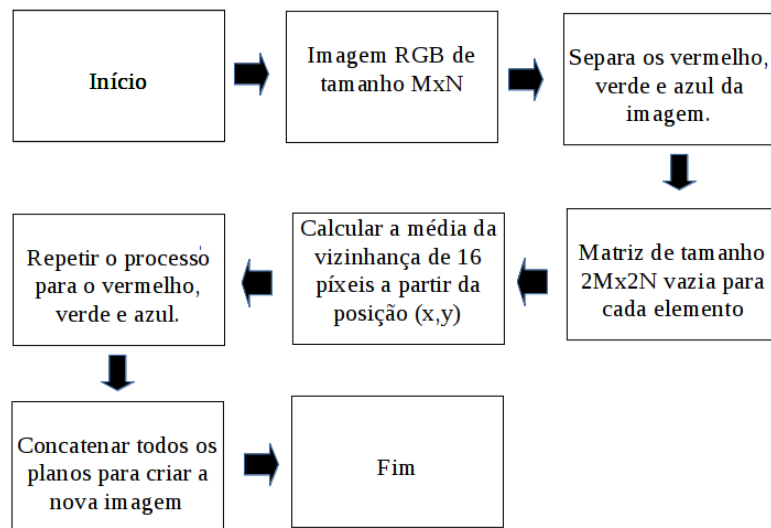
$$R(s) = \frac{1}{6} [P(s+2)^3 - 4P(s+1)^3 + 6P(s)^3 - 4P(s-1)^3] \quad (2.13)$$

$$P(t) = \begin{cases} t, & t > 0 \\ 0, & t \leq 0. \end{cases}$$

A técnica de interpolação bicúbica apresenta o melhor resultado dentre todas as técnicas não adaptativas. Em relação ao seu custo computacional deve ser observado que é a técnica mais custosa computacionalmente, assim quando o tempo não é uma restrição é a técnica recomendada pois possui o melhor resultado entre todas as técnicas não adaptativas (PATEL, 2013).

Kaur, Kaur e Kaur (2016) apresentam na Figura 2.8 um esquema gráfico para ilustrar o algoritmo do modelo de interpolação bicúbica.

Figura 2.8 – Esquema da interpolação bicúbica.



Fonte: Adaptado de Kaur, Kaur e Kaur (2016)

Dourado (2014) demonstra na Figura 2.9 que o modelo de interpolação bicúbica não causa um serrilhamento tão exagerado, como observa-se na interpolação por vizinho próximo, e também não suaviza tanto a imagem interpolada quanto o modelo de interpolação bilinear.

Figura 2.9 – Ampliação da região do rosto pela interpolação bicúbica.



Fonte: Dourado (2014)

Quando o tempo computacional não é uma restrição é recomendado utilizar essa técnica pois proporciona o melhor resultado entre as técnicas não-adaptativas, aos quais as técnicas aqui apresentadas se enquadram.

2.3 Segmentação de Imagens

A segmentação de imagens é uma das principais e mais estudadas operações de processamento de imagens. Define-se como o processo de identificar e separar estruturas e objetos relevantes em uma imagem, sendo uma operação necessária para análise e extração de informações.

Gonzalez e Woods (2010) afirmam que segmentação de imagens não é uma tarefa trivial, sendo uma das mais difíceis do processamento de imagens, não existindo um único modelo ou método que para a variedade de imagens existentes mantenha um determinado nível de acurácia.

Segmentar imagens envolve interpretar e extrair as informações relevantes de como os seres humanos são capazes de executá-la. Fatores de alto nível, como a contextualização da informação inserida em cada imagem ao qual o algoritmo será exposto, dificultam a criação de um modelo que atenda as mais diversas áreas de aplicação de imagens. A ação de interpretar é uma atividade comum para o cérebro humano, mas a sua representação computacional demonstra-se como um problema de resolução complexa.

Outros fatores classificados como de baixo nível, ou seja, informações existentes na imagem (brilho, textura, matiz etc), ruídos ou ainda a qualidade de resolução, também exercem influência no processo de segmentação.

Em uma imagem, uma característica como um canal de cores, pode ser relevante na segmentação daquela imagem específica, mas para outras amostras de imagens, este canal de cores talvez não seja fator relevante para a tomada de decisão do modelo computacional.

Assim para o processo de segmentação, a escolha de um mesmo padrão de características da imagem que sejam aplicáveis a todas as análises, não nos garante efetivamente um resultado adequado diante da diversidade de imagens que podemos encontrar. Faz-se necessário testar diversos algoritmos para verificar qual o de melhor resultado, assim como combinações de características neste processo.

Shi e Malik (2000) afirmam que a análise do detalhamento e a definição de uma divisão hierárquica de aspectos dentro da imagem dificultam a escolha de um algoritmo de segmentação que funcione para as mais diversas imagens e suas diversas áreas de aplicação.

Métodos de segmentação podem ser classificados como interativos ou automáticos, considerando se existe ou não a intervenção de um especialista humano durante o processo (GONZALEZ; WOODS, 2010).

Nos processos de segmentação interativa de imagens ocorre a junção da capacidade humana, auxiliando o processo automatizado de identificação e reconhecimento das partes a serem segmentadas.

A segmentação interativa de imagens, ou semi-automática, consiste na combinação de informações vindas do usuário e de outras extraídas pelo algoritmo. A marcação realizada pelo usuário para diferenciar ou rotular as possíveis classes e divisões da imagem auxilia o processo automatizado do algoritmo no delineamento destas regiões. Esta técnica proporciona um aumento das possibilidades de uma melhor classificação final (MALMBERG, 2011).

Podemos também classificar métodos de segmentação de acordo com a estratégia de representação dos objetos a serem segmentados. Zaitoun e Aqel (2015) descrevem uma classificação para métodos de segmentação, conforme pode-se observar na Figura 2.10.

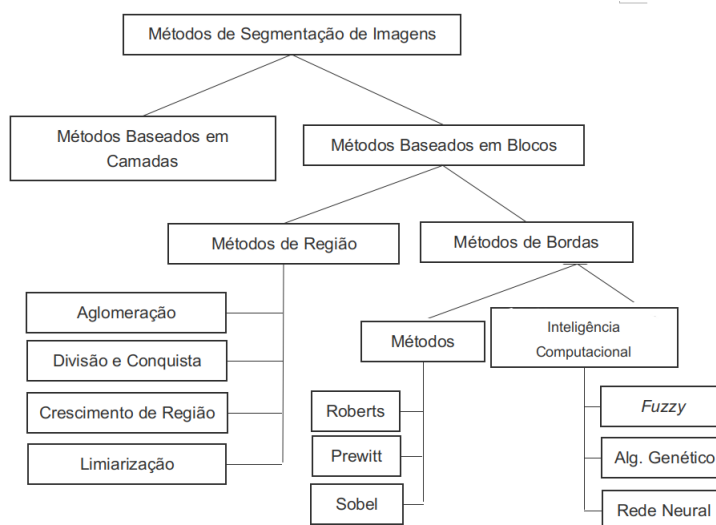
Métodos baseados em camadas (*Layer-based segmentation*) seguem a estratégia de comparação de conjuntos de imagens onde, através de definição de mascaras, é possível realizar a segmentação.

Métodos baseados em blocos (*Block-based segmentation*) baseiam-se nas várias características encontradas na imagem, Este grupo de modelos serão abordados neste trabalho.

Algoritmos baseados em bloco utilizam-se de uma ou duas propriedades dos valores de níveis de cinza: descontinuidade e similaridade. Podem ser divididos em três sub-grupos: Métodos baseados em regiões, métodos baseados em borda e técnicas híbridas.

Métodos orientados a bordas podem detectar bordas de cada objeto de interesse

Figura 2.10 – Classificação dos Métodos de Segmentação.



Fonte: Adaptado de Zaitoun e Aqel (2015)

na imagem, baseando-se nas mudanças abruptas dos níveis de cinza da imagem, sendo utilizada em detecção de pontos isolados, detecção de linhas e detecção de bordas através de máscaras de convolução.

Os métodos orientados a regiões possuem como exemplos de algoritmos, técnicas baseadas em limiarização (*thresholding*), crescimento de regiões (*region growing*), divisão e conquista (*split & merge*) e aglomeração (*clustering*).

Os métodos citados serão discutidos nas próximas seções do trabalho.

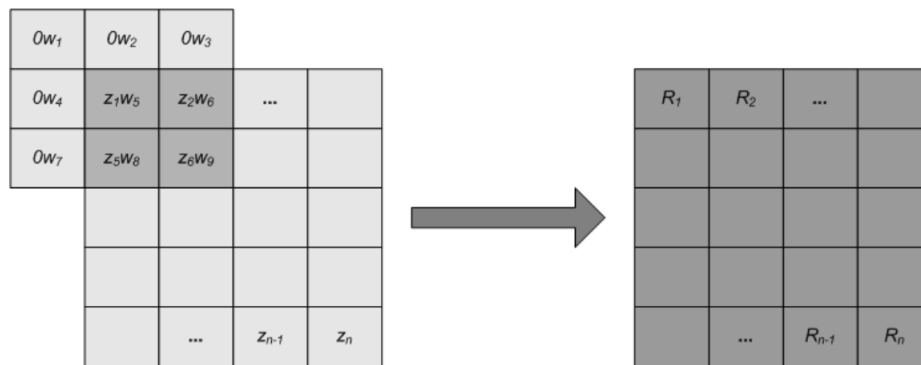
2.3.1 Técnicas Baseadas em Descontinuidade

Os objetos que podem ser localizados por técnicas baseadas em descontinuidade são os pontos, as linhas e as bordas. Tais técnicas são operações lógicas e matemáticas orientadas a vizinhança que utilizam o conceito de convolução de máscaras (GONZALEZ; WOODS, 2010).

As máscaras são conhecidas da literatura em inglês como *templates*, *kernel* ou *masks*.

A convolução envolve percorrer uma imagem, pixel a pixel, utilizando uma máscara, onde o pixel central da máscara se posiciona em cada pixel da imagem original até percorrer todos as posições dessa imagem. A cada posicionamento da máscara é realizada a operação matemática com o pixel e seus vizinhos a partir do peso de cada uma das posições da máscara utilizada e o resultado é gravado na posição em que o pixel central da máscara ocupou, como pode-se observar na Figura 2.11.

Figura 2.11 – Processo de Convolução sobre uma imagem.



Fonte: Adaptado de Gonzalez e Woods (2010)

Na Figura 2.11, a posição do (x, y) da imagem que foi sobreposta pela posição central da máscara será o resultado do cálculo da convolução dos píxeis vizinhos pelos pesos da máscara. Após isso, a máscara é movida para o próximo pixel da imagem original. No final do processo, uma imagem resultante é gerada preservando a original no processo Equação 2.14.

$$R = w_1z_1 + w_2z_2 + \dots + w_9z_9 = \sum_{i=1}^9 w_i z_i \quad (2.14)$$

O processo de detecção de pontos isolados em uma imagem é direto, utilizando-se a máscara da Figura 2.12, onde afirma-se que um ponto é detectado na localização em que a máscara está centralizada se:

$$g(x, y) = \begin{cases} 1 & \text{se } |R(x, y)| \geq T \\ 0 & \text{caso contrário} \end{cases} \quad (2.15)$$

onde T é um limiar não negativo, e R é dado pela Equação 2.15.

Figura 2.12 – Máscara de Convolução para detecção de Pontos Isolados

-1	-1	-1
-1	8	-1
-1	-1	-1

Fonte: Adaptado de Gonzalez e Woods (2010)

Conforme observa-se na Figura 2.12, a máscara mede as diferenças de pesos entre o ponto central e seus vizinhos.

Da mesma forma que se obtêm um ponto isolado, pode-se também detectar uma linha em uma imagem desde que a máscara de peso utilizada seja adequada para realçar as partes de uma reta. As máscaras apresentadas na Figura 2.13 realizam tal detecção.

Figura 2.13 – Máscara de Convolução para detecção de Linhas

-1	-1	-1	-1	-1	-1	2	-1	-1	2	-1	2	-1	-1	-1	-1	2	-1	-1	-1
2	2	2	-1	2	-1	-1	2	-1	-1	-1	2	-1	-1	-1	2	-1	-1	-1	-1
-1	-1	-1	2	-1	-1	-1	2	-1	-1	-1	-1	2	-1	-1	-1	-1	-1	2	-1
Horizontal			+ 45°			Vertical			- 45°										

Fonte: Gonzalez e Woods (2010)

Gonzalez e Woods (2010) descrevem uma borda como uma linha inserida entre dois objetos ou, ainda, uma diferença abrupta entre os píxeis vizinhos de objetos ou regiões distintas.

Os métodos baseados em detectores de borda procuram os píxeis que se encontram em um limite de uma região, partindo do pressuposto de que a mudança de valores entre píxeis vizinhos dentro de uma região não é tão significativa quanto a mudança de valores nos píxeis em uma região de fronteira (NAVON; MILLER; AVERBUCH, 2005).

Durante a iteração do algoritmo de detecção de borda, os valores dos píxeis localizados em limite de uma região crescem, destacando cada vez mais a borda, assim separando as regiões ou objetos internos da imagem (NAVON; MILLER; AVERBUCH, 2005).

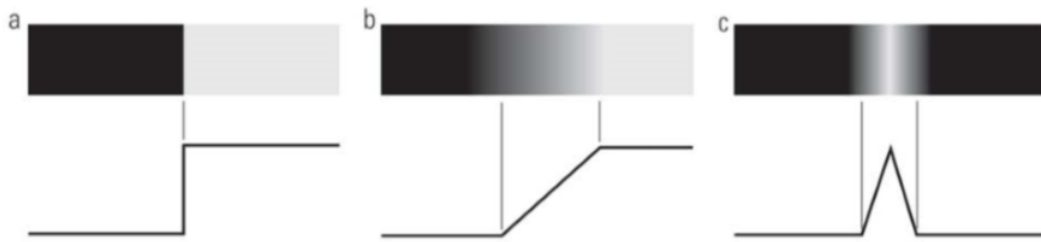
Gonzalez e Woods (2010) demonstram alguns tipos de borda que podem ser encontrados no processo de segmentação, como pode ser visualizado na Figura 2.14.

As bordas de tipo degrau, Figura 2.14a, são as ideais para o processo de segmentação. Este tipo de borda ocorre quando a transição de cores acontece entre dois píxeis separados.

Nas bordas em rampa, Figura 2.14b, diferentemente das bordas em degrau, a transição de cores ocorre em mais de um píxel, tendo a inclinação de rampa como medida de nitidez da imagem.

As bordas em forma de telhado exemplificada na Figura 2.14c, utilizam uma rampa crescente e a seguir uma decrescente, exercendo a função de linha divisória, onde a nitidez é medida segundo a espessura de tal linha.

Figura 2.14 – Estilos de Borda: a)Borda em degrau; b)Borda em Rampa; c)Borda em forma de telhado



Fonte: Gonzalez e Woods (2010, pag.462)

Gonzalez e Woods (2010) descrevem operadores para detecção de bordas horizontais e verticais de dimensão 3×3 , como pode-se observar na Tabela 2.1.

Tabela 2.1 – Operadores de Gradiente 3×3

Operador	Vertical ($ G_y $)	Horizontal ($ G_x $)
Roberts	$\begin{bmatrix} 0 & 0 & -1 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
Sobel	$\frac{1}{4} \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{4} \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

Gonzalez e Woods (2010) também descrevem operadores que não são sensíveis a rotação e detectam bordas em todas as direções, como o operador Laplaciano, que também é descrito em uma máscara de convolução 3×3 , como observa-se na Equação 2.16.

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & -4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \tag{2.16}$$

Castleman (1996) afirma que o operador Laplaciano tem sua utilização restrita devido a sua suscetibilidade a ruídos na imagem.

Gonzalez e Woods (2010) descrevem como a técnica mais avançada para a tarefa de delimitar uma borda nos objetos de uma imagem é a utilização da máscara de convolução Laplaciano da Gaussiana.

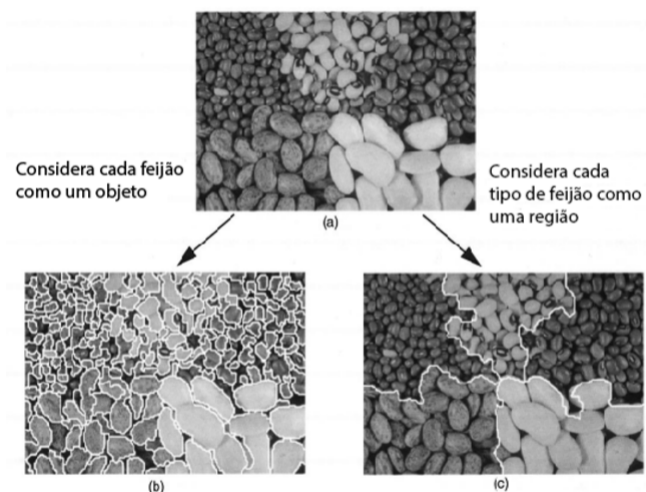
Este operador possui dimensão 5×5 e permite detectar correções de foco e nitidez

presentes na imagem tornando-o muito mais eficiente, conforme descrito na Equação 2.17.

$$\begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix} \quad (2.17)$$

Na figura 2.15 podemos observar o funcionamento do algoritmo onde primeiramente, em uma escala menor, ocorre a detecção das bordas de cada feijão e, a seguir, utilizando uma escala maior, se realiza a detecção da região de cada tipo de feijão.

Figura 2.15 – Detecção de Bordas: a) Apresenta a imagem com cinco tipos de feijões; b) Resultado da segmentação utilizando uma escala pequena c) Resultado de segmentação utilizando uma escala maior



Fonte: Ma e Manjunath (2000)

Em imagens coloridas, a detecção de borda ocorre utilizando o mesmo processo de uma imagem em tons de cinza, devendo-se aplicar o método de detecção de bordas de tal forma que cada componente da imagem seja considerado uma imagem e, a seguir, estas sejam unidas novamente combinando os resultados (GONZALEZ; WOODS, 2010).

2.3.2 Técnicas Baseadas em Similaridades

Nesta seção discute-se técnicas de segmentação de imagens diretamente em regiões com base nos valores de intensidade e/ou propriedades desses valores.

2.3.2.1 Técnicas de Limiarização

Segundo Gonzalez e Woods (2010), as técnicas de limiarização (*thresholding*) possuem um papel central nas aplicações de segmentação de imagens devido a sua simplicidade de implementação e velocidade computacional.

É utilizada principalmente em imagens escuras com fundo claro ou vice-versa onde, neste caso, o histograma da imagem apresenta mais de um pico separado por vales permitindo segmentar regiões ou objetos da imagem original (DASS; PRIYANKA; DEVI, 2012; GONZALEZ; WOODS, 2010). Assim, uma maneira óbvia de extrair objetos do fundo é selecionar um limiar T que separa estes modos. Então, qualquer ponto (x, y) na imagem em que $f(x, y) > T$ é chamado de *ponto do objeto*; caso contrário, o ponto é chamado de *Ponto de fundo* (GONZALEZ; WOODS, 2010). Como podemos verificar na Equação 2.18.

$$g(x, y) = \begin{cases} 1, & \text{se } f(x, y) > T \\ 0, & \text{se } f(x, y) \leq T, \end{cases} \quad (2.18)$$

Segundo Gonzalez e Woods (2010), quanto o valor T é uma constante aplicável a toda a imagem, o processo dado nesta equação é conhecido como limiarização global. Quando existe a variação de T na imagem ocorre a limiarização variável denominada limiarização local ou regional.

O termo limiarização local também é utilizado para caracterizar a limiarização variável, onde o valor de T depende das propriedades de vizinhança (x, y) como, por exemplo, a média da intensidade dos píxeis vizinhos.

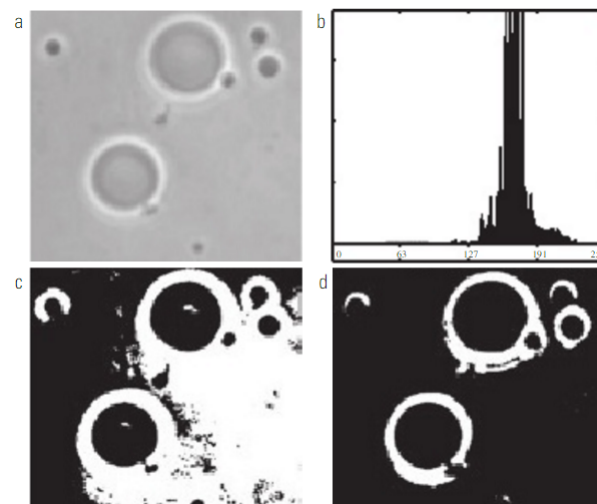
Gonzalez e Woods (2010) discutem que o Método de Otsu é um modelo de segmentação por limiarização global atraente pois maximiza a variância entre as classes da imagem. Seu objetivo é determinar um limiar que separe os elementos de fundo e de frente da imagem a partir de uma imagem em tons de cinza.

O Método de Otsu foi proposto por Nobuyuki Otsu (OTSU, 1979). Em 2001 foi proposta uma variação deste algoritmo para limiarização multinível por Liao, Chen e Chung (2001).

Na figura 2.16 é demonstrado a aplicação de segmentação por limiarização utilizando o Método de Otsu, onde podemos observar a imagem original, o histograma com os picos elevados o resultado para efeito de comparação da segmentação realizada com o algoritmo global básico e o resultado obtido pelo método proposto por Otsu.

Skarbek e Koschan (1994) propõem algumas definições para o uso de limiarização em imagens coloridas, onde regiões com poucas características têm a tendência a pertencer ao fundo da imagem e sendo necessária a normalização dos valores RGB de cada pixel

Figura 2.16 – Limiarização: a) Imagem Original; b) Histograma (os picos elevados foram cortados para realçar os detalhes nos valores mais baixos c) Resultado da segmentação utilizando o algoritmo global básico d) Resultado obtido pelo método Otsu



Fonte: Gonzalez e Woods (2010, Pag. 492)

para assim conseguir obter a segmentação adequada.

2.3.2.2 Crescimento de Regiões

O método de segmentação por crescimento de regiões é um procedimento que agrupa píxeis ou regiões de uma imagem com base em critérios pré estabelecidos. O processo começa a partir de uma semente e o algoritmo vai anexando, de forma iterativa, novos píxeis vizinhos à região de acordo com a avaliação das propriedades de cada novo pixel (GONZALEZ; WOODS, 2010).

A seleção de píxeis da região será realizada a partir de critérios estabelecidos, podendo levar em consideração a análise da natureza do problema no diagnóstico das imagens ou, se não estiver disponível, a seleção considerará apenas as características dos píxeis já selecionados.

Dass, Priyanka e Devi (2012) afirma que este é um algoritmo relativamente simples e imune a ruídos. Este método aplica o conceito de divisão e junção entre as partes de acordo com as propriedades encontradas, tendo como ponto de parada a obtenção de diversas regiões que contenham propriedades idênticas.

A parada do crescimento é um dos problemas deste método. Para melhorar seu desempenho, podemos acrescentar os conceitos de tamanho da região, semelhança entre os píxeis, píxeis selecionados até o momento, além de atributos como intensidade, cor e textura. A desvantagem deste método é a necessidade de pós-processamento, com o

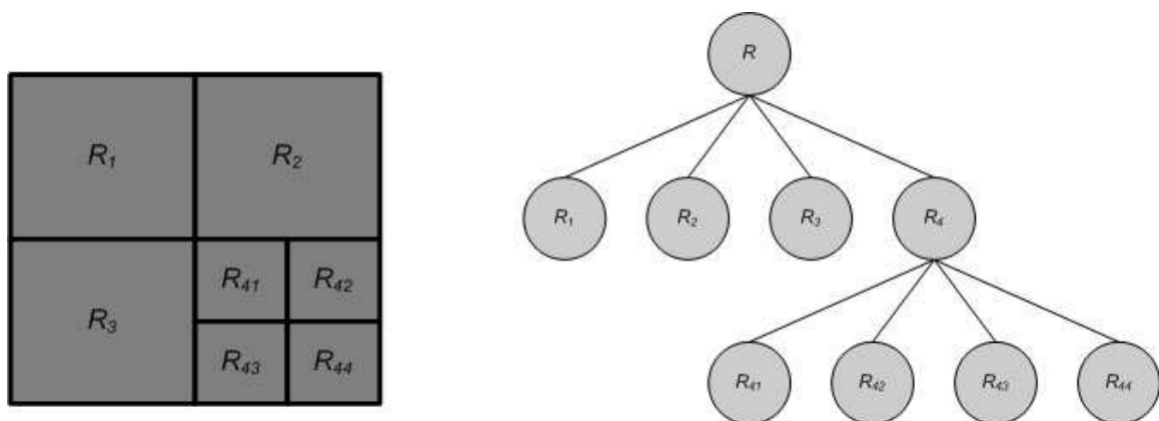
objetivo de unir regiões próximas que possam características semelhantes (GONZALEZ; WOODS, 2010).

2.3.2.3 Técnica de Divisão e Conquista

O processo de crescimento de regiões aumenta o tamanho das regiões a partir de um pixel semente. Uma alternativa é particionar a imagem em regiões distintas e arbitrárias e, em seguida, fundir e/ou dividir regiões em uma tentativa de satisfazer as condições de segmentação.

Gonzalez e Woods (2010) definem o modelo de crescimento de regiões em 3 passos: 1. Dividir em 4 quadrantes separados qualquer região R , para a qual $Q(R_i) = \text{Falso}$; 2. Quando não for possível continuar dividindo, fundir as regiões adjacentes R_j e R_k para as quais $Q(R_j \cup R_k) = \text{Verdade}$; 3. Parar quando a fusão não for mais possível; o processo descrito é demonstrado na Figura 2.17.

Figura 2.17 – Representação Gráfica do Método de Divisão e Conquista



Fonte: Gonzalez e Woods (2010, Pag. 505)

O modelo pode ser computacionalmente intensivo se o processo de divisão começar pela imagem inteira. Por outro lado, se começar com uma árvore extremamente ramificada, ficará suscetível a erros por estar baseada em apenas poucos píxeis vizinhos. Desta forma, a técnica de Divisão e Conquista deve iniciar em um estágio intermediário.

2.3.2.4 Técnica de Bacias Hidrográficas

Com o objetivo de mesclar as vantagens das técnicas de detecção de bordas, limiarização e crescimento de regiões foi criada a técnica de *watersheds* ou bacias hidrográficas. Esta técnica permite a incorporação de uma segmentação estável e com fronteiras de segmentação conectadas (GONZALEZ; WOODS, 2010).

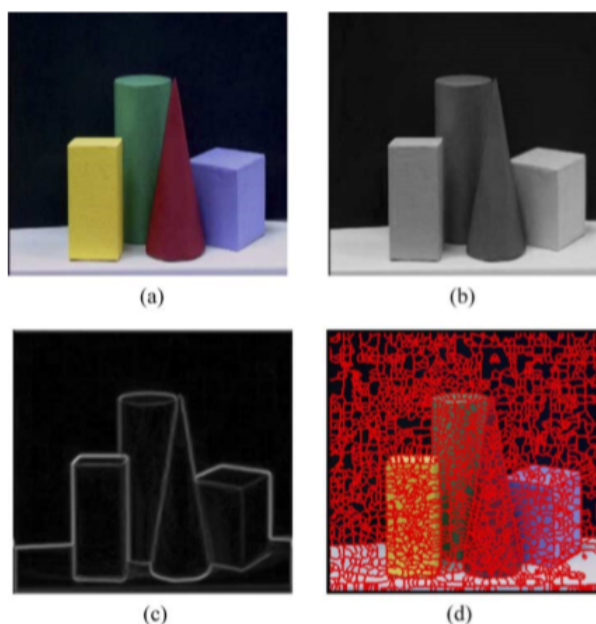
A técnica de segmentação de imagens, que utiliza o conceito de bacias hidrográficas (*watersheds*), faz analogia a um relevo que contenha bacias e montanhas e que, ao serem inundadas, elas formam linhas divisórias.

Esta “inundação” será determinada pelo gradiente, ou seja, a alteração do valor do pixel em uma unidade de espaço. A imagem se torna uma espécie de relevo onde regiões baixas são representadas com menor gradiente e as mais altas com um gradiente maior.

O gradiente morfológico, que representa a transição dos níveis de intensidade na imagem, permite realçar os contornos da imagem, mas é importante suavizar a imagem antes de aplicar o método para evitar uma quantidade excessiva de mínimos locais devido à sensibilidade do operador de gradiente ou à presença de ruídos, podendo gerar o efeito de super-segmentação da imagem (ARAÚJO, 2010).

Na Figura 2.18 podemos observar a execução do algoritmo de bacias hidrográficas e o problema da super-segmentação.

Figura 2.18 – Aplicação de técnica de Bacias Hidrográficas: a) Imagem Original; b) Imagem em escala de cinza c) Imagem após a aplicação do detector de bordas d) Segmentação resultante do algoritmo de bacias hidrográficas



Fonte: Navon, Miller e Averbuch (2005)

No caso de imagens coloridas deve-se realizar uma conversão para tons de cinza, o que permite a melhor identificação de contornos pois estes se tornam mais nítidos. Para esta análise é utilizada a camada de matiz da imagem que possibilita a aplicação do gradiente para detectar as bacias e barreiras da imagem (NAVON; MILLER; AVERBUCH, 2005).

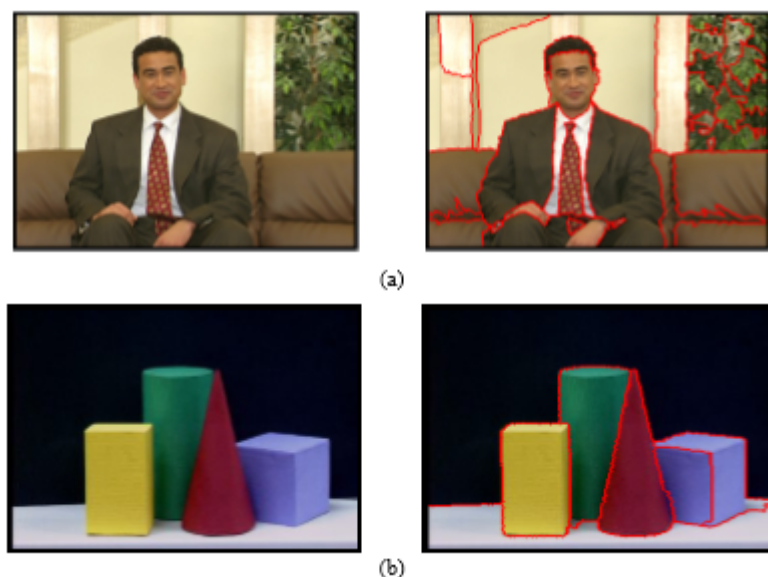
A técnica se inicia com a definição da posição de coordenada (x,y) e a intensidade e, posteriormente, é inserido um gradiente que selecione apenas os contornos relevantes da imagem.

O algoritmo é executado em duas etapas: a primeira é a fase de triagem onde os pixels são classificados em ordem crescente de acordo com a intensidade de cada um.

A fase seguinte é a fase das inundações, que consiste na verificação dos pixels pela ordem crescente e construção de bacias (regiões com pixels de menor gradiente), onde cada uma recebe um rótulo.

Finalizada a fase de construção de bacias, então é avaliada a similaridade entre regiões vizinhas, tomando por referência as cores e bordas das regiões. Através de um grafo construído a partir das regiões identificadas é verificado as características semelhantes entre essas áreas realizando a junção de regiões semelhantes que haviam sido anteriormente segmentadas. Por meio desta solução é possível diminuir o número de regiões que dividem a imagem como pode ser visualizado na Figura 2.18 e na Figura 2.19.

Figura 2.19 – Aplicação de técnica de Bacias Hidrográficas proposta (NAVON; MILLER; AVERBUCH, 2005) : a) Imagem Original; b) Resultado final da imagem após a segmentação



Fonte: Navon, Miller e Averbuch (2005)

Este método de segmentação é alvo de diversos estudos, propondo diversas abordagens de execução como, por exemplo, através do algoritmo de Caminhos Mínimos proposto por Osma-Ruiz et al. (2007), ou do algoritmo *Watershed Cut* proposto por Cousty et al. (2009).

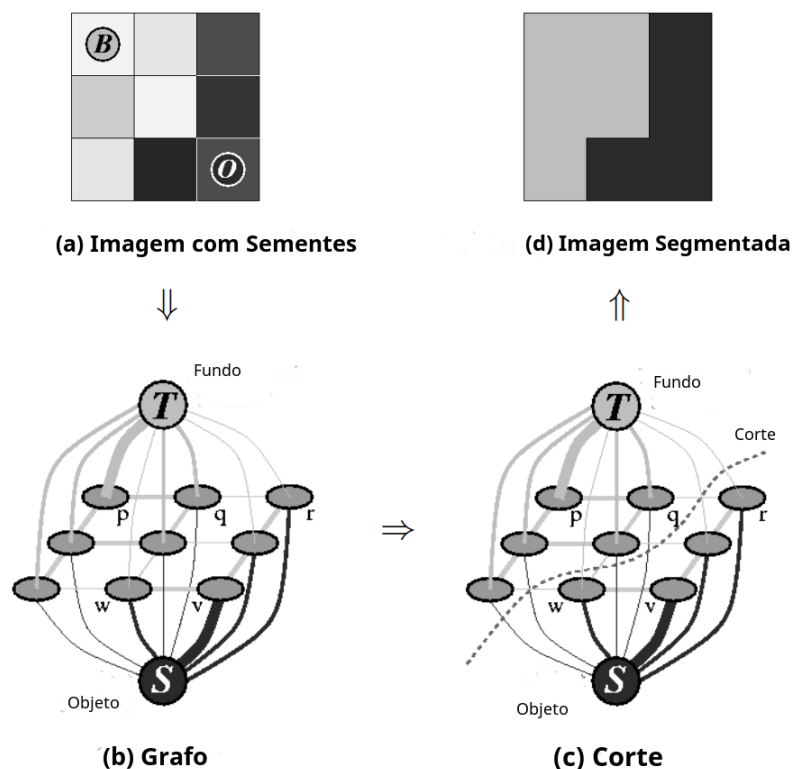
2.3.3 Técnicas Baseadas em Grafos

Segmentação baseada em grafos geralmente descreve o problema como um grafo $G = \{V, E\}$, onde cada nó V corresponde a um pixel na imagem e uma aresta (V_i, V_j) pertencente a E conecta V_i e V_j com um peso associado baseado em alguma propriedade do pixel.

Boykov e Jolly (2001) descrevem uma abordagem de segmentação de propósito geral denominado *Interactive Graph cuts*, onde o usuário marca certos píxeis como “objeto” ou como “fundo” para fornecer restrições para o algoritmo de segmentação. Além das restrições impostas pelo usuário, restrições adicionais são incorporadas como informações de fronteira e regiões e, a partir das informações, é gerado um grafo da imagem.

O algoritmo realiza cortes no grafo para encontrar uma segmentação globalmente ótima da imagem. A solução obtida deve resultar no melhor equilíbrio de fronteira e região satisfazendo as restrições impostas, como Boykov e Jolly (2001) demonstram na Figura 2.20.

Figura 2.20 – Segmentação Imagem utilizando método proposto por Boykov e Jolly (2001)



Fonte: Boykov e Jolly (2001)

No modelo, cada pixel é representado por um vértice e, as arestas, as relações entre os píxeis vizinhos. Para realizar a segmentação de imagens são utilizados cortes no grafo

(*graph cuts*) tendo como referência píxeis especiais que foram previamente rotulados pelo usuário (sementes). Deste modo, um corte no grafo define um subconjunto de vértices que irão representar a imagem final segmentada.

Boykov e Kolmogorov (2004) propuseram melhorias no processo de corte de arestas no modelo melhorando sua eficiência.

O modelo foi estendido por Rother, Kolmogorov e Blake (2004) que propuseram o modelo denominado *GrabCut*. Uma das vantagens deste modelo é a fácil interação com o usuário. O usuário, para realizar uma segmentação, deve desenhar um retângulo do objeto de interesse e o algoritmo analisa a segmentação, como pode-se observar na Figura 2.21.

Figura 2.21 – Exemplo de aplicação do *GrabCut*. O usuário desenha um retângulo ao redor de um objeto, o qual é extraído da imagem inicial



Fonte: Rother, Kolmogorov e Blake (2004)

Como o processo de execução do algoritmo é interativo, é possível que na primeira interação do usuário o modelo não obtenha um resultado satisfatório. A partir de novas interferências do usuário, em áreas classificadas erroneamente, o algoritmo pode ser executado novamente até que a segmentação seja realizada. Este processo está exemplificado na Figura 2.22.

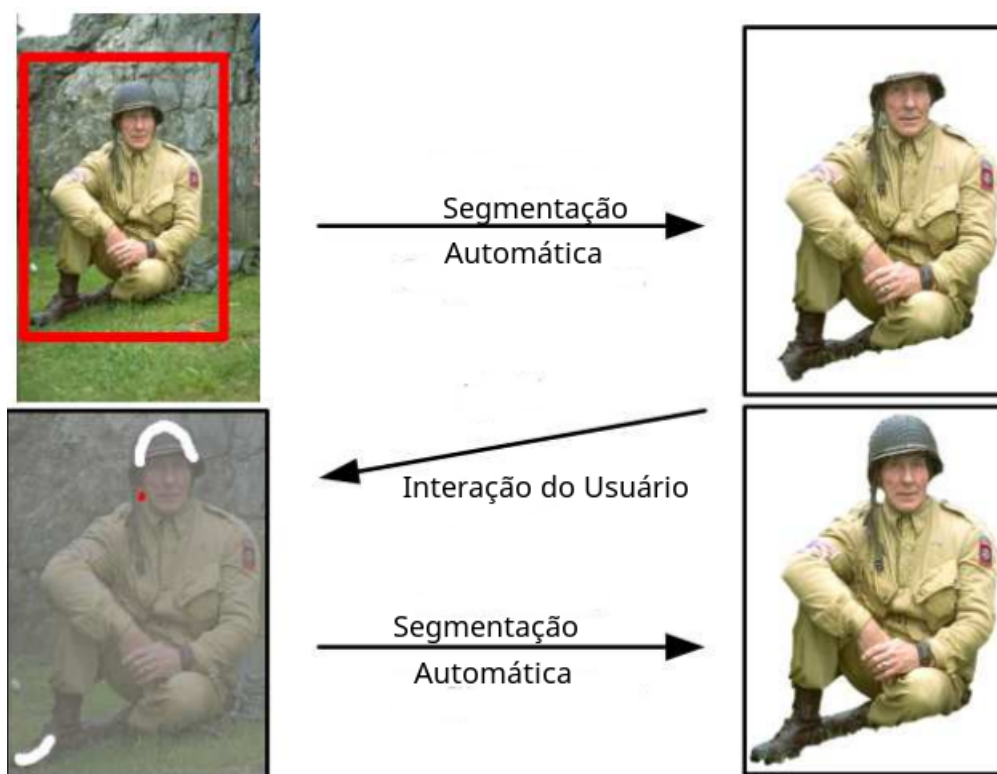
Métodos baseados em grafos também são utilizados em trabalhos em conjunto com métodos denominados *superpixels*, ou supercélulas, essa técnica está associada à redução do custo computacional nas várias tarefas de análise de imagens.

Na segmentação de imagens baseada no agrupamento em grafos, utiliza-se de agrupamentos de píxeis em células.

O algoritmo de supercélulas realiza o pré-processamento agrupando os píxeis nas chamadas células que serão utilizadas para a geração dos vértices do grafo.

Esse processo de pré-processamento permite uma redução considerável na quantidade de nós do grafo.

Figura 2.22 – Exemplo da Interação do Usuário com o algoritmo



Fonte: Rother, Kolmogorov e Blake (2004)

São apresentadas abordagens que utilizam grafos em conjunto com algoritmos de supercélulas (*superpixels*), em Ren e Malik (2003), que apresentam a geração de supercélulas baseadas em diversas características derivadas de Gestalt, Felzenszwalb e Huttenlocher (2004), que apresentam um método que tem a capacidade de preservar detalhes em regiões de imagem de baixa variabilidade. Moore et al. (2008) promove melhorias na geração das supercélulas, evitando a superdivisão da imagem. Veksler, Boykov e Mehrani (2010) propuseram melhorias através da geração de supercélulas mais compactas e de tamanhos mais regulares o ocasionam uma melhora na precisão do processo de segmentação.

No exemplo apresentado na seção anterior do trabalho, Navon, Miller e Averbuch (2005) utilizaram a técnica de grafos como ferramenta de auxílio para eliminar a supersegmentação no algoritmo de bacias hidrográficas (*watersheds*).

Na sequência do trabalho serão discutidos os conceitos de redes complexas.

3 Redes Complexas

Na última década pudemos observar um crescente interesse de pesquisas dentro da área de redes, afastando-se das análises em pequenas redes e focando em redes de grande escala denominadas redes complexas (SILVA; ZHAO, 2016).

São várias as aplicações para utilização de redes complexas em sistemas complexos, podemos citar aplicações em redes de informação, internet, *world wide web* (WWW), redes neurais, área financeira, redes sociais analisando a relação entre pessoas e corporações, metabolismo, redes elétricas, correios e redes de distribuição, redes de tráfego ou redes de interação entre proteínas (SILVA; ZHAO, 2016; BREVE, 2010; NEWMAN, 2010).

As redes complexas podem ser representadas como grafos que apresentam propriedades topográficas bastante particulares, não encontradas em grafos mais simples. O estudo de redes complexas geralmente é interdisciplinar porque abrange diversas áreas do conhecimento como ciência da computação, matemática, física, biologia e sociologia, sendo assim, a oportunidade de podermos representar diversos aspectos do mundo real.

Podemos citar, dentre as características de uma rede complexa, sua estrutura que envolve dados heterogêneos e não triviais; sua evolução com uma constante alteração em sua estrutura com a adição ou remoção de elementos; sua diversidade de conexões onde são estabelecidas ligações entre vértices utilizando diversos critérios como capacidade, comprimento, largura ou direção, possibilitando uma grande diversidade de conexões (SILVA; ZHAO, 2016).

Três dos principais modelos de redes complexas são as redes aleatórias, redes de pequeno-mundo e redes livres de escala.

As redes aleatórias¹ foram propostas por Erdős e Rényi (ERDŐS; RÉNYI, 1961; ERDŐS; RÉNYI, 1960; ERDŐS; RÉNYI, 1959), sendo o modelo mais simples que uma rede complexa pode assumir. Neste modelo, as arestas são geradas de forma aleatória e cada vértice pode possuir $(N - 1)$ arestas, onde N representa o número de vértices.

Devido a aleatoriedade do modelo, Erdős e Rényi concluíram que uma determinada rede possui aproximadamente a mesma quantidade de ligações e a mesma possibilidade de receber novas conexões e, quanto mais ligações eram adicionadas, maiores as probabilidades de serem gerados pequeno grupos de nós coesos. (BARABASI; ALBERT, 1999).

Em 1998, Watts e Strogatz (1998) definiram o conceito de rede pequeno mundo², onde observaram que muitas redes apresentavam padrões altamente conectados e formavam

¹ Do Inglês: *Random Networks*

² Do Inglês: *Small-World Networks*

grupos de vértices próximos, ou seja, a maioria dos vértices se conecta através de um caminho mínimo, gerando o efeito de pequeno mundo.

Determinaram, também, que a distância média entre dois pontos na rede não ultrapassa um pequeno número de nós e que basta que seja estabelecido um número reduzido de ligações entre os pequenos grupos para que isso ocorra.

Em 1999 foi proposto por Barabasi e Albert (1999) um novo modelo de rede denominado Redes Livre de Escala³.

Barabasi e Albert (1999) demonstraram em seu trabalho que algumas redes possuem uma ordem na sua dinâmica de estruturação, onde uma das principais características é denominada conexão preferencial.

A conexão preferencial é definida como a tendência de um vértice se conectar a um outro vértice na rede com um grau elevado de conexões, o que implica na criação de alguns vértices altamente conectados e muitos vértices com poucas conexões.

3.1 Conceitos de Grafos

A teoria dos grafos é a maneira mais natural de representar uma rede complexa, demonstrando seus elementos e suas relações através dos nodos e arestas de um grafo (BOCCALETTI et al., 2006).

Um grafo é definido como um par de conjuntos, $G = \{V, E\}$, onde V é o conjunto de vértices e E é o conjunto de arestas do grafo. Em um grafo não direcionado, cada aresta é um par não ordenado $\{v, e\}$, enquanto em grafos direcionados (dígrafos) os pares são ordenados (SILVA; ZHAO, 2016).

Um grafo pode também ser definido como um grafo ponderado quando as arestas possuem valores numéricos.

O tamanho do grafo é definido pelo número de arestas $m = \|E\|$ e na situação onde as arestas possuam pesos temos a função $w = E \rightarrow \mathbb{R}$, para designá-los. O número de vértices é determinado em $n = \|V\|$.

A densidade é a razão entre a quantidade de arestas e a quantidade de vértices do grafo. Em um grafo não direcionado é dada pela equação 3.1.

$$\delta(G) = \frac{2m}{n(n-1)} \quad (3.1)$$

onde $\delta(G) = 0$ quando $n = \{0, 1\}$, de modo que a densidade mínima será 0 e a máxima será 1 em um grafo completo (BREVE, 2010).

³ Do Inglês: *Scale-Free Networks*

O número de arestas que incidem sobre um dado vértice v é chamado grau de v e é representado por $deg(v)$, onde um grafo é considerado regular se todos os vértices têm o mesmo grau, equação 3.2.

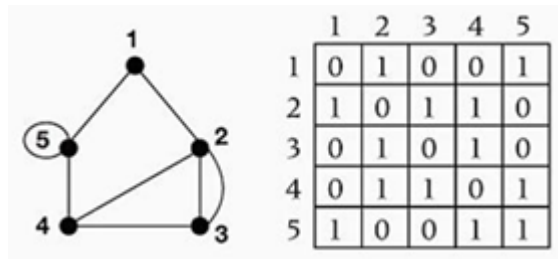
$$deg(S) = \sum_{v \in S} deg\{v\} \tag{3.2}$$

Um grafo pode ser representado adequadamente por meio de matrizes ou listas, assim dado um grafo não direcionado onde dois vértices $\{u, v\} \in E$ são chamados de vizinhos ou adjacentes, a matriz de adjacência A_G de um grafo G de ordem n tem tamanho $n \times n$, $A_G = (a_{u,v}^G)$, onde:

$$a_{u,v}^G = \begin{cases} 1, & \text{se } \{u, v\} \in E, \\ 0, & \text{caso contrario,} \end{cases} \tag{3.3}$$

gerando uma matriz simétrica para esse caso, como observado na Figura 3.1 (BREVE, 2010; ASCENCIO; ARAUJO, 2010).

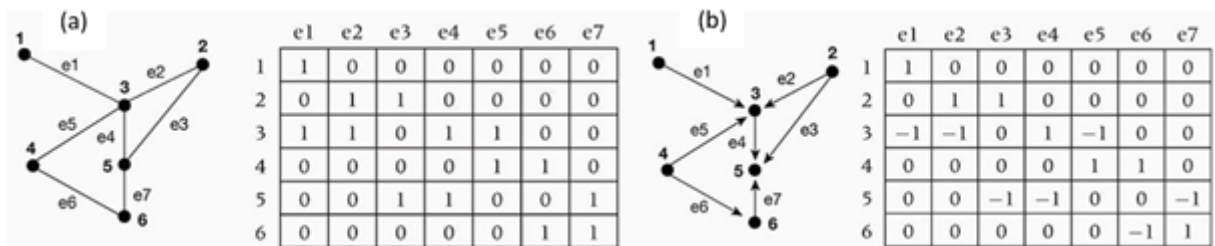
Figura 3.1 – Matriz de Adjacências para grafo não ordenado



Fonte: Ascencio e Araujo (2010, pag. 375)

Podemos representar um grafo a partir de sua matriz de incidências. Um grafo não orientado tem sua matriz de incidência representada na Figura 3.2a, e um grafo orientado tem sua matriz de incidência representada na Figura 3.2b.

Figura 3.2 – Matriz de Incidência: (a) Grafo não Orientado (b) Grafo Orientado



Fonte: Ascencio e Araujo (2010, pags. 376/377)

A matriz de incidência de um grafo não orientado é definida pela Equação 3.4 (ASCENCIO; ARAUJO, 2010).

$$a_{u,v}^G = \begin{cases} 1, & \text{se } (u, v) \in E, \\ 0, & \text{se } (u, v) \notin E, \\ 0, & \text{se } (u, u) \in E, \end{cases} \quad (3.4)$$

E um grafo orientado tem sua matriz de incidências determinada conforme a Equação 3.5 (ASCENCIO; ARAUJO, 2010).

$$a_{u,v}^G = \begin{cases} 1, & \forall u, \text{ se } (u, v) \in E, \\ -1, & \forall v, \text{ se } (u, v) \in E, \\ 0, & \text{se } (u, v) \notin E \vee \text{se } (u, u) \in E, \end{cases} \quad (3.5)$$

3.1.1 Medidas de Dissimilaridade e Similaridade

As medidas de dissimilaridade e similaridade têm grande importância pois são utilizadas para algoritmos de agrupamento como os utilizados em mineração de dados.

A dissimilaridade entre dois objetos é um valor numérico que representa o grau de diferença que existe entre os objetos x e y , e quanto mais baixo for esse valor maior será a semelhança entre eles. A dissimilaridade geralmente é conhecida como distância e, para que esta seja considerada uma métrica, deve satisfazer as seguintes condições:

- A não negatividade: $dist(x, y) \geq 0$
- A distância de um vértice para ele mesmo é zero: $dist(d_i, d_i) = 0$;
- Distâncias são simétricas: $dist(d_i, d_j) = dist(d_j, d_i)$;
- A desigualdade triangular é válida: $dist(d_i, d_j) \leq dist(d_i, d_k) + dist(d_k, d_j)$,

Tan, Steinbach e Kumar (2005) afirmam que a distância de Minkowski é uma métrica em um espaço vetorial normalizado que pode ser considerado uma generalização das distâncias Euclidiana e Manhattan.

A distância de Minkowski é determinada pela Equação 3.6:

$$d(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^r \right)^{1/r}, \quad r \geq 1 \quad (3.6)$$

onde x e y são vetores com n atributos que representam respectivamente 2 vértices do modelo (TAN; STEINBACH; KUMAR, 2005).

Assim quando temos $r = 1$, tem-se o cálculo da distância *Manhattan*, também conhecida como *City Block* e quando r assume valor $r = 2$, tem-se calculado a distância Euclidiana.

Na distância de Chebyshev, também conhecida como a distância do tabuleiro de xadrez (*chessboard*), é gerado um círculo de raio r , que é o conjunto de pontos com distância de Chebyshev r a partir de um ponto central. Por exemplo, em um tabuleiro de xadrez, o raio 1 gera um quadrado de 3x3. A distância de Chebyshev é determinada pela Equação 3.7 (TAN; STEINBACH; KUMAR, 2005).

$$d(x, y) = \max_{i=1}^n \{|x_i - y_i|\} \tag{3.7}$$

A distância Mahalanobis é baseada nas correlações entre variáveis com as quais padrões podem ser identificados. É uma estatística útil para determinar a similaridade entre uma amostra desconhecida e uma conhecida. Os dados nos grupos a serem comparados deverão ter o mesmo número de variáveis (ou seja, o mesmo número de colunas) mas não necessariamente o mesmo número de elementos (o número de linhas pode ser diferente). A distância é determinada pela Equação 3.8 (XIANG; NIE; ZHANG, 2008):

$$d(d, y) = \sqrt{(x - y)^T C^{-1} (x - y)} \tag{3.8}$$

onde C é a matriz de covariância do conjuntos de vetores ao qual x e y pertencem.

A similaridade é um valor numérico que expressa o grau de semelhança entre dois objetos (vértices). Quando o valor é alto, os objetos são semelhantes e geralmente o valor da similaridade não apresenta valores negativos, ficando sempre entre $[0,1]$, onde 0 é não similar e 1 é completamente similar.

Arruda, Costa e Rodrigues (2012), Tan, Steinbach e Kumar (2005) e Shi e Malik (2000) em seus trabalhos descrevem alguns descritores de similaridade que são demonstradas na Tabela 3.1. As funções Exponencial, Fu e Tanimoto foram descritas por Arruda, Costa e Rodrigues (2012), a função Gaussiana por Shi e Malik (2000) e a função Cosseno por Tan, Steinbach e Kumar (2005).

Tabela 3.1 – Exemplos de Função de Similaridade

Nome	Função
Exponencial	$s(x, y) = e^{-d(x,y)}$
Fu	$s(x, y) = 1 - \frac{d_2(x,y)}{\ x\ + \ y\ }$
Tanimoto	$s(x, y) = \frac{x^T y}{\ x\ ^2 + \ y\ ^2 - x^T y}$
Guassiana	$s(x, y) = e^{-\frac{d(x,y)^2}{\sigma^2}}$
Cosseno	$s(x, y) = \frac{x \cdot y}{\ x\ \ y\ } = \frac{\sum_i x_i y_i}{\sqrt{\sum_i (x_i)^2} \sqrt{\sum_i (y_i)^2}}$

3.1.2 Qualidade de agrupamento do grafo

Existem várias métricas para avaliar a qualidade do agrupamento do grafo, não existindo uma melhor métrica, isto dependerá da forma em que o grafo foi construído e da quantidade de nós que apresenta. Duas das métricas mais conhecidas é a *Silhouette Index* (TAN; STEINBACH; KUMAR, 2005) e a *Modularity* (NEWMAN; GIRVAN, 2004).

A métrica *Silhouette Index* é baseada em conceitos de coesão e separação, sua avaliação de qualidade é resultado do cálculo de distâncias entre os vértices do grafo o que permite medir sua similaridade. A métrica está descrita na Equação 3.9:

$$S_{index} = \frac{1}{|V|} \sum_{v \in V} \frac{b_v - a_v}{\max\{a_v, b_v\}} \quad (3.9)$$

onde V é o conjunto de vértices do grafo, a_v é a distância média entre o vértice v e todos os outros nós do mesmo grupo de v e b_v é a distância média entre o vértice v e todos os nós do grupo mais próximo que não seja do seu grupo.

Esta métrica pode assumir valores de $[-1, 1]$, apresentando valores negativos quando a média das distâncias internas dos grupos são maiores do que a média das distâncias externas. Valores próximos a 1 indicam boa qualidade de agrupamento.

A métrica Modularidade (*Modularity*) é uma das mais conhecidas para avaliar a qualidade do agrupamento em um grafo. Considerando a divisão de uma rede em k comunidades, é definida uma matriz simétrica e da ordem $k \times k$, onde os elementos e_{ij} são compostos pela fração de todas as arestas da rede que unem as comunidades i e j . Em relação à matriz e , a modularidade é definida pela Equação 3.10:

$$Q = \sum_i (e_{ii} - a_i^2) \quad (3.10)$$

onde $a_i = \sum_j e_{ij}$, que é o somatório das j linhas ou j colunas da matriz.

A métrica pode variar entre os valores de $[0, 1]$, onde valores próximos a 1 representam uma boa qualidade de agrupamento, o que se define por uma boa quantidade de arestas no interior das comunidades. A métrica de Modularidade pode assumir valor negativo se houver um grupo com apenas um membro (*singleton*).

3.2 Detecção de Comunidades em Redes

As redes no mundo real sofrem interações de forma dinâmica e essas interações podem ser representadas por meio de redes complexas.

Para que consigamos interpretar o comportamento de indivíduos em uma rede complexa, faz-se necessário a detecção de comunidades dentro da rede. Determinamos estas comunidades a partir do agrupamento de vértices que contenham uma alta densidade

de arestas entre eles e baixa densidade quando estas arestas interligam grupos distintos. Isto pode ser comparado a situações de comportamento de grupos de indivíduos, grupos com interesse distintos, como por ocupação profissional ou faixa etária (NEWMAN, 2003; HOPCROFT et al., 2003).

Esta definição inclui as mais diversas redes como *World Wide Web*, redes de citação, redes de transporte, *e-mail* ou redes sociais. Representando estes modelos em redes complexas podemos aplicar algoritmos de aprendizado de máquina para que, através de técnicas de agrupamento, seja possível determinar padrões de comportamento de um usuário de uma rede social ou de telefonia.

A seguir estudaremos conceitos de aprendizado de máquina enfatizando o semi-supervisionado.

4 Aprendizado de Máquina

Aprendizado de máquina é uma subárea dos estudos de Inteligência Artificial que visa a pesquisa de métodos computacionais relacionados à aquisição automática de novos conhecimentos, novas habilidades e formas de organizar o conhecimento já existente de forma mais eficiente (MITCHELL, 1997).

A abordagem conceitual de algoritmos de aprendizado de máquina busca a melhora de desempenho de forma automática a partir da experiência como o que ocorre com humanos. O comportamento do aprendizado nestes algoritmos pode ser obtido a partir de ajustes de parâmetros de processamentos referentes aos dados de entrada ou, ainda, de acordo com as informações desejadas na saída do algoritmo. A grande dificuldade está na diversidade de combinações possíveis para os conjuntos de dados de entrada e em todos os possíveis modelos de comportamentos encontrados, o que inviabiliza meramente sua representação por linguagens de programação tradicionais (MITCHELL, 1997; ALPAYDIN, 2014).

Alguns destes modelos computacionais visam eliminar a necessidade de analistas humanos na análise de dados, outros tem como estratégia a colaboração entre o especialista humano e o modelo computacional.

Para Alpaydin (2014) o aprendizado de máquina é utilizado onde o conhecimento humano não pode ser representado em um algoritmo computacional, pois o especialista não consegue descrever o processo de tomada de decisão para a máquina. O reconhecimento de objetos, por exemplo, é uma tarefa cotidiana em que é difícil de determinar o processo decisório realizado pelo cérebro .

Desta forma a aplicação dos conceitos de aprendizado de máquina objetiva a construção de algoritmos que sejam capazes de se adaptar a novas circunstâncias sem a necessidade de reescrever o modelo computacional. Algoritmos construídos aplicando tais técnicas analisam grandes coleções de dados permitindo reconhecer padrões existentes dentro do conjunto, outros analisam dados pré rotulados de forma a determinar funções que representam tais informações e que permitem ao algoritmo classificar com acurácia novas informações apresentadas ao modelo (ALPAYDIN, 2014).

Dentre as principais categorias de aprendizado de máquina podemos destacar o aprendizado supervisionado, não supervisionado e semi-supervisionado.

No modelo de aprendizado supervisionado o objetivo é determinar uma função que seja capaz de prever a saída válida para qualquer outra entrada de dados apresentada.

Assim, para o treinamento do modelo, tem-se a necessidade de uma base dados

previamente classificada por um especialista de modo que o algoritmo consiga generalizar um padrão dentro de uma função capaz de representar todos os dados pertencentes ao conjunto. A base de treinamento fornecida pelo supervisor são pares de exemplos, sendo um exemplo de entrada para um exemplo de saída esperado (BARBER, 2012).

Os algoritmos de aprendizado não supervisionado buscam determinar como os dados estão organizados, trabalham a partir de dados de entrada não rotulados e sem valores de saída. Objetiva encontrar padrões através da descoberta de relações de semelhança entre as informações recebidas e padrões de separação, de modo a dividir o conjunto de dados originais em sub-conjuntos (JAIN; MURTY; FLYNN, 1999).

O aprendizado não supervisionado dispensa a necessidade de um especialista, contudo esta sujeito a apresentar falhas devido ao fato de certas características serem diferentes dentro de uma mesma classe e semelhantes entre os dados de outra classe (BARBER, 2012).

O aprendizado semi-supervisionado é um paradigma de aprendizado que se utiliza de conceitos do aprendizado supervisionado e do aprendizado não-supervisionado. Por ser o modelo de aprendizado utilizado neste trabalho de pesquisa será abordado de forma mais detalhada na próxima subseção.

4.1 Aprendizado Semi-Supervisionado

O aprendizado de máquina semi-supervisionado encontra-se entre os aprendizados supervisionado e não-supervisionado. O aprendizado supervisionado nos permite classificar dados através de uma função gerada partir da análise de dados previamente rotulados, enquanto o aprendizado não-supervisionado, caracteriza-se por determinar a estrutura entre os dados, em geral esses algoritmos tem sido utilizados para resolver problemas de mineração de dados, reconhecimento de padrões, bioinformática, predição de séries temporais entre outros (ALPAYDIN, 2014; WITTEN; FRANK; HALL, 2011; AGGARWAL; YU, 2009; WOLFF; BHADURI; KARGUPTA, 2009; CIOU et al., 2007; BISHOP, 2006; HINTON; SEJNOWSKI, 1999; MITCHELL, 1997).

O aprendizado semi-supervisionado é uma técnica de aprendizado de máquina que se utiliza tanto das informações contidas nos dados rotulados como nos dados não rotulados, e devido a essa característica, essa técnica tem surgido como um alternativa interessante quando uma grande quantidade de dados não rotulados está disponível e rotular dados manualmente é muito custoso (CHAPELLE; SCHÖLKOPF; ZIEN, 2006).

Sabendo-se do custo da classificação manual de dados, é possível escolher apenas uma parte desses dados e rotula-los como exemplos para os demais. A partir do qual realiza-se a propagação da classificação para o restante dos dados, se aproximando bastante

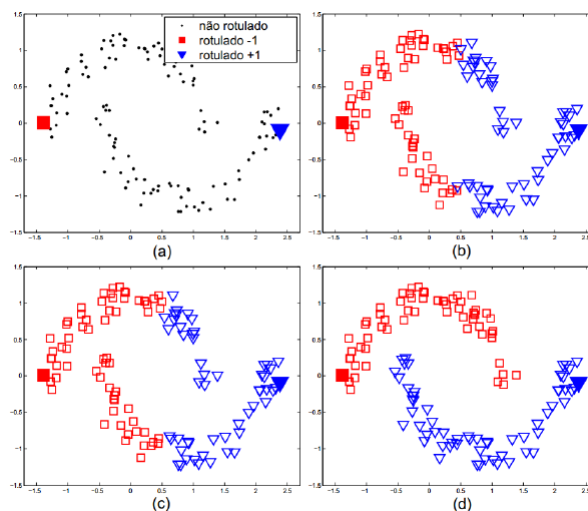
da forma que os seres humanos reconhecem e aprendem com o que o ocorre a sua volta.

Contudo, há dados que não se adéquam perfeitamente ao aprendizado de máquina semi-supervisionado, Segundo Lu (2009), há tipos de dados em que o aprendizado supervisionado não pode ser substituído pelo aprendizado semi-supervisionado devido a distribuição e características presentes na base de dados.

Outro ponto levantado por Lu (2009) é a possibilidade de que em algumas bases de dados a utilização do aprendizado semi-supervisionado possa gerar um aprendizado errôneo quando as suposições criadas pela técnica são deturpadas.

Zhou et al. (2004) demonstra uma distribuição de dados onde o aprendizado supervisionado não apresentaria bons resultados utilizando apenas os dados rotulados para treinamento, conforme pode ser visualizado na Figura 4.1.

Figura 4.1 – Classificação no conjunto de dados com dois objetos. (a) Dados de entrada; (b) resultado da classificação utilizando o Suport Vector Machines (SVM); (c) Classificação utilizando k-vizinhos; (d) classificação ideal.



Fonte: Zhou et al. (2004)

Barber (2012) descreve que o algoritmo trabalha com a oscilação de características presentes entre os dois dados rotulados, onde a classificação ocorre devido a proximidade e/ou distância entre os dados comparados e também utiliza-se de agrupamentos (*clusters*) onde os dados que estejam no mesmo grupo são suscetíveis de estarem na mesma classe.

O aprendizado semi-supervisionado é uma mescla de características entre o aprendizado supervisionado (dados conhecidos) e do aprendizado não-supervisionado em reconhecer um padrão entre os dados desconhecidos (CHAPELLE; SCHÖLKOPF; ZIEN, 2006).

Dentre as principais categorias de métodos de aprendizado semi-supervisionado podemos destacar: Generativos, Clusterização e Rotulação, Co-training, Separação de baixa

intensidade e Baseado em grafos. Tais métodos serão discutidos nas subseções adiante.

4.1.1 Auto Treinamento

O auto treinamento¹ é uma das formas mais primitivas de aprendizado semi-supervisionado, que também é conhecida como auto aprendizado ou *bootstrapping*.

Breve (2010) descreve que este paradigma está apoiado em um classificador que é treinado utilizando poucos dados rotulados e, a partir destes, é utilizado para classificar todos os dados não-rotulados. Os dados rotulados que tiverem maior confiança serão adicionados ao conjunto de treinamento.

O processo é repetido utilizando o novo conjunto de treinamento até que todos os dados tenham sido rotulados. Uma desvantagem é que eventuais erros de classificação serão propagados para todas as iterações subsequentes e o desempenho dependerá do classificador utilizado no procedimento (CHAPELLE; SCHÖLKOPF; ZIEN, 2006; ZHU, 2008).

A ideia do auto treinamento vem aparecendo na literatura há bastante tempo. Yarowsky (1995) utilizou auto treinamento para remoção de ambiguidade de palavras em inglês de mesma grafia mas significados diferentes, posteriormente Riloff, Wiebe e Wilson (2003) utilizaram para identificar substantivos subjetivos para distinguir sentenças entre objetivas ou subjetivas. Haffari e Sarkar (2007), Abney (2004) realizaram uma análise mais profunda do algoritmo de Yarowsky.

4.1.2 Modelos Generativos

Os métodos generativos são uma categoria antiga de aprendizado semi-supervisionado, onde assume-se que um modelo $p(x, y) = p(y) p(x|y)$ onde $p(x|y)$ é uma distribuição com mistura identificável como, por exemplo, um modelo de mistura Gaussiano (ZHU, 2008).

Zhu (2008) demonstra que com uma grande quantidade de dados não rotulados os componentes da amostra podem ser identificados com apenas um exemplo rotulado em cada componente, para determinar completamente a mistura da distribuição.

Deste modo, Nigam et al. (2000) aplicou este modelo em um problema de classificação de texto, mostrando que o desempenho quando são utilizados também os dados não rotulados é melhor que o desempenho obtido quando são utilizados apenas os dados rotulados e Fujino, Ueda e Saito (2005) estenderam esse modelo incluindo um termo de correção.

¹ do inglês: self-training

4.1.3 Agrupar e Rotular

A técnica de agrupar e rotular² tem por objetivo realizar o agrupamento da base de dados e em seguida rotular cada uma dos grupos descobertos (DARA; KREMER; STACEY, 2002; DEMIRIZ; BENNETT; EMBRECHTS, 1999).

Os algoritmos desta classe tendem a funcionar bem quando os algoritmos de agrupamento escolhidos combinam com a distribuição real dos dados (ZHU, 2008; DARA; KREMER; STACEY, 2002; DEMIRIZ; BENNETT; EMBRECHTS, 1999).

Demiriz, Bennett e Embrechts (1999) apresentam um algoritmo que utiliza métodos de agrupamento (aprendizado supervisionado) para determinar os subconjuntos de dados e seus pontos centrais, a partir disto, o algoritmo verifica dos pontos centrais encontrados a sua relação de proximidade com os pontos que foram previamente rotulados. Deste modo, cada um dos subconjuntos de dados são rotulados a partir desta amostra.

4.1.4 Co-Treinamento

No Co-Treinamento³ assume-se que:

1. o conjunto de atributos pode ser dividido em dois subconjuntos;
2. cada subconjunto de atributos é suficiente para treinar um bom classificador;
3. ambos os conjuntos são condicionalmente independentes dada a classe.

Dois classificadores separados são treinados com os dados rotulados de cada um dos dois subconjuntos, estes então classificam os dados não rotulados. A seguir, como no auto treinamento, os dados com rótulos preditos mais confiáveis são utilizados para treinar o outro classificador, assim o processo se repete sempre com dados adicionais. (ZHOU; LI, 2007; ZHOU; LI, 2005; MITCHELL, 1999; BLUM; MITCHELL, 1998; MITCHELL, 1997).

A técnica de Co-treinamento tem o objetivo de modelar a relação entre as variáveis dependentes e independentes, selecionar quais rótulos serão utilizados para um melhor treinamento, a fim de abranger todas as classes e permitir uma boa classificação. Finalizando a etapa anterior, são adicionados pesos às classificações obtidas para que haja uma classificação mediana, a qual será utilizada nos demais dados (HU; YOUN; KIM, 2012).

Os classificadores precisam concordar tanto nos dados rotulados quanto nos dados não rotulados, dependendo a confiabilidade do classificador da qualidade dos subconjuntos de atributos com o qual eles foram treinados. Estes não consideram a existência de redundância apresentada por características ou divisão entre as classes. Para esta identificação

² do inglês: cluster-and-label

³ do inglês: co-training

pode ser necessário o uso de algoritmos para seleção de características, com a finalidade de remover as repetições (LIU, 2011).

O processo termina quando todos os nós receberam uma classificação.

4.1.5 Separação de Baixa Densidade

A categoria de métodos de separação de baixa densidade assume que a separação entre as classes se encontra em uma região de baixa densidade e, portanto, tentam empurrar a fronteira de decisão para longe das áreas de alta densidade.

A abordagem mais comum é usar um algoritmo que maximiza a margem, como as máquinas de vetor de suporte⁴. Tal método, quando usado para maximizar margens para dados rotulados e não rotulados, é chamado Transductive SVM (TSVM) (VAPNIK, 1998).

As TSVM estendem as SVM tradicionais que utilizam somente dados rotulados e o objetivo é encontrar rótulos para os dados não rotulados de modo que a fronteira tenha a máxima margem, o que de forma intuitiva ocorre, os dados não rotulados guiam a fronteira para longe das regiões densas. Este algoritmo encontra-se na categoria de NP-Difícil (ZHU, 2008).

Com o auxílio dos dados rotulados é estudada uma regra para ser utilizada como meio de classificação da base de dados. Este mecanismo difere do aprendizado supervisionado por utilizar um número finito de rótulos para o aprendizado (BARBER, 2012).

Estes métodos reforçam a distinção entre os conjuntos através da melhor classificação dos elementos a margem do grupo em uma região onde temos uma menor densidade de informação.

4.1.6 Métodos Baseados em Grafos

Recentemente observamos uma maior atividade dentre os métodos baseados em grafos. Esta categoria demonstra os dados representados por nós e os relacionamentos, que são as similaridades entre esses dados, estão representados pelas arestas. Tais métodos assumem que existe suavidade dos rótulos no grafo. A maioria dos métodos pode ser vista como um *framework* de regularização que estima uma função f que satisfaça tanto a suavidade dos rótulos quanto a proximidade dos rótulos estimados aos dados pré-rotulados (WANG et al., 2008; WANG; ZHANG, 2008; WU; SCHÖLKOPF, 2007; GRADY, 2006; ZHOU et al., 2004; ZHU et al., 2003; SZUMMER; JAAKKOLA, 2002; ZHU; GHAMRANI, 2002).

⁴ do inglês: support vector machines - SVM

Métodos baseados em grafos fazem uso de matriz de adjacência para realizar as conexões entre os nós. Sendo definida pela Equação 4.1.

$$W_{ij} = \begin{cases} w_{(e)} & \text{se } i \text{ e } j \text{ são vizinhos} \\ 0 & \text{se } i \text{ e } j \text{ não são vizinhos} \end{cases} \quad (4.1)$$

A regra para inferir se i e j são vizinhos ou não, pode variar de acordo com cada algoritmo proposto. Mas utiliza-se o modelo descrito na Equação 4.1, onde define-se $W_{i,j}$ como 0 para representar que os vértices não estão conectados ou $w_{(e)}$ para representar a conexão entre os nós, podendo de acordo com o algoritmo utilizado ser considerado um peso para esta ligação ou ainda o valor 1 se o modelo não trabalhar com pesos nas arestas (CHAPELLE; SCHÖLKOPF; ZIEN, 2006).

Alguns exemplos de modelos de aprendizado semi-supervisionado são os métodos de Propagação de Rótulos, Consistência Global e Local, Caminhadas Aleatórias e Competição e Cooperação de Partículas.

Nos algoritmos de Propagação de Rótulos⁵ o objetivo é propagar o rótulo de um nó para seus vizinhos de acordo com sua proximidade. Os nós rotulados são fonte de informação para nós não rotulados (ZHU; GHAMRANI, 2002).

Neste modelo, os nós possuem pesos que representam a similaridade entre os mesmos, assim quanto maior for o peso da aresta entre dois nós, maior a sua similaridade. A probabilidade de propagação de um rótulo de um nó para outro é determinada pela diferença dos pesos entre os nós, sendo a probabilidade representada por T_{ij} .

O modelo realizará iterações propagando os rótulos entre os nós de acordo com o valor de T e finalizará quando todos os nós estiverem classificados.

O método de Consistência Local e Global é um método similar ao anterior, foi proposto por Zhou et al. (2004), onde em cada iteração do algoritmo o nó recebe contribuição de seus vizinhos, ponderada pelos pesos das arestas, e uma pequena contribuição dada pelos valores iniciais.

Ao final, os nós que não receberam nenhum rótulo serão classificados com o rótulo que foi propagado para o maior número de nós.

Outro método de propagação de rótulos é aquele que utiliza de modelos de Caminhadas Aleatórias, conforme proposto por Szummer e Jaakkola (2002).

Este modelo a partir de um grafo gerado conforme a Equação 4.1, trata da probabilidade de transição de um vértice i para um vértice j , dada pela equação 4.2 que é

⁵ Do Inglês: *Label Propagation*

usada para estimar as probabilidades dos rótulos das classes.

$$P_{ij} = \frac{W_{ij}}{\sum_k W_{ik}} \quad (4.2)$$

De acordo com Zhu (2008), os métodos de aprendizado semi-supervisionado baseados em grafos ainda possuem uma complexidade computacional alta ($O(n^3)$), sendo aplicáveis apenas em bases de dados pequenas.

O modelo de competição e cooperação de partículas é o método que será utilizado neste trabalho de pesquisa, assim será abordado de forma mais detalhada no próximo capítulo.

5 Modelo de Competição e Cooperação de Partículas

O modelo de movimentação de partículas foi desenvolvido com o objetivo de agrupar dados utilizando aprendizado não supervisionado (QUILES et al., 2008). Este modelo foi estendido para o paradigma de aprendizado semi-supervisionado utilizando competição de partículas baseando-se de parte de dados rotulados e dados não rotulados como já foi abordado anteriormente neste trabalho.

Os métodos de aprendizado semi-supervisionado baseados em grafos possuem uma boa identificação de classes, mas são custosos computacionalmente, na ordem de $O(n^3)$, limitando sua utilização a pequenas bases (BREVE, 2015; BREVE et al., 2012; BREVE, 2010).

O modelo de competição e cooperação de partículas se diferencia dos demais métodos de aprendizado semi-supervisionado baseado em grafos, pois estes propagam rótulos de maneira global de acordo com o peso das arestas, enquanto o modelo de partículas realiza uma abordagem local alterando apenas um nó de interesse a cada movimento (BREVE, 2015; BREVE et al., 2012).

Neste modelo, as partículas caminham em uma rede e competem entre si de forma que cada uma tente obter para si a maior quantidade possível de nós, ao mesmo tempo tentam evitar que outras partículas invadam seu território, terminando confinadas dentro de uma comunidade na rede (BREVE et al., 2012; BREVE, 2010).

O algoritmo utiliza-se de um conjunto de dados $X = x_1, x_2, x_3, \dots, x_n \subset R^m$ a ser utilizado e transformado em uma rede sem pesos e não direcionada, e um conjunto de dados de rótulos $L = 1, 2, \dots, c$ onde estes rótulos são escolhidos para que sejam anexadas as marcações de cada um dos times (BREVE et al., 2012).

A partir destas informações é montada uma rede através de um grafo $G = (V, E)$, $V = v_1, v_2, \dots, v_n$ onde cada nó v_i representa um elemento da rede e, através de uma matriz de adjacência, W são definidos quais nós estão interconectados, onde W_{ij} especifica se há uma aresta entre os nós x_i e x_j , sendo representada pela fórmula 5.1:

$$W_{ij} = \begin{cases} 1 & \text{se } \|x_i - x_j\|^2 \leq M \text{ e } i \neq j \\ 0 & \text{se } \|x_i - x_j\|^2 > M \text{ e } i = j \end{cases} \quad (5.1)$$

onde M define a distância máxima permitida para que dois pontos estabeleçam uma conexão, retornando o valor 1 no caso dos pontos se conectarem ou 0 caso o contrário ocorra. A distância é definida pelo cálculo da distância Euclidiana (BREVE et al., 2012; BREVE, 2010).

Uma segunda forma de atribuir a existência de vizinhança é através dos K -vizinhos mais próximos, onde se x_j está entre os k vizinhos mais próximos de x_i , mesmo que o inverso não ocorra, x_i é inserido na lista de vizinhos mais próximos de x_j , representado na Equação 5.2.

$$W_{ij} = \begin{cases} 1 & \text{se } x_j \text{ está entre os } k \text{ vizinhos mais próximos de } x_i \text{ ou vice-versa} \\ 0 & \text{caso contrario} \end{cases} \quad (5.2)$$

Após a montagem da matriz de adjacência é verificado o conjunto de dados rotulados, cada dado rotulado possui um nó correspondente na rede e será uma partícula representada por $P = (p_1, p_2, \dots, p_c)$, portanto, o número de partículas é igual ao número de dados rotulados. Cada partícula tem sua posição inicial definida como v_i , ou seja, o seu nó corresponde na rede. Este nó será nomeado como o nó-casa da partícula p_i (BREVE et al., 2012).

Cada partícula muda de posição a cada iteração e a sua distância do seu nó-casa é registrada. Cada sub-conjunto de partículas geradas a partir de amostras com um mesmo rótulo forma um time, colaborando entre si e competindo com outros times (BREVE et al., 2012; BREVE, 2010).

No modelo existem dois tipos de dinâmicas, a dinâmica de partículas e a dinâmica de nós. Cada partícula p_j possui duas variáveis: $p_j^w(t)$ e $p_j^d(t)$.

A primeira variável $p_j^w(t) \in [0, 1]$ é a força da partícula que caracteriza o quanto a partícula pode dominar o nó no tempo t .

A segunda variável é um vetor de distâncias $p_j^d(t) = \{p_j^{d_1}(t), p_j^{d_2}(t), \dots, p_j^{d_n}(t)\}$, onde cada elemento $p_j^{d_i}(t) \in [0, n-1]$ corresponde à distância medida entre o nó-casa da partícula p_j e o nó v_i .

Cada nó v_i tem uma variável $v_j^w(t) = \{v_j^{w_1}(t), v_j^{w_2}(t), \dots, v_j^{w_n}(t)\}$ onde cada elemento $v_j^w(t) \in [0, 1]$ representa o nível de dominação do time l sobre o nó. A soma desse vetor é sempre uma constante,

$$\sum_{l=1}^e v_i^{w_l} = 1, \quad (5.3)$$

isto ocorre porque quando uma partícula aumenta o seu nível de domínio sobre o nó, este domínio diminui na mesma proporção para os outros times.

Os níveis iniciais do vetor domínio v_i^w de cada nó v_i é configurado como:

$$W_i^{wl} = \begin{cases} 1 & \text{se } y_i = l \\ 0 & \text{se } y_i \neq l \text{ e } y_i \in L \\ \frac{1}{c} & \text{se } y_i = 0 \end{cases} \quad (5.4)$$

deste modo cada nó correspondente ao rótulo é dado o valor mais alto 1 caso contrário 0, para nó correspondente a uma amostra não rotulada o nível de domínio é configurado como $\frac{1}{c}$, onde c é o número de classes ou times de partículas, conforme pode ser visualizado na Figura 5.1.

Durante a execução do algoritmo, as partículas caminham pela rede aumentando o nível de domínio do seu time sobre o nó ao mesmo tempo que diminuem o nível de domínio de outros time, mas o somatório será sempre constante.

As partículas durante a execução do algoritmo também tem alterações em seu nível de força, aumentando quando visita um nó dominado pelo seu time e diminuindo quando visita um nó dominado por outro time. Esta informação é importante porque a mudança no domínio do nó que a partícula causa é proporcional ao seu nível de força (BREVE et al., 2012; BREVE, 2010).

A posição inicial de cada partícula é configurada como $p_i^w(0) = 1$, ou seja, força máxima.

$$p_j^w(0) = 1 \quad (5.5)$$

A tabela de distâncias é inicialmente configurada conforme a Equação 5.6. Esta equação demonstra que as partículas conhecem apenas as distâncias dos nós já visitados ou que tiveram como alvo, desta forma assume-se que a maior distância possível é $n - 1$, exceto para o nó-casa que é zero desde o início.

$$p_j^{d_i}(t) = \begin{cases} 0 & \text{se } i = j \\ n - 1 & \text{se } i \neq j \end{cases} \quad (5.6)$$

Durante as iterações da execução do algoritmo os valores de domínio dos times de partículas nos nós variam de acordo com a Equação 5.7.

$$v_i^{w_l}(t+1) = \begin{cases} \max \left\{ 0, v_i^{w_l}(t) - \frac{\Delta_v p_j^w(t)}{c-1} \right\} & \text{se } y_i = \emptyset \text{ e } l \neq p_j^f \\ v_i^{w_l}(t) + \sum_{q \neq c} v_i^{w_q}(t) - v_i^{w_q}(t+1) & \text{se } y_i = \emptyset \text{ e } l = p_j^f \\ v_i^{w_l}(t+1) & \text{se } y_i \in L \end{cases} \quad (5.7)$$

Na equação 5.7 p_j^f representa o time da partícula p_j , onde cada partícula altera o nó visitado v_i incrementando o nível de dominação do time quando $(v_i^{w_l}(t), l = p_j^f)$, e decrementando este nível quando $(v_i^{w_l}(t), l \neq p_j^f)$, sempre respeitando a lei de conservação definida na Equação 5.3 (BREVE; QUILES; ZHAO, 2015a).

A força da partícula depende do nível de dominação pelo seu time do nó visitado, então a cada iteração a força da partícula $p_j^w(t)$ é atualizada de acordo com a equação:

$$p_j^w(t+1) = v_i^{w_l}(t+1) \quad (5.8)$$

onde, v_i é o nó alvo, e $l = p_j^f$, ou seja l é o rótulo do time da partícula p_j , conseqüentemente cada partícula p_j tem sua força p_j^f configurada com o valor de nível de domínio do seu time $w_i^{w_j}$ no nó v_i (BREVE; QUILES; ZHAO, 2015a; BREVE, 2010).

O modelo condiciona a força da partícula a um redutor (Δ_v), desta forma é possível controlar a velocidade com que as partículas irão dominar os vértices não rotulados.

A tabela de distância introduzida (Equação 5.6) em cada partícula tem como objetivo manter o controle da distância entre o "nó-casa" e o "nó-atual", evitando que as partículas se distanciem muito deixando suas vizinhanças desprotegidas para ataques de outras partículas.

A cada iteração do algoritmo, a tabela de distâncias entre nó-casa e o nó-atual é atualizada, assumindo-se que a partícula tem conhecimento limitado da rede, sabendo apenas o número de nós que foram efetivamente visitados por ela.

Deste modo, inicializa a distância entre o nó-casa e os demais nós com o valor $n - 1$, e a cada iteração t um nó v_i é visitado pela partícula p_j e tem seu valor de distância $p_{(j)}^{d(k)}(t)$ atualizado no vetor de distâncias com base na Equação 5.9 a partir da posição do nó anterior (BREVE et al., 2012).

$$p_j^{d_k}(t+1) = \begin{cases} p_j^{d_i}(t) + 1 & \text{se } p_j^{d_i}(t) + 1 < p_j^{d_k}(t) \\ p_j^{d_k}(t) & \text{caso contrário} \end{cases} \quad (5.9)$$

Deste modo toda vez que a partícula se move do nó-atual para o nó-alvo, ela checa a tabela de distâncias e, se a distância do nó-alvo é maior que a distância do nó atual, ela é incrementada com mais 1. Caso contrário, mantém a distância atual no vetor de distâncias, tendo a vantagem de usar apenas distâncias conhecidas não necessitando recalculá-las.

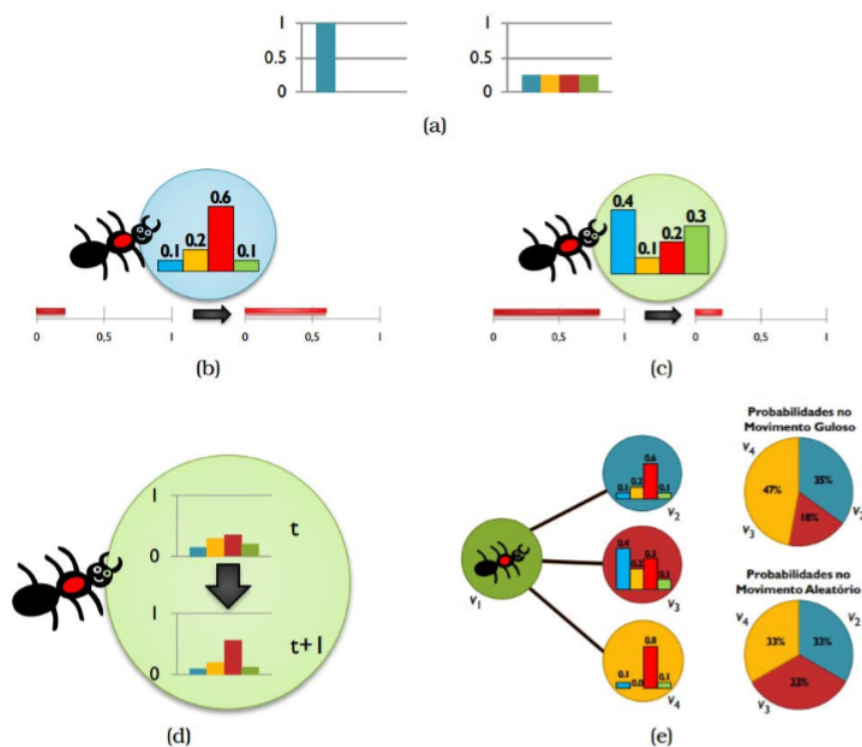
O processo descrito acima pode ser visualizado na Figura 5.1, onde em 5.1(a) pode-se observar os vetores de níveis de domínio inicial dos nós, o histograma à esquerda corresponde a uma amostra rotulada onde sua classe possui domínio total do nó, e no histograma à direita observa-se um nó não rotulado onde todas as classes possuem igual nível de domínio sobre o nó.

Na Figura 5.1(b) está representado o fortalecimento de uma partícula (força) ao visitar um nó que está sendo dominado pelo seu time. Na Figura 5.1(c) temos a situação oposta onde ocorre o enfraquecimento de uma partícula ao receber como objetivo um nó dominado por uma partícula de outra classe.

Se a partícula visitar um nó dominado por outra classe (partícula) e não conseguir dominar este vértice, será repelida para sua posição anterior e sofrerá uma redução de sua força inicial.

Na Figura 5.1(d) observa-se a situação onde uma partícula aumenta o seu domínio sobre um nó previamente dominado por sua classe, como já explicado, se a partícula visitar um nó dominado por outra classe e não conseguir dominá-la, será repelida para sua última posição.

Figura 5.1 – Ilustração da dinâmica de partículas e nós.



Fonte: Breve et al. (2012)

Na Figura 5.1(e) observa-se o processo de movimentação da partícula pela rede, onde temos descrito a probabilidade de serem escolhidos o movimento guloso e o movimento aleatório pela partícula; tal processo será descrito na próxima subseção do trabalho.

5.1 Movimento guloso ou aleatório

A cada iteração, a partícula deve escolher qual nó irá visitar entre os k -vizinhos mais próximos da sua posição atual. As partículas escolhem os nós a serem visitados com base em duas regras: regra aleatória ou regra gulosa.

Na regra aleatória, a partícula escolhe aleatoriamente, com iguais possibilidades, qualquer um dos nós vizinhos a posição em que a partícula se encontra.

Breve (2010) define que esta regra não se preocupa com os níveis de domínio ou distância do nó-casa, sendo útil para exploração e aquisição de novos nós, está definida pela Equação:

$$p(v_i|p_j) = \frac{W_{qi}}{\sum_{\mu=1}^n W_{q\mu}} \quad (5.10)$$

Na Equação 5.10, q é o índice do nó atual da partícula p_j , portanto $W_{qi} = 1$ se existe uma aresta entre o nó atual e qualquer nó v_i e $W_{qi} = 0$ caso contrário.

Na regra gulosa, a partícula escolhe aleatoriamente qualquer um dos nós vizinhos a partir do nó em que a partícula se encontra, com probabilidades calculadas de forma diretamente proporcionais ao nível de dominância do time dessa partícula em cada nó, e inversamente proporcionais à distância de cada nó ao nó-casa dessa partícula. Este movimento é útil para a defesa do território do time, sendo definida pela Equação 5.11.

$$p(v_i|p_j) = \frac{W_{qi}v_i^{w_l} (1 + p_j^{d_i})^{-2}}{\sum_{\mu=1}^n W_{q\mu}v_{\mu}^{w_l} (1 + p_j^{d_{\mu}})^{-2}} \quad (5.11)$$

Na Equação 5.11, q é o índice do nó atual da partícula p_j e $l = p_j^f$, onde p_j^f é o rótulo da classe da partícula p_j .

A cada iteração, cada partícula tem a probabilidade p_{grd} de escolher o movimento guloso e a probabilidade $1 - p_{grd}$ de escolher o movimento aleatório, com $0 \leq p_{grd} \leq 1$.

Deste modo, a partícula decide ficar em territórios já dominados e mais próximos a sua vizinhança, ou seja, uma atitude mais defensiva, quando escolher a regra gulosa. Por outro lado, se a regra aleatória for a escolhida, as partículas ignoram níveis de dominância e distância e tem a maior possibilidade de escolher nós que não dominam ou distantes da sua origem, em um comportamento de exploração.

Deve-se atentar para que a partícula só permaneça no nó escolhido se conseguir que seu time seja dominante, caso contrário, a partícula é expulsa e volta para o nó onde estava até a próxima iteração.

Esta regra evita que a partícula deixe seu território inutilmente, perdendo força sem conseguir provocar mudanças. Garante também que a partícula só poderá dominar um nó se tiver dominado um nó vizinho anteriormente.

O critério de parada acontece quando o algoritmo atinge um número máximo de iterações ($maxIte$), ou a partir do monitoramento dos domínios dos nós, o qual é feito analisando o comportamento das partículas.

Este monitoramento é realizado sob a média de conquistas das partículas, ou seja, quando estas não apresentarem melhoras no critério de conquistas em um determinado intervalo de iterações (*maxParada*), é entendido pelo algoritmo que já ocorreu a separação das classes.

Se houver alguma melhora na média de conquistas das partículas, *maxParada* é definido com o valor zero e reinicia a contagem até que atinja o limite definido pelo modelo. Considera-se como melhoria na conquista das partículas qualquer oscilação positiva na média do domínio dos vértices acima do valor definido pelo *controleParada*.

Normalmente essa parada ocorre após um grande número de iterações dentro da rede, na ordem de 10^5 , dependendo do tamanho da rede (quantidade de amostras) e da quantidade de partículas (quantidade de amostras rotuladas).

Abaixo podemos observar o algoritmo proposto por (BREVE et al., 2012).

Algoritmo 5.1– Aprendizado semi-supervisionado com cooperação e competição entre partículas

1 **início**

2 Construa uma matriz de adjacências W usando a Equação 5.1 ou a Equação 5.2;

3 Ajuste os níveis de domínio dos nós usando a Equação 5.4;

4 Ajuste as posições iniciais das partículas de acordo com seus respectivos nós-casa usando a Equação 5.5;

5 Ajuste a força das partículas e tabelas de distâncias usando a Equação 5.6;

6 **repita**

7 **para** cada partícula **faça**

8 Selecione a regra aleatória ou gulosa com as probabilidades definidas por P_{gd} ;

9 Selecione o nó alvo usando a Equação 5.10 ou a Equação 5.11 para movimento aleatório ou guloso respectivamente;

10 Atualize os níveis de domínio do nó alvo usando a Equação 5.7;

11 Atualize a força da partícula usando a Equação 5.8;

12 Atualize a tabela de distâncias das partículas usando a Equação 5.9;

13 **fim**

14 **até** até que o critério de parada seja satisfeito;

15 Rotule cada item de dado não rotulado pelo time com maior nível de domínio:
 $y_i = \arg \max_l v_i^{\omega_l}(\infty)$;

16 **fim**

5.2 Aprendizado Ativo aplicado modelo de competição e cooperação de partículas

Como melhoria, Breve (2013) apresenta a combinação do modelo de cooperação e competição entre partículas com a metodologia de aprendizado ativo ¹.

O aprendizado ativo é uma forma de aprendizado de máquina onde o algoritmo permite a interação do especialista ou outra fonte de informação, inserindo novas informações durante a execução do algoritmo.

Assim é possível a intervenção do especialista adicionando novos dados rotulados para correção dinâmica durante a execução do algoritmo sem prejuízo de haver reprocessamento de informações.

Este novo algoritmo precisa somente de um nó rotulado de cada classe onde será criada uma partícula para cada classe. Então novas partículas podem ser adicionadas dinamicamente sem precisar reiniciar o algoritmo (BREVE, 2013).

Neste trabalho, Breve (2013) apresenta duas abordagens para o novo algoritmo proposto. Na primeira, o algoritmo consulta pela incerteza, ou seja, na presença de um nó não rotulado duvidoso.

Na segunda abordagem, o algoritmo alterna entre a interrogação do nó de rede não rotulado mais duvidoso e a consulta de nó de rede sem rótulo mais distante de um nó rotulado. Esta versão trata de nós discrepantes, assim como também ajuda a evitar grandes regiões da rede sem nós rotulados.

O método proposto apresentou uma melhor precisão do que quando comparado a versão original do algoritmo com o mesmo número de nós rotulados. A primeira versão funciona melhor onde as fronteiras das classes não possuem muitas discrepâncias, enquanto a segunda abordagem funciona melhor em situações em que as fronteiras das classes não são tão bem definidas e há muitas discrepâncias. Não sendo possível afirmar qual das abordagens irá funcionar de forma mais eficiente em um conjunto de dados específico (BREVE, 2013).

5.3 Segmentação de Imagens através do modelo de Competição e Cooperação de Partículas

O modelo de competição de partículas é uma abordagem de aprendizado semi-supervisionado baseado em grafos, onde partículas movimentam-se na rede através de mecanismos de cooperação e competição a fim de rotular nós não rotulados, demarcando

¹ Do Inglês: *Active Learning*

territórios para sua classe.

Se diferencia dos demais modelos baseados em grafos devido a sua abordagem de propagação local ao invés de uma propagação global.

No modelo original do algoritmo, os dados vetoriais são transformados em uma rede complexa com base na distância Euclidiana entre os pares de amostras.

No caso de imagens, as redes são geradas com base na similaridades e diferenças entre características extraídas dos píxeis, sendo as imagens coloridas mais difíceis de segmentar.

Para este tipo de imagem é necessário extrair um número maior de informações do que em uma imagem em tons de cinza, e a variação das informações contidas em cada uma das variáveis a serem consideradas pelo modelo também é maior. Por exemplo, pode-se estabelecer a correlação entre os valores do modelo RGB de cada pixel utilizando a distância Euclidiana.

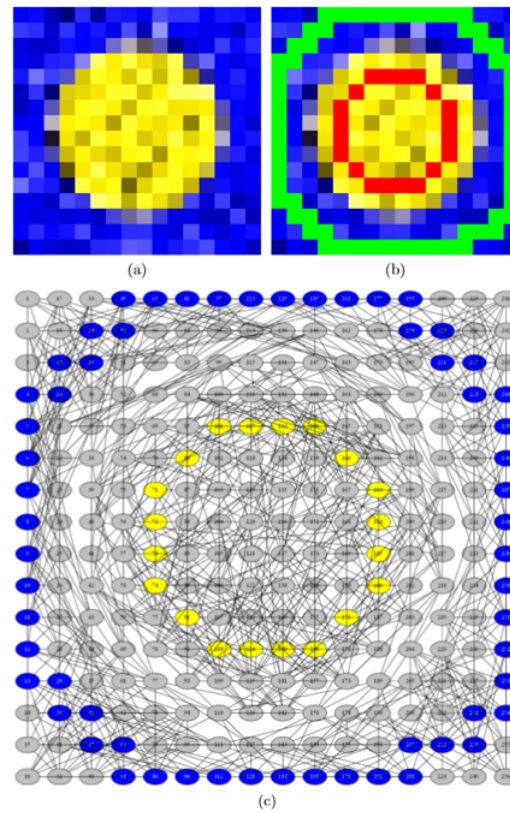
Como pode ser visualizado na Figura 5.2, Breve, Quiles e Zhao (2015a) determinam que dada uma imagem a ser segmentada, o conjunto de píxeis são organizados como $x = \{x_1, x_2, \dots, x_L, x_{L+1}, \dots, x_N\}$, e o conjunto de píxeis rotulados é representado como $x_L = \{x_i\}_{i=1}^L$, e o conjunto de píxeis não rotulados $X_U = \{x_i\}_{i=1}^N$, e $y : x \Rightarrow l$ é a função de associação, onde cada $x_i \in x$ com o rótulo $y(x_i)$ no final da segmentação.

Na Figura 5.2 demonstra-se o método de segmentação proposto, onde na Figura 5.2(a) tem-se a imagem original, na Figura 5.2(b) observa-se a imagem rotulada e na Figura 5.2(c) pode-se observar o grafo gerado com os vértices cinzas representando os píxeis não rotulados, os vértices azuis representando os píxeis do fundo e os vértices amarelos representando os píxeis do objeto a ser segmentado, conforme descrito na imagem rotulada (b).

Foram estabelecidas 20 características que serão extraídas de cada pixel da imagem durante o processo da formação do grafo pelo algoritmo, assim cada pixel x produzirá as características: posição horizontal e vertical do pixel, os três componentes do RGB, três componentes do HSV, média do *Red* nos píxeis adjacentes, média do *Green* nos píxeis adjacentes, média do *Blue* nos píxeis adjacentes, média do *Hue* nos píxeis adjacentes, média do *Saturation* nos píxeis adjacentes, média do *Value* ou brilho nos píxeis adjacentes, desvio padrão de cada componente do RGB com os píxeis adjacentes, desvio padrão de cada componente do HSV com seus píxeis adjacentes (BREVE; QUILES; ZHAO, 2015b).

Cada uma das características extraídas são normalizadas para representar um valor médio 0 e um desvio padrão de 1, em seguida, é escalado um vetor constante λ de pesos que tem a função de enfatizar ou desacentuar cada característica durante o processo de seleção de vizinhança para a geração do grafo (BREVE; QUILES; ZHAO, 2015b).

Figura 5.2 – Representação da geração da rede complexa pelo método de segmentação proposto. (a) Imagem original, (b) Imagem rotulada (c) Rede complexa com as partículas.



Fonte: Breve, Quiles e Zhao (2015a)

A vizinhança de cada vértice do grafo será composta pelos 8 elementos mais similares de acordo com a análise realizada através da distância euclidiana, Equação 5.12, do conjunto de características extraídas ponderada pelo vetor de pesos λ . Deste modo será construída a matriz de adjacências que será utilizada pelo algoritmo de competição e cooperação de partículas (BREVE; QUILES; ZHAO, 2015b).

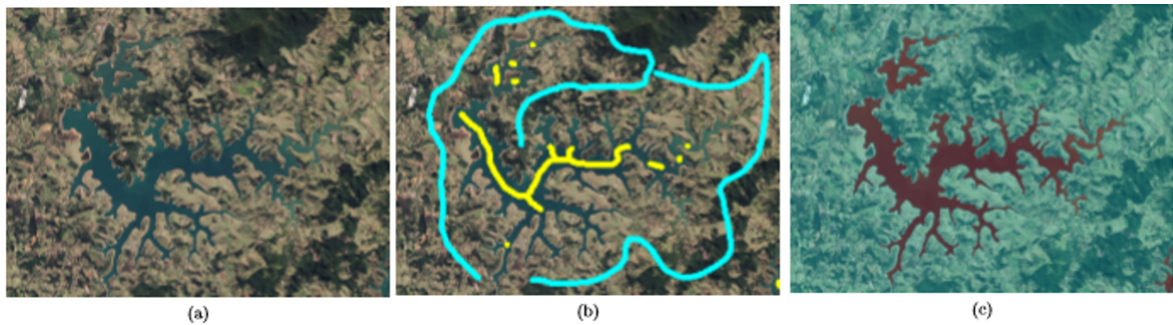
$$D(x, y) = \sqrt{(\lambda(x_1 - y_1))^2 + (\lambda(x_2 - y_2))^2 + \dots + (\lambda(x_n - y_n))^2} \quad (5.12)$$

A partir da montagem da matriz de adjacências, o algoritmo determina a matriz de distâncias de acordo com o descrito na Equação 5.9, define os vetores de partículas e sua força inicial conforme a Equação 5.5, assim como determina o nível de domínio nos vértices como descrito na Equação 5.4 e segue a execução conforme demonstrou-se no Algoritmo 5.1.

A seguir pode-se verificar os resultados dos trabalhos publicados, nas Figuras 5.3 e 5.4 observa-se os resultados do algoritmo, sem a utilização do vetor de pesos λ , conforme

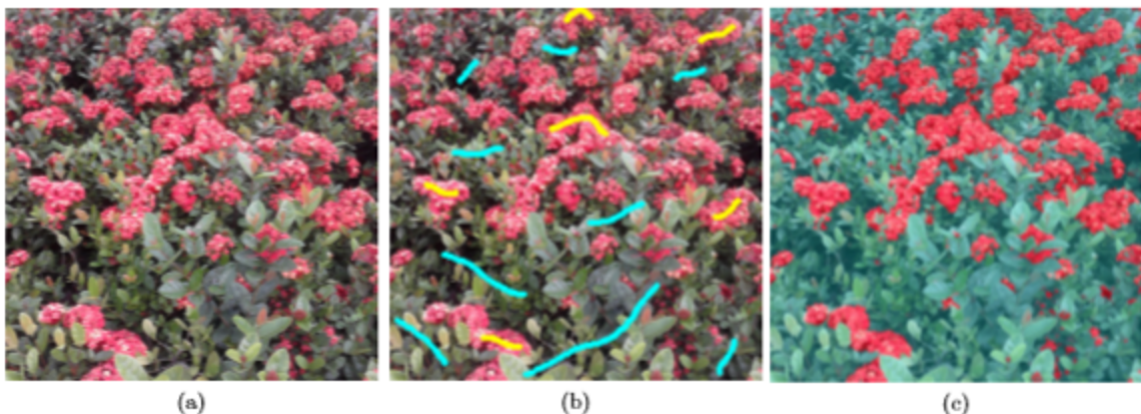
descrito em Breve, Quiles e Zhao (2015a).

Figura 5.3 – Reserva Jaguari: a) Imagem Original b) Imagem com pixels rotulados c) Resultado da Segmentação



Fonte: Breve, Quiles e Zhao (2015a)

Figura 5.4 – Flores: a) Imagem Original b) Imagem com pixels rotulados c) Resultado da Segmentação



Fonte: Breve, Quiles e Zhao (2015a)

Os resultados deste trabalho utilizaram o parâmetro de número de vizinhos próximos fixado em $k = 100$, e, como teste, utilizou-se uma imagem de satélite Landsat (5.3) e outra imagem de algumas flores (5.4), ambas as imagens foram redimensionadas para a execução do algoritmo.

Breve, Quiles e Zhao (2015a) demonstram a eficiência em relação a imagens onde o objeto a ser segmentação não está contíguo como observado nas Figuras 5.3 e 5.4.

Breve, Quiles e Zhao (2015b) introduzem o vetor de pesos λ , como melhoria ao modelo. Este vetor é responsável por enfatizar ou diminuir a importância de uma característica para o processo de segmentação de imagens.

Nesta abordagem foi utilizado algoritmo genético para definir os parâmetros de número de vizinhos próximos k e e e otimizar o vetor de pesos λ . A função de aptidão do algoritmo genético é calculada utilizando como parâmetro a imagem *ground-truth* fornecida pela base *GrabCut* da Microsoft.

Como resultado obteve desempenho melhor em algumas imagens do que o algoritmo de estado da arte utilizado no trabalho.

Breve (2015) propõe melhorias no algoritmo em relação ao seu custo computacional utilizando apenas uma tabela de distâncias para cada classe de partículas, onde essas partículas compartilham a mesma tabela.

O trabalho também aumenta para 23 o conjunto de características a serem avaliadas, sendo esses novos componentes são ExR , ExG e ExB , definidos pela Equação 5.13.

$$\begin{aligned} ExR &= (2R - (G + B)) \\ ExG &= (2G - (R + B)) \\ ExB &= (2B - (G + R)). \end{aligned} \tag{5.13}$$

Breve (2015) observa em seu trabalho que quanto maior a quantidade de atributos, o desempenho de algoritmos de aprendizado de máquina é diminuído, assim é necessário escolher apenas um subconjunto de características que melhor discrimine as classe de interesse, ou mesmo reduzir a sua dimensionalidade.

Em processos de segmentação de imagens, a variedade e a complexidade das mesmas torna a escolha das características ideais ou a determinação de pesos do vetor λ uma tarefa não trivial. Assim um atributo, que pode ser fundamental para uma determinada imagem, pode ter pouca ou nenhuma capacidade de discriminar classes em outra. Deste modo, Breve (2015) propõe a utilização de quatro métodos para tentar determinar o peso dos atributos e melhor definir as características ideais para segmentação das imagens.

O primeiro método utilizado foi a diferença entre as médias intra-classe (DMI), conforme a Equação 5.14, que consiste em obter a média de valores de um atributo f entre os píxeis rotulados de cada classe separadamente e então verificar a diferença entre as médias pois estas serão bastante diferentes se a característica for importante para a segmentação.

$$w_f = \sum_{i|y_i=1} x_{i,f} - \sum_{i|y_i=2} x_{i,f} \tag{5.14}$$

O segundo processo estende o primeiro, as médias intra-classe divididas pela soma do desvio padrão intra-classe (DMI/SDPI), a hipótese é que atributos que discriminem

bem duas terão baixa variância intra-classe, conforme a Equação 5.15.

$$w_f = \frac{\sum_{i|y_i=1} x_{i,f} - \sum_{i|y_i=2} x_{i,f}}{\sigma_{f1} + \sigma_{f2}} \quad (5.15)$$

O terceiro e o quarto métodos baseiam-se em histogramas. A diferença de histogramas (DH) é calculada gerando o histograma de um atributo f para cada subconjunto de píxeis de uma mesma classe. O histograma terá 10 barras e a soma das barras é sempre 1, o peso é definido pela soma das diferenças absolutas de cada par de barras, partindo-se do conceito que bons atributos terão histogramas bem diferentes, conforme a Equação 5.16.

$$w_f = \sum_{b=1}^{10} |h_{b1} - h_{b2}| \quad (5.16)$$

O quarto método é a diferença entre histogramas acumulados, similar ao método anterior, porém utiliza o histograma acumulado, dado por:

$$a_{bc} = \sum_1^b h_{bc} \quad (5.17)$$

onde a_{bc} corresponde à barra b do histograma acumulado da classe c , e h_{bc} corresponde à barra b do histograma original da classe c , sendo o peso W_f determinado por:

$$w_f = \sum_{b=1}^{10} |a_{b1} - a_{b2}| \quad (5.18)$$

onde a_{b1} e a_{b2} correspondem às barras dos histogramas acumulados do atributo f nas classes 1 e 2 respectivamente.

Como resultado do estudo, Breve (2015) afirma que foi possível demonstrar a eficiência dos quatro métodos, pois as simulações determinaram um aumento da taxa de classificação correta dos píxeis, assim sendo, estas são abordagens promissoras, sem no entanto conseguir determinar um único método mais indicado.

Breve (2017) demonstra uma nova abordagem para o problema da definição de importância das características através da atribuição de pesos por meio do vetor λ . Neste trabalho, o processo de otimização dos pesos para cada característica extraída também é realizado através da utilização de um algoritmo genético.

Nesta abordagem, a função de aptidão do algoritmo genético não utiliza da imagem *ground-truth* como ocorria em Breve, Quiles e Zhao (2015b), onde no processo de segmentar imagens em situações reais, tais imagens *ground-truth* não estão disponíveis. A nova função de aptidão proposta utiliza a estrutura do grafo baseado no número de arestas geradas entre nós rotulados em redes candidatas.

Como resultado observou-se uma melhoria da taxa de erro na classificação dos píxeis das imagens trabalhadas.

Nos trabalhos estudados ficou demonstrada o quão complexa é a tarefa de determinar um algoritmo que realize o processo de segmentação de imagens e, principalmente, definir de forma autônoma sem a intervenção de um especialista o conjunto de características ideais para execução desta tarefa.

6 Metodologia

Neste capítulo é apresentada uma descrição detalhada sobre as alterações propostas para o modelo de segmentação de imagens utilizando o algoritmo de Competição e Cooperação de Partículas, tendo como foco a alteração no processo de montagem da rede complexa que será analisada pelo algoritmo.

O objetivo desta alteração é aumentar a automação do processo por meio da eliminação do vetor de pesos λ , melhorar os aspectos de independência do conjunto de características para o processo de segmentação e propor alterações nas ligações dos vértices no grafo.

Com o objetivo de comparar o desempenho do modelo proposto será construído um algoritmo testemunha baseado nos trabalhos descritos no capítulo 5.

6.1 Método Testemunha

Foi construído para a fase de testes um algoritmo baseado nos trabalhos publicados sobre a aplicação do modelo de competição e cooperação de partículas em segmentação de imagens, conforme descrito no capítulo 5, aqui denominado algoritmo testemunha.

O modelo possui duas fases, onde a primeira é a extração de características da imagem e a montagem da rede complexa, a segunda fase é a aplicação do algoritmo de competição e cooperação de partículas.

Para a execução da primeira fase o algoritmo utilizou-se das características das posições horizontal e vertical do pixel dentro da imagem, os componentes vermelho, verde e azul do sistema de cores RGB, a intensidade, matiz e a saturação do sistema de cores HSV, a média e o desvio padrão dos pixels adjacentes na imagem dos componentes dos modelos RGB e HSV e os componentes ExR, ExG e EXB, totalizando 23 características.

O algoritmo testemunha se diferencia do modelo descrito no capítulo 5 pela aplicação dos métodos de interpolação para a redução do tamanho da imagem a ser processada pelo modelo e a sua reconstrução ao final do processo. Esta abordagem será realizada utilizando o mesmo critério aplicado ao modelo proposto por esse trabalho.

Outra abordagem foi a padronização do vetor de pesos λ com valores iguais a 1 para cada uma das características extraídas pelo modelo, de forma a não apresentar nenhum ajuste durante os testes.

Na segunda fase é aplicado o algoritmo de competição e cooperação de partículas sem alterações do modelo estudado anteriormente neste trabalho.

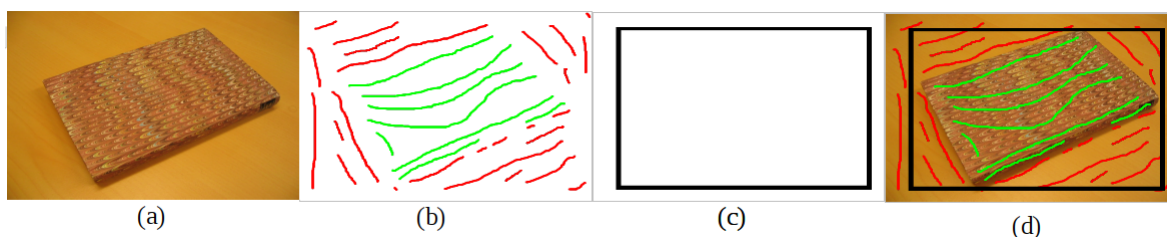
Na finalização do processo de segmentação a imagem será reconstruída e comparada a base de imagens *GrabCut* para a aferição de seu desempenho, conforme será descrito na próxima seção.

6.2 Estruturação do Método Proposto e Justificativas

O método proposto neste trabalho tem como objetivo promover melhorias no modelo de aprendizado semi-supervisionado de Competição e Cooperação entre Partículas aplicado à segmentação de imagens, permitindo uma maior automação do processo e reduzindo a dependência da interferência de especialistas para ajuste da execução do modelo, a partir de alterações no processo de construção da rede complexa.

A Figura 6.1 apresenta as informações de entrada para o algoritmo proposto.

Figura 6.1 – Informações de Entrada: (a) Imagem Original, (b) Marcação de píxeis rotulados, (c) Polígono de corte e (d) Sobreposição de informações para visualização



Fonte: Elaborado pelo autor

O modelo proposto recebe obrigatoriamente a imagem original a ser segmentada 6.1(a) e a marcação dos rótulos efetuado pelo usuário diferenciando as classes a serem segmentadas Figura 6.1(b).

Opcionalmente o usuário ainda pode determinar a região que se encontra o objeto de interesse na imagem através de um polígono Figura 6.1(c), onde a partir desta informação será efetuado um corte na imagem original e somente a região demarcada pelo polígono será processada pelo modelo proposto.

Está informação é opcional pois existem situações que a demarcação da região onde se encontra o objeto de interesse será coincidente ou mesmo muito próxima ao tamanho original da imagem, como pode-se verificar na Figura 6.2, onde o gabarito determina o objeto de interesse para segmentação.

Na Figura 6.1(d) pode-se observar uma montagem com a sobreposição das informações de entrada a serem processadas pelo modelo.

Figura 6.2 – Informações de Entrada: (a) Imagem com utilização de corte, (b) Imagem sem utilização de corte



Fonte: Elaborado pelo autor

Independentemente da escolha de utilizar o polígono para o corte da região o processo restante de montagem da rede complexa ocorrerá sem alterações em relação a este fator.

Uma vez definida as imagens a serem processadas, as mesmas serão submetidas a um processo de interpolação para a redução de sua dimensão (número de píxeis) possibilitando o processamento de imagens maiores.

Os modelos de interpolação foram descritos no capítulo 2 e são amplamente utilizados como descrito por Kaur, Kaur e Kaur (2016), Dourado (2014), Patel (2013), Prajapati, Naik e Mehta (2012), Getreuer (2011), Pedrini e Schwartz (2007).

Não será considerada a técnica de interpolação por vizinho próximo pois é o algoritmo mais simples e gera um efeito de serrilhado na imagem.

Forma selecionados os métodos de interpolação bicúbica e bilinear. A interpolação bilinear gera uma suavização do efeito serrilhado enquanto a interpolação bicúbica também reduz o serrilhamento e não suaviza tanto a imagem interpolada.

Esta etapa consiste em testar os métodos de interpolação Bicúbica e bilinear. Para medir o desempenho das abordagens será executado o algoritmo testemunha 25 vezes utilizando cada um dos métodos de interpolação com cada imagem selecionada.

O teste verificará a redução do tamanho da imagem e, após o processamento, o seu retorno ao tamanho original terá como a abordagem de redução/retorno com a seguinte configuração de testes: Bicúbica/Bicúbica, Bilinear/Bilinear, Bicúbica/Bilinear e Bilinear/Bicubica.

O algoritmo foi configurado para utilizar k vizinhos mais próximos em $k = 200$,

este valor foi determinado de forma empírica observando durante os testes quais valores produziam bons resultados na relação de acurácia e tempo necessário para a segmentação. O modelo não utilizou vetor de pesos λ . O redutor de processamento definido como $\Delta_V = 0.1$ e fator de sorteio do movimento guloso ou aleatório definido como $P_{grd} = 0.5$, para essas variáveis foi mantido o apresentado em trabalhos anteriores. O controle de parada com máximo de iterações em $maxIte = 1.000.000$, número máximo de iterações sem melhorar a média de dominância dos vértices como $maxParada = 15.000$ e o fator de $controleParada = 0,001$ onde é definida qual é a diferença mínima de melhora considerada no domínio dos vértices para zerar a variável $maxParada = 0$, também foram determinados de forma empírica a partir de observações que geraram melhores resultados durante os testes.

Para a realização de testes foi utilizado um notebook Asus modelo S46C com configuração de processador Intel i7 modelo 3537U de 2 Ghz e 4 núcleos, essa máquina durante os testes possuía 16 GB de memória RAM, com sistema operacional Linux Deepin na versão 15.5 Desktop. Para o desenvolvimento dos modelos foi utilizada a linguagem *Python* na versão 2.7 utilizando as bibliotecas *math*, *numpy*, *skimage* e *scipy*. Os resultados dos testes estão demonstrados na Tabela 6.1.

Tabela 6.1 – Resultados Teste de Algoritmos de Interpolação

Abordagem	Imagem	Px Válidos	Px Cinza	Px Erro	% Erros	Desvio Padrão	Desvio Tempo	Desvio Padrão
Bicubica	86016	154401	954	2194	1,43%	0,10%	752,01	49,49
Bicubica	227092	154401	1398	3539	2,31%	0,28%	798,14	81,3
	326038	154401	1606	4053	2,65%	0,07%	803,31	76,45
	Média	154401	1319	3262	2,13%		784,49	
Bilinear	86016	154401	954	2922	1,90%	0,09%	762,93	61,88
Bilinear	227092	154401	1398	3432	2,24%	0,03%	759,14	62,7
	326038	154401	1606	3860	2,53%	0,02%	816,73	93,15
	Média	154401	1319	3405	2,21%		779,60	
Bicubica	86016	154401	954	2202	1,44%	0,11%	752,26	77,71
Bilinear	227092	154401	1398	3417	2,23%	0,26%	774,47	64,6
	326038	154401	1606	3868	2,51%	0,06%	781,42	75,66
	Média	154401	1319	3162	2,05%		769,38	
Bilinear	86016	154401	954	2249	1,47%	0,10%	743,52	97,63
Bicubica	227092	154401	1398	3429	2,24%	0,23%	778,43	54,82
	326038	154401	1606	3977	2,58%	0,09%	790,61	80,54
	Média	154401	1319	3218	2,08%		770,85	

Fonte: Elaborado pelo autor

Pode-se observar que as taxas de erro das abordagens estudadas nesta amostra estão muito próximos com uma ligeira vantagem para a abordagem que utilizou a interpolação bicubica/bilinear. Esta abordagem obteve uma taxa de erro de 2,05%. Em relação ao

tempo, a abordagem bicúbica/bilinear também obteve uma pequena vantagem em relação as outras abordagens estudadas. A partir disto iremos utilizar a abordagem de interpolação bicúbica/bilinear na execução do modelo proposto assim como no algoritmo testemunha.

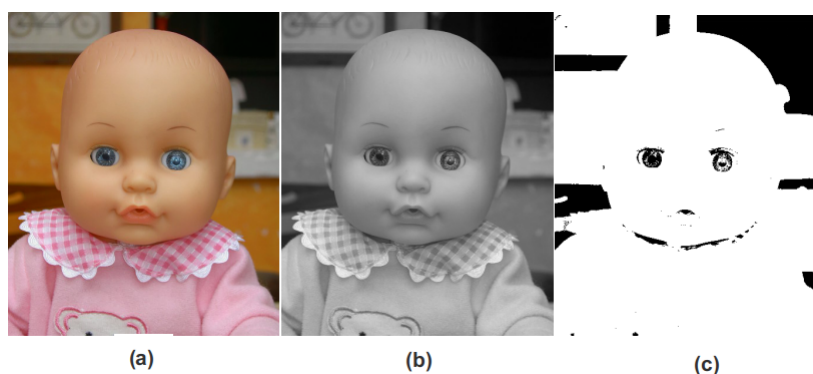
A montagem da rede complexa a partir do momento em que a imagem alvo da segmentação é interpolada, inicia-se o processo de extração do conjunto de características que serão utilizadas para a determinação dos vizinhos próximos.

Propõe-se a redução dessa dimensionalidade a partir da diminuição do número de características extraídas da imagem. Para o modelo proposto serão mantidas as coordenadas x e y referentes a posição do pixel na imagem, serão utilizados os componentes R (Vermelho), G (Verde) e B (Azul) do sistema de cores RGB, será utilizado apenas a intensidade (*value*) do sistema HSV e o componente de cores ExR, ExG, ExB, além de uma nova característica extraída a partir de uma binarização utilizando o algoritmo de Otsu.

Dentre os componentes do sistema HSV foi escolhido o componente de intensidade (*value*) devido ser dentre esses elementos o que representa a maior intensidade de absorção pelo sensor dentro do modelo RGB como é explanado no capítulo 2.

Uma nova característica foi adicionada ao modelo, a informação referente a uma imagem binarizada através do algoritmo de Otsu, que é amplamente utilizado, como descrito por Dass, Priyanka e Devi (2012), Gonzalez e Woods (2010), Skarbek e Koschan (1994). Na figura 6.3 pode-se observar o resultado do processo de limiarização na imagem *doll.png*.

Figura 6.3 – Processo de Limiarização - Otsu: a) Imagem Original b) Imagem em escala de cinza c) Imagem binária



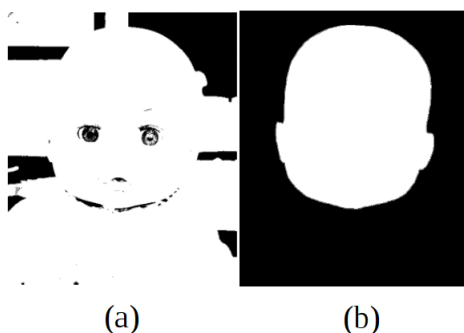
Fonte: Elaborado pelo autor

Para executar o processo de limiarização utilizando o modelo de Otsu, a imagem colorida foi inicialmente convertida para uma imagem em tons de cinza de acordo com a Equação 2.10 e, logo após, aplicado o processo de limiarização onde é definido um valor de $[0 - 255]$ para o limiar (*threshold*) da Imagem. A partir do valor do limiar, todos os píxeis

da imagem menores que este limiar são deslocados para o valor de 0 e os maiores para o valor 255.

A imagem limiarizada contribuirá com mais um elemento na análise para a formação da vizinhança dentro da rede complexa e não representará exatamente a segmentação da imagem, pois, como pode-se observar o gabarito da imagem *doll.png* (Figura 6.4(b)) difere da limiarização gerada pelo algoritmo de Otsu (Figura 6.4(a)).

Figura 6.4 – Comparação: a) Imagem limiarizada b) Gabarito Base *GrabCut*



Fonte: Elaborado pelo autor

Na figura 6.5 é possível visualizar todas as informações de entrada para o cálculo da vizinhança para a montagem do grafo, onde (a) é a imagem limiarizada pelo método de Otsu, (b) é o canal vermelho do RGB, (c) o canal verde do RGB, (d) o canal azul do RGB, (e) a intensidade do HSV, (f) o componente ExR, (g) o componente ExG e (h) o componente ExB.

As informações de entrada são normalizadas para ter uma média de 0 e um desvio padrão de 1. Então ocorre a geração do grafo não orientado que representará em seus vértices os píxeis da imagem e suas arestas as relações de semelhança entre esses vértices.

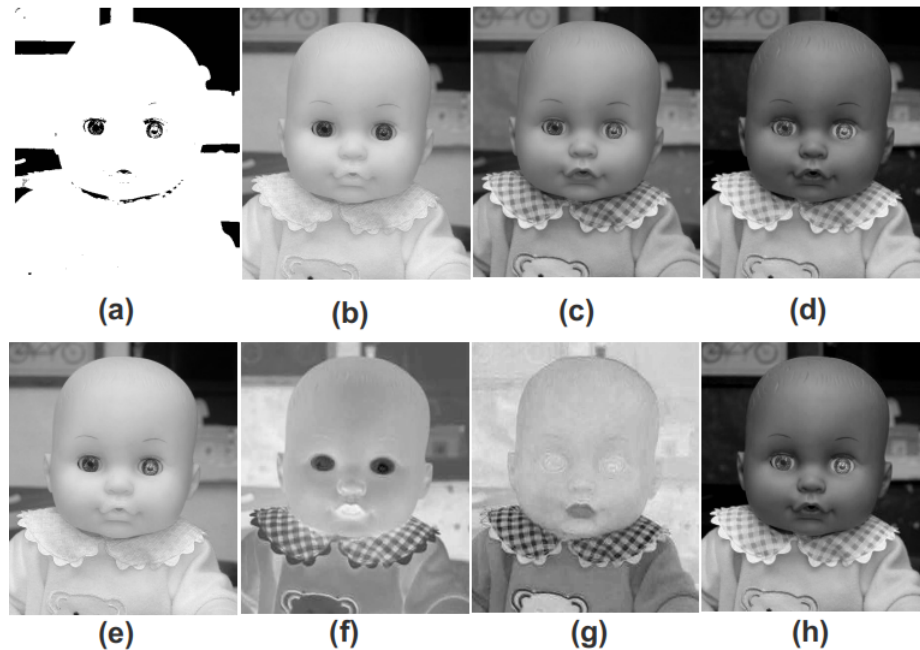
Como nas abordagens anteriores as relações de semelhança são determinadas através da aplicação da Distância Euclidiana entre as características extraídas da imagem a ser segmentada.

O modelo proposto utilizará $k = 200$ vizinhos, mas as ligações entre os vértices devem levar em consideração dois aspectos: o primeiro aspecto é a similaridade entre os vértices como já ocorre no algoritmo padrão através da distância Euclidiana e este processo determinará $k = 192$ vizinhos.

O segundo aspecto a ser considerado é a vizinhança física (espacial) do pixel na imagem, sendo que essa vizinhança é definida em uma janela 3×3 , ou seja, serão ligados os 8 vizinhos mais próximos fisicamente na imagem ao vértice processado.

Este processo visa garantir que os vizinhos físicos do vértice na imagem sejam

Figura 6.5 – Dados de Entrada para cálculo da vizinhança



Fonte: Elaborado pelo autor

selecionados, uma vez que não é possível garantir que isso ocorra pela primeira abordagem. Pode ocorrer também de um mesmo vértice ser selecionado pelas duas abordagens, mas no modelo só ocorrerá a ligação destes vértices apenas uma única vez.

A motivação para essa alteração demonstrar as relações de vizinhança física que ocorrem na imagem dentro da construção da rede complexa, valorizando-as uma vez que a vizinhança local do pixel na imagem tende a pertencer a mesma classe do pixel analisado.

Outra alteração em relação a abordagem anterior é que o modelo proposto verifica se os dois vértices ligados são de classes diferentes (através dos rótulos fornecidos pelos usuário). Se isto ocorrer, a nova aresta não será gerada. Vértices só podem ser interligados com vértices de mesma classe ou com vértices não rotulados.

Esta regra vale tanto para a vizinhança por similaridade como para a vizinhança definida pela distância espacial do pixel na imagem.

Uma vez construído o grafo que representa a imagem e as relações de similaridades entre seus píxeis, este trabalho propõe outra alteração aos modelos estudados, que é considerar a influência de um pixel previamente rotulado pelo usuário (partícula) em relação aos seus píxeis vizinhos na imagem. Está alteração também visa reforçar a informação da relação de vizinhança física entre os píxeis na imagem, só que agora em relação aos píxeis que foram rotulados pelo usuário, em uma janela 5×5 .

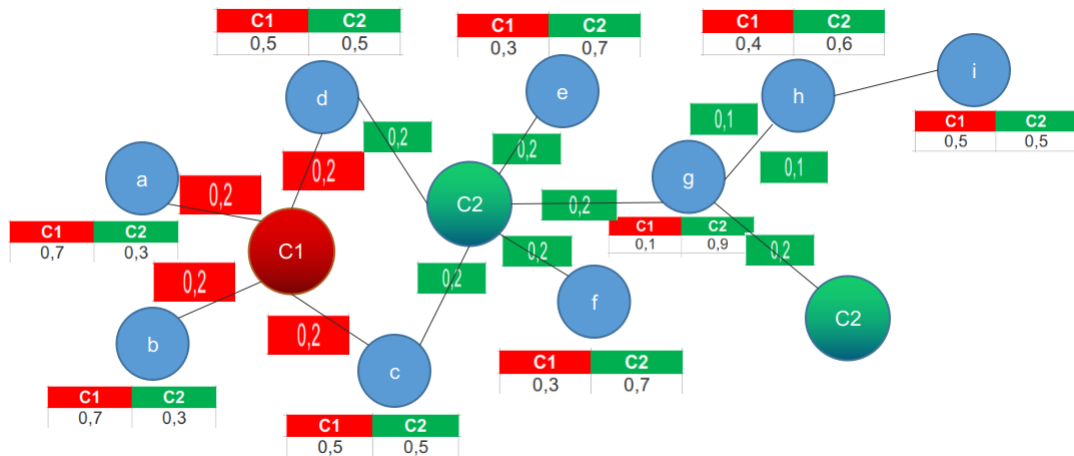
Como os píxeis rotulados são considerados partículas pelo modelo de competição e cooperação, deste modo levando em consideração uma abordagem inspirada na natureza onde uma partícula tende a gerar influência em outros elementos ao seu redor propõe-se representar esse processo dentro desta abordagem.

Considerar-se-a para isto que uma partícula (vértice rotulado) influencia outros vértices até uma distância igual a 2 dentro da estrutura do grafo. Deve-se enfatizar que, nesta abordagem, a distância novamente leva em consideração a distância espacial do pixel na imagem.

As alterações visam valorizar a relação da vizinhança espacial do pixel dentro da imagem a ser analisada pelo o modelo e determinar que um pixel não rotulado vizinho a um pixel rotulado tende a pertencer a mesma classe.

Assim, vértices com distância igual a 1, sofrerão incremento de 0,2 no seu vetor de domínio referente a classe da partícula que estiver próxima, inclusive pode sofrer essa influência de mais de uma partícula, com a possibilidade de ocasionar o domínio daquele vértice por uma determinada classe sem, no entanto, defini-lo como um novo rótulo. Para uma distância igual a 2, a influência é definida com um incremento de 0,1. Pode-se observar o modelo proposto na Figura 6.6.

Figura 6.6 – Modelo para influência de Partícula sobre a vizinhança



Fonte: Elaborado pelo autor

Conforme a Figura 6.6, cada partícula gera uma influência ao seu redor e cada vértice não rotulado (azul) possui seu vetor de domínios. Por exemplo, o vértice “g” está com um domínio na classe verde de 0,9 e na vermelha de 0,1, ou seja, foi praticamente dominado pela classe verde devido a influência de nível 1 de duas partículas.

O vértice “h” também sofreu a influência das mesmas duas partículas, só que em nível 2, desequilibrando seu vetor de domínios a favor da classe verde para 0,6.

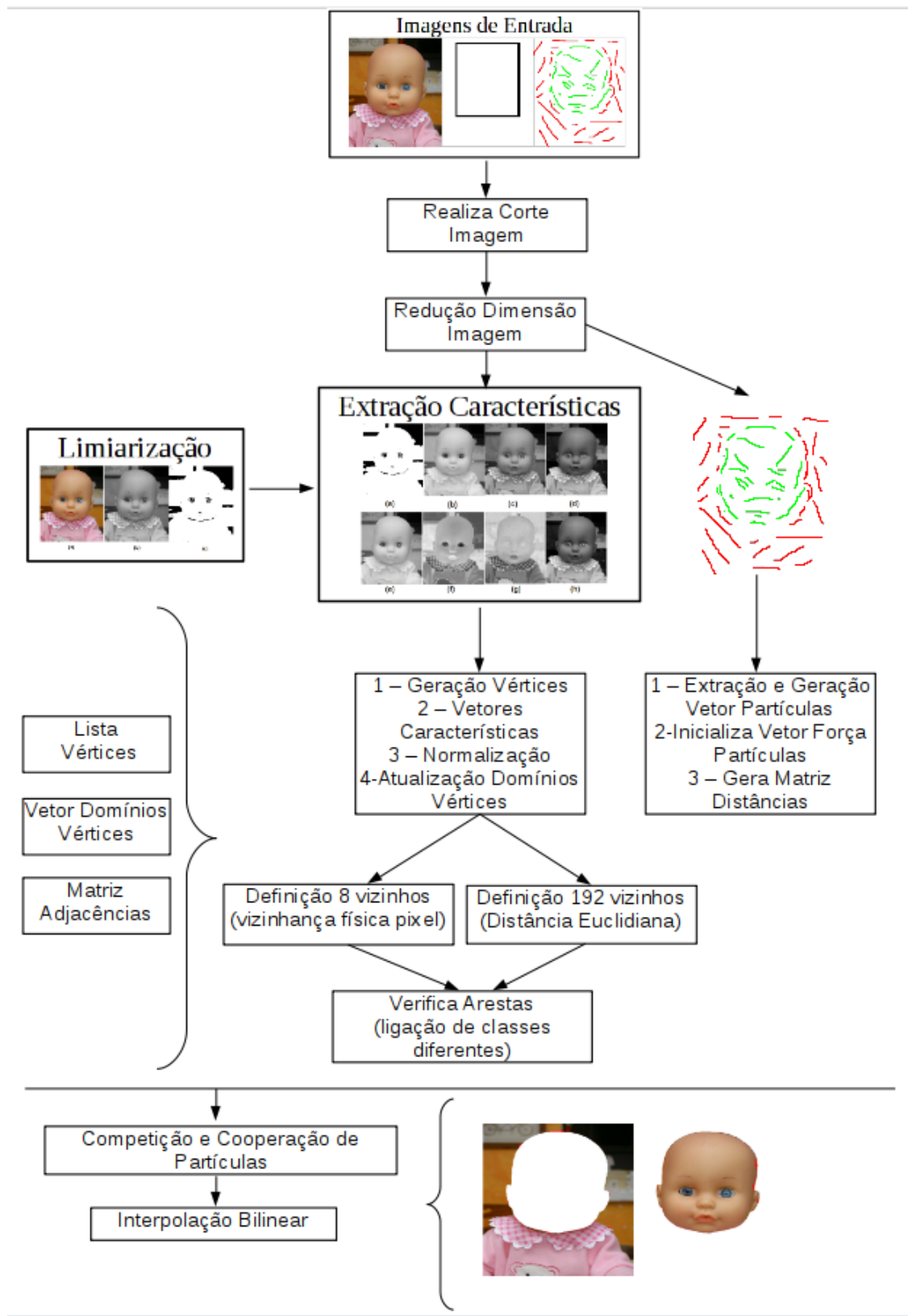
O vértice “c” continua sem uma classe dominante mesmo sofrendo influência de partículas, mas como houve influência de duas partículas de classes diferentes, essas se anulam mantendo o equilíbrio de forças no vértice. O vértice "i" está fora da zona de influência das partículas no modelo.

Os outros processos descritos em trabalhos anteriores em relação a construção da rede complexa serão realizados da mesma forma, como a geração da matriz de distâncias, os vetores de partículas, força de partícula e domínio dos vértices.

A Figura 6.7 resume todo o processo proposto neste capítulo como uma nova abordagem para geração da rede complexa a partir de uma imagem que será utilizada pelo algoritmo de Competição e Cooperação de Partículas para o processo de segmentação. Deve-se enfatizar que o Modelo de Competição e Cooperação de Partículas não sofreu qualquer alteração.

No capítulo 7 deste trabalho serão apresentados os resultados dos testes a partir da execução do algoritmo testemunha e do algoritmo que implementa as alterações aqui propostas.

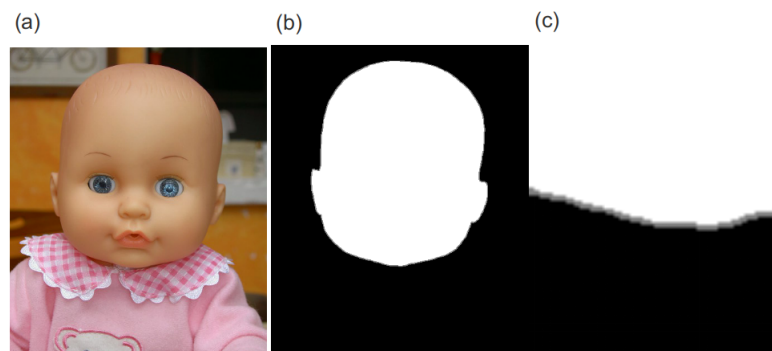
Figura 6.7 – Metodologia Proposta para geração do Grafo a partir de uma imagem



7 Resultados

Para a realização dos testes foram utilizadas imagens da Base *GrabCut* da Microsoft, que é uma base especializada para testes de algoritmos de segmentação. A base possui imagens que tiveram seus píxeis previamente classificados por vários especialistas e, desta forma, indicam três informações: pixel classificado como objeto (píxeis brancos) conforme demonstrado na Figura 7.1a, pixel classificado como fundo (píxeis pretos) que pode ser visualizado em 7.1b e píxeis onde houve divergência entre os especialistas, estando estes píxeis caracterizados na cor cinza, de acordo com o demonstrado na Figura 7.1c.

Figura 7.1 – Imagem Base Grabcut : (a) Imagem Original (b) Gabarito (c) Detalhe de píxeis cinza onde houve divergência de classificação



Fonte: Adaptado da base de imagens *Grabcut*

As imagens da base *Grabcut* foram rotuladas manualmente e submetidas ao algoritmo testemunha e ao algoritmo que implementa a nova abordagem proposta. A taxa de erro é determinada pela comparação da classificação de cada modelo com a imagem pré-classificada da base *Grabcut*, excluindo-se do cálculo os píxeis que causaram divergência entre os especialistas.

Os píxeis cinzas não são contabilizados nem como acerto e nem como erro do algoritmo, assim como são subtraídos do total de píxeis da imagem para efeito do cálculo da taxa de erro do modelo.

Para realizar os testes de desempenho será utilizado o algoritmo conforme descrito no capítulo 5 com todas as 23 características descritas e sem qualquer tipo de ajustes do vetor de pesos λ , que será inicializado com valor 1 para todas características, sem enfatizar ou minimizar nenhuma delas.

Ambos os modelos, o testemunha e o proposto pelo trabalho, serão configurados com os parâmetros ajustados para utilizar k vizinhos próximos em $k = 200$, não utilizar

vetor de pesos, redutor de processamento definido como $\Delta_v = 0,1$, fator de sorteio do movimento guloso ou aleatório como $Pgrd = 0,5$ e controle de parada com máximo de iterações em $maxIte = 1.000.000$, número máximo de iterações sem melhorar a média de dominância dos vértices como $maxParada = 15.000$ e o fator do $controleParada = 0,001$, onde é definida qual é a diferença mínima de melhora considerada no domínio dos vértices para zerar a variável $maxParada = 0$.

O critério de número de vizinhos próximos assim como os valores adotados como controle de parada do modelo foram definidos de forma empírica a partir de observações onde se obtinha um melhor resultado na relação de tempo de execução e acurácia na segmentação. Os parâmetros $Delta_v$ e $Pgrd$ foram definidos com valores já apresentados com eficientes em trabalhos anteriores.

Foram utilizados durante os testes as mesmas imagens para ambos os modelos, assim como o mesmo esquema de píxeis rotulados aplicados às imagens escolhidas.

Foram estudadas todas as 151 imagens da base apresentando as taxas de erro de classificação para os modelos estudados. Cada imagem foi executada 30 vezes pelo algoritmo proposto e 30 vezes pelo algoritmo testemunha visando demonstrar a melhora da taxa de erro média do modelo proposto em relação ao algoritmo testemunha sem ajustes no vetor de pesos λ . Deste modo, cada um dos modelos foi executado 4530 vezes. As marcações utilizadas para os testes de execução estão disponibilizadas no *GitHub*¹

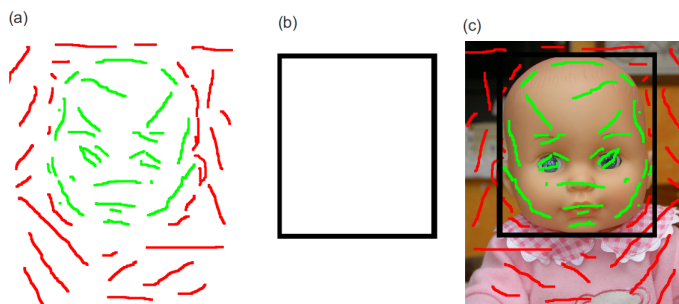
Para a realização dos testes foi utilizado um notebook Asus modelo S46C com configuração de processador Intel i7 modelo 3537U de 2 Ghz e 4 núcleos, essa máquina durante os testes possuía 16 GB de memória RAM, com sistema operacional Linux Deepin na versão 15.5 Desktop. Para o desenvolvimento dos modelos foi utilizada a linguagem *Python* na versão 2.7 utilizando as bibliotecas *math*, *numpy*, *skimage* e *scipy*.

Para todas as imagens foram gerados arquivos adicionais com rótulos e com a marcação de corte da imagem. O arquivo de rótulos possui as marcações efetuadas pelo usuário informando para o modelo que tais píxeis são do objeto a ser extraído (rótulos verdes) ou do fundo da imagem (rótulo vermelho). Para a marcação de corte da imagem, o arquivo gerado possui um retângulo com bordas pretas que determinará uma área dentro da imagem onde o objeto de interesse se encontra, reduzindo o escopo de processamento do algoritmo.

Os rótulos são as marcações que o usuário realiza para gerar as amostras de frente e fundo da imagem, conforme pode ser visualizado na Figura 7.2a e a marcação retangular que pode ser visualizada na Figura 7.2b o usuário demonstra a região de interesse na imagem e na Figura 7.2c pode-se verificar uma montagem realiza das marcações sobre a imagem original, sendo esta visualização meramente ilustrativa.

¹ Base de marcações para testes, disponível em: <https://github.com/jeffersonarpasserini/dataset-interactive-algorithms.git>

Figura 7.2 – Exemplos de Rótulos Gerados pelo Usuário: (a) Rótulos de Frente e Fundo (b) Marcação da área de corte na imagem (c) Montagem dos rótulos sobre a imagem original para visualização



Fonte: Elaborado pelo autor

O modelo proposto se utiliza da marcação de corte da imagem, enquanto o modelo testemunha não possui esse recurso implementado. Assim no modelo proposto o algoritmo considera apenas as informações constantes do perímetro do retângulo informado pelo usuário, tanto para extração de informações dos píxeis, como para a verificação de píxeis rotulados.

Durante a execução dos testes verificou-se a impossibilidade de utilização do corte em cinco imagens na base analisada, devido ao tamanho ou formato do objeto a ser segmentado, conforme pode-se observar na Figura 7.3.

Assim nestas imagens o modelo proposto realizou o processamento sem a utilização do recurso de corte.

Pode-se verificar na Tabela 7.1 as características das imagens demonstradas em seu tamanho, número de píxeis cinza em seu gabarito e as informações das redes geradas a partir dessas imagens.

Tabela 7.1 – Características das redes geradas a partir de imagens sem o recurso de Corte

Nome Imagem	Pixel	Pixel	Partículas	Método Proposto		Testemunha	
	Original	Cinza		Vértices	Arestas	Vértices	Arestas
Baby_2007_006647	187500	7670	6659	16800	1796483	16800	2206608
cross	270000	1870	4206	24300	2883050	24300	3148781
gt02	419200	2229	6294	37680	4371763	37680	4772882
gt07	494400	6656	7598	44400	5247685	44400	5721712
gt13	476000	12429	5314	42720	5149011	42720	5536232
Média	369420	6171	6014	33180	3889598	33180	4277243

Fonte: Elaborado pelo autor

Figura 7.3 – Imagens sem utilização de marcação de corte



Fonte: Elaborado pelo autor

Observa-se que as redes geradas para o algoritmo testemunha e para o algoritmo proposto não possuem alterações em relações ao número de vértices pois não houve corte a ser realizado no modelo proposto, mas pode-se verificar que houve uma redução significativa no número de arestas encontradas na rede gerada pelo modelo proposto neste trabalho.

Na Tabela 7.2 é demonstrada a comparação da taxa de erro de classificação dos píxeis na segmentação das imagens que não tiveram a marcação de corte pelo modelo proposto e pelo modelo testemunha.

Tabela 7.2 – Taxa de erro dos modelos a partir de imagens sem o recurso de Corte

Nome Imagem	Pixel		Método Proposto			Método Testemunha		
	Original	Cinza	Px Erro	% Erro	Desvio Padrão	Px Erro	% Erro	Desvio Padrão
Baby_2007_006647	187500	7670	2102	1,17%	0,02%	8213	4,57%	0,07%
cross	270000	1870	1276	0,48%	0,03%	4796	1,79%	0,07%
gt02	419200	2229	2180	0,52%	0,02%	5282	1,27%	0,13%
gt07	494400	6656	1005	0,21%	0,03%	3127	0,64%	0,06%
gt13	476000	12429	4995	1,08%	0,06%	9797	2,11%	0,16%
Média	369420	6171	2312	0,64%		6243	1,72%	

Fonte: Elaborado pelo autor

Pode-se verificar que o método proposto obteve um melhor desempenho na taxa de classificação dos píxeis em comparação ao método testemunha, onde na média das imagens

o método proposto obteve uma taxa de erro de 0,64% e o método testemunha obteve uma taxa de erro de 1,72%. Observa-se também que na média dos testes realizados em cada uma das imagens a execução do método proposto apresenta um desvio padrão menor do que o método testemunha demonstrando uma maior estabilidade nos resultados obtidos na segmentação das amostras em relação ao método testemunha.

O resultado da segmentação das imagens analisadas pode ser visualizada na Figura 7.4, pode-se observar que quanto maior a diferença entre o objeto a ser segmentado e o fundo da imagem o desempenho dos modelos se aproximam como nas imagens gt02 e gt07 em situações contrárias a diferença na taxa de erro aumenta na comparação dos modelos.

Na Tabela 7.3 verifica-se uma pequena vantagem no tempo de execução médio deste conjunto de imagens do método proposto em relação ao método testemunha, apesar de existirem situações onde o método testemunha obteve um desempenho de tempo melhor em relação ao método proposto como imagem *cross*.

Tabela 7.3 – Tempo de execução dos modelos a partir de imagens sem o recurso de Corte

Nome Imagem	Px Original	Método Proposto		Método Testemunha	
		Tempo(seg)	D.Padrão	Tempo(seg)	D.Padrão
Baby_2007_006647	187500	615,65	53,28	1179,81	108,06
cross	270000	1238,05	238,78	1123,14	67,23
gt02	419200	1098,05	38,84	1146,92	68,42
gt07	494400	1567,95	20,95	1637,93	109,4
gt13	476000	1505,62	90,38	1767,62	337,78
Média	369420	1205,06		1371,08	

Fonte: Elaborado pelo autor

Na Tabela 7.4 pode-se observar as cinco imagens da base estudada onde obteve-se a melhor taxa de erro durante a segmentação a partir do método proposto. As cinco imagens durante o processamento pelo método proposto sofreram o corte da região de interesse selecionada pelo usuário, assim determinando redes diferentes para os métodos proposto e testemunha.

A rede complexa que foi processada pelo método proposto é menor em número de vértices, uma média de 2089 vértices nas cinco imagens selecionadas, enquanto para o método testemunha observa-se um número de vértices muito mais elevado com 16003 vértices.

O mesmo conseqüentemente ocorre com o número de arestas onde na rede complexa do método proposto temos uma média de 554445 arestas enquanto na rede do método testemunha observa-se um número médio de arestas de 2102821, o que determinará diferenças no tempo final de execução dos modelos.

Esse processo de redução de escopo de análise também influencia a segmentação na

Figura 7.4 – Resultado da segmentação das imagens sem utilização de marcação de corte



Fonte: Elaborado pelo autor

taxa de erro de classificação obtida. Como podemos observar na Tabela 7.5, onde pode-se verificar uma taxa de erro média na classificação das 5 melhores imagens processadas pelo modelo proposto de 0,08%, comparativamente a taxa de erro das mesma imagens no método testemunha é de 1,94%.

Observa-se também, assim como ocorreu com a primeira análise realizada com as imagens onde não houve o corte da região, uma maior estabilidade durante o processamento das amostras no método proposto. A variação do desvio padrão no método proposto é pequena dentro do conjunto de 30 análises realizadas para cada imagem, diferentemente

Tabela 7.4 – Características das redes geradas a partir das imagens com cinco melhores taxas de erro no método proposto

Nome Imagem	Pixel		Método Proposto			Método Testemunha		
	Original	Cinza	Part.	Vért.	Arestas	Part.	Vért.	Arestas
Monitor_2007_003011	187500	1614	949	1428	114506	5176	16800	2237507
Train_2007_004627	187500	1180	1199	2220	2023323	5276	16800	2213111
Car_2008_001716	143424	3028	2127	4028	373544	4468	12814	1649199
Monitor_2007_004193	187500	800	877	1287	117952	4119	16800	2180800
Person_2007_002639	187500	1507	803	1482	142950	3921	16800	2233489
Média	178685	1626	1191	2089	554455	4592	16003	2102821

Fonte: Elaborado pelo autor

Tabela 7.5 – Taxa de erro dos modelos a partir dos cinco melhores resultados do método proposto

Nome Imagem	Pixel		Método Proposto			Método Testemunha		
	Original	Cinza	Px	%	Desv.	Px	%	Desv.
			Erro	Erro	Padr.	Erro	Erro	Padr.
Monitor_2007_003011	187500	1614	29	0,02%	0,00%	2021	1,09%	0,12%
Train_2007_004627	187500	1180	162	0,09%	0,01%	1409	0,76%	0,02%
Car_2008_001716	143424	3028	143	0,10%	0,01%	3528	2,51%	0,10%
Monitor_2007_004193	187500	800	198	0,11%	0,00%	5601	3,00%	0,28%
Person_2007_002639	187500	1507	215	0,12%	0,00%	4592	2,47%	0,10%
Média	178685	1626	149	0,08%		3430	1,94%	

Fonte: Elaborado pelo autor

do método testemunha onde a variação no número de píxeis classificados de forma errônea é maior.

Na Figura 7.5 pode-se observar as imagens segmentadas pelos métodos proposto e testemunha, onde os píxeis em vermelho são os que foram classificados erroneamente pelos modelos estudados.

Na imagem *monitor_2007_003011* pode-se observar na análise da amostra testemunha que houve claramente um problema na classificação dos píxeis azulados da janela com os píxeis azulados do monitor, assim como, os píxeis escuros do monitor com faixas escuras da imagem próximo aos estofados, o que foi evitado na imagem processada pelo modelo proposto onde foi demarcada a área de interesse de processamento dentro da imagem.

Nas outras amostras da Figura 7.5 pode-se observar o mesmo ocorrido com a imagem *monitor_2007_003011*.

Figura 7.5 – Resultado da segmentação das cinco imagens com melhores taxas de erro do método proposto



Fonte: Elaborado pelo autor

Na Tabela 7.6 observa-se o tempo de execução para as 5 imagens do conjunto analisado, para os métodos proposto e testemunha, onde o método proposto executou a segmentação das imagens em um tempo médio de 255,46 segundos enquanto o método testemunha levou 823,30 segundos. O resultado de tempo esta diretamente relacionado ao

tamanho das redes analisadas por cada modelo conforme demonstrado na Tabela 7.4.

Tabela 7.6 – Tempo de execução dos modelos a partir dos cinco melhores resultados do método proposto

Nome Imagem	Pixel Original	Método Proposto		Método Testemunha	
		Tempo	D.Padrão	Tempo	D.Padrão
Monitor_2007_003011	187500	227,00	48,0	777,1	90,34
Train_2007_004627	187500	323,41	79,8	757,83	140,97
Car_2008_001716	143424	273,70	34,1	842,3	97,41
Monitor_2007_004193	187500	199,79	8,84	927,99	161,08
Person_2007_002639	187500	253,41	50,84	811,27	58,1
Média	178684,8	255,46		823,30	

Fonte: Elaborado pelo autor

A próxima análise relaciona o comparativo dos resultados das cinco melhores taxas de erro do modelo testemunha com os mesmos resultados do modelo proposto. Na Tabela 7.7 pode-se visualizar as informações das redes geradas para processamento dos modelos.

Novamente todas as cinco imagens selecionadas no modelo proposto tiveram corte da área de interesse selecionada por anotação do usuário.

Tabela 7.7 – Características das redes geradas a partir das imagens com cinco melhores taxas de erro no método testemunha

Nome Imagem	Pixel Original	Pixel Cinza	Método Proposto			Método Testemunha		
			Part.	Vért.	Arestas	Part.	Vért.	Arestas
person2	270000	1951	2720	7000	770573	6294	24300	3212419
book	307200	2566	4163	20664	2481056	4731	27648	3605820
aero_2007_000032	140500	2863	1426	2856	297608	4187	12600	1656481
fullmoon	154700	423	906	1680	184919	3038	13860	1516112
Plant_2007_004121	166000	1391	682	952	68462	4547	14850	1982433
Média	207680	1839	1979	6630	760524	4559	18652	2394653

Fonte: Elaborado pelo autor

Através deste processo observa-se que as redes processadas pelo método proposto são sensivelmente menores do que as redes processadas pelo método testemunha. A média do número de vértices processados pelo método proposto na amostra analisada é de 6630 vértices enquanto no método testemunha observa-se uma rede média de 18652 vértices.

A imagem *book* tem o tamanho de sua rede mais próxima entre os dois modelos devido a característica da imagem do objeto de interesse para segmentação ocupar uma grande parte de seu tamanho total.

O número de arestas também é bem menor se comparado os métodos. O método proposto trabalhou com uma rede com 760524 arestas enquanto o método testemunha trabalhou com 2394653 arestas.

Na Tabela 7.8 observa-se o desempenho da taxa de classificação dos píxeis para os dois métodos dentro do conjunto analisado.

Tabela 7.8 – Taxa de erro dos modelos a partir dos cinco melhores resultados do método testemunha

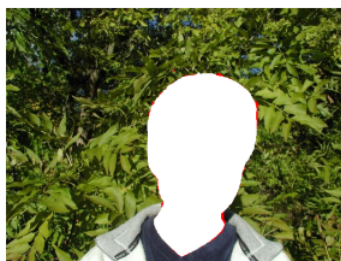
Nome Imagem	Pixel		Método Proposto			Método Testemunha		
			Px	%	Desv. Padr.	Px	%	Desv. Padr.
	Original	Cinza	Erro	Erro	Erro	Erro	Erro	Erro
person2	270000	1951	604	0,23%	0,01%	521	0,19%	0,01%
book	307200	2566	857	0,28%	0,02%	990	0,32%	0,02%
aero_2007_000032	140500	2863	190	0,14%	0,01%	449	0,33%	0,05%
fullmoon	154700	423	253	0,16%	0,02%	593	0,38%	0,01%
Plant_2007_004121	166000	1391	192	0,12%	0,00%	768	0,47%	0,18%
Média	207680	1839	419	0,20%		664	0,32%	

Fonte: Elaborado pelo autor

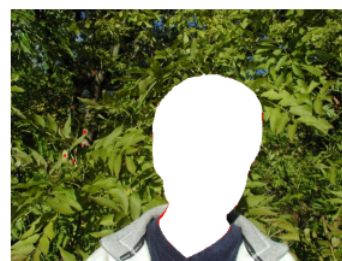
A média obtida pelo método proposto dentro do conjunto de processamento analisado é uma taxa de erro de 0,20% comparativamente ao método testemunha temos uma taxa de erro de 0,32%. Observa-se novamente a imagem *book* que tem uma taxa de erro muito próxima entre os métodos analisados, o método proposto apresenta uma taxa de erro de 0,28% enquanto o método testemunha uma taxa de erro de 0,32%.

Outro fato que merece destaque neste conjunto analisado é a imagem *person2* onde o método proposto obteve um desempenho ligeiramente pior do que o método testemunha, conforme pode-se visualizar na Figura 7.6.

Figura 7.6 – Imagem *person2* com melhor resultado no método testemunha



person2 - Proposto - Taxa Erro : 0,23%



person2 - Testemunha - Taxa Erro : 0,19%

Fonte: Elaborado pelo autor

Na Figura 7.6 não percebe-se diferenças entre a segmentação dos métodos proposto

e testemunha devido a taxa de erro muito próximo entre os métodos. A imagem *person2* foi a única imagem analisada entre as 151 imagens da base estudada onde o método testemunha obteve um desempenho melhor do que o método proposto.

Na Figura 7.7 pode-se verificar o resultado da segmentação das outras quatro imagens do conjunto estudado pelo método proposto e testemunha. A imagem *book* obteve uma taxa de erro de 0,28% para o método proposto e 0,32% para o método testemunha, a imagem *aero_2007_000032* o modelo proposto obteve uma taxa de erro de 0,14% contra 0,33% para o modelo testemunha, o mesmo ocorrendo para a imagem *fullmoon* onde a relação foi de 0,16% para o modelo proposto contra 0,38% da abordagem testemunha e a mesma situação é observada na imagem *plant_2007_004121* onde temos o modelo proposto com taxa de erro na classificação em 0,12% e o modelo testemunha com uma taxa de erro de 0,47%.

Nas outras imagens demonstradas na Figura 7.7 temos desempenhos de segmentação semelhantes, conforme as taxas de erro de classificação dos píxeis que foram verificadas na Tabela 7.8.

Nesta amostra das imagens com as cinco melhores taxas de erro do modelo testemunha, temos na Tabela 7.9 tempo de execução da segmentação pelos métodos proposto e testemunha. O método proposto por processar uma rede menor obteve um tempo médio de processamento para o conjunto das imagens de 383,95 segundos, enquanto o método testemunha obteve o tempo médio de 1144,39 segundos.

Tabela 7.9 – Tempo de execução dos modelos a partir dos cinco melhores resultados do método testemunha

Nome Imagem	Pixel Original	Método Proposto		Método Testemunha	
		Tempo	D.Padrão	Tempo	D.Padrão
person2	270000	588,06	96,62	1387,25	67,8
book	307200	677,41	79,5	1131,46	51,81
aero_2007_000032	140500	226,3	39,14	718,76	79,08
fullmoon	154700	298,05	58,66	1384,66	76,13
Plant_2007_004121	166000	129,94	2,64	1099,84	139,52
Média	207680	383,95		1144,39	

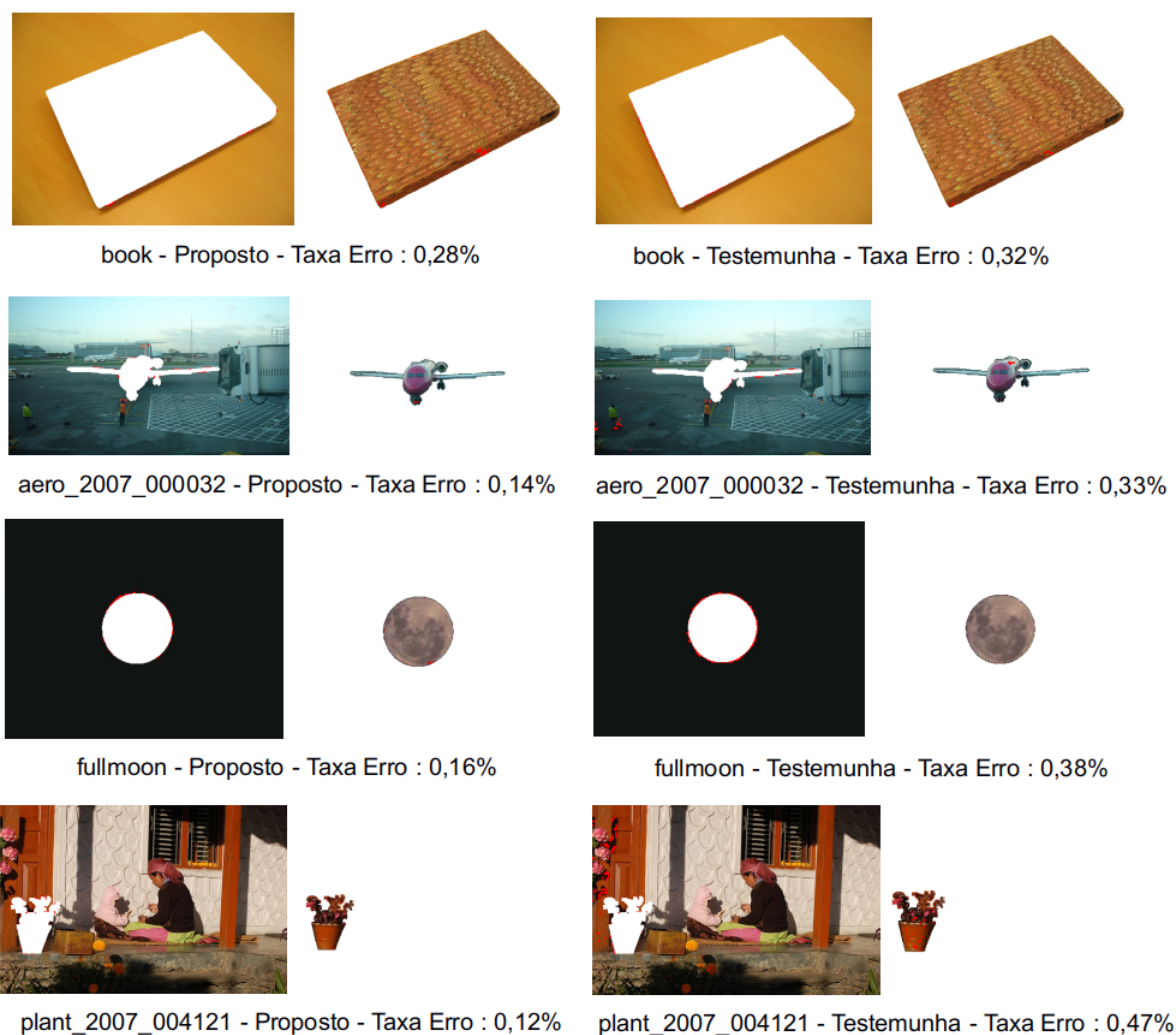
Fonte: Elaborado pelo autor

Na próxima análise o conjunto das cinco imagens estudadas serão as imagens com piores desempenhos de taxa de erro na classificação dos píxeis durante o processo de segmentação realizado pelo método proposto.

As cinco imagens geraram redes com características de vértices e arestas de acordo com que se apresenta na Tabela 7.10.

As imagens analisadas neste conjunto sofreram corte de acordo com a marcação da

Figura 7.7 – Resultado da segmentação das cinco imagens com melhores taxas de erro do método testemunha



Fonte: Elaborado pelo autor

área de interesse na imagem realizada pelo usuário.

Assim, pode-se verificar que as redes geradas pelo método proposto são de menores dimensões do que as redes geradas pelo método testemunha. A média de vértices no conjunto das cinco imagens analisadas para o método proposto é de 8125 vértices enquanto observa-se 13082 vértices para a rede gerada pelo método testemunha.

O mesmo ocorre com o número de arestas onde o método proposto produziu uma rede média com 909931 arestas enquanto o método testemunha gerou uma rede média para o grupo de imagens analisada de 1707808 arestas.

Na Tabela 7.11 observa-se a taxa de erro dos métodos proposto e testemunha para o conjunto de imagens caracterizados anteriormente.

Tabela 7.10 – Características das redes geradas a partir das imagens com cinco piores taxas de erro no método proposto

Nome Imagem	Pixel		Método Proposto			Método Testemunha		
	Original	Cinza	Part.	Vért.	Arestas	Part.	Vért.	Arestas
teddy	113032	0	1335	4674	556958	2351	10115	1261233
189080	154401	0	3630	10640	1194831	4274	13824	1767638
208001	154401	0	2344	5632	611561	4549	13824	1786864
69020	154401	2211	6524	12576	1418336	7091	13824	1864183
376043	154401	0	2453	7104	767971	3254	13824	1859124
Média	146127	442	3257	8125	909931	4304	13082	1707808

Fonte: Elaborado pelo autor

Tabela 7.11 – Taxa de erro dos modelos a partir dos cinco piores resultados do método proposto

Nome Imagem	Pixel		Método Proposto			Método Testemunha		
	Original	Cinza	Px	%	Desv.	Px	%	Desv.
			Erro	Erro	Padr.	Erro	Erro	Padr.
teddy	113032	0	1383	1,22%	0,00%	2600	2,30%	0,12%
189080	154401	0	1994	1,29%	0,15%	2693	1,74%	0,18%
208001	154401	0	1999	1,29%	0,04%	2179	1,41%	0,09%
69020	154401	2211	2240	1,47%	0,02%	3055	2,01%	0,06%
376043	154401	0	3244	2,10%	0,09%	7125	4,61%	0,36%
Média	146127	442	2172	1,49%		3530	2,42%	

Fonte: Elaborado pelo autor

O pior desempenho do método proposto foi observado na segmentação da imagem *376043* de todo o conjunto 151 imagens estudadas neste trabalho.

Nesta imagem temos uma taxa de erro na classificação dos píxeis de 2,10% um desempenho superior ao do método testemunha, como pode-se observar na Figura 7.8. Observa-se que o corte realizado na imagem pelo método proposto evita confusões entre píxeis do objeto a ser segmentado e o fundo da imagem, através da redução do escopo de trabalho do algoritmo.

Apesar de reduzir o problema de classificação errônea de píxeis com características próximas entre o fundo e o objeto através do corte da área de interesse, o método proposto, ainda que em menor proporção do que o método testemunha, gera erros de classificação deste tipo na imagem analisada.

Na imagem *208001* observa-se taxas de erro muito próximas entre os métodos, o método proposto obteve uma taxa de erro de 1,29% enquanto o método testemunha obteve uma taxa de erro de 1,41%, neste caso a área de corte auxiliou a melhora do resultado da classificação, mas existem áreas de píxeis que pertencem ao fundo com características

muito próximas aos píxeis do objeto em regiões anexas a este, o que ocasionou erros de classificação.

Nas imagens *69020* e *189080* o corte realizado pelo método proposto devido as características dos objetos a serem segmentados geram pouca interferência no resultado em relação ao método testemunha que utiliza-se da imagem completa para a realização da segmentação.

Observa-se que o método proposto conseguiu classificar com melhor eficácia os píxeis do objeto e do fundo em ambos os casos resultando em resultados melhores do que os apresentados pelo método testemunha.

Na imagem *teddy* apesar visualmente ter-se a impressão de características bem distintas entre os píxeis do objeto e do fundo, o método proposto ainda teve uma taxa de erro de classificação de 1,22%. Talvez essa taxa de classificação fosse melhorada se a imagem não tivesse passado pelo processo de interpolação inicial, como é uma imagem pequena tenha perdido características que permitiriam uma melhor classificação.

Em relação ao tempo de execução a construção de redes de menores dimensões pelo método proposto também representou ganho de tempo em relação ao método testemunha, conforme pode-se verificar na Tabela 7.12.

Tabela 7.12 – Tempo de execução dos modelos a partir dos cinco piores resultados do método proposto

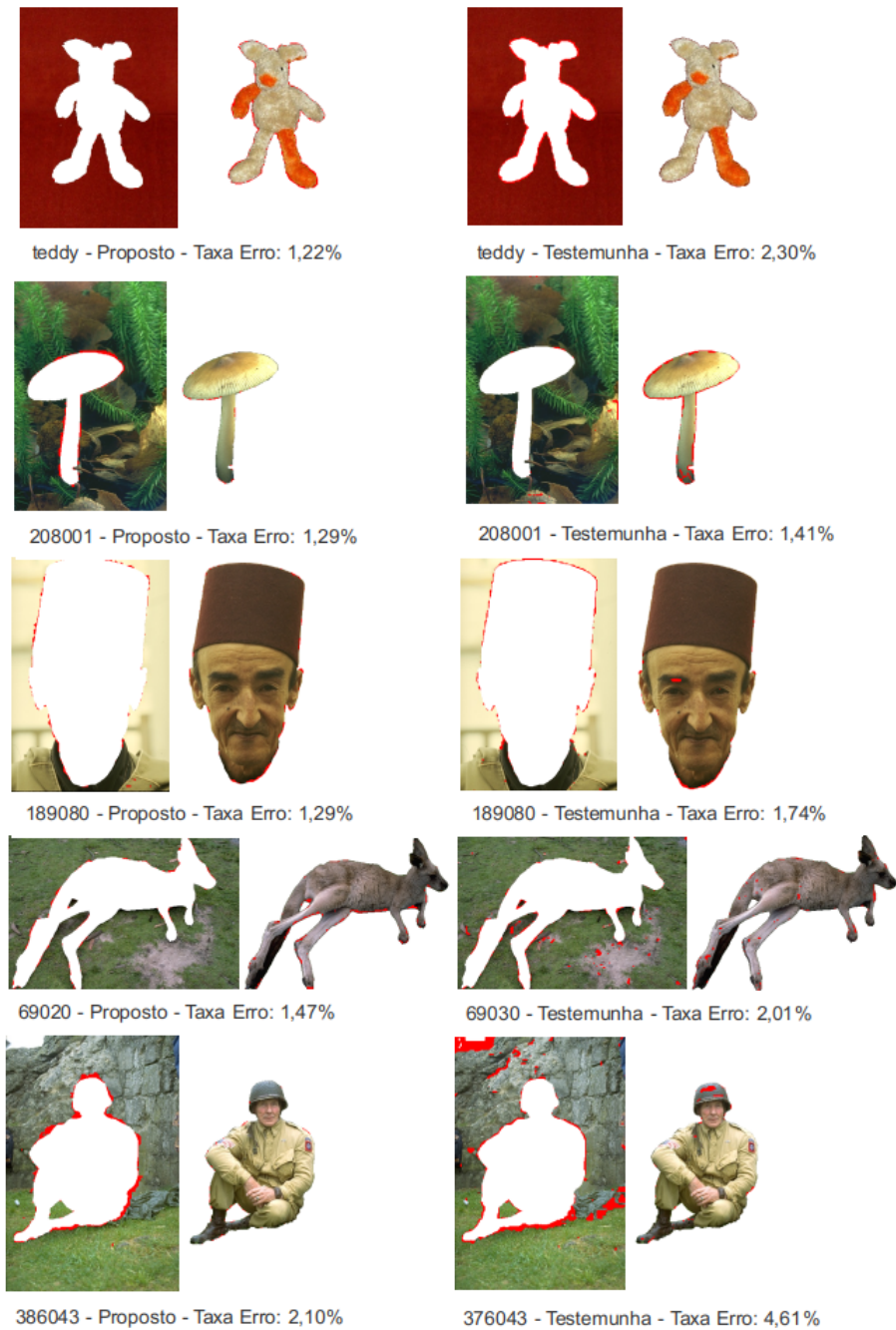
Nome Imagem	Pixel Original	Método Proposto		Método Testemunha	
		Tempo	D.Padrão	Tempo	D.Padrão
teddy	113032	571,4	124,79	583,3	74,55
189080	154401	537,23	157,8	838,11	169,39
208001	154401	407,56	132	756,87	65,5
69020	154401	472,67	18,87	1160,63	133,45
376043	154401	647,5	139,18	1068,84	112,95
Média	146127,2	527,27		881,55	

Fonte: Elaborado pelo autor

O método proposto conseguiu executar as segmentações para o conjunto analisado em um tempo médio de 527,27 segundos enquanto o método testemunha realizou a mesma tarefa em um tempo médio de 881,55 segundos. Desta forma, foi possível observar o melhor desempenho do método proposto em comparação ao método testemunha mesmo quando comparado ao seu pior desempenho.

A seguir iremos comparar o desempenho geral dos métodos proposto e testemunha em relação ao processamento das 30 amostras de cada uma das 151 imagens. Neste capítulo será demonstrado apenas os resultados médios de desempenho e tempo de execução, os resultados individuais de cada imagem podem ser verificados no apêndice deste trabalho.

Figura 7.8 – Resultado da segmentação das cinco imagens com melhores taxas de erro do método testemunha



Fonte: Elaborado pelo autor

O resultado individual de todas as imagens processadas podem ser visualizados na Tabela A.1 onde apresenta-se os dados das redes geradas a partir das imagens. Na Tabela B.1 pode-se verificar o resultado da taxa de erro da segmentação realizada pelos métodos estudados, assim como na Tabela B.2 temos os tempos de processamento da segmentação das imagens estudadas. Ambas as tabelas encontram-se no apêndice deste trabalho.

O desempenho geral dos métodos proposto e testemunha pode ser observado na Tabela 7.13.

Considerando a média do tamanho das imagens em número de píxeis da base analisada neste trabalho obtêm-se uma imagem média com 200124 píxeis, se considerarmos a utilização do recurso de corte implementado no modelo proposto a imagem média da base tem tamanho de 84320 píxeis, o que acarretará uma redução significativa do escopo de processamento pelo algoritmo.

Tabela 7.13 – Resultado da execução dos Métodos Proposto e Testemunha na Base Estudada

Método	Pixel	Pixel	Características			Taxa de Erro		
	Original	Cinza	Part.	Vért.	Arestas	Px.Err.	%Erro	Tempo
Proposto	200124	2783	2860	7538	838564	963	0,49%	432,54
Testemunha	200124	2783	5487	17946	2354555	5856	3,14%	1082,94

Fonte: Elaborado pelo autor

Na Tabela 7.13 pode-se visualizar o desempenho determinado pela taxa de erro de ambos os modelos. O método proposto obteve uma taxa de erro na classificação dos píxeis durante a segmentação de 0,49% enquanto o mesmo índice para o método testemunha ficou em 3,14%.

Observa-se também a diminuição do tempo de execução da segmentação, o método proposto obteve uma média de processamento de 432,64 segundos enquanto o método testemunha realizou a tarefa de segmentar as 151 imagens da base estudada com um tempo médio de 1082,94.

A redução do escopo de processamento através da demarcação da região de interesse na imagem original permitiu essa vantagem ao modelo proposto. Na Figura 7.9 pode-se observar a dispersão dos pontos de cada uma das 151 imagens com seus resultados de taxa de erro e tempo de execução para os métodos testemunha e proposto.

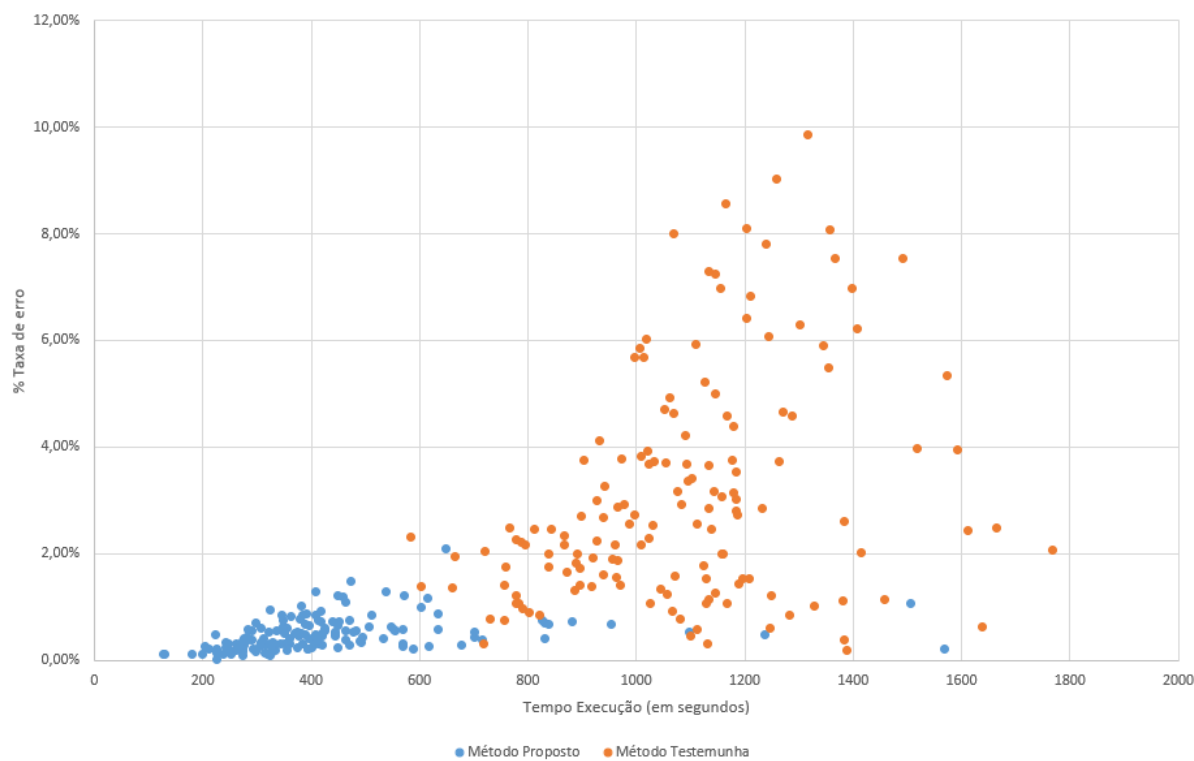
Nesta análise, observa-se que as imagens processadas pelo algoritmo que implementa o método proposto estão concentradas em uma região específica do gráfico apresentado, enquanto o resultado do processamento das imagens pelo método testemunha estão dispersas dentro do gráfico.

Este fato demonstra uma maior estabilidade na execução do modelo proposto em relação ao modelo testemunha, estando menos suscetível as variações de características entre as diferentes imagens apresentadas pela base estudada.

No gráfico também pode-se observar o melhor desempenho do método proposto sobre o método testemunha, assim como já havia sido apresentado na Tabela 7.13.

Em relação ao desempenho do modelo proposto quanto a percentual da taxa de

Figura 7.9 – Análise da relação % Taxa de Erro e Tempo de Processamento da base de imagens processadas - Método Proposto e Método Testemunha



Fonte: Elaborado pelo autor

erro pode-se observar na Figura 7.9 que apenas uma imagem ficou acima dos 2% de erro na classificação dos píxeis como foi demonstrada na Tabela 7.11 foi a imagem *376043.png* e que pode ser observada na Figura 7.8.

Quando é analisado o fator tempo de execução verifica-se que também uma concentração no desempenho do método proposto, neste quesito variando entre 200 e 500 segundos, enquanto no método testemunha os tempos de execução encontram-se mais dispersos pelo gráfico. Além de obter-se um melhor tempo médio de processamento da segmentação das imagens estudadas através do método proposto, neste observa-se também uma maior estabilidade em relação ao método testemunha.

Na Figura 7.9 observa-se a existência de algumas imagens que realizaram a segmentação utilizando o algoritmo proposto com tempo de execução acima de 800 segundos, a Tabela 7.14 demonstra as 5 imagens que se destacaram negativamente em relação ao tempo médio geral no estudo apresentado.

Dentre as imagens relacionadas a única que teve a marcação da área de interesse foi a imagem *banana1*, o que pode ser verificado comparando o número de píxeis da imagem entre original e corte. A seguir na Tabela 7.15 pode-se verificar o tempo de cada uma das

Tabela 7.14 – Características das redes geradas a partir da imagens com pior tempo de processamento pelo método proposto

Nome Imagem	Pixel Original	Pixel Corte	Pixel Cinza	Método Proposto			Método Testemunha		
				Part.	Vért.	Arestas	Part.	Vert.	Arestas
gt07	494400	494400	6656	7598	44400	5247685	7598	44400	5721712
gt13	476000	476000	12429	5314	42720	5149011	5314	42720	5536232
cross	270000	270000	1870	4206	24300	2883050	4206	24300	3148781
gt02	419200	419200	2229	6294	37680	4371763	6294	37680	4772882
banana1	307200	236925	2562	3987	21175	2434701	4812	27648	3547442

Fonte: Elaborado pelo autor

imagens processadas pelo método proposto e pelo método testemunha

Tabela 7.15 – Características das redes geradas a partir da imagens com pior tempo de processamento pelo método proposto

Nome Imagem	Pixel Original	Pixel Corte	Método Proposto		Método Testemunha	
			Tempo	D.Padrão	Tempo	D.Padrão
gt07	494400	494400	1567,95	20,95	1637,93	109,4
gt13	476000	476000	1505,62	90,38	1767,62	337,78
cross	270000	270000	1238,05	238,78	1123,14	67,23
gt02	419200	419200	1098,05	38,84	1146,92	68,42
banana1	307200	236925	953,00	182,00	1248,77	60,78
Média	393360		1272,53		1384,88	

Fonte: Elaborado pelo autor

Apesar das imagens apresentadas estarem fora do padrão médio de tempo de processamento utilizando o método proposto, se comparado ao método testemunha ainda observa-se um desempenho melhor. As imagens analisadas são as imagens que geraram grafos com o maior número de vértices de todas as imagens da base analisada.

Especificamente em relação as imagens *gt07*, *gt13* e *gt02* observa-se que são imagens que geraram redes complexas com o maior número de vértices, respectivamente 44400, 42720 e 37680. Da mesma forma essas imagens geraram redes com o maior número de arestas, com 5247685 para a imagem *gt07*, 5149011 para *gt13* e 4371763 para a imagem *gt02*.

Nesta situação, durante a execução dos testes, observou-se que a capacidade de memória do hardware empregado era totalmente utilizada pelo algoritmo em execução o que exigia a utilização de memória virtual, este fato afetou negativamente os tempos das referidas imagens o que justificaria o desempenho fora do padrão demonstrado pelo algoritmo no processamento das demais imagens da base.

8 Conclusão

A segmentação autônoma representa uma das tarefas mais complexas quando trabalhamos com processamento de imagens, pois não existe ainda um algoritmo de segmentação que possa ser utilizado em qualquer situação. Na aplicação do modelo de competição e cooperação entre partículas, a tarefa de segmentação de imagens aplica recursos de inteligência computacional a esta atividade, através de aprendizado semi-supervisionado.

As aplicações em segmentação de imagens que até este momento utilizavam esta abordagem para segmentar imagens, empregavam de recursos computacionais como algoritmos genéticos, análise de histograma ou ainda análise de indicadores de qualidade em redes complexas de forma a determinar qual a combinação de características previamente extraída da imagem era de maior relevância para segmentação daquela imagem específica (BREVE, 2017; BREVE; QUILES; ZHAO, 2015a; BREVE; QUILES; ZHAO, 2015b; BREVE, 2015; BREVE et al., 2012).

Uma vez determinada a relevância, era necessário ajustar o peso das características extraídas dos píxeis da imagem, representada no modelo pelo vetor de pesos λ , de forma automática a partir das análises realizadas, ou ainda, em uma outra abordagem necessitavam de ajustes manuais efetuados por um especialista para esse fim.

Deste modo, o presente trabalho demonstrou que a proposta de metodologia apresentada, melhora o nível de automação da tarefa de segmentação de imagens utilizando o modelo de competição e cooperação de partículas, para isto, realizou uma alteração na forma em que a rede complexa foi construída permitindo a eliminação do vetor de pesos λ do modelo.

A metodologia aplicou interpolação bicúbica/bilinear para trabalhar com imagens maiores, implementou uma nova marcação que permite ao usuário indicar a área de interesse dentro da imagem, deste modo, promoveu uma redução do tamanho da imagem diminuindo o escopo que foi analisado pelo modelo proposto.

A técnica apresentada também alterou o conjunto de características extraídas de cada pixel, reduzindo o escopo das características que são utilizadas para determinar as ligações dos vértices dentro da rede complexa e inseriu uma nova característica baseada no modelo de limiarização OTSU. Alterou a construção da rede complexa, que é a base para o modelo de competição e cooperação entre partículas, determinando uma nova regra de geração de arestas, onde baseou-se na similaridade entre as características do píxeis e também enfatizou a importância da vizinhança espacial entre os píxeis na imagem, representando essa informação relevante no modelo.

Após a construção da rede, a técnica que foi proposta trabalhou também a influência inicial das partículas (píxeis rotulados) dentro do modelo da rede complexa, onde vértices que foram gerados a partir de píxeis espacialmente próximos sofrem influência daquela partícula em até segundo grau de vizinhança. A influência aumentou o domínio da classe da partícula sobre o vértice dentro do seu raio de ação na rede antes de iniciar o processo de competição e cooperação entre as partículas.

A metodologia proposta demonstrou resultados satisfatórios quando comparados ao modelo tradicional de preparo da rede complexa para a segmentação utilizando competição e cooperação de partículas. Comparando as taxas de erros do modelo proposto e do modelo original sem utilizar ajustes de pesos no vetor λ , obteve-se uma taxa de erro de 0,49% dos píxeis para o método proposto e o método original sem ajustes chegou a índice de 3,14%.

Além de obter uma melhor média final no conjunto de dados de testes, a técnica também mostrou-se mais estável em relação as diferentes imagens utilizadas. Na Figura 7.9 pode-se verificar que o método proposto obteve em seu desempenho uma menor influência das características e variedade de tipos de imagens que foram analisadas dentro do conjunto apresentado no trabalho, o que ficou caracterizado na relação do percentual da taxa de erro e o tempo de execução necessário para a segmentação.

O modelo proposto também obteve uma melhora no tempo de execução em relação ao modelo testemunha, onde pode-se observar uma média de 432,54 segundos para o modelo proposto, contra 1082,94 segundos do modelo testemunha, sendo este um dos principais problemas da utilização do modelo de competição e cooperação entre partículas dentro da aplicação de segmentação de imagens.

O modelo proposto ainda precisa de melhorias em situações de características muito próximas entre o fundo da imagem e o objeto a ser segmentado, estas deficiências demandam continuação dos estudos da aplicação do modelo de competição e cooperação entre partículas, mas isto, não minimiza o sucesso da metodologia proposta por neste trabalho. Onde a técnica apresenta melhora o nível de automação do processo de segmentação, senda essa a sua principal vantagem sobre outras abordagens de aplicação de competição e cooperação entre partículas na segmentação de imagens previamente estudadas.

Referências

- ABNEY, S. Understanding the yarowsky algorithm. *Computational Linguistics*, MIT Press, Cambridge, MA, EUA, v. 30, n. 3, p. 365–395, set 2004. ISSN 0891-2017. Citado na página 52.
- AGGARWAL, C. C.; YU, P. S. A survey of uncertain data algorithms and applications. *IEEE Transactions on Knowledge and Data Engineering*, IEEE Educational Activities Department, v. 21, n. 5, p. 609–623, maio 2009. Citado na página 50.
- ALPAYDIN, E. *Introduction to Machine Learning*. 3. ed. Cambridge, MA: MIT Press, 2014. (Adaptive Computation and Machine Learning). Citado 2 vezes nas páginas 49 e 50.
- ARAUJO, A. F. *Método para extração e caracterização de lesões de pele usando difusão anisotrópica, crescimento de regiões, watershed e contornos ativos*. São José do Rio Preto, SP: Universidade Estadual Paulista, 2010. Citado na página 37.
- ARRUDA, G. F. de; COSTA, L. da F.; RODRIGUES, F. A. A complex networks approach for data clustering. *Physica A: Statistical Mechanics and its Applications*, v. 391, n. 23, p. 6174 – 6183, 2012. Citado na página 46.
- ASCENCIO, A.; ARAUJO, G. *Estrutura de Dados - Algoritmos, análise da complexidade e implementações*. 1. ed. São Paulo: Pearson Prentice Hall, 2010. Citado 2 vezes nas páginas 44 e 45.
- BARABASI, A.-L.; ALBERT, R. Emergence of scaling in random networks. *Science*, v. 286, n. 5439, p. 509–512, 1999. Citado 2 vezes nas páginas 42 e 43.
- BARBER, D. *Bayesian Reasoning and Machine Learning*. [S.l.]: Cambridge University Press, 2012. Citado 3 vezes nas páginas 50, 51 e 54.
- BISHOP, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006. Citado na página 50.
- BLUM, A.; MITCHELL, T. Combining labeled and unlabeled data with co-training. In: *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*. New York, NY, EUA: ACM, 1998. (COLT' 98), p. 92–100. Citado na página 53.
- BOCCALETTI, S. et al. Complex networks : Structure and dynamics. *Physics Reports*, v. 424, n. 4-5, p. 175–308, 2006. Citado na página 43.
- BOYKOV, Y.; JOLLY, M. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In: *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*. [S.l.: s.n.], 2001. v. 1, p. 105–112. Citado 2 vezes nas páginas 8 e 39.
- BOYKOV, Y.; KOLMOGOROV, V. An experimental comparison of min-cut max-flow algorithms for energy minimization in vision. *IEEE Transaction Pattern Analysis Machine*

- Intelligence*, IEEE Computer Society, v. 26, n. 9, p. 1124–1137, set 2004. Citado na página 40.
- BREVE, F.; QUILES, M. G.; ZHAO, L. Interactive image segmentation of non-contiguous classes using particle competition and cooperation. In: *15th International Conference on Computational Science and Its Applications, ICCSA 2015, 2015, Computational Science and Its Applications - ICCSA 2015 - 15th International Conference, Proceedings, Part I*. [S.l.]: Springer International Publishing, 2015. v. 9155, p. 203 – 216. Citado 7 vezes nas páginas 15, 59, 60, 65, 66, 67 e 99.
- BREVE, F.; QUILES, M. G.; ZHAO, L. Interactive image segmentation using particle competition and cooperation. *Lecture Notes in Computer Science*, Springer International Publishing, v. 9155, p. 203–216, out 2015. Citado 7 vezes nas páginas 14, 15, 65, 66, 67, 69 e 99.
- BREVE, F. et al. Particle competition and cooperation in networks for semi-supervised learning. *Knowledge and Data Engineering*, IEEE Transactions, n. 24(9), p. 1686–1698, 2012. Citado 8 vezes nas páginas 15, 57, 58, 59, 60, 61, 63 e 99.
- BREVE, F. A. *Aprendizado de máquina utilizando dinâmica espaço temporal em redes complexas*. Tese (Doutorado) — Universidade de São Paulo, São Carlos, 2010. Citado 9 vezes nas páginas 42, 43, 44, 52, 57, 58, 59, 60 e 62.
- BREVE, F. A. Active semi-supervised learning using particle competition and cooperation in networks. *Neural Networks (IJCNN)*, The 2012 International Joint Conference, 2013. Citado na página 64.
- BREVE, F. A. Auto feature weight for interactive image segmentation using particle competition and cooperation. In: *Proceedings - XI Workshop de Visão Computacional WVC2015*. [S.l.]: XI Workshop de Visão Computacional (WVC2015), 2015. p. 164 – 169. Citado 5 vezes nas páginas 15, 57, 68, 69 e 99.
- BREVE, F. A. Building networks for image segmentation using particle competition and cooperation. In: *Gervasi O. et al (eds) Computational Science and Its Applications, ICCSA 2017, International Conference, Proceedings*. [S.l.]: Springer International Publishing, 2017. v. 10404, p. 217–231. Citado 3 vezes nas páginas 15, 69 e 99.
- BRYNS, L. M. *Página dinâmica para aprendizado do sensoriamento remoto*. Porto Alegre: UFRGS, 2008. Citado na página 20.
- CASTLEMAN, K. R. *Digital Image Processing*. Upper Saddle River, NJ: Prentice-Hall, 1996. Citado na página 32.
- CHAPELLE, O.; SCHÖLKOPF, B.; ZIEN, A. *Semi-Supervised Learning*. Cambridge, MA, USA: MIT Press, 2006. 508 p. (Adaptive computation and machine learning). Citado 5 vezes nas páginas 15, 50, 51, 52 e 55.
- CIE. Proceedings of the commission internationale de l éclairage - 8a sessão. In: . [S.l.]: CIE, 1931. p. 326. Citado na página 22.
- CIOS, K. J. et al. *Data Mining: A knowlegde Discovery Approach*. [S.l.]: Springer-Verlag New York, Inc., 2007. Citado na página 50.

- COUSTY, J. et al. Watershed cuts: Minimum spanning forests and the drop of water principle. *IEEE transactions on pattern analysis and machine intelligence*, v. 31, p. 1362–74, 09 2009. Citado na página 38.
- DARA, R.; KREMER, S. C.; STACEY, D. A. Clustering unlabeled data with soms improves classification of labeled real-world data. In: *Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on*. [S.l.: s.n.], 2002. v. 3, p. 2237–2242. Citado na página 53.
- DASS, R.; PRIYANKA; DEVI, S. Image segmentation techniques. *International Journal of Electronics and Communication Technology*, v. 3, n. 1, p. 66–70, Jan-Mar 2012. Citado 3 vezes nas páginas 34, 35 e 75.
- DEMIRIZ, A.; BENNETT, K.; EMBRECHTS, M. J. Semi-supervised clustering using genetic algorithms. In: *Artificial Neural Networks in Engineering*. [S.l.]: ASME Press, 1999. p. 809–814. Citado na página 53.
- DOURADO, W. B. *Avaliação de Técnicas de Interpolação de Imagens Digitais*. São José do Rio Preto, SP: Universidade Estadual Paulista, 2014. Citado 6 vezes nas páginas 23, 24, 25, 26, 27 e 73.
- DUDA, R. O.; HART, P. E.; STORK, D. G. *Pattern Classification*. 2. ed. [S.l.]: Wiley-Interscience, 2000. ISBN 0471056693. Citado na página 14.
- ERDŐS, P.; RÉNYI, A. On random graphs. *Publicationes Mathematicae*, n. 6, p. 290–297, 1959. Citado na página 42.
- ERDŐS, P.; RÉNYI, A. On evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, n. 5, p. 17–61, 1960. Citado na página 42.
- ERDŐS, P.; RÉNYI, A. On the strength of connectedness of a random graph. *Acta Mathematica Scientia Hungary*, n. 12, p. 261–267, 1961. Citado na página 42.
- FELZENSZWALB, P. F.; HUTTENLOCHER, D. P. Efficient graph-based image segmentation. *International Journal of Computer Vision*, v. 59, n. 2, p. 167–181, Set 2004. Citado na página 41.
- FUJINO, A.; UEDA, N.; SAITO, K. A hybrid generative/discriminative approach to semi-supervised classifier design. In: *Proceedings of the 20th National Conference on Artificial Intelligence - Volume 2*. [S.l.]: AAAI Press, 2005. (AAAI'05), p. 764–769. Citado na página 52.
- GETREUER, P. Linear methods for image interpolation. *Image Processing On Line*, n. 1, p. 238–259, 2011. Citado 5 vezes nas páginas 22, 23, 24, 25 e 73.
- GONZALEZ, R. C.; WOODS, R. E. *Processamento Digital de Imagens*. 3. ed. [S.l.]: Longman do Brasil., 2010. Citado 15 vezes nas páginas 14, 17, 18, 19, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36 e 75.
- GRADY, L. Random walks for image segmentation. *IEEE Transactions Pattern Analysis Machine Intelligence*, IEEE Computer Society, v. 28, n. 11, p. 1768–1783, nov. 2006. Citado na página 54.

- GUIJARRO, M. et al. Automatic segmentation of relevant textures in agricultural images. *Computers and Electronics in Agriculture*, Elsevier, v. 75, n. 1, p. 75–83, 2011. Citado na página 21.
- HAFFARI, G.; SARKAR, A. Analysis of semi-supervised learning with the yarowsky algorithm. In: *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*. Arlington, Virginia, EUA: AUAI Press, 2007. (UAI'07), p. 159–166. Citado na página 52.
- HINTON, G. E.; SEJNOWSKI, T. J. *Unsupervised Learning: Foundations of Neural Computation*. [S.l.]: MIT Press, 1999. Citado na página 50.
- HOPCROFT, J. et al. Natural communities in large linked networks. In: *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM, 2003. p. 541–546. Citado na página 48.
- HU, C.; YOUN, B. D.; KIM, T. Semi-supervised learning with co-training for data-driven prognostics. *Prognostics and Health Management (PHM)*, IEEE Conference on, p. 1–10, 2012. Citado na página 53.
- JAIN, A. K.; DUBES, R. C. *Algorithms for Clustering Data*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1988. ISBN 0-13-022278-X. Citado na página 14.
- JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data clustering: a review. *ACM Computing Surveys*, v. 31, n. 3, p. 264–323, 1999. Citado 2 vezes nas páginas 14 e 50.
- KAUFMAN, L.; ROUSSEEUW, P. J. *Finding Groups in Data: An Introduction to Cluster Analysis*. [S.l.]: John Wiley, 1990. Citado na página 14.
- KAUR, K.; KAUR, I.; KAUR, J. Survey on image interpolation. *International Journal of Advanced Research in Computer Science and Software Engineering*, v. 6, n. 5, p. 613–616, 2016. Citado 6 vezes nas páginas 22, 23, 24, 25, 26 e 73.
- LIAO, P.-S.; CHEN, T.-S.; CHUNG, P.-C. A fast algorithm for multilevel thresholding. *Journal of Information Science and Engineering*, v. 17, p. 713–727, 2001. Citado na página 34.
- LICHMAN, M. *UCI Machine Learning Repository*. 2013. Disponível em: <<http://archive.ics.uci.edu/ml/datasets/image+segmentation>>. Citado na página 21.
- LIU, B. *Web data mining; Exploring hyperlinks, contents, and usage data*. 2. ed. [S.l.]: Springer, 2011. Citado na página 54.
- LU, T. T. *Fundamental limitations of semi-supervised learning*. Waterloo, Ontario, Canada: University of Waterloo, 2009. Citado na página 51.
- MA, W.-Y.; MANJUNATH, B. Edgeflow: A technique for boundary detection and image segmentation. *Image Processing, IEEE Transactions*, n. 9(8), p. 1375–1388, 2000. Citado na página 33.
- MALMBERG, F. *Graph-based Methods for Interactive Image Segmentation*. [S.l.]: Uppsala University, Centre for Image Analysis, Computerized Image Analysis and Human Computer Interaction, 2011. Citado na página 28.

- MARQUES, O. F.; VIEIRA, H. N. *Processamento Digital de Imagens*. Rio de Janeiro: Brasport, 1999. Citado 3 vezes nas páginas 17, 18 e 22.
- MEYER, G. E.; HINDMAN, T. W.; LAKSMI, K. Machine vision detection parameters for plant species identification. *Proceedings of SPIE - The international Society for Optical Engineering*, v. 3543, p. 327–335, 1999. Citado na página 21.
- MITCHELL, T. M. *Machine Learning*. 2. ed. New York: McGraw-Hill, 1997. Citado 4 vezes nas páginas 14, 49, 50 e 53.
- MITCHELL, T. M. The role of unlabeled data in supervised learning. In: *Proceedings of the Sixth International Colloquium on Cognitive Science*. [S.l.: s.n.], 1999. Citado na página 53.
- MOORE, A. et al. Superpixel lattices. In: *IEEE Conference on Computer Vision and Pattern Recognition*. Anchorage, AK, EUA: [s.n.], 2008. Citado na página 41.
- NAVON, E.; MILLER, O.; AVERBUCH, A. Color image segmentation based on adaptive local thresholds. *Image and Vision Computing*, v. 23, n. 1, p. 69 – 85, 2005. Citado 5 vezes nas páginas 8, 31, 37, 38 e 41.
- NEWMAN, M. The structure and function of complex networks. *SIAM review*, JSTOR, v. 45, n. 2, p. 167–256, 2003. Citado na página 48.
- NEWMAN, M. E. J. *Networks: An Introduction*. New York: Oxford, 2010. Citado na página 42.
- NEWMAN, M. E. J.; GIRVAN, M. Finding and evaluating community structure in networks. *Physical Review E*, American Physical Society, v. 69, n. 2, fev 2004. Citado na página 47.
- NIGAM, K. et al. Text classification from labeled and unlabeled documents using em. *Machine Learning*, Kluwer Academic Publishers, Hingham, MA, EUA, v. 39, n. 2-3, p. 103–134, maio 2000. Citado na página 52.
- OSMA-RUIZ, V. et al. An improved watershed algorithm based on efficient computation of shortest paths. *Pattern Recogn.*, Elsevier Science Inc., New York, NY, USA, v. 40, n. 3, p. 1078–1090, mar. 2007. Citado na página 38.
- OTSU, N. A threshold selection method from gray-level histograms. *Systems, Man and Cybernetics, IEEE Transactions on*, v. 9, p. 62–66, 01 1979. Citado na página 34.
- PATEL, K. M. V. A review on different image interpolation techniques for image enhancement. *International Journal of Emerging Technology and Advanced Engineering*, v. 3, n. 12, p. 129–133, 2013. Citado 6 vezes nas páginas 22, 23, 24, 25, 26 e 73.
- PEDRINI, H.; SCHWARTZ, W. R. *Análise de Imagens Digitais: Princípios, Algoritmos e Aplicações*. São Paulo: Thomson Learning, 2007. Citado 2 vezes nas páginas 24 e 73.
- PRAJAPATI, A.; NAIK, S.; MEHTA, S. Evaluation of different image interpolation algorithms. *International Journal of Computer Applications*, v. 58, n. 12, p. 6–12, 2012. Citado 5 vezes nas páginas 22, 23, 24, 25 e 73.

- QUILES, M. G. et al. Particle competition for complex network community detection. *Chaos*, v. 18, n. 3, 2008. Citado na página 57.
- REN, X.; MALIK, J. Learning a classification model for segmentation. In: *Proceedings of the 9th IEEE International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 2003. (ICCV '03, v. 2). Citado na página 41.
- RILOFF, E.; WIEBE, J.; WILSON, T. Learning subjective nouns using extraction pattern bootstrapping. In: *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*. Stroudsburg, PA, EUA: Association for Computational Linguistics, 2003. (CONLL '03), p. 25–32. Citado na página 52.
- ROTHER, C.; KOLMOGOROV, V.; BLAKE, A. "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, ACM, New York, NY, USA, v. 23, n. 3, p. 309–314, ago 2004. ISSN 0730-0301. Citado 2 vezes nas páginas 40 e 41.
- SCARANTI, A.; BERNARDI, R.; PLOTZE, R. O. Identificação de Órgãos foliares utilizando as wavelets de daubechies. *VI Workshop de Visão Computacional de Presidente Prudente*, Presidente Prudente -SP, v. 6, p. 211–216, Julho 2010. Citado na página 22.
- SCHAEFFER, S. E. Survey: Graph clustering. *Computer Science Review*, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 1, n. 1, p. 27–64, ago 2007. Citado na página 14.
- SHI, J.; MALIK, J. Normalized cuts and image segmentation. *IEEE Transactions - Pattern Analysis and Machine Intelligence*, IEEE Computer Society, v. 22, n. 8, p. 888–905, ago 2000. ISSN 0162-8828. Citado 2 vezes nas páginas 28 e 46.
- SILVA, T. C.; ZHAO, L. *Machine Learning in Complex Networks*. [S.l.]: Springer International Publishing, 2016. Citado 2 vezes nas páginas 42 e 43.
- SKARBEEK, W.; KOSCHAN, A. *Colour Image Segmentation: A Survey*. [S.l.]: Leiter der Fachbibliothek Informatik, 1994. Citado 2 vezes nas páginas 34 e 75.
- SZUMMER, M.; JAAKKOLA, T. Partially labeled classification with markov random walks. In: DIETTERICH, T. G.; BECKER, S.; GHAMRANI, Z. (Ed.). *Advances in Neural Information Processing Systems 14*. [S.l.]: MIT Press, 2002. p. 945–952. Citado 2 vezes nas páginas 54 e 55.
- TAN, P. N.; STEINBACH, M.; KUMAR, V. *Introduction to data mining*. [S.l.]: Addison Wesley, 2005. Citado 3 vezes nas páginas 45, 46 e 47.
- VAPNIK, V. N. *Statistical Learning Theory*. [S.l.]: Wiley-Interscience, 1998. Citado na página 54.
- VEKSLER, O.; BOYKOV, Y.; MEHRANI, P. Superpixels and supervoxels in an energy optimization framework. In: *Proceedings of the 11th European Conference on Computer Vision: Part V*. Berlin, Heidelberg: Springer-Verlag, 2010. (ECCV'10), p. 211–224. Citado na página 41.
- WANG, F. et al. Semi-supervised classification using local and global regularization. In: *AAAI'08: Proceedings of the 23rd national conference on Artificial intelligence*. [S.l.]: AAAI Press, 2008. p. 726–731. Citado na página 54.

- WANG, F.; ZHANG, C. Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, v. 20, n. 1, p. 55–67, Jan 2008. Citado na página 54.
- WATTS, D. J.; STROGATZ, S. Collective dynamics of 'small-world' networks. *Nature*, v. 393, p. 440–442, June 1998. Citado na página 42.
- WITTEN, I. H.; FRANK, E.; HALL, M. A. *Data Mining: Practical Machine Learning Tools and Techniques*. 3. ed. São Francisco-CA, EUA: Morgan Kaufmann Publishers Inc., 2011. Citado na página 50.
- WOEBBECKE, D. M. et al. Shape features for identifying young weeds using image analysis. *Transactions of the ASAE*, v. 38, p. 271–281, 01 1995. Citado na página 21.
- WOLFF, R.; BHADURI, K.; KARGUPTA, H. A generic local algorithm for mining data streams in large distributed systems. *IEEE Transactions on Knowledge and Data Engineering*, IEEE Educational Activities Department, v. 21, n. 4, p. 465–478, abr 2009. Citado na página 50.
- WU, M.; SCHÖLKOPF, B. Transductive classification via local learning regularization. In: MAX-PLANCK-GESELLSCHAFT. *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics (AISTATS 2007)*. [S.l.], 2007. v. 2, p. 628–635. Citado na página 54.
- XIANG, S.; NIE, F.; ZHANG, C. Learning a mahalanobis distance metric for data clustering and classification. *Pattern Recognition*, v. 41, n. 12, p. 3600 – 3612, 2008. Citado na página 46.
- YAROWSKY, D. Unsupervised word sense disambiguation rivaling supervised methods. In: *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. [S.l.: s.n.], 1995. p. 189–196. Citado na página 52.
- ZAITOUN, N. M.; AQEL, M. J. Survey on image segmentation techniques. *Procedia Computer Science*, v. 65, p. 797 – 806, 2015. Citado 3 vezes nas páginas 14, 28 e 29.
- ZHOU, D. et al. Learning with local and global consistency. In: *Proceedings of the 16th International Conference on Neural Information Processing Systems*. [S.l.]: MIT Press, 2004. p. 321–328. Citado 3 vezes nas páginas 51, 54 e 55.
- ZHOU, Z.-H.; LI, M. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge and Data Engineering*, IEEE Educational Activities Department, v. 17, n. 11, p. 1529–1541, nov 2005. Citado na página 53.
- ZHOU, Z.-H.; LI, M. Semisupervised regression with cotraining-style algorithms. *IEEE Transactions on Knowledge and Data Engineering*, IEEE Computer Society, v. 19, n. 11, p. 1479–1493, 2007. Citado na página 53.
- ZHU, X. Semi-supervised learning literature survey. *Computer Science, University of Wisconsin-Madison*, v. 2, 07 2008. Citado 4 vezes nas páginas 52, 53, 54 e 56.
- ZHU, X.; GHAHRAMANI, Z. Learning from labeled and unlabeled data with label propagation. *Technical Report CMU-CALD-02-107*, Carnegie Mellon University, 2002. Citado 2 vezes nas páginas 54 e 55.

ZHU, X. et al. Semi-supervised learning using gaussian fields and harmonic functions. In: *Proceedings of the 20th International Conference on Machine Learning (ICML-2003)*. [S.l.: s.n.], 2003. v. 3, p. 912–919. Citado na página 54.

APÊNDICE A – Características das redes geradas a partir das imagens

Tabela A.1 – Características das redes geradas a partir das imagens

Nr	Imagem	Pixel		Proposto			Testemunha	
		Orig.	Cinza	Corte	Vért	Arestas	Vért	Arestas
	Média	200124	2783	84320	7538	838564	17946	2354555
1	106024	154401	0	40310	3567	366313	13824	1803219
2	153077	154401	2116	97440	8712	944290	13824	1825881
3	153093	154401	0	87822	7811	884217	13824	1781415
4	181079	154401	0	120000	10800	1239991	13824	1815101
5	189080	154401	0	119808	10640	1194831	13824	1767638
6	208001	154401	0	63936	5632	611561	13824	1786864
7	209070	154401	2047	58240	5208	486052	13824	1801476
8	21077	154401	928	27416	2420	232090	13824	1767511
9	227092	154401	1398	92512	8241	929373	13824	1775637
10	24077	154401	1173	41789	3705	345729	13824	1806492
11	271008	154401	1072	47888	4263	454991	13824	1788435
12	304074	154401	1656	32702	2891	278120	13824	1858725
13	326038	154401	1606	58608	5192	528539	13824	1815465
14	37073	154401	1288	63664	5665	624396	13824	1825048
15	376043	154401	0	80104	7104	767971	13824	1859124
16	388016	154401	1615	74370	6600	766777	13824	1821046
17	65019	154401	1022	66005	5888	616903	13824	1748331
18	69020	154401	2211	140160	12576	1418336	13824	1864183
19	86016	154401	954	43099	3822	425613	13824	1763023
20	aero_2007 _000032	140500	2863	32148	2856	297608	12600	1656481
21	Aero_2007 _001884	187500	2282	46968	4140	397966	16800	2175929
22	Aero_2007 _006232	166500	7266	75658	6750	685366	14850	1955296
23	Aero_2007 _007341	187500	3112	73305	6534	681072	16800	2187675

Continua na próxima página

Tabela A.1 – continuação da página anterior

Nr	Imagem	Pixel		Proposto			Testemunha	
		Orig.	Cinza	Corte	Vért	Arestas	Vért	Arestas
24	Aero_2007 _009030	187500	3180	124704	11094	1213068	16800	2244825
25	Aero_2008 _002358	166500	2620	83832	7450	797159	14850	1941604
26	Baby_2007 _005227	167000	5293	89698	8010	865896	15000	2017835
27	Baby_2007 _006647	187500	7670	157500	16800	1796483	16800	2206608
28	Baby_2007 _007203	187500	1124	32088	2850	280418	16800	2194209
29	banana1	307200	2562	236925	21175	2434701	27648	3547442
30	banana2	307200	2942	224010	20007	2306610	27648	3658525
31	banana3	307200	3567	215232	19292	2223337	27648	3645747
32	Bike_2007 _000364	187500	6436	132680	11904	1267308	16800	2274510
33	Bike_2007 _000822	187000	1563	30225	2668	262268	16800	2224116
34	Bike_2007 _004003	187500	9989	130180	11592	1156895	16800	2252193
35	Bike_2007 _005354	187500	3289	57856	5092	511750	16800	2249509
36	Bike_2007 _005878	166000	1318	36792	3225	332614	14850	1952731
37	Bike_2007 _008994	187500	1202	63484	5600	565966	16800	2228176
38	Bike_2007 _009084	187500	9841	134368	12012	1275188	16800	2268613
39	Bike_2008 _000782	166500	3038	101200	9000	977125	14850	1952846
40	Bike_2008 _002772	187500	5180	75632	6693	625414	16800	2203296
41	Bird_2007 _000363	166500	5166	87000	7800	850352	14850	1924230
42	Bird_2008 _002200	209500	3146	74782	6640	715036	18750	2481982

Continua na próxima página

Tabela A.1 – continuação da página anterior

Nr	Imagem	Pixel		Proposto			Testemunha	
		Orig.	Cinza	Corte	Vért	Arestas	Vért	Arestas
43	Boat_2007 _000061	166500	1792	53504	4732	495385	14850	1928865
44	Boat_2007 _000837	137500	11802	12862	1120	103383	12300	1564052
45	Boat_2007 _009527	187500	4333	83358	7434	833304	16800	2190192
46	Boat_2008 _001135	187500	1877	34105	2996	307618	16800	2212739
47	book	307200	2566	230720	20664	2481056	27648	3605820
48	bool	234000	1888	112029	9984	1119617	21060	2799193
49	Bus_2007 _000727	187500	1675	28542	2520	226865	16800	2206324
50	Bus_2007 _002216	170000	4503	87216	7810	833732	15300	1985979
51	Bus_2007 _003101	166500	1326	62700	5576	605688	14850	1973506
52	Bus_2007 _003503	167000	1831	56992	5084	501331	15000	1991699
53	Bus_2007 _003786	140500	753	14384	1258	126808	12600	1612983
54	Bus_2007 _005262	187500	1329	43710	3864	377132	16800	2229348
55	bush	270000	2237	103950	9324	993519	24300	3219373
56	Car_2007 _006900	187500	3235	100738	9000	937371	16800	2188902
57	Car_2007 _009779	187500	2030	38078	3384	309558	16800	2149284
58	Car_2008 _000595	206000	2350	51009	4576	450120	18450	2417036
59	Car_2008 _001716	143424	3028	45312	4028	373544	12814	1649199
60	Car_2008 _002834	187500	1138	39114	3431	361172	16800	2155874
61	Cat_2007 _000528	187500	2498	59706	5312	550462	16800	2299345

Continua na próxima página

Tabela A.1 – continuação da página anterior

Nr	Imagem	Pixel		Proposto			Testemunha	
		Orig.	Cinza	Corte	Vért	Arestas	Vért	Arestas
62	Cat_2007 _000876	167000	6958	85050	7614	775282	15000	1929666
63	Cat_2007 _001185	187500	3802	71610	6417	727323	16800	2128321
64	Cat_2007 _007530	189000	1196	24424	2142	241698	16950	2236032
65	Cat_2008 _000345	173500	2285	93800	8400	930022	15600	2058881
66	Cat_2008 _001592	187500	1932	55948	5015	511732	16800	2196625
67	Cat_2008 _001885	187500	4058	117640	10506	1162463	16800	2157652
68	ceramic	307200	3196	167217	15000	1721171	27648	3530174
69	Chair_2008 _000107	207000	2087	73914	6612	718659	18600	2402026
70	Cow_2007 _000464	187500	876	11850	1035	93503	16800	2184694
71	cross	270000	1870	270000	24300	2883050	24300	3148781
72	Cyclist_2007 _000129	167000	5329	101088	8960	918476	15000	1971149
73	Dog_2007 _001239	187500	6980	85824	7654	847992	16800	2219165
74	Dog_2008 _000080	187500	3617	56404	4970	502233	16800	2238903
75	doll	253638	2238	123279	11074	1252248	22632	2890827
76	elefant	307200	3405	178893	16000	1842439	27648	3520160
77	flower	270000	1740	113850	10197	1199909	24300	3247886
78	fullmoon	154700	423	18760	1680	184919	13860	1516112
79	grave	270000	1268	51182	4512	527824	24300	3285360
80	gt02	419200	2229	419200	37680	4371763	37680	4772882
81	gt07	494400	6656	494400	44400	5247685	44400	5721712
82	gt13	476000	12429	476000	42720	5149011	42720	5536232
83	Horse_2007 _003889	152640	4766	52920	4704	481240	13680	1826082

Continua na próxima página

Tabela A.1 – continuação da página anterior

Nr	Imagem	Pixel		Proposto			Testemunha	
		Orig.	Cinza	Corte	Vért	Arestas	Vért	Arestas
84	Horse_2007 _006134	187500	3406	74793	6624	705953	16800	2237109
85	Horse_2008 _000696	143000	1515	29524	2592	256479	12750	1724429
86	llama	190323	0	75348	6642	721396	16983	2269842
87	memorial	270000	1242	103251	9234	1004260	24300	3223000
88	Monitor_2007 _000039	187500	2143	50190	4473	480203	16800	2207294
89	Monitor_2007 _000121	187500	2046	60395	5390	559755	16800	2209146
90	Monitor_2007 _000187	187500	2230	60993	5400	587195	16800	2180160
91	Monitor_2007 _001678	166500	808	14260	1258	99753	14850	1996401
92	Monitor_2007 _003011	187500	1614	16188	1428	114506	16800	2237507
93	Monitor_2007 _004193	187500	800	14690	1287	117952	16800	2180800
94	music	307200	2596	186172	16640	1913332	27648	3650930
95	Person_2007 _000170	187500	4328	170973	15228	1764527	16800	2218212
96	Person_2007 _000323	187500	4118	90508	8064	866025	16800	2190656
97	Person_2007 _000392	166000	7575	86086	7650	755623	14850	1976273
98	Person_2007 _000480	187500	1809	32625	2881	291017	16800	2092859
99	Person_2007 _000733	195750	6490	105060	9348	988113	17550	2365584
100	Person_2007 _000783	209760	5723	104300	9345	1002771	18864	2497448
101	Person_2007 _001423	166500	4437	77406	6873	736966	14850	1931896
102	Person_2007 _001430	187500	1812	43524	3850	381775	16800	2212902

Continua na próxima página

Tabela A.1 – continuação da página anterior

Nr	Imagem	Pixel		Proposto			Testemunha	
		Orig.	Cinza	Corte	Vért	Arestas	Vért	Arestas
103	Person_2007_001526	149000	2210	31472	2772	267389	13350	1779658
104	Person_2007_002378	187500	5316	68200	6138	602270	16800	2279804
105	Person_2007_002426	187500	3659	116998	10496	1152178	16800	2235057
106	Person_2007_002639	187500	1507	16999	1482	142950	16800	2233489
107	Person_2007_002823	187500	1859	29700	2640	252767	16800	2170149
108	Person_2007_004328	155500	940	21054	1872	187687	13950	1846781
109	Person_2007_004722	187500	1710	31464	2788	269795	16800	2218159
110	Person_2007_005086	187500	3410	80817	7242	740489	16800	2268369
111	Person_2007_005430	187500	4218	42660	3807	343903	16800	2261672
112	Person_2007_006364	172000	4118	50856	4462	460473	15450	1976194
113	Person_2007_007996	166500	1388	24396	2176	209857	14850	1990361
114	Person_2007_008218	187500	2993	96990	8645	921099	16800	2198220
115	Person_2007_009327	167000	3179	74636	6664	736513	15000	1937660
116	Person_2007_009687	187500	3500	87975	7828	836463	16800	2233687
117	Person_2008_000474	187500	2867	88788	7920	885573	16800	2217802
118	Person_2008_000510	166500	3436	63960	5694	612904	14850	1929024
119	Person_2008_000584	187500	871	26880	2412	234188	16800	2228236

Continua na próxima página

Tabela A.1 – continuação da página anterior

Nr	Imagem	Pixel		Proposto			Testemunha	
		Orig.	Cinza	Corte	Vért	Arestas	Vért	Arestas
120	Person_2008 _001375	166500	3888	49419	4386	443841	14850	1928752
121	Person_2008 _001547	246000	2396	40890	3654	372406	22050	2842521
122	Person1	270000	1977	213328	19040	2266194	24300	3197730
123	person2	270000	1951	78725	7000	770573	24300	3212419
124	person3	270000	1210	67450	6042	709867	24300	3212676
125	person4	270000	1455	99682	8850	1011339	24300	3186445
126	person5	270000	1169	40817	3570	382414	24300	3163682
127	person6	270000	1273	73800	6642	726740	24300	3280895
128	person7	270000	1024	43792	3850	426825	24300	3279375
129	person8	307200	2364	105501	9375	1051546	27648	3684274
130	Plant_2007 _002967	168000	4163	25600	2280	218663	15000	1972769
131	Plant_2007 _004121	166000	1391	10810	952	68462	14850	1982433
132	Plant_2007 _004948	187500	801	7957	672	45981	16800	2220946
133	Plant_2008 _000391	187500	7621	93922	8370	868360	16800	2167171
134	Plant_2008 _002273	187500	1368	36284	3192	284536	16800	2230984
135	Plant_2008 _002379	187500	5551	103009	9159	1034542	16800	2187968
136	scissors	307200	2499	227154	20241	2326134	27648	3631247
137	sheep	270000	832	47288	4235	488474	24300	3320604
138	Sheep_2007 _001416	164000	2017	23316	2080	194216	14700	1921648
139	Sheep_2007 _003593	166500	2669	53000	4740	535302	14850	1948950
140	Sofa_2007 _000661	187500	1329	25641	2277	202793	16800	2186248
141	Sofa_2007 _001149	187500	1686	50778	4455	473030	16800	2236414

Continua na próxima página

Tabela A.1 – continuação da página anterior

Nr	Imagem	Pixel		Proposto			Testemunha	
		Orig.	Cinza	Corte	Vért	Arestas	Vért	Arestas
142	Sofa_2007_006530	187500	1044	22400	1976	212305	16800	2233438
143	stone1	307200	1269	112252	9954	1154833	27648	3545810
144	stone2	307200	1543	164980	14715	1689370	27648	3583419
145	teddy	113032	0	52250	4674	556958	10115	1261233
146	tennis	236000	1601	65514	5777	637870	21150	2794239
147	Train_2007_000042	167500	3850	69936	6216	673616	15000	1965734
148	Train_2007_000333	166500	4060	96607	8547	892879	14850	1974394
149	Train_2007_002462	187500	3401	66672	5934	635874	16800	2276762
150	Train_2007_003286	187500	4566	114904	10287	1125750	16800	2268694
151	Train_2007_004627	187500	1180	25149	2220	2023323	16800	2213111

Fonte: Elaborado pelo autor

APÊNDICE B – Resultados: taxa de erro e tempos de execução

Tabela B.1 – Segmentação as imagens - taxa de Erro

Nr	Imagem	Proposto			Testemunha		
		Px Erro	% Erro	Dv Pad	Px Erro	% Erro	Dv Pad
1	106024	1476	0,96%	0,01%	4214	2,73%	0,18%
2	153077	1399	0,92%	0,01%	4442	2,92%	0,11%
3	153093	1190	0,77%	0,00%	3618	2,34%	0,07%
4	181079	1874	1,21%	0,02%	2531	1,64%	0,03%
5	189080	1994	1,29%	0,15%	2693	1,74%	0,18%
6	208001	1999	1,29%	0,04%	2179	1,41%	0,09%
7	209070	900	0,59%	0,02%	9130	5,99%	0,16%
8	21077	640	0,42%	0,01%	12315	8,02%	0,46%
9	227092	1574	1,03%	0,06%	3426	2,24%	0,31%
10	24077	1069	0,70%	0,01%	11243	7,34%	0,73%
11	271008	1681	1,10%	0,04%	9264	6,04%	0,20%
12	304074	914	0,60%	0,03%	4182	2,74%	0,13%
13	326038	959	0,63%	0,01%	3826	2,50%	0,07%
14	37073	1285	0,84%	0,03%	2048	1,34%	0,46%
15	376043	3244	2,10%	0,09%	7125	4,61%	0,36%
16	388016	1132	0,74%	0,01%	1303	0,85%	0,03%
17	65019	864	0,56%	0,04%	5786	3,77%	0,19%
18	69020	2240	1,47%	0,02%	3055	2,01%	0,06%
19	86016	900	0,59%	0,07%	2150	1,40%	0,10%
20	aero_2007_000032	190	0,14%	0,01%	449	0,33%	0,05%
21	Aero_2007_001884	620	0,33%	0,01%	2462	1,33%	0,14%
22	Aero_2007_006232	677	0,43%	0,01%	7622	4,79%	0,38%
23	Aero_2007_007341	1324	0,72%	0,01%	13031	7,07%	0,26%
24	Aero_2007_009030	1047	0,57%	0,01%	5317	2,88%	0,07%
25	Aero_2008_002358	1395	0,85%	0,03%	6114	3,73%	0,20%
26	Baby_2007_005227	639	0,40%	0,02%	4865	3,01%	0,08%
27	Baby_2007_006647	2102	1,17%	0,02%	8213	4,57%	0,07%

Continua na próxima página

Tabela B.1 – continuação da página anterior

Nr	Imagem	Proposto			Testemunha		
		Px Erro	% Erro	Dv Pad	Px Erro	% Erro	Dv Pad
28	Baby_2007_007203	383	0,21%	0,00%	2933	1,57%	0,06%
29	banana1	2073	0,68%	0,03%	3733	1,23%	0,10%
30	banana2	2161	0,71%	0,03%	3102	1,02%	0,03%
31	banana3	2175	0,72%	0,03%	4716	1,55%	0,04%
32	Bike_2007_000364	1781	0,98%	0,01%	11602	6,41%	0,19%
33	Bike_2007_000822	272	0,15%	0,01%	4054	2,19%	0,15%
34	Bike_2007_004003	1457	0,82%	0,01%	11359	6,40%	0,14%
35	Bike_2007_005354	562	0,31%	0,01%	9748	5,29%	0,11%
36	Bike_2007_005878	765	0,46%	0,01%	3221	1,96%	0,13%
37	Bike_2007_008994	1221	0,66%	0,01%	10609	5,69%	0,22%
38	Bike_2007_009084	1302	0,73%	0,01%	15127	8,51%	0,19%
39	Bike_2008_000782	1939	1,19%	0,02%	8167	5,00%	0,42%
40	Bike_2008_002772	929	0,51%	0,01%	3598	1,97%	0,05%
41	Bird_2007_000363	1375	0,85%	0,01%	1475	0,91%	0,05%
42	Bird_2008_002200	673	0,33%	0,01%	2947	1,43%	0,08%
43	Boat_2007_000061	494	0,30%	0,00%	1769	1,07%	0,08%
44	Boat_2007_000837	320	0,25%	0,00%	1854	1,47%	0,05%
45	Boat_2007_009527	921	0,50%	0,02%	6090	3,32%	0,12%
46	Boat_2008_001135	1102	0,59%	0,02%	3269	1,76%	0,10%
47	book	857	0,28%	0,02%	990	0,32%	0,02%
48	bool	1749	0,75%	0,01%	7865	3,39%	0,15%
49	Bus_2007_000727	615	0,33%	0,03%	5946	3,20%	0,26%
50	Bus_2007_002216	528	0,32%	0,02%	13219	7,99%	0,21%
51	Bus_2007_003101	916	0,55%	0,02%	12519	7,58%	0,32%
52	Bus_2007_003503	734	0,44%	0,00%	7778	4,71%	0,12%
53	Bus_2007_003786	298	0,21%	0,01%	2884	2,06%	0,10%
54	Bus_2007_005262	600	0,32%	0,01%	12020	6,46%	0,27%
55	bush	1416	0,53%	0,01%	8296	3,10%	0,11%
56	Car_2007_006900	904	0,49%	0,01%	16901	9,17%	0,19%
57	Car_2007_009779	533	0,29%	0,01%	10959	5,91%	0,21%
58	Car_2008_000595	649	0,32%	0,00%	7533	3,70%	0,25%
59	Car_2008_001716	143	0,10%	0,01%	3528	2,51%	0,10%
60	Car_2008_002834	552	0,30%	0,02%	5472	2,94%	0,59%
61	Cat_2007_000528	1585	0,86%	0,02%	2016	1,09%	0,64%

Continua na próxima página

Tabela B.1 – continuação da página anterior

Nr	Imagem	Proposto			Testemunha		
		Px Erro	% Erro	Dv Pad	Px Erro	% Erro	Dv Pad
62	Cat_2007_000876	399	0,25%	0,01%	5279	3,30%	0,25%
63	Cat_2007_001185	576	0,31%	0,01%	10644	5,79%	0,26%
64	Cat_2007_007530	400	0,21%	0,00%	2610	1,39%	0,30%
65	Cat_2008_000345	778	0,45%	0,01%	5903	3,45%	0,18%
66	Cat_2008_001592	454	0,24%	0,01%	6902	3,72%	0,15%
67	Cat_2008_001885	1018	0,55%	0,01%	4786	2,61%	0,24%
68	ceramic	2044	0,67%	0,03%	6151	2,02%	0,05%
69	Chair_2008_000107	1048	0,51%	0,02%	12988	6,34%	0,45%
70	Cow_2007_000464	403	0,22%	0,00%	4245	2,27%	0,14%
71	cross	1276	0,48%	0,03%	4796	1,79%	0,07%
72	Cyclist_2007_000129	850	0,53%	0,01%	16413	10,15%	0,32%
73	Dog_2007_001239	954	0,53%	0,01%	4036	2,24%	0,08%
74	Dog_2008_000080	950	0,52%	0,01%	7051	3,83%	0,12%
75	doll	984	0,39%	0,06%	5789	2,30%	0,24%
76	elefant	1299	0,43%	0,02%	3523	1,16%	0,06%
77	flower	1676	0,62%	0,02%	10037	3,74%	0,08%
78	fullmoon	253	0,16%	0,02%	593	0,38%	0,01%
79	grave	747	0,28%	0,01%	7703	2,87%	0,12%
80	gt02	2180	0,52%	0,02%	5282	1,27%	0,13%
81	gt07	1005	0,21%	0,03%	3127	0,64%	0,06%
82	gt13	4995	1,08%	0,06%	9797	2,11%	0,16%
83	Horse_2007_003889	317	0,21%	0,01%	11483	7,77%	0,35%
84	Horse_2007_006134	419	0,23%	0,01%	7033	3,82%	0,26%
85	Horse_2008_000696	442	0,31%	0,00%	3182	2,25%	0,10%
86	llama	1669	0,88%	0,01%	7465	3,92%	0,09%
87	memorial	1088	0,40%	0,01%	6856	2,55%	0,05%
88	Monitor_2007_000039	294	0,16%	0,00%	2970	1,60%	0,06%
89	Monitor_2007_000121	740	0,40%	0,01%	5227	2,82%	0,25%
90	Monitor_2007_000187	647	0,35%	0,00%	4792	2,59%	0,12%
91	Monitor_2007_001678	206	0,12%	0,00%	3034	1,83%	0,24%
92	Monitor_2007_003011	29	0,02%	0,00%	2021	1,09%	0,12%
93	Monitor_2007_004193	198	0,11%	0,00%	5601	3,00%	0,28%
94	music	2266	0,74%	0,04%	8021	2,63%	0,07%
95	Person_2007_000170	791	0,43%	0,02%	4615	2,52%	0,13%

Continua na próxima página

Tabela B.1 – continuação da página anterior

Nr	Imagem	Proposto			Testemunha		
		Px Erro	% Erro	Dv Pad	Px Erro	% Erro	Dv Pad
96	Person_2007_000323	597	0,33%	0,01%	10246	5,59%	0,28%
97	Person_2007_000392	1093	0,69%	0,02%	11982	7,56%	0,26%
98	Person_2007_000480	331	0,18%	0,01%	8793	4,74%	0,14%
99	Person_2007_000733	948	0,50%	0,02%	16730	8,84%	0,33%
100	Person_2007_000783	935	0,46%	0,02%	6564	3,22%	0,19%
101	Person_2007_001423	798	0,49%	0,01%	3608	2,23%	0,21%
102	Person_2007_001430	559	0,30%	0,01%	6609	3,56%	0,18%
103	Person_2007_001526	843	0,57%	0,01%	8794	5,99%	0,72%
104	Person_2007_002378	1331	0,73%	0,03%	12752	7,00%	0,28%
105	Person_2007_002426	1087	0,59%	0,01%	7876	4,28%	0,16%
106	Person_2007_002639	215	0,12%	0,00%	4592	2,47%	0,10%
107	Person_2007_002823	573	0,31%	0,06%	5025	2,71%	0,18%
108	Person_2007_004328	368	0,24%	0,01%	1222	0,79%	0,14%
109	Person_2007_004722	308	0,17%	0,00%	2659	1,43%	0,16%
110	Person_2007_005086	1595	0,87%	0,01%	13032	7,08%	0,16%
111	Person_2007_005430	343	0,19%	0,01%	2254	1,23%	0,06%
112	Person_2007_006364	633	0,38%	0,01%	3721	2,22%	0,08%
113	Person_2007_007996	804	0,49%	0,00%	6374	3,86%	0,21%
114	Person_2007_008218	1376	0,75%	0,03%	5662	3,07%	0,13%
115	Person_2007_009327	649	0,40%	0,02%	3308	2,02%	0,10%
116	Person_2007_009687	1022	0,56%	0,01%	7376	4,01%	0,19%
117	Person_2008_000474	1028	0,56%	0,03%	2696	1,46%	0,17%
118	Person_2008_000510	868	0,53%	0,01%	2861	1,75%	0,05%
119	Person_2008_000584	238	0,13%	0,00%	2337	1,25%	0,17%
120	Person_2008_001375	548	0,34%	0,00%	6209	3,82%	0,41%
121	Person_2008_001547	479	0,20%	0,01%	1935	0,79%	0,04%
122	Person1	1105	0,41%	0,01%	2268	0,85%	0,08%
123	person2	604	0,23%	0,01%	521	0,19%	0,01%
124	person3	838	0,31%	0,01%	3038	1,13%	0,09%
125	person4	1612	0,60%	0,01%	14416	5,37%	0,16%
126	person5	1120	0,42%	0,00%	5430	2,02%	0,08%
127	person6	1301	0,48%	0,01%	6583	2,45%	0,27%
128	person7	844	0,31%	0,01%	3051	1,13%	0,03%
129	person8	1780	0,58%	0,03%	7603	2,49%	0,17%

Continua na próxima página

Tabela B.1 – continuação da página anterior

Nr	Imagem	Proposto			Testemunha		
		Px Erro	% Erro	Dv Pad	Px Erro	% Erro	Dv Pad
130	Plant_2007_002967	360	0,22%	0,00%	1541	0,94%	0,07%
131	Plant_2007_004121	192	0,12%	0,00%	768	0,47%	0,18%
132	Plant_2007_004948	230	0,12%	0,00%	1111	0,60%	0,19%
133	Plant_2008_000391	1156	0,64%	0,01%	5092	2,83%	0,13%
134	Plant_2008_002273	392	0,21%	0,00%	7440	4,00%	0,19%
135	Plant_2008_002379	1153	0,63%	0,01%	1980	1,09%	0,05%
136	scissors	1617	0,53%	0,01%	1852	0,61%	0,03%
137	sheep	584	0,22%	0,01%	4129	1,53%	0,04%
138	Sheep_2007_001416	190	0,12%	0,00%	2615	1,61%	0,11%
139	Sheep_2007_003593	324	0,20%	0,00%	6153	3,76%	0,21%
140	Sofa_2007_000661	506	0,27%	0,02%	3528	1,90%	0,13%
141	Sofa_2007_001149	600	0,32%	0,02%	7704	4,15%	0,16%
142	Sofa_2007_006530	255	0,14%	0,00%	1818	0,98%	0,08%
143	stone1	837	0,27%	0,01%	4686	1,53%	0,02%
144	stone2	772	0,25%	0,02%	3271	1,07%	0,03%
145	teddy	1383	1,22%	0,02%	2600	2,30%	0,12%
146	tennis	893	0,38%	0,00%	4692	2,00%	0,08%
147	Train_2007_000042	404	0,25%	0,00%	8346	5,10%	0,10%
148	Train_2007_000333	516	0,32%	0,01%	13411	8,26%	0,63%
149	Train_2007_002462	736	0,40%	0,01%	3575	1,94%	0,17%
150	Train_2007_003286	652	0,36%	0,01%	8557	4,68%	0,09%
151	Train_2007_004627	162	0,09%	0,01%	1409	0,76%	0,02%
		963	0,49%		5.856	3,14%	

Fonte: Elaborado pelo autor

Tabela B.2 – Segmentação das imagens - tempo (Segundos)

Nr	Imagem	Proposto		Testemunha	
		Tempo	Dv. Padrão	Tempo	Dv. Padrão
1	106024	323,19	58,5	997,39	102,67
2	153077	417,77	88,23	966,19	124,38
3	153093	380	10,94	867,65	100,9

Continua na próxima página

Tabela B.2 – continuação da página anterior

Nr	Imagem	Proposto		Testemunha	
		Tempo	Dv. Padrão	Tempo	Dv. Padrão
4	181079	448,75	105	872,39	104,12
5	189080	537,23	157,8	838,11	169,39
6	208001	407,56	132	756,87	65,5
7	209070	346,12	60,36	1109,79	237,78
8	21077	275,39	82,36	1069,18	161,4
9	227092	382,62	73,9	788,3	77,51
10	24077	296,77	45,24	1134,14	138,95
11	271008	463,77	108	1017,69	131,42
12	304074	356,16	84,13	897,56	113,01
13	326038	348,17	69,89	767	73
14	37073	385,52	69,96	1045	199,91
15	376043	647,5	139,18	1068,84	112,95
16	388016	347,51	69,03	820,87	143,3
17	65019	290,37	80,31	903,7	93,46
18	69020	472,67	18,87	1160,63	133,45
19	86016	634,87	129,15	601,82	72,8
20	aero_2007_000032	226,3	39,14	718,76	79,08
21	Aero_2007_001884	242,39	46,17	885,99	95,31
22	Aero_2007_006232	413,05	50,5	1288,55	194,13
23	Aero_2007_007341	417,51	49,7	1156,53	99,9
24	Aero_2007_009030	462,86	47,67	1132,68	212,96
25	Aero_2008_002358	510,81	83,66	1092,34	126,32
26	Baby_2007_005227	464,52	95,11	977,92	85,31
27	Baby_2007_006647	615,65	53,28	1179,81	108,06
28	Baby_2007_007203	293,78	56,12	964,22	178,21
29	banana1	953	182	1248,77	60,78
30	banana2	831,72	152,84	1328,55	112,64
31	banana3	881,29	110,12	1197,16	146,09
32	Bike_2007_000364	602,75	61,38	1408,59	148,92
33	Bike_2007_000822	315,26	51,27	1009,25	119,16
34	Bike_2007_004003	363,47	55,18	1243,45	127,24
35	Bike_2007_005354	359,61	59,02	1125,78	113,4
36	Bike_2007_005878	377,16	89,96	665,75	115,47
37	Bike_2007_008994	396,37	53,89	1012,45	73,65
38	Bike_2007_009084	440,69	6,37	1203,22	140,2

Continua na próxima página

Tabela B.2 – continuação da página anterior

Nr	Imagem	Proposto		Testemunha	
		Tempo	Dv. Padrão	Tempo	Dv. Padrão
39	Bike_2008_000782	458,2	77,49	1061,41	149,98
40	Bike_2008_002772	382,65	36,01	920,9	125,87
41	Bird_2007_000363	346,57	64,77	803,4	102,07
42	Bird_2008_002200	414,27	57,68	970,21	87,31
43	Boat_2007_000061	277,38	2,56	783,68	74,98
44	Boat_2007_000837	203,37	41,95	659,5	84,11
45	Boat_2007_009527	420,97	70,87	940,74	111,31
46	Boat_2008_001135	307,22	31,47	758,11	98,49
47	book	677,41	79,5	1131,46	51,81
48	bool	470,74	3,65	1096,11	106,92
49	Bus_2007_000727	328,1	73,47	1076,37	180,34
50	Bus_2007_002216	414,2	70,65	1238,54	239,37
51	Bus_2007_003101	425,24	76,36	1490,64	257,45
52	Bus_2007_003503	279,93	59,79	1270,3	265,49
53	Bus_2007_003786	273,66	43,42	719,74	134,02
54	Bus_2007_005262	260,94	38,51	1204,31	142,65
55	bush	373,7	34,99	1158,36	89,07
56	Car_2007_006900	409,43	42,89	1257,72	125,73
57	Car_2007_009779	316,82	65,76	1006,82	172,04
58	Car_2008_000595	323,18	33,49	1132,81	153,73
59	Car_2008_001716	273,7	34,1	842,3	97,41
60	Car_2008_002834	420,82	65,29	1082,84	76,29
61	Cat_2007_000528	407,99	77,86	1168,41	150,03
62	Cat_2007_000876	374,65	46,34	1143,49	166,56
63	Cat_2007_001185	405,62	60,68	996,66	87,35
64	Cat_2007_007530	328,42	51,12	918,4	114,35
65	Cat_2008_000345	444,4	68,25	1101,69	175,26
66	Cat_2008_001592	400,47	78,18	1022,32	85,3
67	Cat_2008_001885	555,53	68,92	1112,56	165,72
68	ceramic	838,99	171,41	1158,08	63,3
69	Chair_2008_000107	351,23	47,02	1302,21	110,39
70	Cow_2007_000464	224,88	26,77	778,75	69,76
71	cross	1238,05	238,78	1123,14	67,23
72	Cyclist_2007_000129	443,79	48,54	1316,71	131,05
73	Dog_2007_001239	478,45	62,25	866,47	93,39

Continua na próxima página

Tabela B.2 – continuação da página anterior

Nr	Imagem	Proposto		Testemunha	
		Tempo	Dv. Padrão	Tempo	Dv. Padrão
74	Dog_2008_000080	370,48	60,38	972,13	71,08
75	doll	716,31	123,9	1024	111
76	elefant	701,87	72,69	1133,08	100,47
77	flower	547,64	39,37	1031,89	111,06
78	fullmoon	298,05	58,66	1384,66	76,13
79	grave	471,56	100,21	1232,76	127,92
80	gt02	1098,05	38,84	1146,92	68,42
81	gt07	1567,95	20,95	1637,93	109,4
82	gt13	1505,62	90,38	1767,62	337,78
83	Horse_2007_003889	394,45	53,43	1365,88	155,93
84	Horse_2007_006134	449,69	65,13	1176,09	112,59
85	Horse_2008_000696	247,57	40,4	926,22	109,95
86	llama	633,84	187,16	1019,77	115,27
87	memorial	532,11	135,41	1029,79	78,18
88	Monitor_2007_000039	225,96	36,22	1071	158
89	Monitor_2007_000121	359,91	61,56	1184,12	126,6
90	Monitor_2007_000187	274,05	34,66	987,67	71,78
91	Monitor_2007_001678	181	25	888,54	103,48
92	Monitor_2007_003011	227	48	777,1	90,34
93	Monitor_2007_004193	199,79	8,84	927,99	161,08
94	music	826,29	176,05	1384,35	114,35
95	Person_2007_000170	495	67,75	1137,77	161,39
96	Person_2007_000323	491,95	55,87	1355,67	184,08
97	Person_2007_000392	389,74	80,89	1144,66	111,28
98	Person_2007_000480	334,91	56,53	1051,48	114,7
99	Person_2007_000733	445,2	67,6	1165,92	138,65
100	Person_2007_000783	419,53	60,87	1179,52	158,32
101	Person_2007_001423	376,81	47,12	961,11	83,13
102	Person_2007_001430	383,97	67,35	1183,74	81,88
103	Person_2007_001526	282,38	48,22	1346,05	171,93
104	Person_2007_002378	451,96	79,36	1211,02	157,01
105	Person_2007_002426	551,85	67,92	1091,6	88,32
106	Person_2007_002639	253,41	50,84	811,27	58,1
107	Person_2007_002823	356,11	89,23	939	164
108	Person_2007_004328	242,44	36,98	730,22	80,66

Continua na próxima página

Tabela B.2 – continuação da página anterior

Nr	Imagem	Proposto		Testemunha	
		Tempo	Dv. Padrão	Tempo	Dv. Padrão
109	Person_2007_004722	274,07	49,15	896,68	102,07
110	Person_2007_005086	392,03	62,44	1397,78	198,47
111	Person_2007_005430	260,91	46,52	778,44	92,79
112	Person_2007_006364	291,88	41,56	794,72	73,67
113	Person_2007_007996	223,04	35,81	1007,89	86,5
114	Person_2007_008218	412,99	61,03	1183,19	117,45
115	Person_2007_009327	411,41	72,91	839,46	102,6
116	Person_2007_009687	337,4	62,51	1591,45	222,81
117	Person_2008_000474	482,2	79,78	1189,22	133,39
118	Person_2008_000510	321,85	36,68	896,44	135,67
119	Person_2008_000584	319,63	46,23	1057,93	89,2
120	Person_2008_001375	308,55	61,3	1262,74	209,2
121	Person_2008_001547	356,11	66,3	1080,09	148,97
122	Person1	831,64	92,35	1283,2	110,93
123	person2	588,06	96,62	1387,25	67,8
124	person3	568,96	146,19	1380,57	167,87
125	person4	424,32	57,06	1573,89	115,1
126	person5	313,01	47,76	1415,13	157,58
127	person6	388,62	55,18	1610,56	126,98
128	person7	351,3	63	1458,77	185,74
129	person8	569,57	99,92	1665,37	129,63
130	Plant_2007_002967	212,26	39,4	1065,9	156,12
131	Plant_2007_004121	129,94	2,64	1099,84	139,52
132	Plant_2007_004948	128,21	8,75	1111,48	171,38
133	Plant_2008_000391	448,46	84,81	1186,25	173,93
134	Plant_2008_002273	264,22	47,07	1518,41	188,17
135	Plant_2008_002379	507,75	135,53	1024,99	117,11
136	scissors	701,84	64,16	1247,32	59,08
137	sheep	394,27	64,21	1128,85	103,78
138	Sheep_2007_001416	238,59	45,38	939,71	92,3
139	Sheep_2007_003593	250,77	2,3	1054,47	98,18
140	Sofa_2007_000661	305,94	59,29	964,99	130,24
141	Sofa_2007_001149	330,58	58,57	931,81	105,74
142	Sofa_2007_006530	325,56	75,82	789,55	85,88
143	stone1	570,21	116,35	1207,11	95,55

Continua na próxima página

Tabela B.2 – continuação da página anterior

Nr	Imagem	Proposto		Testemunha	
		Tempo	Dv. Padrão	Tempo	Dv. Padrão
144	stone2	617,33	82,4	1129,39	97,26
145	teddy	571,4	124,79	583,3	74,55
146	tennis	283,51	30,13	891,11	73,48
147	Train_2007_000042	266,94	2,1	1145,21	131,29
148	Train_2007_000333	342,93	62,31	1356,86	163,46
149	Train_2007_002462	395,91	68,6	955,59	123,94
150	Train_2007_003286	490,33	86,31	1166,96	161,83
151	Train_2007_004627	323,41	79,8	757,83	140,97
		432,54		1.082,94	

Fonte: Elaborado pelo autor