

Rafael Floriani Bertoldi

Método de Rede de Gradientes: um novo algoritmo de segmentação de imagens coloridas

Florianópolis – SC

Abril, 2008

Rafael Floriani Bertoldi

Método de Rede de Gradientes: um novo algoritmo de segmentação de imagens coloridas

Tese de mestrado de curso apresentado como
parte dos requisitos para obtenção do grau
de Mestre em Ciência da Computação.

Orientador:
Prof. Dr.rer.nat. Aldo von Wangenheim

UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

Florianópolis – SC

Abril, 2008

*Dedico à minha família
que, sem importância a distância
que nos separa, reside no
meu coração. A luz que me guia
pode estar distante, mas eu sempre
a vejo no horizonte e
é motivo o suficiente
pra eu seguir adiante.*

Agradecimentos

Dedico meus agradecimentos:

- ao professor doutor Aldo von Wangenheim, pela orientação, oportunidade e paciência;
- ao professor doutor Eros Comunello, pelo apoio, ajuda e compreensão;
- ao doutorando Daniel Abdalla, meu amigo Caju, que, mesmo com toda minha desorganização, me ajudou em muito momentos;
- aos colegas e amigos do LAPIX que me ajudaram dando conselhos, solucionando dúvidas ou simplesmente me divertindo;
- a todos os meus outros amigos. Vocês foram essenciais para eu conseguir superar os momentos mais difíceis;
- e, por fim, aos meus irmãos. Falar com eles sempre é uma lembrança importante daquilo que me forneceu forças pra conseguir chegar até aqui e o que me motiva a seguir adiante.

Sinceramente, obrigado a todos.

Sumário

Lista de Figuras

Lista de Tabelas

Resumo

Abstract

1	Introdução	p. 15
1.1	Motivação	p. 15
2	Estado da arte	p. 18
2.1	Segmentação de imagens	p. 18
2.2	Métodos de avaliação da qualidade de segmentação de imagens	p. 24
3	Metodologia	p. 29
3.1	Princípio geral	p. 29
3.2	Análise de requisitos	p. 29
3.2.1	Representação inicial da imagem	p. 30
3.2.2	Estruturação dos dados	p. 31
3.2.3	Medida de similaridade	p. 32

3.3	Descrição do Método de Rede de Gradientes - GNM	p. 36
3.3.1	Pós-processamento: eliminação de fragmentos	p. 42
4	Modelagem e Implementação	p. 45
4.1	Cumprimento dos requisitos	p. 46
4.1.1	Portabilidade	p. 46
4.1.2	Suporte aos formatos de imagem desejados	p. 46
4.1.3	Configuração de parâmetros	p. 47
4.1.4	Documentação	p. 47
4.2	Implementação da versão de testes	p. 48
4.2.1	Detalhes sobre módulo liblabeling	p. 48
4.2.2	Detalhes sobre módulo libgnm	p. 50
4.2.3	Detalhes sobre módulo gnmexe	p. 58
4.3	Eficiência e outros requisitos não-funcionais	p. 59
4.3.1	A complexidade do GNM	p. 59
4.3.2	A complexidade do passo de pós-processamento	p. 60
4.3.3	Sobre os algoritmos usados como pré-segmentação	p. 61
5	Resultados	p. 63
5.1	Avaliação da qualidade de imagem das segmentações	p. 63
5.1.1	Experimento	p. 63
5.1.2	Avaliação	p. 83
5.2	Resultados dos testes de desempenho de velocidade	p. 89

6 Discussão	p. 92
7 Conclusão	p. 94
Anexo A - Anexos	p. 96
A.1 Parâmetros dos algoritmos testados com as imagens selecionadas e avaliadas	p. 96
A.2 Resultados obtidos com cada conjunto de imagens para cada um de seus respectivos ground truths	p. 98
A.2.1 Rand	p. 98
A.2.2 BGM	p. 110
A.3 Artigo aceito pela publicação Pattern Recognition Letters	p. 122
A.4 Artigo submetido para a publicação Pattern Recognition	p. 132
Referências	p. 146

Listas de Figuras

- 1 Uma imagem bem-organizada onde a ordem é dada por gradientes estruturados mais do que por mudanças abruptas, mas duas diferentes segmentações dessa mesma imagem. p. 16
- 2 Estrutura de ilhas hierárquicas do *Color Structure Code* (PRIESE; STURM, 2003) p. 21
- 3 Mapa de classes e seus valores de J correspondentes (DENG; MANJUNATH, 2001). Observado, especialmente, o mapa de classes no centro pode se ver por que o JSEG é robusto a presença de texturas. p. 24
- 4 Imagem comparando a pré-segmentação de uma mesma imagem entre as técnicas Mumford & Shah e Color Structure Code. A segmentação fornecida por M&S em geral tende a ser menos agressiva e conservando mais gradientes suaves, dando um aspecto mais “escamoso” a imagem. CSC produz resultados menos conservadores, mas bons o suficiente para serem usados como passo inicial, especialmente levando em consideração a velocidade desta técnica. p. 31
- 5 Uma amostra de como fica a organização das imagens após serem rotuladas e estruturadas como grafos conexos pelo algoritmo GNM. p. 32
- 6 Visão geral do espaço de cores HSL. A relação entre seus eixos faz este espaço de cores tomar a forma de um duplo cone. p. 34

- 7 Outra forma de visualizar o espaço de cores HSL. À esquerda pode ser observada a natureza cíclica das matizes de cores e como a variação de saturação as afetam. À direita a influência da variação de luminosidade e saturação. (Imagen criada por Mike Horvath sob a licença Creative Commons.) p. 36
- 8 Diagrama representando o processo do algoritmo do Método de Redes de Gradientes (GNM) p. 42
- 9 GNM comparado com outros métodos de segmentação usando a imagem 2092 da base de dados de testes de segmentação (MARTIN et al., 2001). . . p. 67
- 10 GNM comparado com outros métodos de segmentação usando a imagem 15088 da base de dados de testes de segmentação (MARTIN et al., 2001). p. 68
- 11 GNM comparado com outros métodos de segmentação usando a imagem 22090 da base de dados de testes de segmentação (MARTIN et al., 2001). p. 69
- 12 GNM comparado com outros métodos de segmentação usando a imagem 24004 da base de dados de testes de segmentação (MARTIN et al., 2001). p. 70
- 13 GNM comparado com outros métodos de segmentação usando a imagem 42049 da base de dados de testes de segmentação (MARTIN et al., 2001). p. 71
- 14 GNM comparado com outros métodos de segmentação usando a imagem 46076 da base de dados de testes de segmentação (MARTIN et al., 2001). p. 72
- 15 GNM comparado com outros métodos de segmentação usando a imagem 48055 da base de dados de testes de segmentação (MARTIN et al., 2001). p. 73
- 16 GNM comparado com outros métodos de segmentação usando a imagem 60079 da base de dados de testes de segmentação (MARTIN et al., 2001). p. 74
- 17 GNM comparado com outros métodos de segmentação usando a imagem 68077 da base de dados de testes de segmentação (MARTIN et al., 2001). p. 75

Listas de Tabelas

1	Valores médios obtidos nos testes com os conjuntos de imagens utilizando o índice de avaliação Rand.	p. 84
2	Valores médios obtidos nos testes com os conjuntos de imagens utilizando o índice de avaliação BGM.	p. 85
3	Valores médios e desvios padrões dos algoritmos testados com os índices de avaliação Rand e BGM.	p. 86
4	O tempo de execução total para o conjunto de imagens do conjunto de dados de Berkeley processados usando um conjunto de técnicas selecionadas. O tempo médio e desvio padrão são mostrados nas últimas linhas da tabela.	p. 90
5	Os parâmetros específicos de cada algoritmo testado que foram selecionados como aqueles que produzem uma melhor segmentação seguindo critérios apresentados em seções anteriores.	p. 97
6	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por CSC+GNM.	p. 99
7	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por M&S+GNM.	p. 100
8	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por Edison+GNM.	p. 101
9	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por WS+GNM.	p. 102

10	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por <i>Color Structure Code</i>	p. 103
11	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por Mumford & Shah.	p. 104
12	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por Edison.	p. 105
13	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por <i>Watershed</i>	p. 106
14	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por JSEG.	p. 107
15	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por RHSEG.	p. 108
16	Valores da avaliação de Rand para todos os <i>ground-truths</i> comparados com os resultados produzidos por Blobworld.	p. 109
17	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por CSC+GNM.	p. 111
18	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por M&S+GNM.	p. 112
19	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por Edison+GNM.	p. 113
20	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por WS+GNM.	p. 114
21	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por <i>Color Structured Code</i>	p. 115

22	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por Mumford & Shah.	p. 116
23	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por Edison.	p. 117
24	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por <i>Watershed</i>	p. 118
25	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por JSEG.	p. 119
26	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por RHSEG.	p. 120
27	Valores da avaliação de BGM para todos os <i>ground-truths</i> comparados com os resultados produzidos por Blobworld.	p. 121

Resumo

Um novo algoritmo de segmentação por crescimento de regiões para imagens coloridas é proposto. Diferente de outras técnicas, especialmente de crescimento de regiões, o foco principal não será a homogeneidade de pixels ou texturas. O algoritmo proposto é uma solução robusta para a segmentação de imagens que apresentem objetos formados por variações lentas e contínuas de cores ao longo da cena de uma imagem. Um exemplo é o céu azul em um dia ensolarado e seus vários tons de “azuis”. Muitas técnicas apresentam segmentações incorretas pela dificuldade da existência de objetos representados por uma faixa tão extensa de cores. A técnica proposta será apresentada e validada empiricamente, demonstrando que mesmo com sua abordagem simples, resultados satisfatórios são obtidos, mostrando ser comparativamente tão bom quanto outras técnicas tradicionais de crescimento de regiões. Os resultados da implementação do algoritmo também serão avaliados comparativamente em relação a seu desempenho com outras técnicas do estado da arte de crescimento de regiões.

Abstract

A new region growing segmentation algorithm for color images is proposed. Different to other techniques, especially region growing ones, the main goal is not the homogeneity of pixels or textures. The proposed algorithm is a robust solution to the segmentation of images that show objects formed by slowly varying and continuous color changes in their scenes. An example is the blue sky in a sunny day and its various shades of “blues”. Many techniques produce incorrect segmentation because of the existence of objects represented by a range so long of colors. The proposed technique is presented and validated empirically, showing that is as good as other traditional region-growing techniques. The algorithm implementation performance results will also be evaluated in comparison to state-of-art region-growing algorithms.

1 Introdução

Algoritmos de segmentação de imagens coloridas são um campo de aplicação muito importante em análise de imagens e visão computacional. Seu principal objetivo é identificar regiões homogêneas ou regiões de características similares em uma imagem de acordo com um aspecto predefinido, especialmente em relação a seus valores de cor. Esta família de algoritmos torna-se muito importante quando se busca atingir um resultado que alguém possua comparar diretamente com o obtido pela visão humana. Um problema essencial de algoritmos atuais é que a identificação de similaridade é calculada sobre a continuidade simples de similaridade entre pixels. Esta abordagem leva a erros quando se considera, por exemplo, o céu azul num dia sem nuvens. Nessas condições, geralmente o céu apresenta um azul de intensidade mais escura nas partes mais altas da imagem e gradualmente e continuamente decrescem para um azul de intensidade mais clara nas partes inferiores. Esta variação de intensidade poderá ser identificada como numerosas regiões em muitos dos algoritmos atuais, enquanto o olho humano vê somente uma região, o céu especificamente, reconhecendo a ordem na variação. O conceito de uma segmentação de imagens baseada na análise das relações de gradientes contínuos busca transpor esta dificuldade. A abordagem que será apresentada se chama Método de Rede de Gradientes (*Gradient Network Method - GNM*).

1.1 Motivação

Algoritmos tradicionais de segmentação por crescimento de regiões, de modo a produzir o melhor resultado possível de uma imagem se baseiam em uma busca por ilhas

de alta entropia em uma imagem, ou seja, regiões homogêneas onde nenhuma ou pouca mudança ocorrem. O algoritmo de segmentação Mumford & Shah (MUMFORD; SHAH, 1989) é o melhor exemplo desta metodologia. O mais comum algoritmo de *Watershed* (VINCENT; SOILLE, 1991) e suas implementações mais elaboradas (MITTELHAUSSER; KRUGGEL, 1995), mesmo se elas usassem gradientes de imagens para definir fronteiras de regiões, não são muito diferentes. Isto leva a uma situação onde imagens que mostram uma clara ordem, mas onde esta ordem é representada pela mudança de padrões de cores ou intensidade de luz, tem resultados de segmentação fracos, veja Figura 1.

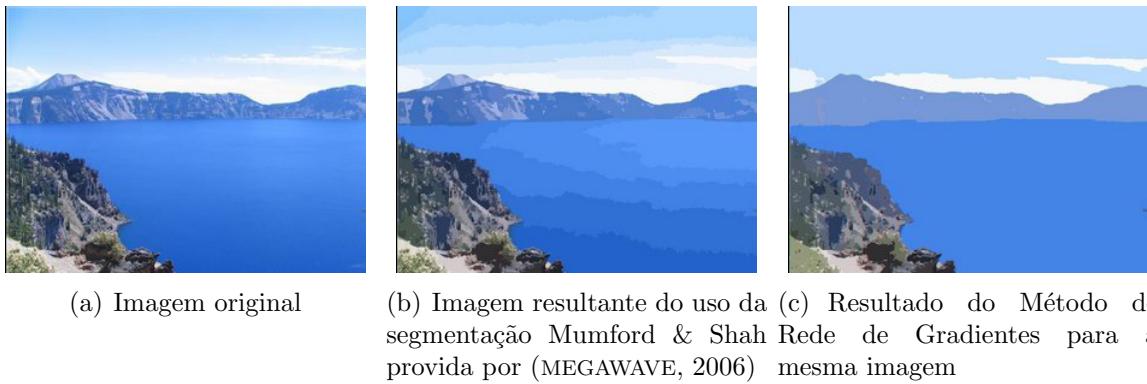


Figura 1: Uma imagem bem-organizada onde a ordem é dada por gradientes estruturados mais do que por mudanças abruptas, mas duas diferentes segmentações dessa mesma imagem.

Por outro lado, quando nós humanos observamos uma cena com o céu azul, onde os tons de azul mudam gradualmente a partir do horizonte, a ordem é óbvia e nós nem mesmo tentamos interpretar o céu como uma série de regiões de diferentes “azuis”. Nós reconhecemos a ordem na dinâmica das cores e identificamos o céu como uma simples estrutura na imagem. Isto ocorre também como imagens similares onde nós não temos conhecimento adicional, tais como formas curvas coloridas que nunca foram vistas anteriormente, onde uma iluminação irregular produz diferentes formas de cor e intensidade de luz. Mas algoritmos tradicionais de segmentação não são capazes de realizar esta operação simples e independente de conhecimento prévio. A cena da Figura 1(b) ilustra isso, segmentada com a implementação do algoritmo Mumford & Shah para espaço de cores encontrada no mais recente pacote de ferramentas de processamento de imagens

Megawave (MEGAWAVE, 2006), onde o resultado mostra o lago com mais segmentos do que o esperado. Se nós modificarmos os parâmetros de modo radical para que o céu ou lago sejam segmentados como um único objeto, isto produzirá vazamentos em outras partes da imagem. Porém, se a mudança gradual de padrões da imagem for levada em conta, pode se obter um resultado de segmentação tal como mostrado na Figura 1(c). Muitas abordagens foram desenvolvidas nos últimos 15 anos para melhorar a performance da tradicional segmentação por crescimento de regiões em cenas naturais, indo desde simples abordagens hierárquicas (PRIESE; REHRMANN, 1993)(REHRMANN; PRIESE, 1998) até algumas mais elaboradas (DUPUIS; VASSEUR, 2006). Métodos mais recentes focam não somente na identificação de regiões de cores bastante similares, mas também objetos formados por texturas (DENG; MANJUNATH, 2001)(DUPUIS; VASSEUR, 2006)(KATO; PONG, 2006). Mas mesmo com o uso de técnicas capazes de identificar objetos de cor similar e mesmo texturizados, há ainda outra característica que atrapalha algoritmos de segmentação de imagens: luminosidade e a existência de extensas mas lentas e graduais variações de cor para um mesmo objeto em uma imagem, resultando em grandes gradientes de baixa variação local. Algumas abordagens têm buscado resolver o problema usando diferentes métodos de avaliação de cromacidade e luminosidade (DONY; WESOLKOWSKI, 1999)(SCHNEIDER et al., 2000), mas as soluções propostas não são totalmente robustas. Aqui é proposta uma nova técnica pós-segmentação para processar imagens pré-segmentadas através da análise de redes de gradientes que descrevem padrões de variação de cores entre segmentos, um método especialmente robusto quando aplicado para objetos de grande e contínua variação de cor em uma imagem, tal como ocorre como imagens ao ar livre.

2 Estado da arte

2.1 Segmentação de imagens

As primeiras pesquisas em segmentação de imagens focaram-se em imagens em tons de cinza devido a exigência computacional imposta por algoritmos de processamento digital de imagens e a dificuldade de obter recursos computacionais suficientes pra lidar com problemas mais complexos. Bons resultados foram atingidos (VINCENT; SOILLE, 1991), mas com a evolução tecnológica aliada a necessidade de métodos capazes de tratar imagens coloridas, algoritmos preparados para processar problemas de maior dimensionalidade foram desenvolvidos. Todavia, tal como no caso de imagens de tons de cinza, uma solução de fato ainda não foi encontrada. O problema de segmentação de imagens coloridas pode ser considerado como um super-conjunto ou uma abordagem de maior dimensionalidade da segmentação em tons de cinza. Na verdade, uma considerável quantidade de algoritmos para imagens coloridas são simplesmente extensões de métodos para tons de cinza. A complexidade maior da maior dimensionalidade das imagens coloridas torna ainda mais difícil a busca por uma solução, especialmente porque algoritmos para imagens coloridas têm que lidar simultaneamente com várias características, como cromacidade e luminosidade. A presença simultânea dessas diferentes características dificulta extensões diretas de métodos em tons de cinza para imagens coloridas. O modo como os algoritmos lidam com esta relação entre as características de cor é essencial para a qualidade dos resultados das técnicas para imagens coloridas.

As primeiras tentativas de resolver o problema de segmentação de imagens coloridas

tentaram aperfeiçoar técnicas usadas em algoritmos para tons de cinza, tais como, por exemplo, limiarização de histograma, clusterização, detecção de bordas, crescimento de regiões, técnicas *fuzzy* e redes neurais. Em vez de usar somente o gradiente da escala de cinza, eles usam todos os canais que representam a cor, geralmente representados por três eixos. Boas revisões sobre técnicas podem ser encontradas em (SKARBEK; KOSCHAN, 1994) (PAL; PAL, 1993) (LUCCHESE; MITRA, 2001) (CHENG et al., 2001). A insatisfação com os resultados iniciais, especialmente em relação ao uso do espaço de cores RGB, levou a pesquisa de técnicas em diferentes espaços de cores. (GAUCH; HSIA, 1992) apresenta uma comparação entre três diferentes algoritmos em quatro espaços de cores. Algoritmos de segmentação tenderam cada vez mais a usar espaços de cores mais perceptivamente próximos àquele observado pelo ser humano. Espaços de cores diferenciando cromacidade e características de iluminação de diversas formas se tornaram populares. Exemplos são HSL, HSV, CIE l^*a^*b e CIE l^*u^*v . Uma avaliação interessante desses espaços de cores e alguns outros podem ser vistas em (CHENG et al., 2001). Em (GEVERS; SMEULDERS, 1999) um estudo sobre características encontradas em diversos espaços de cores, como tonalidade e luminosidade, são classificados de acordo com a propriedades como invariância da direção de visão ou iluminação e controle sobre condições como ofuscamentos. (ANGULO; SERRA, 2007) também faz uma avaliação interessante quanto a espaços de cores perceptivos, inclusive propondo também uma técnica de segmentação baseada no espaço de cores L1 de forma a produzir uma segmentação robusta combinando fatores de cromacidade e iluminação.

Abordagens baseadas em modelos físicos tornaram-se interessantes com a popularização de imagens coloridas, devido ao acréscimo de informações disponíveis. A modelagem e representação de fenômenos físicos necessitam, porém, de um detalhamento elevado, prejudicando a generalização dessas técnicas. Vale salientar que técnicas baseadas em modelos físicos, mesmo que restritivas, são bastante eficientes em relação a problemas como ofuscamento e sombreamento para o conjunto de imagens a que se propõem trabalhar. Um exemplo tradicional é (KLINKER; SHAFER; KANADE, 1990), onde uma técnica de

segmentação baseada no modelo dicromático de reflexão é apresentada. Outros trabalhos seguiram e tentaram aperfeiçoar o uso desse modelo como (TSANG; TSANG, 1996) e outros modelos também foram apresentados (HEALEY, 1992). A dificuldade de se encontrar um modelo físico para descrever segmentação de imagens pode também ser associado ao fato que “o problema de segmentação de imagens está basicamente na percepção psicológica, e dessa forma não é susceptível a uma análise puramente analítica” (FU; MUI, 1981), restando a questão se algum dia será possível atingir “imitações” de sistemas biológicos de visão.

Uma das primeiras abordagens e bastante interessante para imagens coloridas foi o algoritmo de *watershed* (VINCENT; SOILLE, 1991). É uma técnica que foi desenvolvida a partir de técnicas anteriores usadas em imagens em tons de cinza. Classifica pixels em regiões levando em consideração descidas de gradiente de modo a preencher as bordas encontradas na imagem segmentada. Essa abordagem quando aplicada a imagens com objetos formados por longos e suaves gradientes, porém, encontra problemas pela definição pouco clara das fronteiras entre as regiões da imagem. Isso pode resultar em segmentações que, dependendo da agressividade aplicada, resultam na identificação de objetos não existentes ou a fusão errônea de objetos.

Uma abordagem de destaque para segmentação de imagens coloridas é o *Color Structure Code* (CSC) (PRIESE; REHRMANN, 1993) (REHRMANN; PRIESE, 1998). É uma técnica de crescimento de regiões que usa uma topologia hierárquica formada por ilha, topologia essa introduzida por (HARTMANN, 1987). Estas ilhas têm diferentes níveis. Por exemplo, uma ilha de nível 0 é um hexágono, composto pelos seis vértices mais o ponto central. Durante o processo, algumas ilhas sobrepõem outras de forma que ilhas de nível $n + 1$ são compostas pela composição de sete ilhas de nível n sobrepostas. Este procedimento é repetido até que uma ilha cubra a imagem inteira.

O primeiro passo, então, é a divisão particionada de uma imagem em ilhas de nível 0. Um passo de fusão seguirá onde as ilhas cresceram e serão sobrepostas iterativa-

mente. Em seguida, ocorre um passo de divisão, onde correções serão feitas através do uso de informação global da imagem. Assim, CSC combina passos iniciais baseados em informação local seguido por uma avaliação baseada em informação global, buscando segmentar regiões com o maior grau de similaridade possível.

Em relação a imagens ao ar livre, CSC é uma boa alternativa para o tratamento deste tipo de imagem. Porém, como a abordagem não tem enfoque algum em observar variações extensas porém lentas de cores em um mesmo objeto, problemas de segmentação podem ocorrer. As consequências, como na maioria dos algoritmos, são a fusão desnecessária de regiões com gradientes mais sensíveis ou parâmetros cautelosos produzem a divisão de um mesmo objeto em vários segmentos.

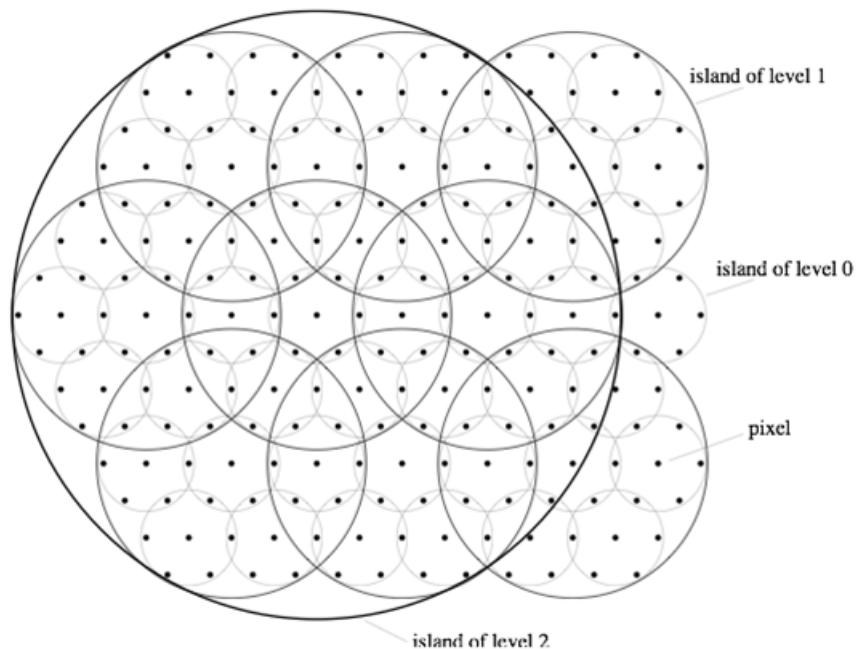


Figura 2: Estrutura de ilhas hierárquicas do *Color Structure Code* (PRIESE; STURM, 2003)

A mais tradicional abordagem em termos de segmentação em termos de detecção de homogeneidade por ilhas de similaridade é a técnica baseada na funcional proposta por Mumford e Shah (MUMFORD; SHAH, 1989). Quanto menor for a energia obtida nesta funcional, melhor é a segmentação, lidando com o problema de segmentação na elegante forma da redução de energia global. Tal funcional de energia é mostrada na equação 2.1.

$$E(f, \Gamma) = \int_{\Omega} |f - g|^2 dx dy + \mu \int_{\Omega \setminus \Gamma} |\nabla f|^2 dx dy + \nu |\Gamma| \quad (2.1)$$

A equação tem como três principais objetivos forçar a aproximação entre f e g , não permitindo uma grande variação de f para cada região e forçando as fronteiras entre as regiões para serem reduzidas o quanto for possível. O resultado da equação equivalerá a imagem segmentada em regiões bem definidas e semelhantes.

A dificuldade da utilização da equação 2.1 em segmentação de imagens reside no desconhecimento da função f e da possibilidade de existência de mínimos locais na funcional.

Uma versão simplificada da equação, proposta pelos próprios autores, é mostrada na equação 2.2. A existência e regularidade dos minimizadores foram provadas teoricamente pelos autores.

$$E(f, \Gamma) = \sum_i \int_{\Omega_i} (g - c_i)^2 dx dy + \nu |\Gamma| \quad (2.2)$$

A abordagem da funcional de Mumford&Shah resulta em uma segmentação que tende formar regiões de homogeneidade em relação até um certo limite de “elasticidade” de similaridade. Essa é uma abordagem bastante interessante e, da forma como é realizada pela funcional, poderosa, mas para alguns tipos de cenas essa solução não é ideal. Imagens que apresentem objetos formados por vários tons variando ao longo da imagem costumam apresentar dificuldade para ser corretamente identificados, pois para um correta segmentação necessitam de uma extração de parâmetros que acaba vazando para regiões mais sensíveis. Parâmetros cautelosos causam a separação errada de um único objeto em várias regiões.

Uma abordagem mais recente de destaque na formação de ilhas de semelhança como segmentação é realizada através de uma técnica de reconhecimento de padrões chamada de deslocamento-médio (*mean-shift*). (COMANICIU; MEER, 2002) usaram essa abordagem para a criação de um método adaptativo de ascendência de gradientes, buscando regiões

que convirjam para um máximo local na função de densidade criada a partir da imagem a ser segmentada. A capacidade de identificação de tendências semelhantes na variação de cores torna os resultados obtidos robustos em relação a textura e também a objetos formados por uma longa e lenta variação de cores.

Segmentações conhecidas como contornos ativos (XU; PRINCE, 1998) (CASELLES; KIM-MEL; SAPIRO, 1997) (WEICKERT, 2001) (CHAN; VESE, 2002), baseadas em level sets ou *snakes*, fornecem uma estruturação semelhante a provida por um *framework* para segmentações em geral semi-automáticas, atingindo várias resultados interessantes. Contornos ativos permitem a combinação de um modelo que define contração e expansão em busca da identificação de uma região de acordo com a funcional de avaliação especificada. É reconhecido que técnicas desse tipo, em geral, são extremamente independentes da inicialização. Também necessitam de aperfeiçoamentos, extensões para que segmentam uma imagem em mais que dois objetos.

Outro ramo da segmentação de imagens coloridas bastante pesquisado é a questão da segmentação robusta na presença de imagens fortemente texturizadas. Exemplos de abordagem são técnicas usando campos aleatórios de Markov (KATO; PONG, 2006) e utilizando conjuntos de pistas para obter tal propósito (DUPUIS; VASSEUR, 2006). Em geral, estes algoritmos atuam bem na detecção de objetos texturizados, mas ainda apresentam problemas com a questão de detectar objetos formados por longas e contínuas variações de cores. Outro exemplo interessante de técnicas robustas a textura é o JSEG (DENG; MANJUNATH, 2001).

O JSEG age em dois passos principais: uma quantização de cores que produz uma classificação da imagem, seguida de uma avaliação realizando um crescimento de regiões. A quantização de cores fará com que os valores de cor sejam substituídos por classes. A boa organização espacial dessas classes é avaliada através da determinação de um chamado valor J , valor esse que remete diretamente ao nome da técnica. Quanto menor o valor de J mais provável é que um conjunto de classes corresponda a uma região homogênea

na cena da imagem. O crescimento de regiões será realizado de forma a unir regiões em termos de semelhança de cor, usando o espaço de cores CIE l^*u^*v , que reduzam o valor de J , indicando uma boa segmentação. Esse processo é repetido iterativamente em várias escalas até o melhor resultado esperado ou possível for encontrado.

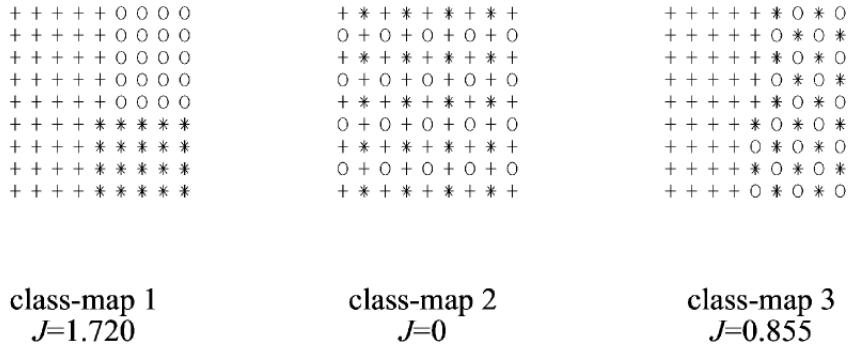


Figura 3: Mapa de classes e seus valores de J correspondentes (DENG; MANJUNATH, 2001). Observado, especialmente, o mapa de classes no centro pode se ver por que o JSEG é robusto a presença de texturas.

A avaliação espacial realizada no cálculo dos valores J torna o JSEG uma técnica robusta a objetos texturizados. Porém, essa avaliação não é suficiente para tornar esta uma técnica robusta a presença de fatores de iluminação e sombreamento. O reconhecimento dessa limitação é feito pelos próprios autores, avaliando a dificuldade da identificação de objetos formados por várias faixas de cores semelhantes e da ambigüidade de como proceder para identificar tais regiões sem fazer avaliações equivocadas.

2.2 Métodos de avaliação da qualidade de segmentação de imagens

Sabendo que técnicas de segmentação de imagens buscam particionar cenas em regiões correlacionadas de alguma maneira, a qualidade de uma segmentação pode ser classificada quanto correlacionadas as partições criadas são em relação aos objetos realmente presentes na cena da imagem.

A questão de acessar diretamente o nível dessa qualidade do resultado em segmentação de imagens, porém, não é trivial. O problema essencial é a definição clara do que car-

acteriza uma segmentação como melhor que outra. Por exemplo, diferentes observadores podem dar diferentes graus de qualidade para uma mesma segmentação, vendo diferentes objetos ou um mesmo objeto de diferentes formas em uma mesma cena.

Outro problema é que a simples superposição dos objetos segmentados e dos objetos esperados não é uma solução viável. A dificuldade nesse caso provém de como e quanto se deve penalizar regiões super ou sub-segmentadas ou pequenos vazamentos que as regiões segmentadas possam apresentar, especialmente porque o resultado de segmentações difficilmente será completamente preciso em relação as posições de pixels.

A análise matemática das segmentações também é inviável em geral devido a complexidade dos métodos usados no particionamento das imagens. Assim, avaliações genéricas usando esse tipo de abordagem são desconsideradas.

Uma última questão que deve ser levada ainda em consideração é que além de permitir avaliar o resultado da própria segmentação, espera-se que o resultado avaliado seja compatível e fácil de ser comparado com avaliações geradas a partir de outras segmentações, mesmo que estas usem paradigmas diferentes.

A motivação de criar medidas para simplificar a comparação de segmentações resultantes de diferentes algoritmos levou a criação de várias medidas de avaliação, apesar de o número destas medidas seja proporcionalmente muito menor que o número de diferentes técnicas e formas de segmentações disponíveis. Uma razão provável para esse fato é que ainda atualmente muitos autores ainda não utilizam medidas de avaliação na validação do resultados de suas técnicas, geralmente ainda encontrando muitos trabalhos que usam avaliações empíricas e visuais para esse fim.

Uma forma de avaliação de segmentação de imagens é a avaliação experimental. Essa forma de avaliação pode ser separada em dois grupos:

- **Baseada em características:** a avaliação realiza-se verificando a diferença entre a segmentação e um conjunto de propriedades desejáveis.

- **Baseada em tarefa:** considerando que a segmentação de imagens é um passo usado para atingir uma meta de maior nível, a segmentação é avaliada pela verificação da influência desta no sistema em questão.

Avaliações baseada em tarefa são específicas em relação a aplicação, enquanto as baseadas em características são mais versáteis. Em geral, as avaliações baseadas em características fazem uso de imagens de *ground truth* para realizar a verificação da qualidade. Essas imagens são imagens onde os objetos na cena são marcados por observadores humanos. Duas características que implicam diretamente desse tipo de avaliação são a fácil correlação da avaliação obtida com o resultado esperado pelo observador e fornece ainda uma forma simples de comparar algoritmos de segmentação quaisquer, pois qualquer seja o algoritmo a avaliação é feita em relação a um mesmo alvo e o resultado desta pode facilmente ser comparado. Exemplos dessas avaliações podem ser vistos em (JIANG et al., 2006), com destaque ao Rand, uma medida tradicional para esse tipo de comparação, e a correspondência por grafos bipartidos (*Bipartite Graph Matching - BGM*), que utiliza uma abordagem distinta a do Rand.

O índice Rand (RAND, 1971) é uma medida de similaridade criada para avaliação da qualidade de algoritmos que produzam resultados na forma de *clusters*. Para esse fim, realiza a comparação com outros resultados ou com um padrão-ouro, como, por exemplo, um *ground truth*. A comparação entre os 2 conjuntos de *clusters* $C_1 = \{c_{11}, c_{12}, \dots, c_{1N}\}$ e $C_2 = \{c_{21}, c_{22}, \dots, c_{2M}\}$ sobre uma mesma imagem $P = \{p_1, p_2, \dots, p_K\}$ onde cada elemento de C_1 ou C_2 é um subconjunto de P e $c_{1j} = \{p_{1j}, p_{2j}, \dots, p_{lj}\}$, calculam-se as seguintes quantidades:

- N_{11} : o número de pixels no mesmo *cluster* em ambos C_1 e C_2 .
- N_{00} : o número de pixels em diferentes *clusters* em ambos C_1 e C_2 .

O índice Rand é, então, definido pela equação 2.3, sabendo que n é a cardinalidade do conjunto P :

$$R(C_1, C_2) = 1 - \frac{N_{11} + N_{00}}{\frac{n(n-1)}{2}} \quad (2.3)$$

O índice BGM (JIANG et al., 2006) verifica a correlação de um-para-um entre os clusters enquanto simultaneamente tenta maximizar esta relação. O BGM considera cada *cluster* de C_1 e C_2 como vértices de um grafo bipartido. As arestas são adicionadas entre cada vértice e então recebem o valor $|c_{1i} \cap c_{2j}|$, um valor que pode ser adquirido diretamente da matriz de correspondência. Então, o grafo bipartido de peso máximo é definido com um subgrafo $\{(c_{1i1}, c_{2j1}), \dots, (c_{1ir}, c_{2jr})\}$, onde estarão presentes somente as arestas de c_{1i} até c_{2j} com máximo peso. Após todas essas arestas serem encontradas, o valor do índice BGM 2.4 será calculado dividindo a soma do peso das arestas restantes w pela cardinalidade n do conjunto P .

$$BGM(C_1, C_2) = 1 - \frac{w}{n} \quad (2.4)$$

Os dois índices apresentados, Rand e BGM, realizam a avaliação da qualidade de uma imagem em relação a um único alvo, em uma relação de um-para-um. Uma medida de avaliação derivada do Rand foi desenvolvida de modo a permitir a comparação direta de uma imagem com um ou mais *ground truths*, uma qualidade bastante desejável, pois a comparação com um único alvo pode ser muito limitada dependendo da aplicação. Essa medida de avaliação, chamada Rand Probabilístico Normalizado (*Normalized Probabilistic Rand - NPR*) (UNNIKRISHNAN; PANTOFARU; HEBERT, 2007), cria um teto máximo para a qualidade da avaliação de acordo com o nível de acordância entre os diferentes *ground truths*, fornecendo uma avaliação que automaticamente oferece uma forma de lidar com a possível ambigüidade entre esses diferentes alvos.

Existem ainda formas de avaliação baseadas em características experimentais não baseadas em *ground truths*. Um exemplo é (FERNANDEZ-GARCIA et al., 2008), onde a avaliação da qualidade do resultado de segmentações é verificada de acordo com um definido nível de consenso entre o resultado de diversos algoritmos. O nível de consenso

é usado na comparação entre os resultados de um definido grupo de segmentações e, de acordo com este, define-se um *ground truth* automaticamente, refletindo a coerência entre os resultados dos diversos algoritmos. Esse alvo gerado automaticamente pode, então, ser usado para realizar a comparação com outras imagens segmentadas. Esse tipo de medida oferece a propriedade de não sofrer da ambigüidade da avaliação humana na definição de *ground truths*, mas ao mesmo tempo terá sua qualidade dependente diretamente da qualidade geral dos algoritmos utilizados na geração das imagens alvo.

3 Metodologia

3.1 Princípio geral

A meta principal da abordagem por redes de gradientes é descrever a dinâmica da mudança de cores e fazer uso dessa informação em um algoritmo de segmentação, tentando reconhecer uma ordem superior de organização em uma imagem, especificamente a organização das mudanças ou a estrutura das tendências em uma imagem. A idéia geral para alcançar tal objetivo é usar um grafo que representa a natureza das modificações entre regiões da imagem na forma de uma função de custo variável para diferentes intervalos. Este grafo é então processado através de um algoritmo que funde regiões que apresentem uma mesma dinâmica dentro da imagem, isto é, que estejam dentro de uma tendência aceitável, tal como uma longa todavia contínua variação de subida ou descida. O algoritmo foi desenvolvido para ser capaz de representar as diferenças cromáticas entre regiões como custos e para reconhecer ordem mesmo na presença de mudança de padrões dentro de uma imagem, unindo regiões que pertencem a uma mesma “estrutura ordenada” do grafo. O método proposto é um *framework* geral, e alguns de seus componentes citados a seguir podem ser mudados através de outras técnicas.

3.2 Análise de requisitos

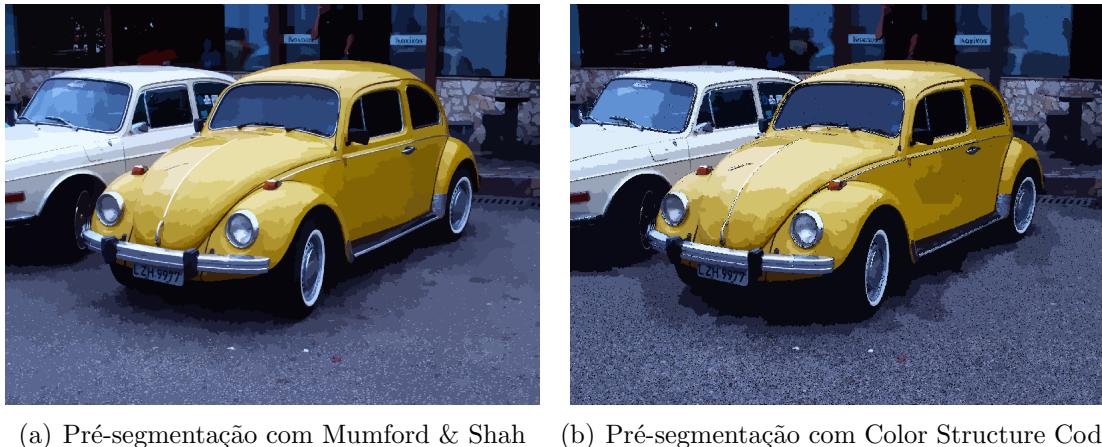
De forma a tentar atingir a meta descrita na seção anterior, foram definidos um conjunto de requisitos envolvendo o algoritmo. A seguir, estes requisitos e como foram abordados:

3.2.1 Representação inicial da imagem

Representar cada simples pixel como um nodo em um grafo com um conjunto de parâmetros é uma solução custosa e pode introduzir ruídos locais que não contribuem para a estimativa regional da tendência da variação de cores. Assim será feito o uso de uma segmentação por regiões como um passo de pré-processamento, que irá representar cada região que é estatisticamente bastante semelhante, ou seja, homogênea como um simples objeto dentro um conjunto determinado de parâmetros. Este conjunto de parâmetros será dependente do domínio e do método de pré-segmentação e será discutido adiante. Cada objeto gerado por esta pré-segmentação será considerado uma região atômica. Esta segmentação deve ser realizada com parâmetros parcimônios que resultem em uma imagem super-segmentada e com o menor número possível de vazamentos, idealmente, nenhum. Espera-se que as regiões atômicas sejam similares a ponto que o valor de cor médio destas seja uma representação válida da cor do grupo de pixels agrupados.

Os testes realizados e que serão apresentados em seções adiantes foram realizados com duas técnicas: a funcional de Mumford & Shah (M&S) functional (MUMFORD; SHAH, 1989) e o método chamado *Color Structure Code* (CSC) (PRIESE; REHRMANN, 1993). Ambas as técnicas são técnicas focadas em forte similaridade regional na identificação de objetos homogêneos, M&S através da representação de elasticidade através da funcional e CSC através da abordagem de ilhas de similaridade. Esse enfoque é ideal para o que se espera da pré-segmentação e os resultados apresentados por ambas validam a escolha, como será mostrado adiante. Nada impede, porém, que técnicas como, por exemplo, *watershed* (VINCENT; SOILLE, 1991) pudessem ser usadas, afinal o objetivo deste passo não é obter uma segmentação final, mas uma pré-classificação de pixels como estruturas homogêneas.

Para a validação, buscou-se produzir um conjunto de parâmetros para o M&S e CSC que não produzissem ou evitassem ao máximo vazamentos, de forma a usar esses parâmetros ao longo dos experimentos.



(a) Pré-segmentação com Mumford & Shah (b) Pré-segmentação com Color Structure Code

Figura 4: Imagem comparando a pré-segmentação de uma mesma imagem entre as técnicas Mumford & Shah e Color Structure Code. A segmentação fornecida por M&S em geral tende a ser menos agressiva e conservando mais gradientes suaves, dando um aspecto mais “escamoso” a imagem. CSC produz resultados menos conservadores, mas bons o suficiente para serem usados como passo inicial, especialmente levando em consideração a velocidade desta técnica.

3.2.2 Estruturação dos dados

É necessária uma estrutura que permita que, dada uma imagem que siga a representação inicial esperada, consiga representar claramente as regiões atômicas de similaridade identificadas e também as relações de vizinhança presentes entre essas regiões. A representação de vizinhanças é necessária para passos posteriores do algoritmo, onde gradientes entre diferentes regiões atômicas acordantes com as métricas do algoritmo proposto serão buscados.

Essa definição de nodos conexos e com representação do relacionamento de vizinhança levou a seleção de grafos como a estrutura para o método. Com grafos, permite-se representar a topologia da imagem inicial através de vértices que representarão as regiões atômicas de similaridade e de arestas não-direcionadas que mostrarão as vizinhanças presentes entre tais vértices. O uso de grafos como forma de estruturação de dados auxiliando técnicas de segmentação também pode ser visto em (TRÉMEAU; COLANTONI, 2000).

Grafos admitem um topologia natural, onde cada aresta $\{v_i, v_j\}$ do grafo G é identificada com um valor no intervalo unitário $I = [0, 1]$ e unidos em vértices coincidentes, isto

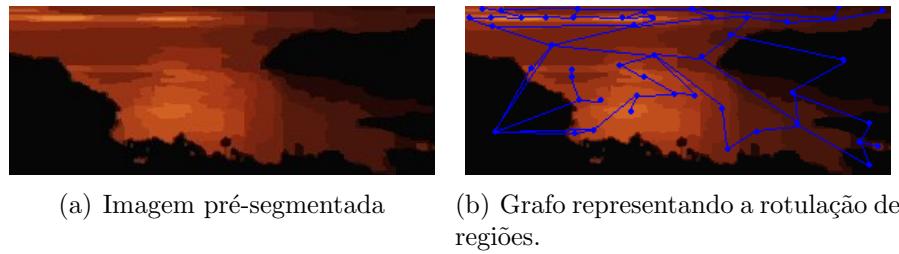


Figura 5: Uma amostra de como fica a organização das imagens após serem rotuladas e estruturadas como grafos conexos pelo algoritmo GNM.

é, duas arestas $\{v_1, v_2\}$ e $\{v_2, v_3\}$ unidas por um nodo v_2 são consideradas equivalentes por uma aresta simples $\{v_1, v_3\}$.

A noção de grafos conexos também apresenta uma propriedade adicional muito desejada, que é a correlação direta com conectividade topológica. Essa propriedade garante que um espaço topológico conexo não possui pares de subconjuntos U e V não-vazios, tal que ambos U e V são conjuntos abertos pertencentes a X e $U \cap V = \emptyset$ e $U \cup V = X$. Também pode ser definido como um espaço que quaisquer dois pontos podem ser unidos por um caminho. Um caminho é definido pela existência de uma função contínua f do intervalo unitário $[0, 1]$ para X com $f(0) = x$ e $f(1) = y$.

3.2.3 Medida de similaridade

A medida de similaridade deverá cumprir os seguintes objetivos:

- 1 Regiões vizinhas que não sejam separadas por bordas salientes e obedecem um gradiente aceitável devem sempre ser consideradas homogêneas e unidas
- 2 Regiões vizinhas separadas por bordas muito salientes jamais devem ser unidas diretamente
- 3 Todas as regiões do grafo conexo devem ter sua vizinhança verificada para possíveis uniões uma única vez
- 4 Regiões que não possuam vizinhança direta podem ser unidas se houver um caminho entre estas regiões que venha a ser considerado homogêneo

O que é considerado como homogêneo será definido por um limiar. Em princípio, a medida de similaridade compara duas regiões atômicas consideradas homogêneas, medindo quão próximas as regiões estão com respeito ao grau de homogeneidade. Estas regiões são representadas em um grafo e um papel da medida de similaridade é prover um custo para cada aresta conectando regiões adjacentes. Como no caso da segmentação inicial, o cálculo da medida de similaridade é uma parte do *framework* onde muitas medidas de similaridade diferentes poderiam ser usadas. Nos experimentos descritos adiante, a diferenciação de regiões através de percepção clara e prejudicada na cena de uma imagem, como discutido , por exemplo, por (HUANG; WANG; WU, 2006), melhorando a robustez da medida de similaridade de cores para luzes especulares e sombras. A idéia é melhorar a robustez na presença de fortes variações de luminosidade, algo comum com imagens do mundo real.

Para conseguir melhor representar estas características de percepção os valores de cor no algoritmo serão convertidos para o espaço de cores HSL. De modo geral, as imagens digitais são armazenadas usando o espaço de cores RGB, porém, este espaço não é interessante para uma abordagem perceptual. As principais razões para isso são:

- Os eixos R, G e B são fortemente correlacionados e não apresentam de forma direta propriedades como cor predominante ou luminosidade. A extração de propriedades como essas é possível, mas não é natural desta representação.
- Apesar de ser uma modelagem que remete diretamente a forma como o olho humano faz a identificação de cores, é muito mais simples realizar uma avaliação semântica ou perceptual de cores através de sistemas como HSL, onde propriedades mais diretamente relacionadas a nossa percepção consciente, e não inconsciente, de cores são usadas.

O espaço de cores HSL, especificamente, representa diretamente três características que simplificam a avaliação perceptual da cor. Os eixos que compõem este espaço representam a matiz ou cor predominante (*hue*), saturação de iluminação (*saturation*) e a

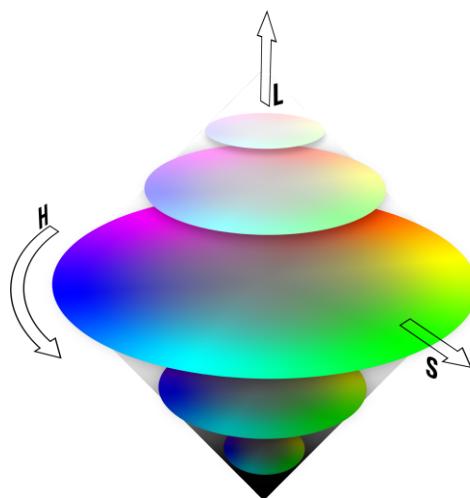


Figura 6: Visão geral do espaço de cores HSL. A relação entre seus eixos faz este espaço de cores tomar a forma de um duplo cone.

própria iluminação (*luminance*). Esta separação de propriedades permite uma avaliação mais específica sobre certas características diante de condições onde serão consideradas mais relevantes certas características sobre outras. Essa relevância das diferentes propriedades se dá da seguinte forma:

- 1 Variações no canal de tom devem ser sensíveis numa verificação de gradientes tênues, pois a cor predominante em uma região deve sempre possuir uma semelhança elevada.
- 2 Saturação determina a importância maior dada ou ao canal de tom ou ao canal de luminosidade para a avaliação. De forma geral, em condições de baixa saturação o tom da cor começa a perder importância na representação da própria cor. No algoritmo proposto, será considerado que a partir de certo ponto de determinada baixa saturação o tom de uma cor será irrelevante e a luminosidade ganhará importância na avaliação. Acima deste ponto de saturação, o tom de uma cor será considerado muito mais importante, dando uma pequena importância somente a luminosidade.
- 3 A luminosidade pode variar bastante de forma elevada entre regiões semelhantes vizinhas, contanto que ainda seja possível uma observar um tom predominantemente semelhante estas regiões ou uma variação gradual aceitável de luminosidade.

4 Sobre a luminosidade vale ressaltar que esta também afetará a importância dada ao tom na avaliação perceptiva. Em condições de luminosidade muito baixa ou muito alta o tom também será considerado irrelevante. Isso busca representar condições com excesso ou falta de iluminação onde as cores não terão muita relação aos tons presentes no disco de matiz do HSL e serão basicamente diferentes variações de cor bastante próximas do branco ou preto, respectivamente.

Deve se observar que para representar essa diferente relevância entre propriedades no RGB seria muito mais difícil. Por exemplo, na questão de matiz de cor, uma avaliação como essas fica prejudicada pela quantidade de informações não interessantes diretamente correlacionadas. O critério de avaliação seria o mesmo, mas a forma de se avaliar sobre este sistema seria muito mais complexo tendo que levar em conta que outras propriedades sempre estarão presentes mesmo quando não desejadas.

O HSL não é o único espaço de cores que forneça características perceptuais de uma forma mais simples. Outros modelos como HSV, CIE L*a*b e CIE L*u*v podem ser usados nesse tipo de avaliação, oferecendo diferentes propriedades e vantagens. O HSL foi escolhido neste trabalho devido a uma representação direta e usual das propriedades desejadas e necessárias, além de ser um espaço de cores usados também já por diversos outros autores. A escolha foi feita mesmo sabendo-se da questão da relação não-linear do HSL com o RGB, da questão da representação cíclica do tom de cores ou de não ser possível aplicar distâncias euclidianas diretamente na diferenciação de valores como em um espaço como o CIE L*a*b, pois estas características não afetam ou são contornadas com a abordagem realizada. Diversas discussões sobre seleção de espaço de cores já foram realizadas e aqui referencia-se especialmente para (CHENG et al., 2001).

Com o uso do espaço de cores HSL e seguindo as restrições de avaliação apresentadas acima, a medida de similaridade proposta busca-se então apresentar uma maior correlação com a forma como analisamos uma imagem e também se tornar mais robusta a presença de fatores como variação de luminosidade e sombras nas imagens.

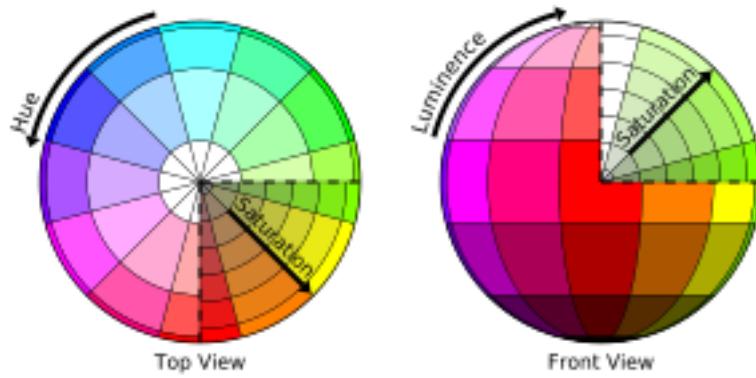


Figura 7: Outra forma de visualizar o espaço de cores HSL. À esquerda pode ser observada a natureza cíclica das matizes de cores e como a variação de saturação as afetam. À direita a influência da variação de luminosidade e saturação. (Imagen criada por Mike Horvath sob a licença Creative Commons.)

3.3 Descrição do Método de Rede de Gradientes - GNM

1 Uma imagem pré-segmentada é usada como entrada. É produzida com parâmetros conservativos e bastante sensíveis para evitar que um segmento nesta imagem vaze de um objeto sobre o outro na imagem, evitando possíveis problemas com ruído local induzido por alta granularidade. Esses *clusters* de semelhança na equação 3.1 são representados como conjuntos de coordenadas no espaço Ω de valores \mathbb{R}^n e faixas de cor dentro do espaço \mathbb{R}^m , onde m é determinado pela dimensionalidade do espaço de cores escolhido.

$$C = \{(x_1, r_1), \dots, (x_i, r_j) | x_1, \dots, x_i \in \Omega \wedge \Omega \subseteq \mathbb{R}^n \wedge r_1, \dots, r_j \in \mathbb{R}^m\} \quad (3.1)$$

Cada *cluster* $C = \{c_1, \dots, c_n\}$ é um conjunto discreto em que cada pixel da imagem de entrada é inequivocamente contido em único e específico *cluster*, tal que $\bigcap_{i=1}^n c_i = \emptyset$ e $\bigcup_{i=1}^n c_i = \Omega$.

Junto com a imagem de entrada, dois limiares são passados como parâmetros. Estes limiares, T_{cp} e T_{rp} , serão usados para verificar se o resultado de um gradiente de cores aplicado a função de similaridade está aceitável de acordo com a percepção associada ao limiar usado.

- 1.1 Cada um dos *clusters* identificados na pré-segmentação recebem rótulos únicos.
- 2 Regiões e suas relações de vizinhança são representadas através de um grafo conexo $G(V, E)$. O grafo $G(V, E)$ serve de estrutura para os *clusters* encontrados no primeiro passo, provendo uma forma simples de descrever a topologia destes. O grafo é construído de forma que os *clusters* correspondem aos vértices $V = \{v_1, \dots, v_n\}$ (3.2) e as arestas $E = \{e_1, \dots, e_m\}$ (3.3) representam a existência de uma relação de vizinhança entre um par de vértices e, por consequência, entre pelo um ou mais pixels dos *clusters* relacionados. Cada aresta é valorada com o resultado da função de percepção para o gradiente de cores entre seu par de vértices. A função de percepção será apresentada adiante.
- $$V = \{v_i \in V | v_i = c_k \wedge c_k \in C\} \quad (3.2)$$
- $$E = \{e_i \in E | e_i = (v_n, v_m, \phi) \wedge v_n \neq v_m \wedge v_n, v_m \in V \wedge \phi = f_p(v_n, v_m)\} \quad (3.3)$$
- 2.1 Este grafo será sempre um grafo conexo porque todas regiões representadas pelos vértices são contidas em uma mesma imagem, e através das arestas que ligam regiões vizinhas é possível estabelecer caminhos entre quaisquer dois vértices v_1 e v_2 tal como é possível encontrar caminhos ligando quaisquer dois pixels p_1 e p_2 em uma mesma imagem.
- 3 No próximo estágio do algoritmo, cada vértice $v \in V$ é associada com uma única e nova *meta-região* $m \in M$. Meta-regiões são repositórios lógicos que serão usados para armazenar grupos de regiões que possuem um caminho entre elas e que estão ligados através de gradientes de cores que são considerados aceitáveis de acordo com a medida de similaridade proposta, através da função de percepção f_p e de um limiar de percepção T_p . A meta do GNM é unir meta-regiões.

3.1 Todas meta-regiões se comportarão como sub-grafos conectados, significando que qualquer vértice contido por uma certa meta-região tem pelo menos um caminho para qualquer outro vértice também contido na mesma meta-região 3.4.

$$M = \{m_i \in M \mid \bigcup_{j=1}^n (v_j \in V) \wedge \exists (e_k \in E \wedge v_n, v_m \in V \wedge \phi = f_p(v_n, v_m) \wedge (v_n, v_m, \phi) = e_k)\} \quad (3.4)$$

3.2 Qualquer vértice $v \in V$ pode pertencer a somente uma meta-região, ou seja, meta-regiões são conjuntos particionados de vértices e consequentemente irão também dividir a imagem final em partições.

4 O objetivo do algoritmo agora é minimizar $\|G(V, E)\|$, isto é, a cardinalidade dos vértices através de um procedimento de união que busca subgrafos $S_i \subseteq G$ que representem $\min(\sum_{k=1}^N f_p(v_1, v_2), v_1, v_2 \subset e_k)$, sabendo que N é a cardinalidade das arestas do grafo $G(V, E)$. O algoritmo então percorre o grafo $G(V, E)$ procurando nas arestas que conectam vértices $v_1, v_2 \in V$ por gradientes suaves suficientes para serem considerados aceitáveis de acordo com a percepção para esta aresta, resultando na união de tais vértices em uma mesma meta-região $m \in M$.

4.1 A meta é verificar se estas duas regiões podem ser colocadas em uma mesma meta-região $m \in M$, significando que ambas são consideradas similares pela medida. Algumas medidas são tomadas para garantir uma seqüência determinística e que gere resultados independentes de inicialização. Cada aresta $e \in E$ passará por uma avaliação inicial. Essa avaliação é realizada por uma função chamada *função de percepção* f_p , que é aplicada após a classificação de tal aresta em relação a percepção, clara ou prejudicada, que esta se encontra. Se, de acordo com tal percepção, com a função de percepção e os parâmetros de aceitação estabelecidos, for verificada uma continuidade de cor suficientemente suave na aresta avaliada essa aresta então é aceita. Aceitação significa

que esta aresta será agora contida em um conjunto auxiliar, correspondente a sua percepção, onde será ordenada crescentemente dentro desse conjunto de acordo com o valor resultante da função f_p .

As arestas avaliadas, aceitas e ordenadas, então, são agrupadas em dois conjuntos separados, um conjunto para avaliação clara e outro para prejudicada. O conjunto de percepção clara será percorrido inicialmente, seguido pelo de percepção prejudicada. Essa separação é feita para não haver uma ambigüidade em relação a ordenação pelo resultado da função de percepção f_p , decorrente do fato que, como será explicado logo a seguir, arestas são avaliadas diferentemente por esta função de acordo com a percepção em que se encontram tais arestas. Essa escolha de ordem de varredura não afetará o resultado, pois arestas que são suficientemente contínuas e similares serão unidas porque a) são avaliadas por percepções independentes b) a vizinhança representada por tal só pode ser excluída caso de união entre as regiões envolvidas pela avaliação anterior de que tais regiões pertencem a um caminho contínuo e semelhante, tornando desnecessária avaliação posterior e tal relação de vizinhança c) a união das regiões envolvidas nestas arestas não afeta a avaliação da função de percepção f_p .

Essa combinação de medidas garantirá que uma mesma ordem seja sempre obedecida na varredura de arestas e na fusão de meta-regiões que sejam semelhantes, não havendo consequências a avaliação das arestas iniciando em diferentes pontos ou em ordens diferentes.

O conceito relacionado a percepções foi desenvolvido a partir do que foi visto em (HUANG; WANG; WU, 2006), onde a diferença de percepção de cor é usada para aperfeiçoar a avaliação de contraste em imagens. Com estas percepções e as propriedades elas provêm, fica mais fácil tornar o método robusto a mudanças de iluminação e presença de sombras na cena. Também é um conceito interessante porque a questão da variação suave de cores nos objetos é direta-

mente conectada a iluminação e este tipo de avaliação através de percepções oferece uma propriedade interessante para lidar com este tipo de característica.

No algoritmo serão usados dois tipos de percepções:

- **Percepção clara**, onde há uma boa saturação de cor e níveis razoáveis de luminosidade;
- **Percepção prejudicada**, onde há uma baixa saturação de cor e/ou intensidades muito altas ou baixas de luz.

Os valores de cor dos vértices serão convertidos para o espaço de cores HSL, composto por três eixos ortogonais que representam, respectivamente, matiz (*hue*), saturação (*saturation*) e luminosidade (*luminance*). Na versão que os testes foram realizados os níveis de aceitação para ambas as percepções são definidos parametricamente como intervalos de valores de aceitação individualmente para matiz, saturação e luminosidade. Estes intervalos devem partitionar o intervalo que os valores de HSL podem assumir de forma que uma ligação seja classificada somente como percepção clara ou prejudicada.

As equações precisam de dois vértices como parâmetros de entrada. Devem sempre ser regiões atômicas vizinhas, como consequência do fato que os vértices são extraídos das arestas do grafo e herdam tal propriedade destas. Dados ambos os vértices e o tipo de percepção que o gradiente se encaixa, é checada se o valor encontrada com a avaliação da função de percepção f_p é aceitável em relação ao limiar T_p definido parametricamente para a percepção usada. A definição da função é feita em 3.5.

$$f_p(v_1, v_2) = \begin{cases} \alpha_{cp} * \min(|\int_{H(v_1)}^{H(v_2)} d\theta|, 1.0 - |\int_{H(v_1)}^{H(v_2)} d\theta|) + \beta_{cp} * |S(v_1) - S(v_2)| + \\ \gamma_{cp} * |L(v_1) - L(v_2)|, & \text{clear perception} \\ \beta_{rp} * |S(v_1) - S(v_2)| + \gamma_{rp} * |L(v_1) - L(v_2)|, & \text{rough perception} \end{cases} \quad (3.5)$$

onde f_p é a função de avaliação de suavidade de gradientes e toma dois vértices $v_1, v_2 \in V$ como parâmetros. As funções H, S, L tomam um vértice $v \in V$ como parâmetro e fornecem o valor médio de, respectivamente, matriz, saturação e luminosidade normalizados no intervalo $[0, 1]$ dos valores de cor das posições dos pixels na imagem que o vértice v contém. Os coeficientes α, β and γ correspondem à percepção com a qual o gradiente entre os vértices deve ser avaliado. Todos os testes apresentados adiante foram realizados com os seguintes valores:

$$\text{clear perception} : \alpha_{cp} = 0.7, \beta_{cp} = 0.2, \gamma_{cp} = 0.1$$

$$\text{rough perception} : \beta_{rp} = 0.2, \gamma_{rp} = 0.8$$

No caso de sucesso, significando que o resultado é menor que o limiar T_p , a aresta é adicionada ao conjunto correspondente a percepção respectiva à aresta. Em passos posteriores esse conjunto será percorrido e a união entre as meta-regiões envolvidas será realizada. A equação 3.6 representa essa relação.

$$f_p(v_1, v_2) < T_p \rightarrow m_1, m_2 \in M | v_1 \subset m_1 \wedge v_2 \subset m_2, m_1 \cup m_2 \quad (3.6)$$

Se não for menor que o limiar T_p , então nada é feito e a aresta avaliada não é adicionada aos conjuntos que serão percorridos posteriormente, por consequência não tendo possibilidade de, neste ponto, pelo menos, haver a união entre as meta-regiões dos vértices que tem sua relação de vizinhança representada por tal aresta.

4.2 Criados esses dois conjuntos ordenados de arestas, o percorrimento respectivo desses conjuntos é iniciado. Para cada aresta $e \in E$ avaliada, se os vértices já são contidos por uma mesma meta-região não há razão para continuar verificando, pois as meta-regiões $m_1, m_2 \in M$ a que os vértices vizinhos $v_1, v_2 \in V$ correspondem já foram identificados como suficientemente semelhantes em um momento anterior. O próximo passo, nesse caso, é ir para a aresta seguinte no conjunto ordenado. Cada aresta é examinada somente uma vez.

4.3 Se ambos vértices não pertencem a uma mesma meta-região $m \in M$, uma união é realizada entre as meta-regiões $m_1, m_2 \in M$ que contêm os vértices $v_1, v_2 \in V | v_1 \subset m_1 \wedge v_2 \subset m_2$, porque o gradiente entre estas regiões foi considerado suficientemente suave dentro da percepção observada e então estas regiões correspondem na verdade a um único objeto na cena da imagem. Nenhum outro tipo de mudança é realizada nos vértices ou nas arestas.

5 Ao final da realização do processo de minimização da cardinalidade do grafo $G(V, E)$, as partições representando os objetos encontrados na imagem equivalem às meta-regiões $m \in M$ restantes após as uniões realizadas ao longo do processo.

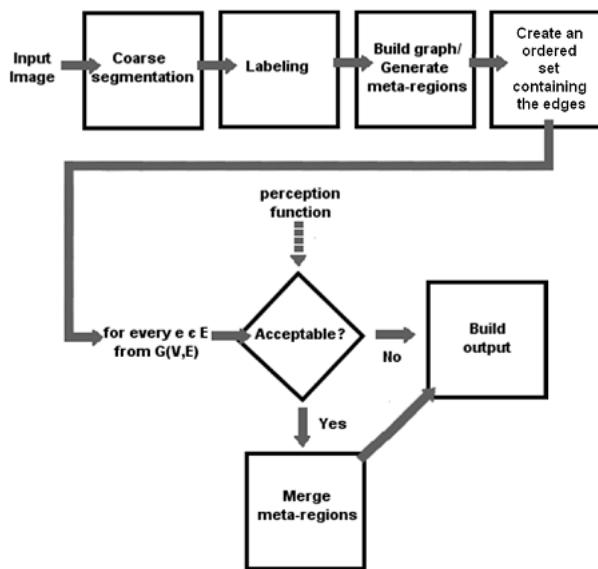


Figura 8: Diagrama representando o processo do algoritmo do Método de Redes de Gradientes (GNM)

3.3.1 Pós-processamento: eliminação de fragmentos

A segmentação realizada pelo GNM apresenta um problema: regiões razoavelmente pequenas, aqui chamadas de fragmentos, quando não conseguem ser fundidas em nenhum momento através da avaliação de gradientes suaves e contínuos persistem na imagem resultante e prejudicam a avaliação desta imagem. Tais fragmentos, em geral, apresentam cores bastante diferentes de regiões ao seu redor e surgem em geral na pré-segmentação,

especialmente no caso do CSC onde pequenos resíduos acontecem mais facilmente. A falta de uma medida que em qualquer momento leve tamanho de regiões atômicas em consideração é o principal fator causador desta característica.

Como estas regiões são muito pequenas e muito provavelmente não possuem significado na cena da imagem, foi desenvolvido um passo de pós-processamento para eliminar estes fragmentos. Assim, haverá uma redução no número de objetos na cena final e objetos de menor relevância ou nenhuma relevância se fundirão a outros mais importantes e representativos, permitindo uma melhor avaliação dos resultados. Técnicas apresentadas em (ANGULO; SERRA, 2007) e (BOSCH; MUÑOZ; FREIXENET, 2007) realizam processamentos semelhantes também em busca de aperfeiçoar a qualidade da segmentação nas imagens resultantes.

Esse pós-processamento se baseia na definição de limites de tamanhos e um limiar de similaridade de cor. São definidos dois limiares de tamanho. O maior limiar define a partir de qual tamanho regiões serão avaliadas pelo pós-processamento. O menor define a partir de qual tamanho, se caso não houver sucesso na fusão através da avaliação por similaridade de cor, regiões que se encaixem nesse limiar serão fundidas a maior região com a qual compartilharem um relacionamento de vizinhança. Todas regiões resultantes da segmentação do GNM serão testadas sob estas condições. O algoritmo para este pós-processamento então é:

- 1 Selecionar uma região presente no resultado da segmentação do GNM ainda não pós-processada.
- 2 Avaliar se esta região é pequena o suficiente, de acordo com o maior limiar de tamanho, para ser avaliada.
 - 2.1 Se não, selecionar outra região não avaliada, se houver, ou encerrar, caso não existam mais regiões a serem avaliadas.
- 3 Caso seja suficientemente pequena, avaliar todas as regiões vizinhas e verificar aquela

que apresenta maior semelhança e de forma que tal semelhança seja aceitável de acordo com o limiar de semelhança de cor estabelecido. Se houver uma região que seja qualificada sob estas condições esta região será selecionada para ser fundida com a região atual.

3.1 Se nenhuma região vizinha for suficientemente semelhante em termos de cor,

avalia-se se a região atual é menor que o segundo limiar de tamanho. Se sim, esta região será fundida à região de maior tamanho que for sua vizinha.

4 Se ainda houveram regiões ainda não avaliadas, voltar ao primeiro passo. Se não, encerrar.

4 Modelagem e Implementação

De modo a realizar a validação do algoritmo através de testes, foi implementado um programa que utiliza o GNM para produzir segmentação sobre imagens digitais. Os requisitos para essa implementação são:

- 1 A implementação do algoritmo deve ser portável entre diversas plataformas e sistemas, de acordo com o que todos projetos atuais do grupo Cyclops e seus laboratórios vêm desenvolvendo.
- 2 A implementação deve suportar os tipos mais comuns de imagens digitais usadas em processamento de imagens, nominalmente do tipo *Portable Any Map* (PNM).
 - 2.1 Esse suporte deve ser tanto para abertura quanto para armazenamento das imagens em arquivo.
 - 2.1 Deve ser feito com uso de alguma biblioteca selecionada. A disponibilidade de diversas soluções para esse problema torna a reimplementação de soluções desse tipo algo não desejado.
- 3 Deve ser possível configurar os diversos parâmetros envolvidos na segmentação do GNM.
 - 3.1 Também devem ser possível configurar outros parâmetros ligados diretamente a implementação, como tipos de arquivo de imagem usados no momento de armazenamento em arquivo.

4 O código feito deve estar documentado e comentado para que seja facilitada sua compreensão e uso por outros futuros colaboradores.

4.1 O modo de uso e as funcionalidades do programa de testes criado deve ser também documentado.

4.1 Cumprimento dos requisitos

4.1.1 Portabilidade

Essa implementação desta versão para testes foi feita com o uso da linguagem de programação C/C++ (ISO, 1998). Não há nenhuma característica intrínseca do algoritmo que necessite diretamente ou seja beneficiada pelo uso dessa linguagem, mas C/C++ é uma linguagem que, além de oferecer a desejada portabilidade, também oferece compiladores capazes de gerarem programas de boa performance (obviamente, somente se o programa em questão cumprir as condições para isso). Grande parte dos projetos atuais do Cyclops também fazem uso dessa linguagem, facilitando uma futura integração e uma maior base de conhecimento prévio da linguagem em questão.

4.1.2 Suporte aos formatos de imagem desejados

Para conseguir o suporte aos formatos desejados fazendo uso da linguagem de programação escolhida, neste caso o C/C++, foi utilizada a biblioteca de ferramentas wxWidgets (WXWIDGETS, 2007). Além de oferecer a capacidade de abrir e salvar os formatos esperados, oferece suporte também para diversos outros tipos de imagem que são interessantes para publicação ou armazenamento de arquivos de imagens, como PNG e JPEG. O acesso a essas funcionalidades é simples e já existem outros projetos no Cyclops que fazem uso dessa biblioteca, tornando ainda mais prático seu uso.

É importante ressaltar, porém, que o wxWidgets não oferece uma performance tão boa quanto outras bibliotecas na questão de abertura e armazenamento de imagens e,

devido a isso, nos testes realizados o tempo destes processos não será considerado.

4.1.3 Configuração de parâmetros

A configuração dos parâmetros de modo a atender as necessidades funcionais e não-funcionais do GNM é feita da seguinte forma:

- Parâmetros como nome da imagem de entrada e nome da imagem de saída são passados como parâmetros na linha de comando do programa implementado.
- Os parâmetros ligados diretamente ao algoritmo, como configuração das percepções e limiares, é feito em um arquivo com formato definido e que terá seu nome passado como parâmetro na linha de comando do programa implementado.
- Parâmetros não-funcionais adicionais, como seleção do formato de imagem usado no armazenamento do arquivo de saída, são armazenados também em um arquivo que será verificado automaticamente pelo programa sempre que executado.

4.1.4 Documentação

Novamente seguindo o modelo do Cyclops e seus projetos recentes, todo o código desenvolvido relacionado a implementação dessa versão de testes para o GNM foi comentada seguindo o modelo Doxygen (DOXYGEN, 2007). Este modelo oferece formas específicas de comentários, restringindo e definindo uma forma a ser seguida, facilitando uma maior relação entre o modo de documentar entre diferentes desenvolvedores. Esse modelo também permite a geração automática de diversos tipos de documentos, uma funcionalidade bastante interessante.

Quanto a forma de uso e funcionalidades dessa versão implementada, foi redigido um pequeno manual com este intuito.

4.2 Implementação da versão de testes

A implementação realizada seguindo todos os requisitos mostrados é dividida basicamente em quatro módulos:

- 1 *libutils*: Uma biblioteca contendo classes e métodos auxiliares. Especificamente, conversão entre diferentes espaços de cores e funcionalidades relacionadas a leitura de parâmetros em linhas de comando e arquivos.
- 2 *liblabeling*: Uma biblioteca contendo classes e métodos relacionados ao processo de *labeling*. Responsável pela definição das classes relacionadas as regiões atômicas e a representação de seus relacionamentos de vizinhança.
- 3 *libgnm*: Uma biblioteca contendo classes e métodos específicos do Método de Rede de Gradientes. Define também a representação de estruturas como meta-regiões. Adicionalmente realiza o pós-processamento mostrado anteriormente.
- 4 *gnmexe*: Um módulo responsável pela geração do programa de testes. Utiliza-se de todas as três bibliotecas citadas anteriormente para isso.

Essa separação foi feita para que houvesse uma diferenciação clara entre as funcionalidades de cada uma das partes do algoritmo e facilitando possíveis extensões e modificações futuras. A seguir um detalhamento das características mais relevantes de alguns desses módulos.

4.2.1 Detalhes sobre módulo liblabeling

A principal funcionalidade presente aqui é um método de rotulação de imagens que identificará unicamente regiões de diferentes cores. Objetivando facilitar passos posteriores do algoritmo do GNM esse método é capaz de estruturar o resultado dessa rotulação como um grafo conexo não-direcionado. Esse módulo define também as estruturas de

dados usadas com esse intuito. Para essa implementação, grafos foram modelados da seguinte forma:

- Vértices: definidos individualmente por uma *struct*, devido a natureza pública de seus atributos membros. Os atributos membros são um conjunto sem duplicatas de posições na imagem rotulada, indicando todas posições contidas nessa região, e outro conjunto sem duplicatas de ponteiros para outros vértices, mostrando a vizinhança para este vértice. Ambos os conjuntos sem duplicatas são implementados com o uso da estrutura de dados *set*.

```

1  namespace Labeling
2  {
3      struct Region
4      {
5          Region();
6          Region(const Region& other_region);
7          ~Region();
8
9          std::set<Index> positions;
10         std::set<Region*> neighbors;
11     };
12 }
```

Um conjunto de vários vértices é definido um conjunto de ponteiros para objetos Vértice, usando a estrutura de dados *vector*.

```

1  namespace Labeling
2  {
3      typedef std::vector<Labeling::Region*> Vertices;
4  }
```

- Arestas: são representadas individualmente através da definição de um par de ponteiros para vértices, implementado na forma de um *typedef* da estrutura *pair*.

```

1  namespace Labeling
2  {
3      typedef std::pair<Region*,Region*> Edge;
4  };

```

Um conjunto de várias arestas é definido como um conjunto sem duplicatas (*set*) de objetos Aresta.

```

1  namespace Labeling
2  {
3      typedef std::set<Labeling::Edge> Edges;
4  };

```

Um grafo, associando um conjunto de vértices com outro conjunto de arestas, é definido também através de um *typedef* para uma estrutura *pair*:

```

1  namespace Labeling
2  {
3      typedef std::pair<Vertices, Edges> Graph;
4  };

```

Todas as estruturas de dados usadas são providas de modo padrão pela linguagem C/C++.

4.2.2 Detalhes sobre módulo libgnm

Meta-regiões são estruturas de dados essenciais para o algoritmo do GNM. O funcionamento destas é análogo ao de um grafo e a definição será bastante semelhante a

usado para definir grafos na seção anterior. Os vértices das meta-regiões são definidas também com *struct*, provendo de forma direta acesso ao seus membros. Os membros de um objeto da classe meta-região são 1) um conjunto de vértices de grafo, representando o agrupamento de regiões atômicas pela qual as meta-regiões são responsáveis; 2) um atributo de identificação única para cada meta-região 3) um arranjo de três posições usado para armazenamento da soma dos valores RGB de cada uma das regiões contidas pela meta-região; 4) um contador do total de posições somadas de todas regiões contidas. Esses dois últimos atributos têm como utilidade otimizar acesso a dados usados na geração da imagem de resultado do programa.

```

1 struct Metaregion
2 {
3     Metaregion(const Label l);
4     ~Metaregion();
5
6     Labeling::Vertices regions;
7     unsigned int rgb_sum[3];
8     unsigned int pos_sum;
9
10    Label label_id;
11};

```

Não há manutenção de meta-regiões vizinhas devido ao elevado custo de ficar atualizando e verificando esta propriedade durante a aplicação do algoritmo. As muitas fusões entre meta-regiões fazem desta uma operação custosa e, que para a atual implementação, não é necessária. Essa falta de necessidade se deve ao fato de meta-regiões somente agruparem as regiões atômicas e o algoritmo atuar somente sobre vizinhança entre regiões atômicas.

Mostrada a definição usada para meta-regiões, agora será apresentado como o algoritmo do GNM é implementado. Para o processo do algoritmo ser iniciado apropriadamente certas informações devem ser passadas. Uma informação necessária é a configuração de percepção. Essa configuração é fornecida através de uma *struct* que possui um arranjo em que cada uma das posições corresponde a uma informação específica, representada por um valor flutuante em *double*, quanto a percepção. Os limiares desejados para avaliação também são passados através dessa mesma estrutura.

```

1 #define GNM_TOTAL_PARAMS 11
2
3 enum GNMIndex { ALPHA_CP = 0,BETA_CP = 1,GAMMA_CP = 2, THRESHOLD_CP = 3,
4
5                 ALPHA_RP = 4,BETA_RP = 5,GAMMA_RP = 6, THRESHOLD_RP = 7,
6
7                 ROUGH_SAT = 8,MIN_LUMINANCE = 9,MAX_LUMINANCE = 10 };
8
9
10 namespace gnm{
11     struct Configuration{
12         double params[GNM_TOTAL_PARAMS];
13     };
14 };

```

Além da passagem da configuração de percepção, espera-se como parâmetro de entrada uma imagem rotulada na forma de um grafo e propriedades como as dimensões da imagem, que serão usadas em diversos pontos da implementação do algoritmo.

Fornecidas todas as informações esperadas, o passo seguinte é a inicialização das meta-regiões. Condizendo com a descrição do algoritmo, cada região atômica presente na rotulação da imagem de entrada vai se tornar uma meta-região, sendo implementado de modo que haverá inicialmente um objeto da classe meta-região contendo a região atômica correspondente no seu conjunto de regiões. Como este passo necessita da varredura de cada uma das regiões atômicas, por questão de otimização, serão realizadas outras operações a cada iteração: cada meta-região terá seu valor já convertido para o espaço

de cores HSL, sendo que cada tupla de valores HSL será associada através de um *map* chamado simplesmente de *hsl* a cada região atômica, e os atributos auxiliares de contagem de posições e acumulo de valores RGB também são inicializados.

Outra característica importante a citar nesse passo é que para cada região atômica será feito um mapeamento em qual meta-região esta está contida, através de uma estrutura do tipo *map* que será referenciada como *meta_track* ao longo do algoritmo. No momento de inicialização esse mapeamento é quase redundante, mas o principal objetivo desta estrutura é auxiliar adiante no momento em que as fusões entre meta-regiões ocorrerem e houver a mudança de domínio sobre muitas regiões atômicas.

Ainda no passo de inicialização outra estrutura será criada. Essa estrutura, chamada *post_track*, tem como objetivo manter uma lista das meta-regiões que deverão, ao final do processo normal do algoritmo, passar pela avaliação realizada pelo passo de pós-processamento descrito no capítulo de metodologia. O *post_track* é um *set* de ponteiros para meta-regiões. Os ponteiros para meta-regiões serão inseridos nesse repositório sempre que, no momento de criação de uma meta-região, se identificar que ela é pequena o suficiente para ser pós-processada. Esse repositório será atualizada conforme fusões de meta-regiões ocorrerem.

Com as meta-regiões inicializadas, são preparadas as estruturas auxiliares que serão usadas na varreduras das arestas do grafo. São usadas duas estruturas de armazenamento do tipo *set* para manter as arestas avaliadas como suficientemente suaves e contínuas, de modo a mantê-las armazenadas de modo ordenado. São duas as estruturas: *clearSet*, para as arestas de percepção clara, e *roughSet*, para as arestas de percepção prejudicada. Para a construção dessas estruturas, é realizada uma varredura sobre todas as arestas do grafo.

Em cada uma destas iterações, as regiões atômicas que são apontadas pelo par de ponteiro presente na aresta sendo avaliada são checadas tem seus valores de cor em HSL passados como parâmetro para a função *LookForGradient*. Esse método fará a avaliação de acordo com as equações mostradas na descrição do algoritmo do GNM. Essa função

retorna, respectivamente, o valor da diferença perceptual entre essas regiões, o limiar da percepção apropriada e um booleano indicando se a percepção é clara ou não. Essa avaliação é realizada uma única vez para todas as arestas do grafo.

```

1 std::set<EvalEdge> clearSet;
2 std::set<EvalEdge> roughSet;
3 for (Labeling::Edges::iterator i=edges.begin(); i != edges.end(); i++){
4     Labeling::Edge current_edge = *i;
5     Labeling::Region *r1 = current_edge.first;
6     Labeling::Region *r2 = current_edge.second;
7
8     /*
9      * Check if the gradient between these two regions is acceptable.
10     */
11    std::pair< std::pair<double,double> , bool> tmp =
12    LookForGradient(hsi[r1],hsi[r2]);
13    std::pair<double,double> gradient = tmp.first;
14    if (gradient.first < gradient.second){
15        EvalEdge ee;
16        ee.first = current_edge;
17        ee.second = gradient.first;
18
19        (tmp.second) ? clearSet.insert(ee) : roughSet.insert(ee);
20    } //end if
21 } //end for

```

A avaliação positiva de uma aresta em termos de semelhança indica que essa aresta deve ser inserida no *set* correspondente a sua percepção. Então, se positiva, é criado um objeto da classe *EvalEdge*. Objetos dessa classes são um par contendo, na ordem, a aresta que deve ser armazenada e o valor da diferença perceptual proveniente da função de

percepção. Esse valor de diferença será usado para ordenar de modo crescente as arestas nesses *sets* auxiliares, devido ao fato de os operadores de menor igual e igual, usados na ordenação de objetos por *sets*, são sobreescritos para realizarem a ordenação baseada nesse valor. Dado então um objeto da classe *EvalEdge* com as informações necessárias associadas, ele é inserido em *clearSet* ou *roughSet*, dependente da percepção da avaliação.

```

1  namespace gnm{
2
3      class EvalEdge : public std::pair<Labeling::Edge,double>{
4
5          public:
6
7              bool operator==(const EvalEdge& ee) const {
8
9                  return second == ee.second;
10
11             }
12
13     };
14
15 };

```

Após cada uma das arestas do grafo ter sido visitada e avaliada, o próximo passo é realizar a varredura das arestas contidas nas estruturas auxiliares *clearSet* e *roughSet*. O procedimento é totalmente análogo para esses dois conjuntos. Essa varredura é um laço de repetição que iterará uma única vez sobre cada uma das arestas destes *sets*. Em cada uma destas iterações, as regiões atômicas que são apontadas pelo par de ponteiro presente na aresta atual são checadas se já não pertencem a uma mesma meta-região. Essa verificação é realizada para evitar reavaliar desnecessariamente uniões já presentes e, para que seja feita, se faz uso da estrutura *meta_track* que indicará se tais regiões não estão mapeadas para uma mesma meta-região. Se a verificação mostrar que existe uma confirmação do mapeamento para uma mesma meta-região, o laço de repetição procede para a seguinte iteração.

Caso tais regiões atômicas não estejam contidas por uma mesma meta-região, o resultado será a fusão entre as meta-regiões que contêm ambas as regiões atômicas envolvidas na avaliação usando o método *MergeMetaRegionsOf*.

```
1  for (std::set<EvalEdge>::iterator i=clearSet.begin(); i != clearSet.end(); i++){
2      Labeling::Edge current_edge = i->first;
3      Labeling::Region *r1 = current_edge.first;
4      Labeling::Region *r2 = current_edge.second;
5
6      /*
7       * Regions are not already in the same meta-region
8       */
9      if (meta_track.find(meta_regions[r1]->label_id) !=
10         meta_track.find(meta_regions[r2]->label_id)){
11          /*
12           * If the gradient is acceptable, merge the meta-regions which contain
13           both regions.
14           */
15
16          MergeMetaRegionsOf(r1,r2,regions,meta_regions,meta_track);
17      } //end if
18  } //end for
19
20  for (std::set<EvalEdge>::iterator i=roughSet.begin(); i != roughSet.end();
21 i++){
22      Labeling::Edge current_edge = i->first;
23      Labeling::Region *r1 = current_edge.first;
24      Labeling::Region *r2 = current_edge.second;
25
26      /*
```

```

27     * Regions are not already in the same meta-region
28
29     */
30
31     if (meta_track.find(meta_regions[r1]->label_id) !=  

32         meta_track.find(meta_regions[r2]->label_id)){  

33
34     /*  

35      * If the gradient is acceptable, merge the meta-regions which contain  

36 both regions.  

37
38     */
39
40     MergeMetaRegionsOf(r1,r2,regions,meta_regions,meta_track);  

41 } //end if  

42 } //end for

```

A união entre meta-regiões realiza diversas operações. A essencial, obviamente, é que a partir deste ponto as regiões atômicas envolvidas sejam contidas e apontem a uma mesma meta-região. O resultado será que a partir deste momento a região fundida conterá todas as posições de pixel destas regiões atômicas e de todas outras regiões atômicas que já pertenciam as meta-regiões envolvidas. O mapeamento realizado por *meta_track* será atualizado para manter consistência, de forma que as regiões atômicas contidas nessa nova meta-região apontem devidamente a esta. Os atributos auxiliares de acumulo de valores RGB e contagem de posições também são atualizados, evitando posteriores varreduras redundantes.

O passo de fusão de meta-regiões realiza outra operação importante, que é a atualização do *post_track*, repositório de regiões que devem ser pós-processadas. As regiões que desaparecem devido à fusão, caso estejam contidas no repositório, são retiradas. A região resultante, caso se torne maior do que o necessário para ser avaliada no pós-processamento, também deve ser eliminada do repositório.

Concluída a iteração sobre todas as arestas do grafo e realizadas todas as uniões perceptivamente aceitáveis, se faz necessário realizar a verificação do pós-processamento sobre as meta-regiões que foram suficientemente pequenas para serem adicionadas a *post-track*. Este passo é bastante semelhante ao do algoritmo do GNM, com a exceção que vértices serão percorridos em vez de arestas e as regras de fusão de regiões são próprias ao estabelecido anterior em ao pós-processamento. Todas essas meta-regiões serão avaliadas de acordo com o que é proposto pelo pós-processamento. Sempre que houver o cumprimento da condição de união de regiões, os vértices envolvidos serão unidos tal como na segmentação normal do GNM, através do uso do método *MergeMetaRegionsOf*.

Por fim, resta a geração do resultado. O resultado do algoritmo é fornecido através de um arranjo do mesmo tamanho que a imagem de entrada onde as regiões encontradas durante a segmentação serão representadas pelo valor RGB médio de cada uma das meta-regiões e suas regiões atômicas contidas. Nesse ponto que se faz o uso da informação de acumulo de valores RGB e contagem de posições para cada meta-região, evitando uma nova varredura e gerando o valor médio através de uma simples divisão.

4.2.3 Detalhes sobre módulo gnmexe

O programa gerado não possui interface gráfica, considerado algo desnecessário por ser somente um programa de testes. É um programa de console e todos parâmetros referentes a ele serão passados diretamente através de linhas de comando.

Esse módulo gerencia a conversão dos dados adquiridos através de arquivos de entrada e configuração para que estes possam ser passados como parâmetros para os outros módulos. Também aqui que será feita a conversão do arranjo de valores resultantes da segmentação do GNM para o arquivo de imagem digital do tipo desejado pelo usuário.

Por questões de avaliação de desempenho, esse módulo também rastreia o tempo gasto especificamente pelo *labeling* e pelo algoritmo do GNM.

4.3 Eficiência e outros requisitos não-funcionais

Algoritmos de segmentação de alta velocidade vêm sendo pesquisados para satisfazer a demanda de aplicações que requerem resultados em tempo real ou performances semelhantes. Processos de segmentação rápida podem ser usados em várias situações como orientação autônoma de veículo (REHRMANN; PRIESE, 1998). Outra aplicação possível é guiar cirurgias e outros procedimentos médicos. Um exemplo é dado em (KIM; PARK, 2005), onde segmentar uma artéria carótida é um passo útil para imagens médicas. Requerimentos de eficiência podem ser encontrados ainda em diversas áreas. (WIT, 2006) apresenta uma técnica desenvolvida para aplicações de tempo real como análise de clima espacial. Em (FIGUEROA; LEITE; BARROS, 2006) uma abordagem é proposta para rastrear jogadores em um campo de futebol. Outras aplicações podem ser encontradas na detecção de defeitos em linhas de produção (ABOUELELA et al., 2005). Abordagens de segmentação rápida são um tópico recorrente e várias otimizações ou especializações de técnicas conhecidas foram desenvolvidas (PAN; BIRDWELL; DJOUADI, 2006)(WONG; SPETSAKIS, 2006)(SUN; YANG; REN, 2005).

Enquanto diversos algoritmos obtêm bons resultados negligenciando velocidade, GNM e CSC são ambas técnicas de segmentação genéricas que provêm uma boa performance tanto em termos de qualidade de imagem quanto em termos de velocidade.

4.3.1 A complexidade do GNM

O GNM obtém um desempenho em termos de velocidade combinando uma série de estratégias. Primeiro, um otimizado algoritmo de *labeling* realiza o processamento inicial da imagem pré-processada e garantindo uma solução rápida para este passo intermediário. A complexidade do algoritmo de *labeling* usado é $O(n^2)$.

Depois de classificar a informação, a construção do grafo é realizada. Este passo vai estruturar a informação já que cada região encontrada pelo *labeling* corresponderá a um vértice do grafo. A geração de grafo tem complexidade $O(n)$. Para aperfeiçoar o

desempenho e evitar repetições redundantes, os cálculos de valor de cor média para cada região e a conversão de valores de RGB para HSL (CHENG et al., 2001) são feitas ao longo da geração do grafo.

Em seguida à construção do grafo é construído, ele é percorrido para buscar regiões similares de acordo com a percepção correspondente, verificando a necessidade de uniões. Primeiro, todas as arestas são avaliadas e classificadas se são semelhantes ou não. Se consideradas semelhantes, arestas devem ser armazenadas de forma ordenada. A complexidade geral da ordenação usada é $O(\log m)$. Então, o percorramento de todas as arestas e ordenamento destas terá complexidade esperada de $O(m \log m)$.

As arestas aceitas e ordenadas são varridas novamente, agora para realizar as fusões necessárias para produzir o resultado do algoritmo. A quantidade de arestas pode variar tanto entre a aceitação de nenhuma quanto, no pior caso em termos de complexidade, a aceitação de todas em relação a sua semelhança. Como não é possível estimar sequer uma percentagem esperada ou geral encontrada pela avaliação de semelhança do algoritmo, será considerado o pior caso para a definição de complexidade desta parte, isto é, $O(m)$.

A complexidade total do GNM é $O(n(1 + n) + m(1 + \log m))$, onde n é o número de vértices e m é o número de arestas. Este método apresenta uma solução simples que é somente dependente do tamanho da imagem e da complexidade da cena da imagem resultante do pré-processamento por uma segmentação selecionada. É importante notar, mesmo que não considerada diretamente na complexidade do GNM, que o algoritmo selecionado para a pré-segmentação tem um efeito cumulativo no tempo total de processamento. Assim, uma técnica apropriada deve ser selecionada de forma a privilegiar a característica de desempenho.

4.3.2 A complexidade do passo de pós-processamento

Diferente do processo do GNM, o pós-processamento varre diretamente as meta-regiões, não as arestas do grafo. Essa varredura será feita para cada meta-região consid-

erada como necessária de reavaliação pelas condições do pós-processamento. Para cada uma dessa meta-regiões selecionadas, todos as meta-regiões vizinhas serão percorridas e verificadas em busca de unir as regiões que produzam a menor interferência no resultado. A complexidade desse passo então é $O(n^2)$, onde n é o número de meta-regiões que necessitam de pós-processamento. Como a classificação das meta-regiões que necessitam de pós-processamento é feita ao longo do processo normal do algoritmo, a detecção das meta-regiões que serão avaliadas neste passo não resulta em acréscimo de complexidade.

4.3.3 Sobre os algoritmos usados como pré-segmentação

O foco da discussão nesta seção é obter bons resultados de imagem combinados a performance em termos de velocidade, lembrando que a segmentação escolhida deve obedecer a expectativa que o GNM tem de sua pré-segmentação, que é fazer um reconhecimento grosso das regiões de similaridade mais óbvia sem comprometer os gradientes de cores mais suaves.

Duas técnicas foram usadas nos testes realizados: Mumford&Shah e *Color Structure Code*. Ambos os algoritmos apresentam uma filosofia de segmentação que condiz com aquilo que o GNM espera. Porém, em termos de eficiência, o CSC se destaca em relação ao M&S.

Apesar do CSC ser uma técnica focada em velocidade e ter sido desenvolvida com propósitos específicos, ainda obtém bons resultados em termos de qualidade de imagem e prova ser uma boa solução em casos genéricos também. A abordagem de ilhas de similaridade ajusta-se bem com a propriedade esperada do ponto de partida do GNM, que regiões de similaridade alta sejam encontradas evitando vazamentos. O algoritmo do Mumford&Shah não apresenta preocupações específicas com desempenho e necessita de diversas iterações e reinicializações de valores. Outra característica desfavorável é que a implementação de técnicas baseadas em Mumford&Shah devem discretizar ou realizar o cálculo numérico da equação diferencial que é a função de avaliação de segmentação

de regiões, uma tarefa custosa. Essas características colocam o CSC em vantagem, pelo menos quanto ao aspecto de velocidade.

5 Resultados

5.1 Avaliação da qualidade de imagem das segmentações

5.1.1 Experimento

De modo a conseguir objetivamente selecionar a pré-segmentação ideal para ser usada com a abordagem do GNM e, assim, escolher os melhores parâmetros de segmentação para cada algoritmo, assume-se que imagens de *ground truth* representam o julgamento de um observador apto, provavelmente um especialista, servindo como um padrão ouro a ser atingido. O processo de avaliação dos resultados de segmentação de imagens foi investigado por vários pesquisadores como (UNNIKRISHNAN; PANTOFARU; HEBERT, 2007), (SAHASRABUNDHE et al., 1999) e (GESU; STAROVOITOV, 1999).

Como fonte de imagens de *ground truth* foi selecionada a base de imagens de Berkeley (MARTIN et al., 2001). Esse conjunto de imagens é reconhecido na comunidade de processamento de imagens e cada imagem contida nessa base possui entre cinco a sete diferentes *ground truths* gerados por diferentes indivíduos. A avaliação adotada aqui será realizada com um conjunto de 16 imagens selecionadas desta base de imagens, de forma que as imagens selecionadas apresentam fortes gradientes que resultam em variações de cor contínuas para um mesmo objeto.

A avaliação objetiva da qualidade dos resultados do GNM seguirá um procedimento comparativo a outras técnicas do estado da arte. O primeiro passo foi a geração de uma série de resultados de segmentação com os algoritmos *Color Structured Code* (CSC) (REHRMANN; PRIESE, 1998), Mumford & Shah (M&S) (MUMFORD; SHAH, 1989), *Wa-*

tershed (WS) (VINCENT; SOILLE, 1991), RHSEG (TILTON, 2006), JSEG (MARTIN et al., 2001), Blobworld (CARSON et al., 2002), uma segmentação que utiliza maximização de expectativa voltada especialmente para pesquisa e busca de imagens, e Edison, que é uma implementação da segmentação baseada em *mean-shift* apresentada em (COMANICIU; MEER, 2002). Adicionalmente aos resultados individuais de cada uma destas técnicas foram obtidas segmentações com o GNM combinado com pré-segmentações obtidas com os algoritmos CSC, M&S, Edison e WS. O propósito é validar o algoritmo GNM como uma técnica de segmentação tão capaz quanto as outras selecionadas para os testes.

Em relação aos testes realizados com o GNM, todas as pré-segmentações foram obtidas de um conjunto de diferentes faixas de parâmetros. Apesar de os parâmetros serem específicos de acordo com cada pré-segmentação utilizada, a meta é a mesma, adquirir uma imagem super-segmentada evitando vazamentos e preservando gradientes suaves. Os parâmetros usados nas pré-segmentações foram os seguintes:

- *Color Structured Code*: limiar igual a 30;
- Mumford & Shah: *lambda* igual a 600;
- Edison: parâmetros $SS = 3$ e $SR = 8$;
- *Watershed*: condutância igual a 2, número de iterações igual a 10 e limiar igual a 0.01.

Sobre cada uma destas pré-segmentações foi realizada uma série de segmentações com o GNM. Dois parâmetros somente foram variados nestes testes, t_{cp} e t_{rp} , ambos variando no intervalo [0.01, 0.07], utilizando incrementos de 0.005. Todos os testes usam ainda os seguintes intervalos para os parâmetros restantes, com intervalos semelhantes aos encontrados em (HUANG; WANG; WU, 2006). Com todos os valores de HSL normalizados dentro do intervalo [0, 1]:

$$\text{Percepção clara: } = \{(S > 0.1) \wedge (0.2 < L < 0.95)\}$$

$$\text{Percepção prejudicada: } = \{(S \leq 0.1) \vee (L \leq 0.2) \vee (L \geq 0.95)\}$$

Os parâmetros usados no passo de pós-processamento foram os seguintes:

- Maior limite de tamanho: 50
- Menor limite de tamanho: 10
- Similaridade de cor: 0.5

As técnicas do estado da arte que tiveram as segmentações individuais selecionadas para comparação também foram testadas com uma série de parâmetros. Os intervalos de parâmetros e os incrementos usados são os seguintes:

- *Color Structured Code*: $20 \leq limiar \leq 100$, incremento igual a 10;
- Mumford & Shah: $1000 \leq lambda \leq 15000$, incremento igual a 500;
- Edison: $3 \leq SS \leq 30$, com incremento de 1, e $SR = 8$;
- *Watershed*: condutância igual a 2, número de iterações igual a 10 e $0 \leq limiar \leq 0.5$.

Os algoritmos Blobworld e JSEG são técnicas não-supervisionadas e não requerem a passagem de parâmetros. O algoritmo RHSEG é supervisionado, porém, com é intrinsecamente hierárquico não requereu a passagem de um intervalo de parâmetros. As segmentações com o RHSEG foram feitas com a similaridade de entropia, com um fator de convergência igual a 1.75 e com uma importância de 0.1 a *clusterização* espectral.

Após a geração das séries de imagens com as técnicas selecionadas com os parâmetros listados, foi realizada uma inspeção visual para a seleção daqueles que foram os melhores resultados para cada imagem do conjunto. Os resultados selecionados foram aqueles que melhor particionaram as imagens em regiões relevantes. Esta seleção com intervenção manual é considerada um filtragem simples das segmentações e o grupo de imagens selecionadas será em seguida avaliada por medidas específicas para este fim para que seja possível quantificar a qualidade das imagens segmentadas obtidas.

As imagens selecionadas de cada um dos conjuntos extraídos da base de imagens de Berkeley testados podem ser vistas a partir da figura 9 até a figura 24. Os parâmetros com que essas imagens foram obtidas podem ser vistos no anexo A.1.

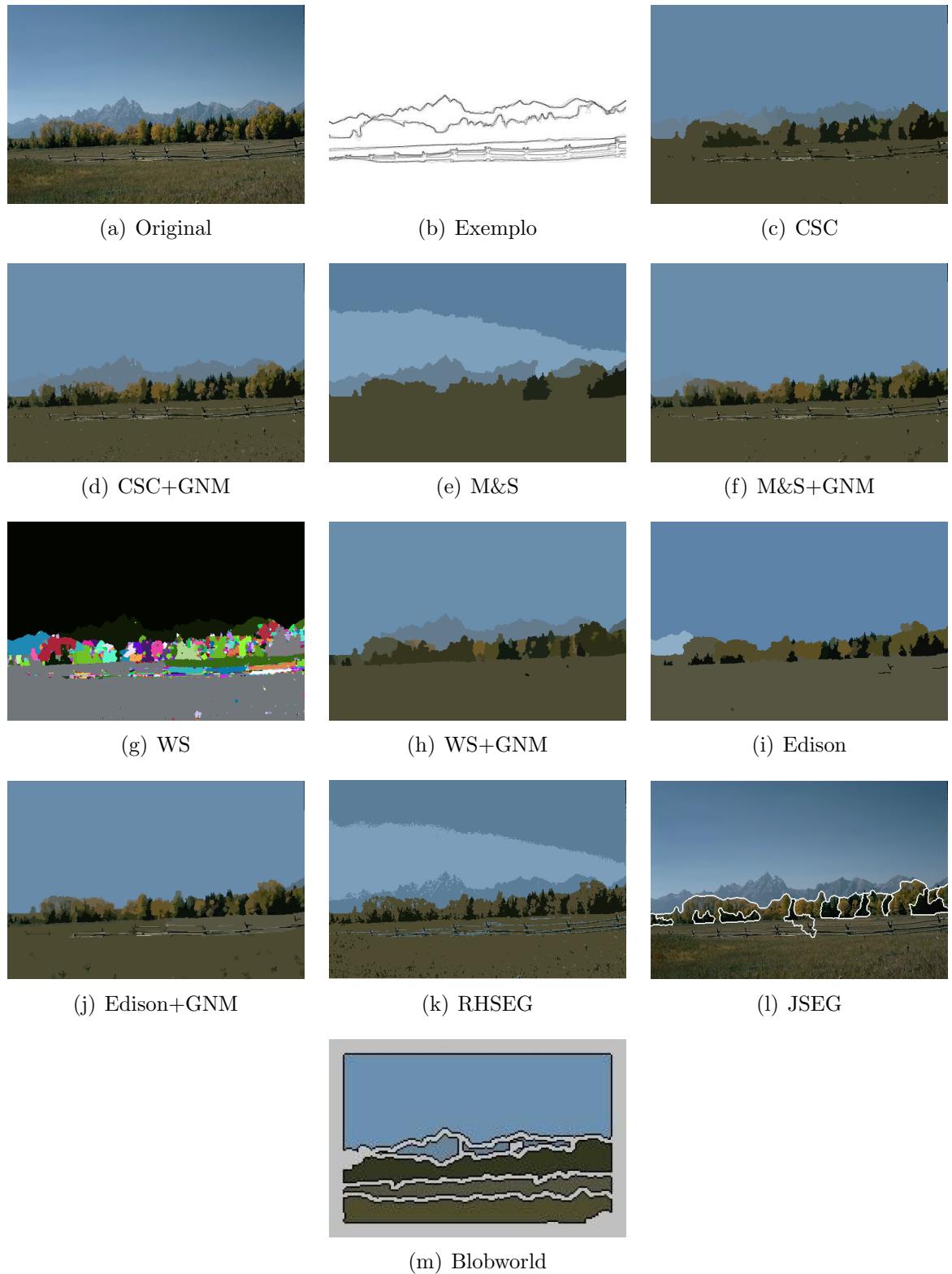


Figura 9: GNM comparado com outros métodos de segmentação usando a imagem 2092 da base de dados de testes de segmentação (MARTIN et al., 2001).

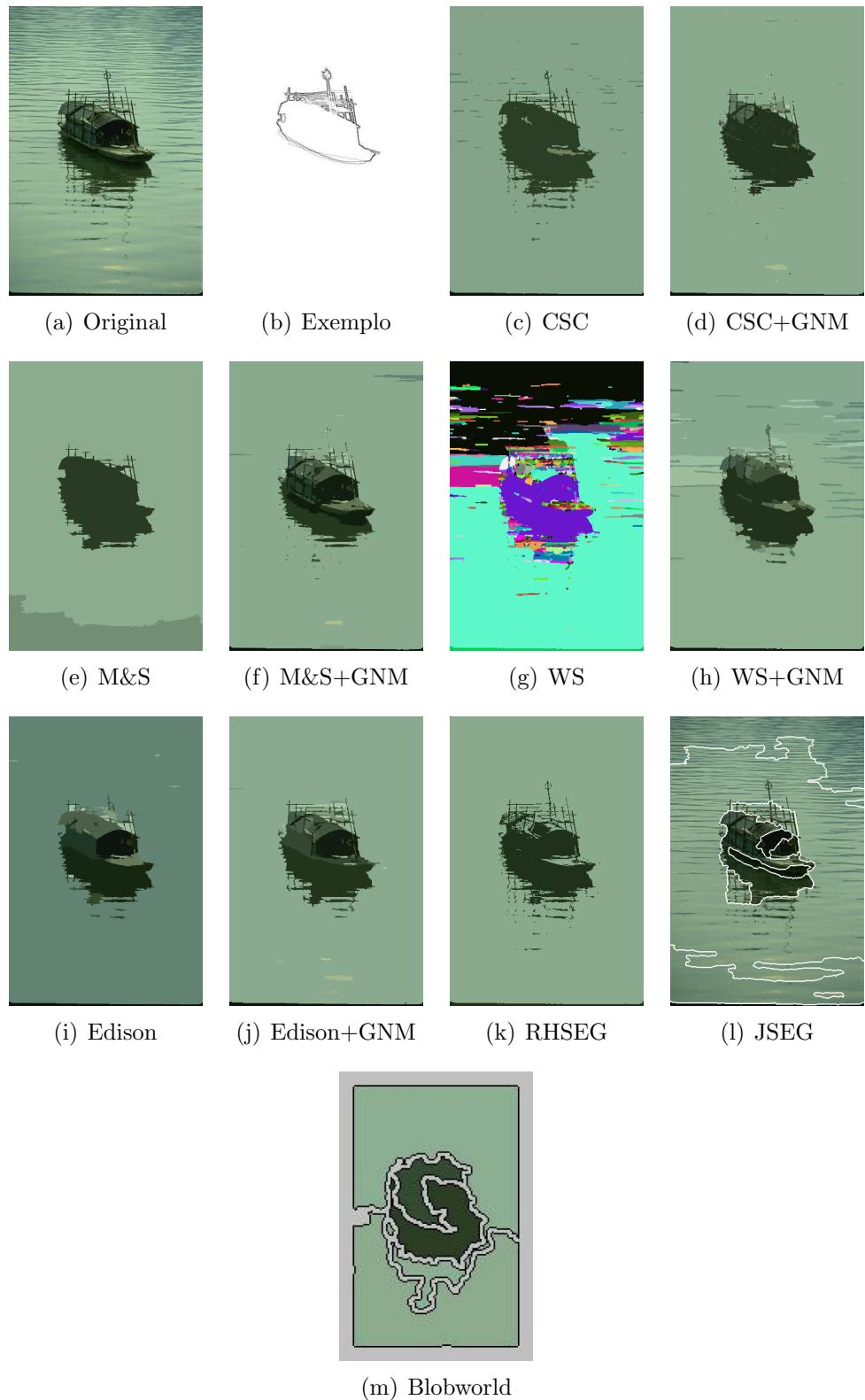


Figura 10: GNM comparado com outros métodos de segmentação usando a imagem 15088 da base de dados de testes de segmentação (MARTIN et al., 2001).

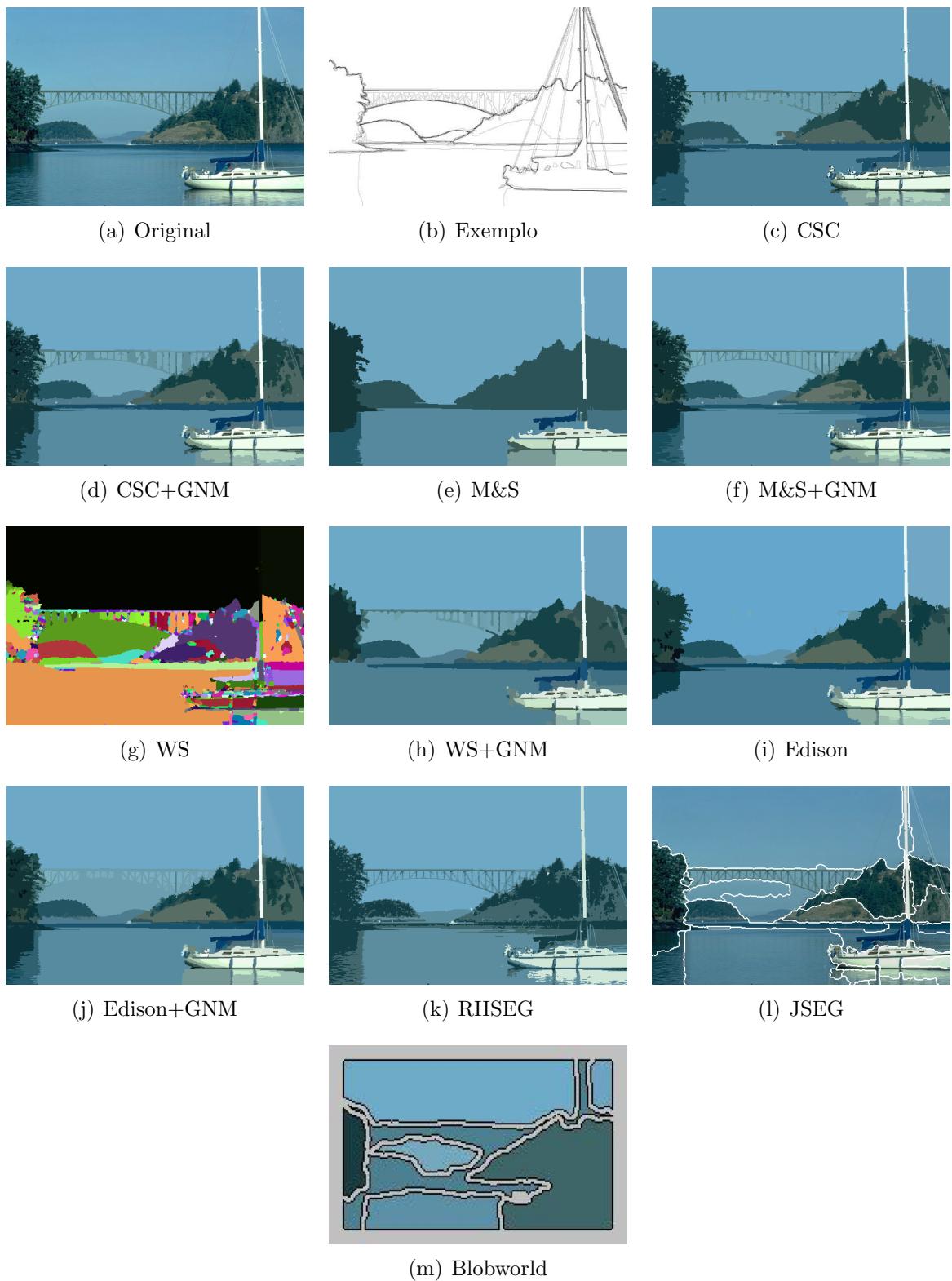


Figura 11: GNM comparado com outros métodos de segmentação usando a imagem 22090 da base de dados de testes de segmentação (MARTIN et al., 2001).

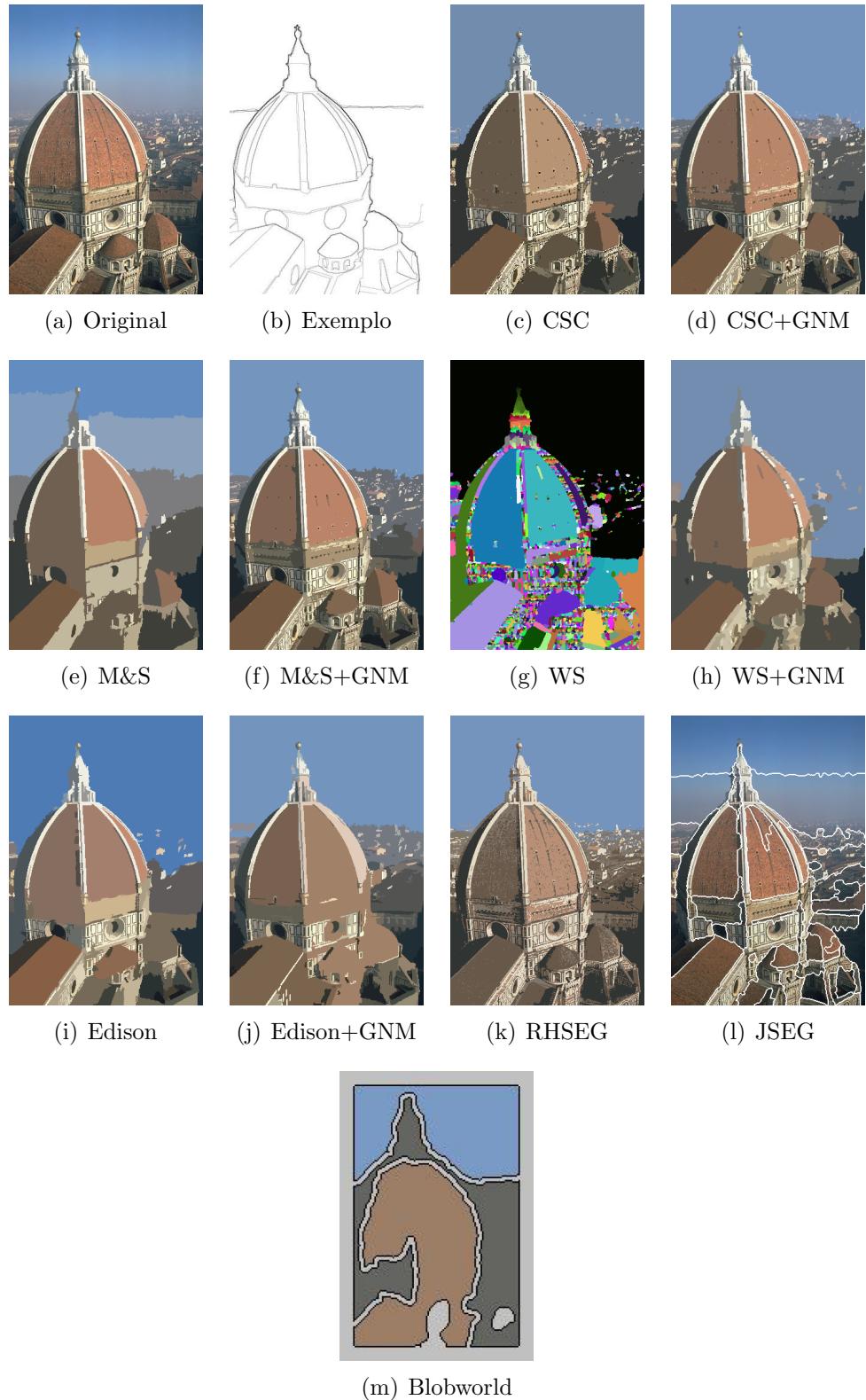


Figura 12: GNM comparado com outros métodos de segmentação usando a imagem 24004 da base de dados de testes de segmentação (MARTIN et al., 2001).

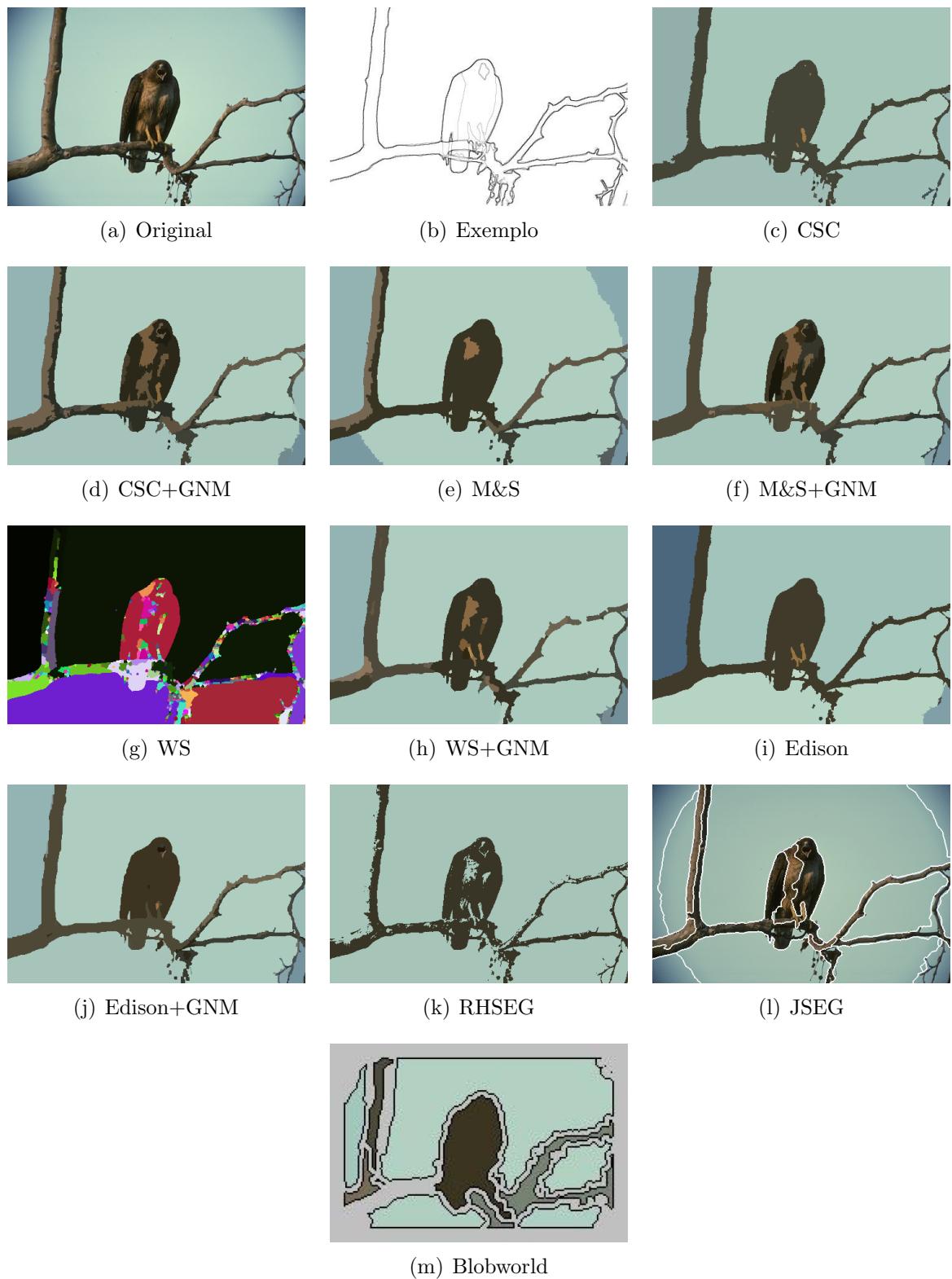


Figura 13: GNM comparado com outros métodos de segmentação usando a imagem 42049 da base de dados de testes de segmentação (MARTIN et al., 2001).

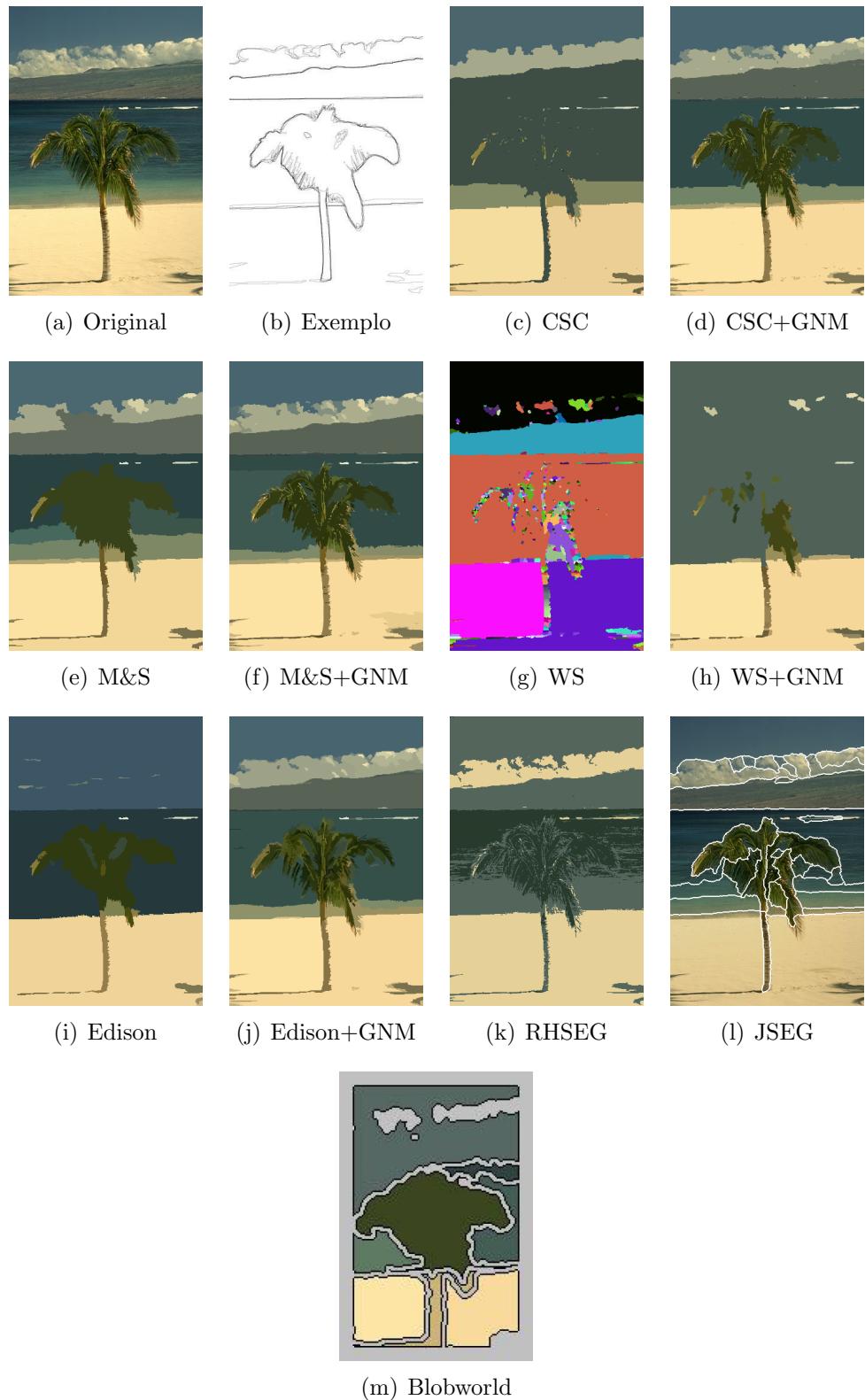


Figura 14: GNM comparado com outros métodos de segmentação usando a imagem 46076 da base de dados de testes de segmentação (MARTIN et al., 2001).

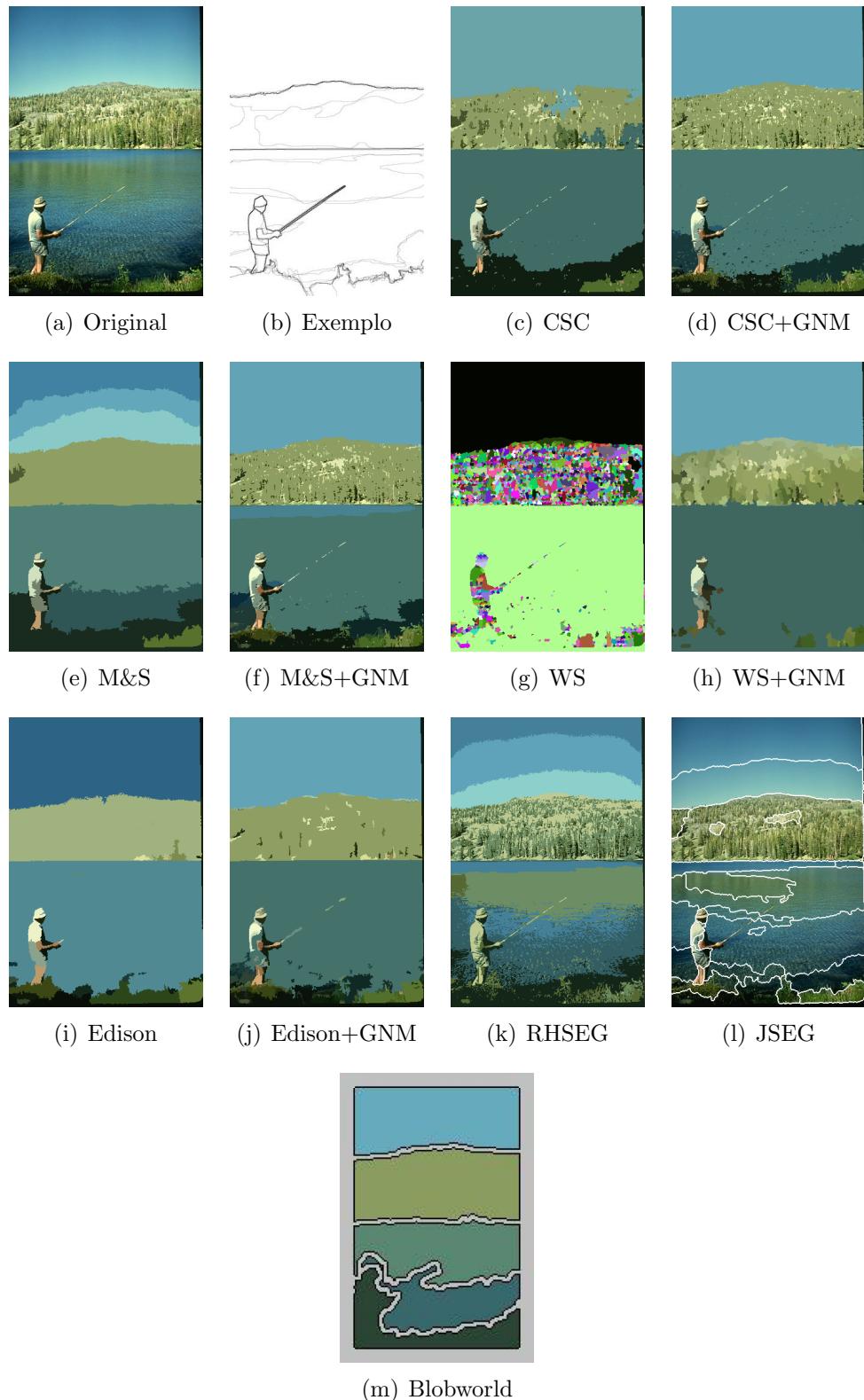


Figura 15: GNM comparado com outros métodos de segmentação usando a imagem 48055 da base de dados de testes de segmentação (MARTIN et al., 2001).

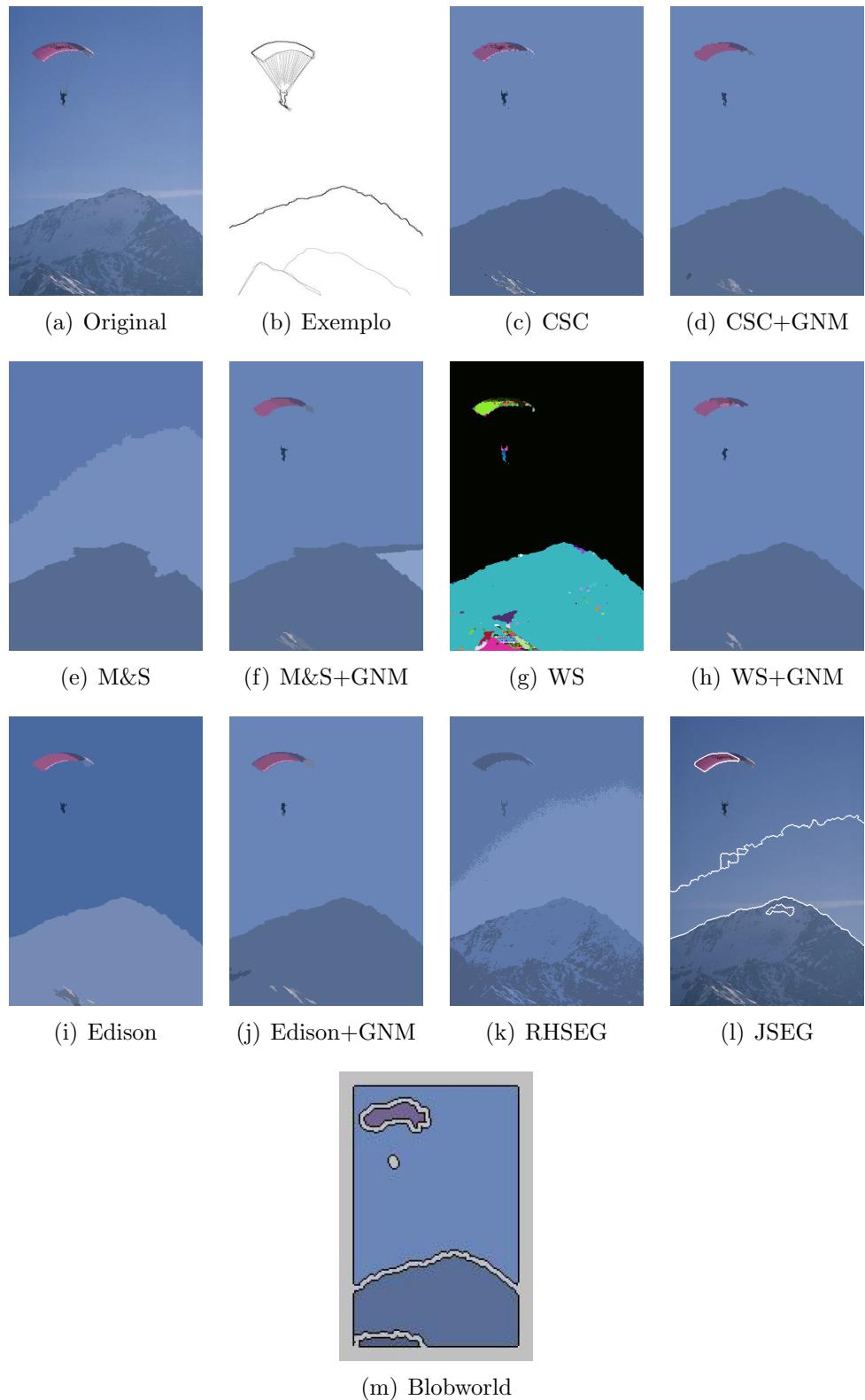


Figura 16: GNM comparado com outros métodos de segmentação usando a imagem 60079 da base de dados de testes de segmentação (MARTIN et al., 2001).

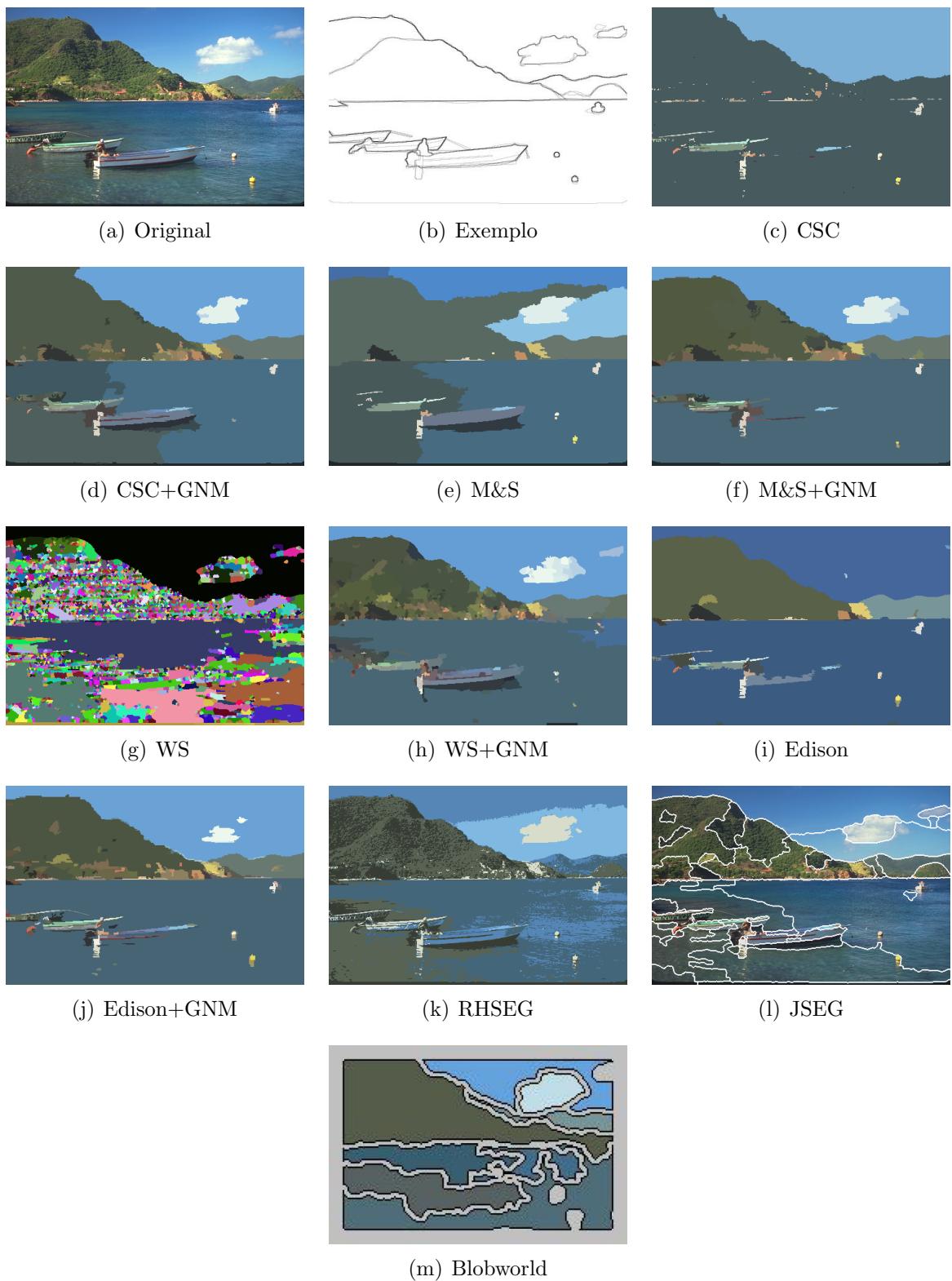


Figura 17: GNM comparado com outros métodos de segmentação usando a imagem 68077 da base de dados de testes de segmentação (MARTIN et al., 2001).

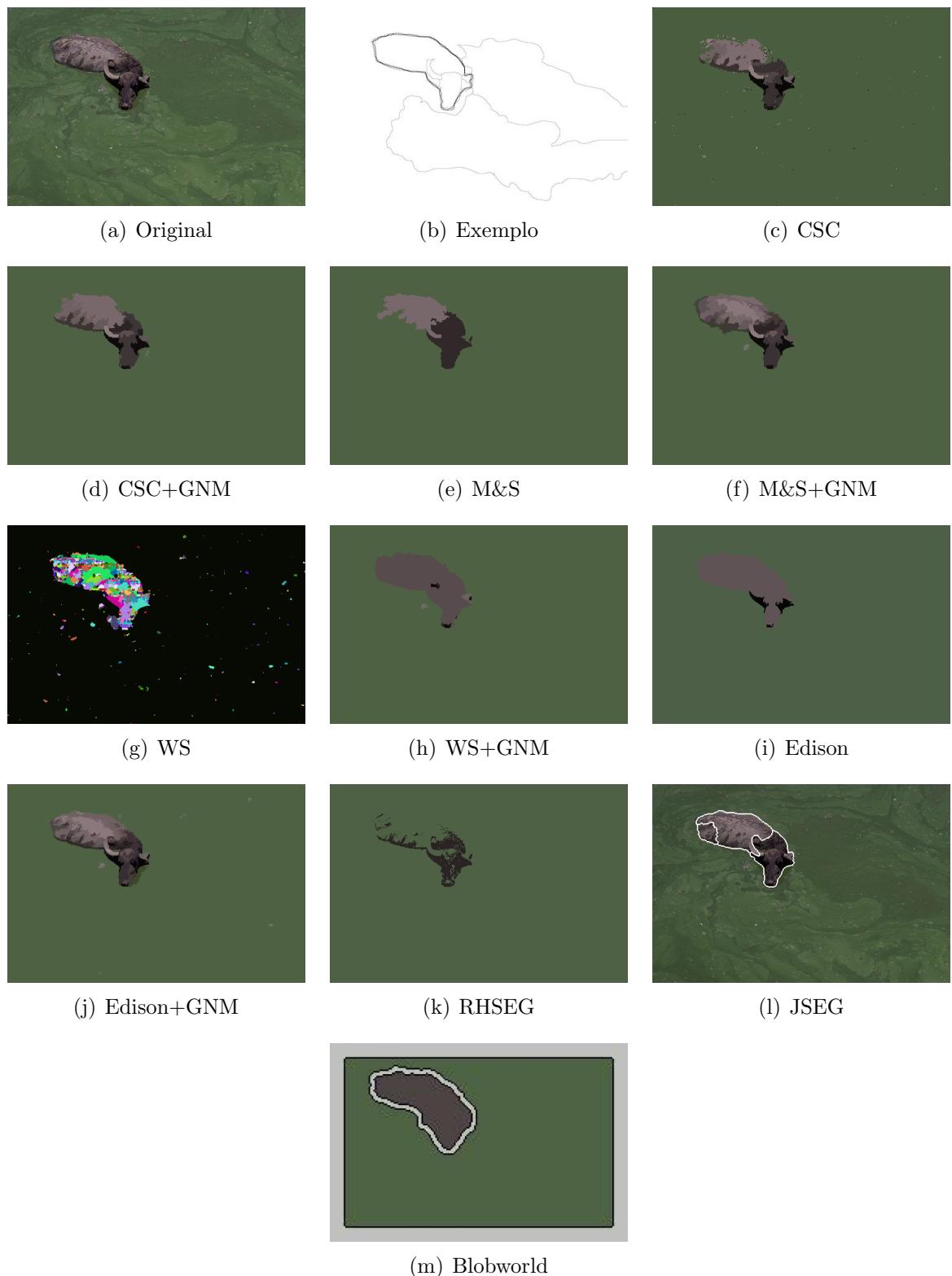


Figura 18: GNM comparado com outros métodos de segmentação usando a imagem 80099 da base de dados de testes de segmentação (MARTIN et al., 2001).



Figura 19: GNM comparado com outros métodos de segmentação usando a imagem 118035 da base de dados de testes de segmentação (MARTIN et al., 2001).

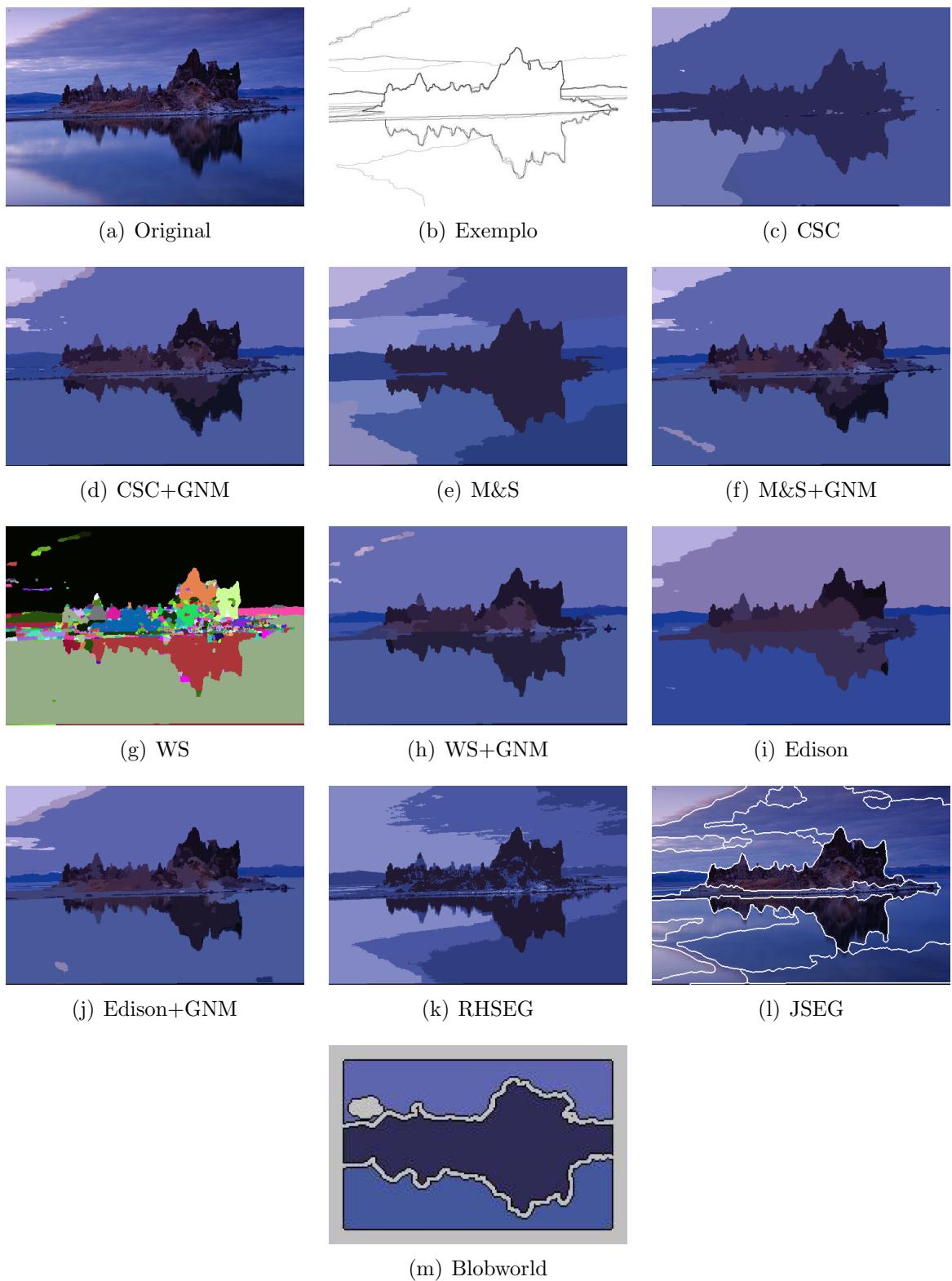


Figura 20: GNM comparado com outros métodos de segmentação usando a imagem 143090 da base de dados de testes de segmentação (MARTIN et al., 2001).

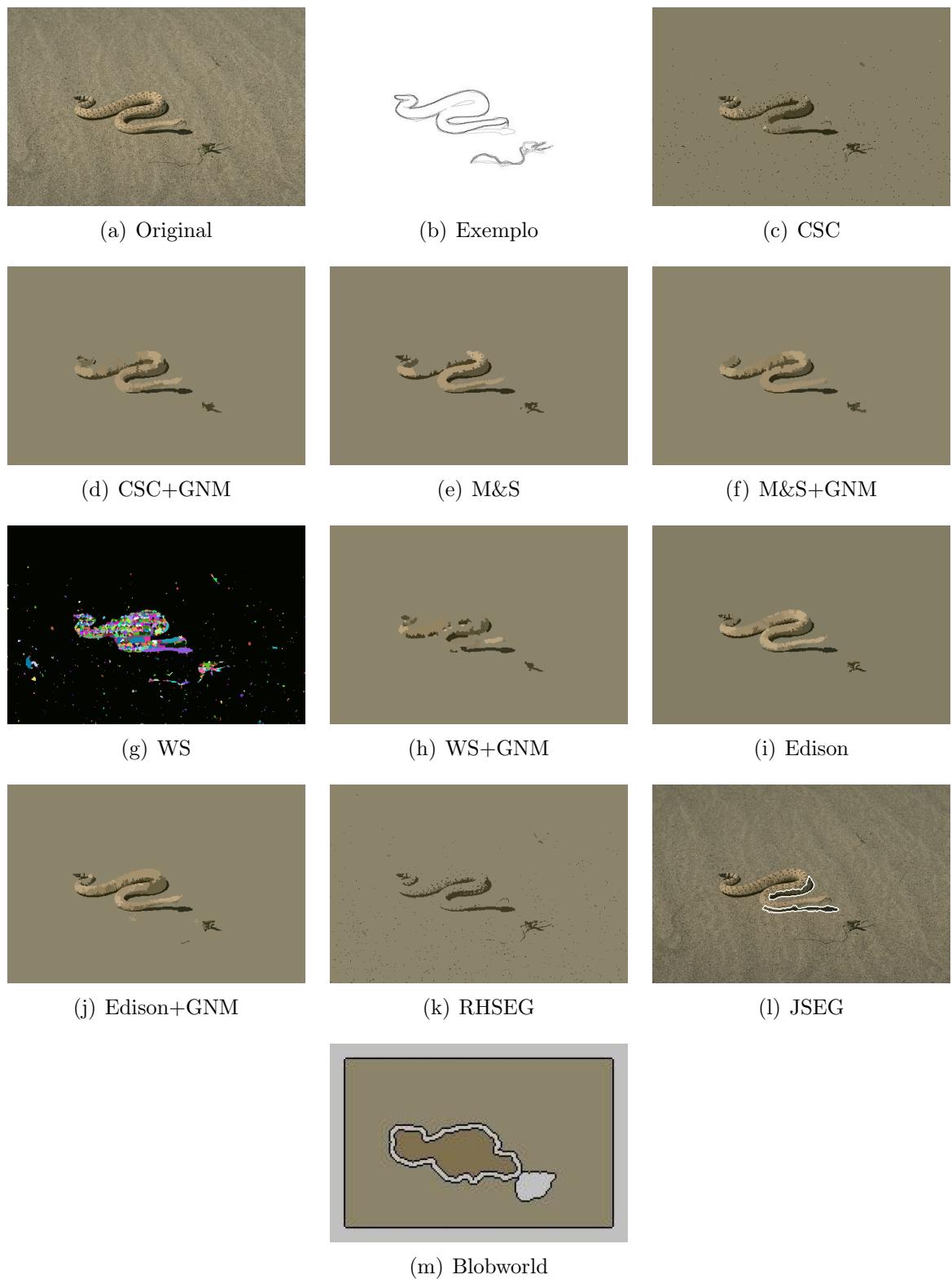


Figura 21: GNM comparado com outros métodos de segmentação usando a imagem 196073 da base de dados de testes de segmentação (MARTIN et al., 2001).

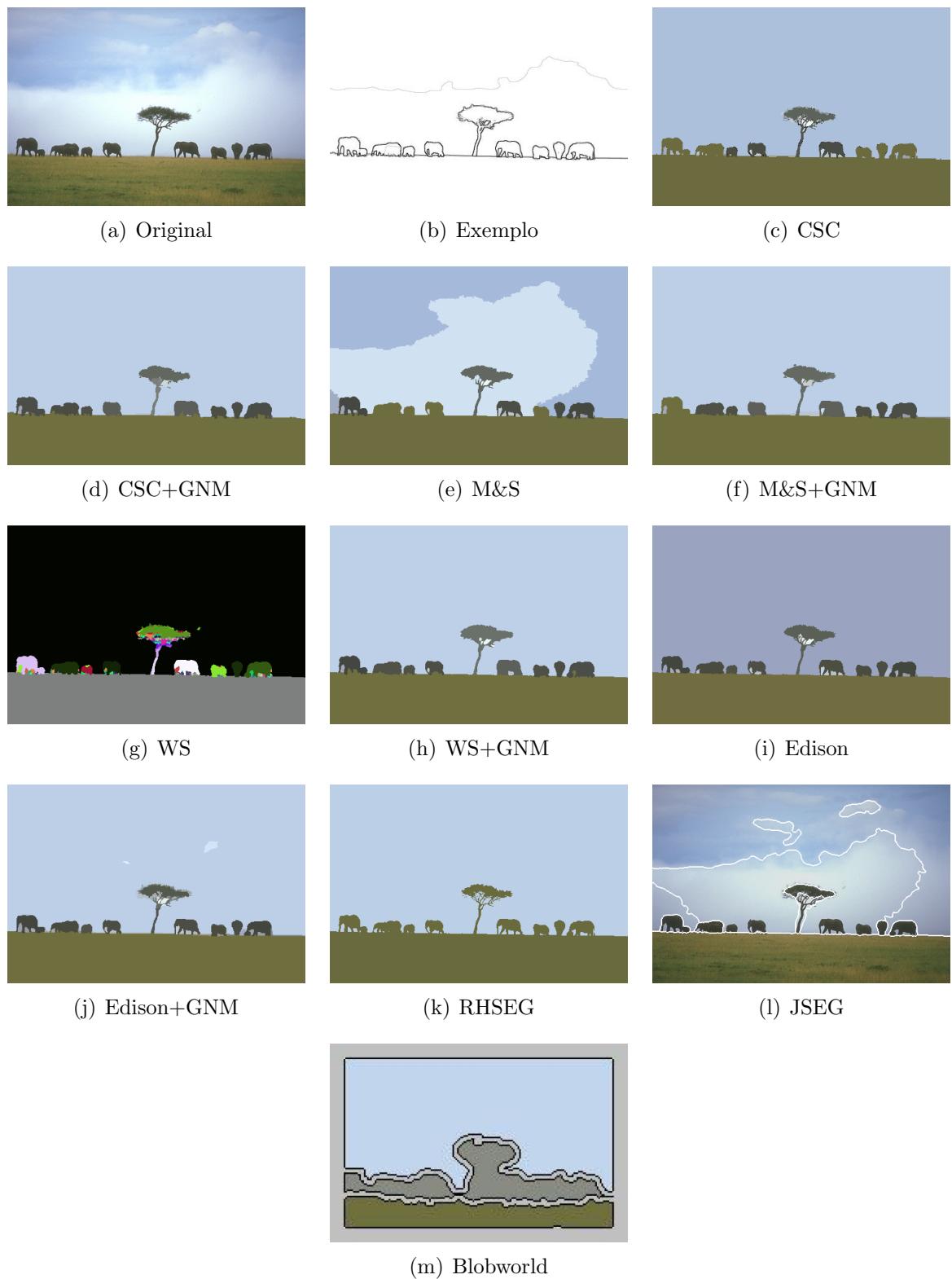


Figura 22: GNM comparado com outros métodos de segmentação usando a imagem 253036 da base de dados de testes de segmentação (MARTIN et al., 2001).

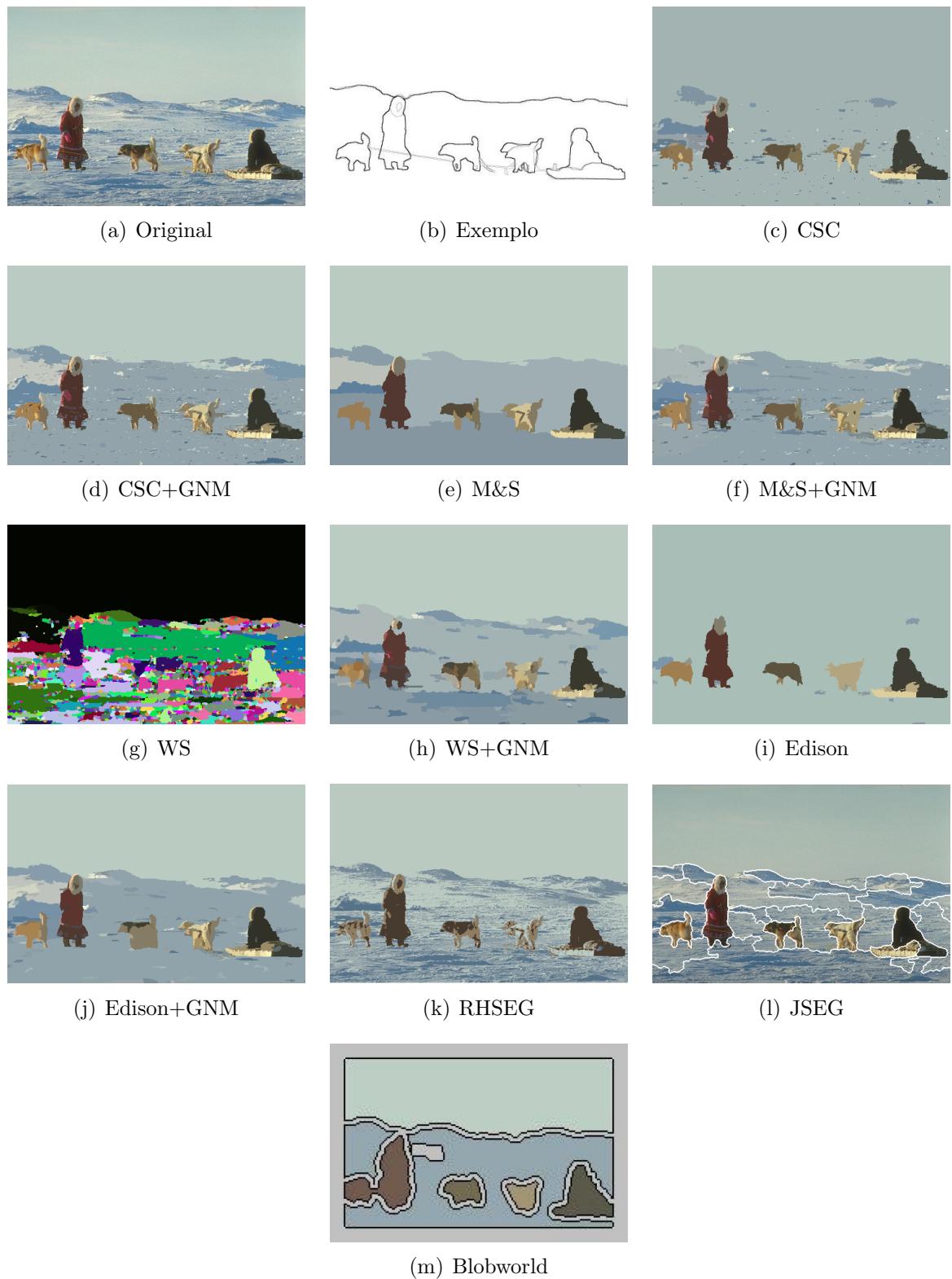


Figura 23: GNM comparado com outros métodos de segmentação usando a imagem 310007 da base de dados de testes de segmentação (MARTIN et al., 2001).

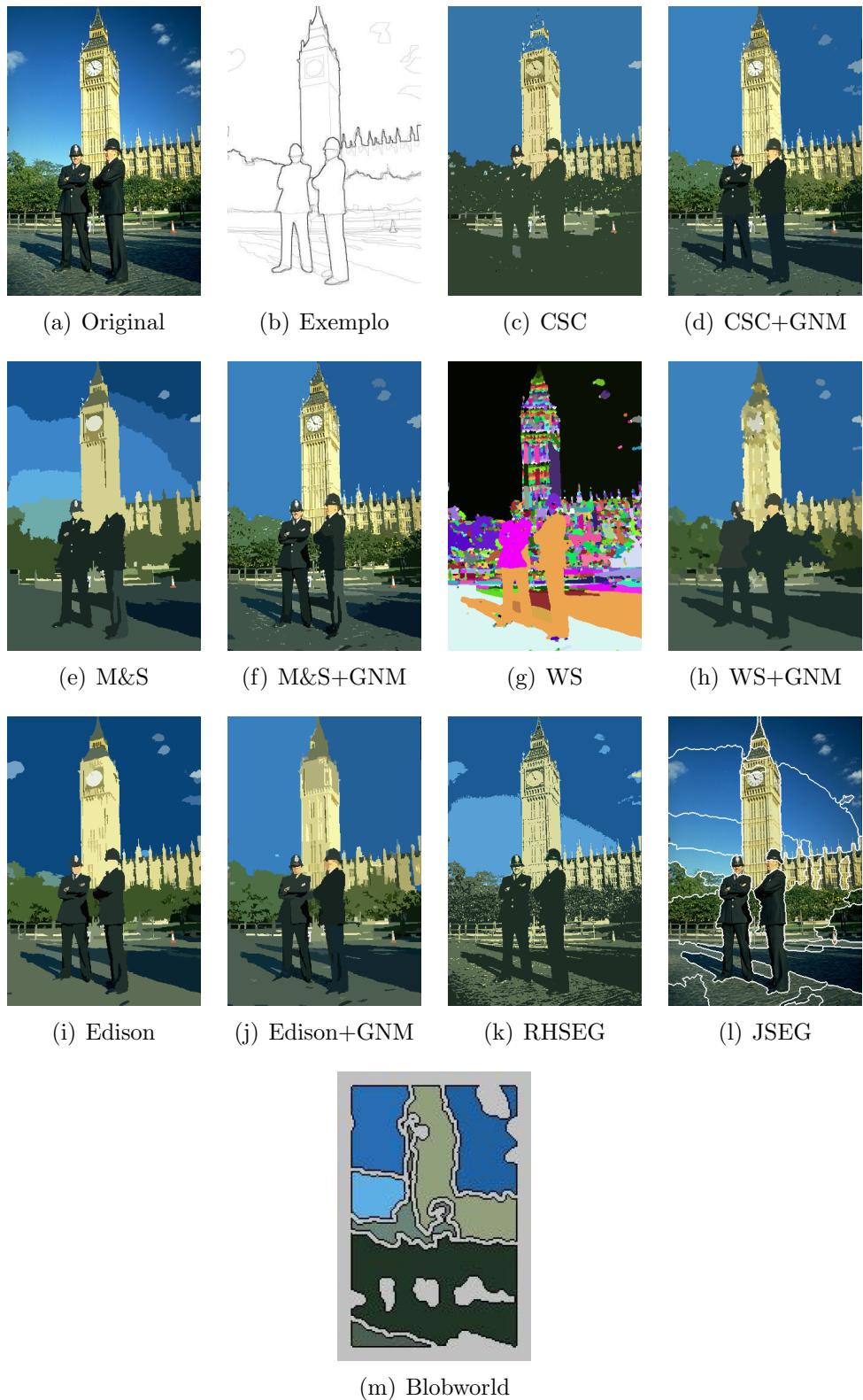


Figura 24: GNM comparado com outros métodos de segmentação usando a imagem 368078 da base de dados de testes de segmentação (MARTIN et al., 2001).

5.1.2 Avaliação

As segmentações de imagens obtidas e selecionadas de acordo com os critérios apresentados foram objetivamente avaliadas com medidas de distância desenvolvidas para a tarefa de avaliar a qualidade de segmentação de imagens. A meta destas medidas é comparar uma segmentação a uma dada imagem de *ground truth* e quantificar o qualidade da identificação dos objetos realizada pela segmentação. O esperado é conseguir a segmentação mais semelhante às regiões esperadas definidas pelo *ground truth*. Nos testes realizados, foram usadas as medidas Rand (RAND, 1971) e BGM (JIANG et al., 2006), respectivamente, uma medida baseada em contagem de pares e outra medida baseada em correspondência de conjuntos. As imagens de *ground truth* providas pela base de imagens de Berkeley foram testadas com ambas as medidas de avaliação com o intuito de prover a qualidade obtida com as segmentações com o GNM. Tanto o Rand quanto o BGM produzem valores dentro do intervalo [0, 1], onde 0 significa uma perfeita correspondência entre segmentação e *ground truth* e 1 significando nenhuma relação entre ambos. O fato que ambas as medidas se encontram dentro de um intervalo específico facilita a comparação entre algoritmos com diferentes abordagens e características, além de permitir a comparação se resultados foram melhor avaliados com uma ou outra medida.

Cada uma das imagens selecionadas foi, então, avaliada com ambas as medidas Rand e BGM para cada um dos 5 a 7 *ground truths* disponibilizados para cada conjunto. Os valores médios de Rand e BGM resultantes para cada um dos pares de conjunto de imagens e algoritmos testados podem ser vistos nas tabelas 1 e 2, respectivamente. A tabela 3 apresenta os valores médios e desvio padrão para cada um dos algoritmos testados para ambas as medidas de avaliação, permitindo uma comparação direta e geral de quais algoritmos obtiveram melhores resultados. As informações dessas tabelas podem ser vista de forma condensada no gráfico presente na figura 25. No anexo A.2 encontram-se os valores de Rand e BGM obtidos para cada *ground truth* de cada conjunto de imagens.

Tabela 1: Valores médios obtidos nos testes com os conjuntos de imagens utilizando o índice de avaliação Rand.

<i>Rand</i>	196073	253036	42049	60079	68077	80099
CSC+GNM	0.033	0.052	0.022	0.037	0.177	0.119
M&S+GNM	0.027	0.055	0.021	0.074	0.085	0.113
Edison+GNM	0.027	0.053	0.019	0.035	0.080	0.114
WS+GNM	0.060	0.048	0.024	0.037	0.097	0.113
CSC	0.064	0.059	0.027	0.037	0.332	0.128
M&S	0.044	0.248	0.051	0.261	0.186	0.127
JSEG	0.083	0.248	0.052	0.239	0.221	0.108
Edison	0.027	0.052	0.043	0.036	0.065	0.109
WS	0.049	0.048	0.022	0.041	0.240	0.124
RHSEG	0.084	0.073	0.311	0.289	0.237	0.174
Blobworld	0.421	0.315	0.283	0.272	0.300	0.391
	2092	15088	24004	46076	310007	368078
CSC+GNM	0.056	0.119	0.183	0.105	0.097	0.079
M&S+GNM	0.106	0.119	0.184	0.123	0.112	0.079
Edison+GNM	0.102	0.125	0.185	0.119	0.095	0.073
WS+GNM	0.070	0.512	0.213	0.369	0.179	0.091
CSC	0.091	0.133	0.188	0.194	0.429	0.235
M&S	0.231	0.278	0.237	0.112	0.134	0.127
JSEG	0.152	0.481	0.232	0.079	0.227	0.138
Edison	0.124	0.120	0.200	0.096	0.437	0.073
WS	0.036	0.549	0.210	0.122	0.203	0.079
RHSEG	0.214	0.127	0.189	0.135	0.248	0.139
Blobworld	0.242	0.606	0.305	0.272	0.265	0.281
	22090	48055	118035	143090		
CSC+GNM	0.049	0.080	0.109	0.091		
M&S+GNM	0.051	0.086	0.102	0.090		
Edison+GNM	0.051	0.079	0.099	0.097		
WS+GNM	0.046	0.139	0.099	0.088		
CSC	0.051	0.124	0.108	0.316		
M&S	0.141	0.152	0.096	0.218		
JSEG	0.054	0.160	0.107	0.235		
Edison	0.108	0.070	0.105	0.094		
WS	0.043	0.131	0.101	0.086		
RHSEG	0.054	0.201	0.101	0.192		
Blobworld	0.230	0.232	0.328	0.266		

Tabela 2: Valores médios obtidos nos testes com os conjuntos de imagens utilizando o índice de avaliação BGM.

<i>Rand</i>	196073	253036	42049	60079	68077	80099
CSC+GNM	0.039	0.012	0.123	0.014	0.309	0.043
M&S+GNM	0.037	0.013	0.103	0.043	0.170	0.054
Edison+GNM	0.038	0.017	0.068	0.014	0.110	0.052
WS+GNM	0.027	0.006	0.132	0.012	0.268	0.017
CSC	0.031	0.014	0.023	0.016	0.032	0.046
M&S	0.033	0.313	0.129	0.330	0.341	0.032
JSEG	0.010	0.282	0.192	0.258	0.501	0.028
Edison	0.042	0.010	0.022	0.014	0.062	0.008
WS	0.062	0.016	0.140	0.034	0.611	0.065
RHSEG	0.019	0.010	0.076	0.492	0.578	0.022
Blobworld	0.319	0.345	0.345	0.308	0.362	0.283
	2092	15088	24004	46076	310007	368078
CSC+GNM	0.124	0.073	0.430	0.318	0.137	0.355
M&S+GNM	0.129	0.109	0.463	0.409	0.181	0.385
Edison+GNM	0.159	0.121	0.429	0.417	0.158	0.328
WS+GNM	0.104	0.439	0.391	0.195	0.345	0.350
CSC	0.114	0.084	0.412	0.245	0.086	0.154
M&S	0.262	0.170	0.557	0.331	0.292	0.489
JSEG	0.056	0.375	0.555	0.259	0.340	0.455
Edison	0.103	0.114	0.424	0.072	0.038	0.314
WS	0.171	0.458	0.405	0.223	0.440	0.369
RHSEG	0.323	0.078	0.487	0.360	0.448	0.389
Blobworld	0.323	0.635	0.433	0.400	0.376	0.439
	22090	48055	118035	143090		
CSC+GNM	0.243	0.109	0.202	0.206		
M&S+GNM	0.284	0.132	0.275	0.206		
Edison+GNM	0.255	0.116	0.226	0.227		
WS+GNM	0.213	0.229	0.233	0.170		
CSC	0.242	0.193	0.192	0.173		
M&S	0.147	0.328	0.205	0.465		
JSEG	0.214	0.413	0.238	0.512		
Edison	0.204	0.080	0.066	0.157		
WS	0.238	0.262	0.241	0.181		
RHSEG	0.257	0.597	0.185	0.447		
Blobworld	0.354	0.383	0.444	0.378		

Tabela 3: Valores médios e desvios padrões dos algoritmos testados com os índices de avaliação Rand e BGM.

	Rand		BGM	
	Média	D. Padrão	Média	D. Padrão
CSC+GNM	0.088	0.047	0.171	0.130
M&S+GNM	0.089	0.040	0.187	0.140
Edison+GNM	0.085	0.042	0.171	0.133
WS+GNM	0.137	0.132	0.196	0.140
CSC	0.157	0.118	0.129	0.111
M&S	0.165	0.074	0.276	0.152
JSEG	0.176	0.107	0.293	0.171
Edison	0.110	0.097	0.108	0.117
WS	0.130	0.130	0.245	0.172
RHSEG	0.173	0.076	0.298	0.208
Blobworld	0.313	0.094	0.383	0.082

Os resultados obtidos com as segmentações combinadas com o GNM obtiveram uma melhor média com a medida Rand, exceto pela combinação WS+GNM, que mostrou valores comparáveis entre ambas medidas de avaliação. O valor médio de Rand para CSC+GNM foi 0.087, para M&S+GNM for 0.089, EDISON+GNM foi 0.085 e para o WS+GNM foi 0.137. O melhor algoritmo após o GNM de acordo com Rand foi Edison com uma média de 0.111, enquanto a pior média foi obtida por Blobworld, 0.313. O testes com o GNM também apresentam a menor variação em termos de desvio padrão, todos por volta de 0.04, exceto para o WS+GNM que teve um desvio padrão de 0.132. O menor desvio padrão depois do GNM foi obtido por Mumford & Shah, 0.074. Então, de acordo com a avaliação feita medida Rand, as combinações de GNM mostraram-se, em geral, as mais eficientes e estáveis.

Os valores médios de BGM para as segmentações resultantes das combinações com GNM não foram os melhores resultados, mas foram próximos ao melhor. O melhor resultado médio obtido com BGM foi obtido por Edison, que teve uma média de 0.108, enquanto o pior foi o Blobworld com 0.383. Os resultados do GNM foram os melhores após àqueles com Edison e CSC, que teve média de 0.129. As médias obtidas com GNM na avaliação realizada pela medida BGM foram 0.171 para CSC+GNM e Edison+GNM, 0.187 para M&S+GNM e 0.196 para WS+GNM. O desvio padrão para todas as om-

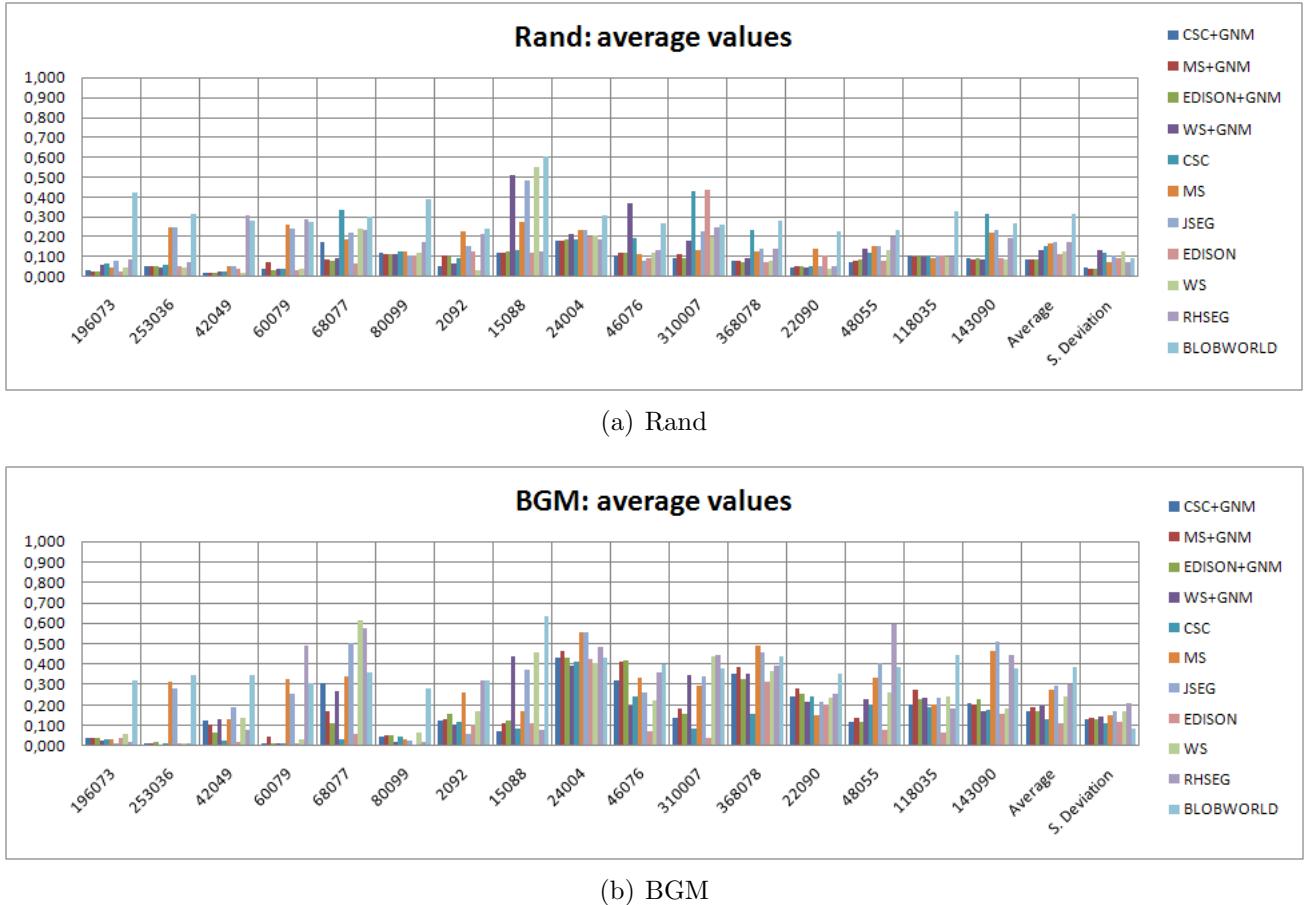


Figura 25: Gráficos sumarizando os dados apresentados nas tabelas 1, 2 e 3.

binações de GNM foram similares, entre 0.130 a 0.140. O menor desvio padrão obtido com BGM pertence ao Blobworld com 0.082, sendo que este também obteve o pior valor médio nesta avaliação. Os próximos melhores valores de desvio padrão foram 0.111 e 0.117 para, respectivamente, CSC e Edison. Considerando que ambas as segmentações também obtiveram as melhores avaliações médias, em termos da avaliação de BGM, CSC e Edison mostraram ser as técnicas mais estáveis e eficientes.

Comparando os resultados do GNM combinado com CSC, M&S, Edison e WS com os resultados obtidos individualmente por estas técnicas (figura 26), o GNM mostra uma melhora sensível para todas as técnicas quando tomado como referência a avaliação da medida Rand, exceto para WS/WS+GNM, onde o valor médio de 0.130 foi um pouco melhor que o obtido com a combinação com GNM. Ainda em relação aos valores obtidos com Rand, M&S/M&S+GNM e CSC/CSC+GNM mostraram o maior ganho de quali-

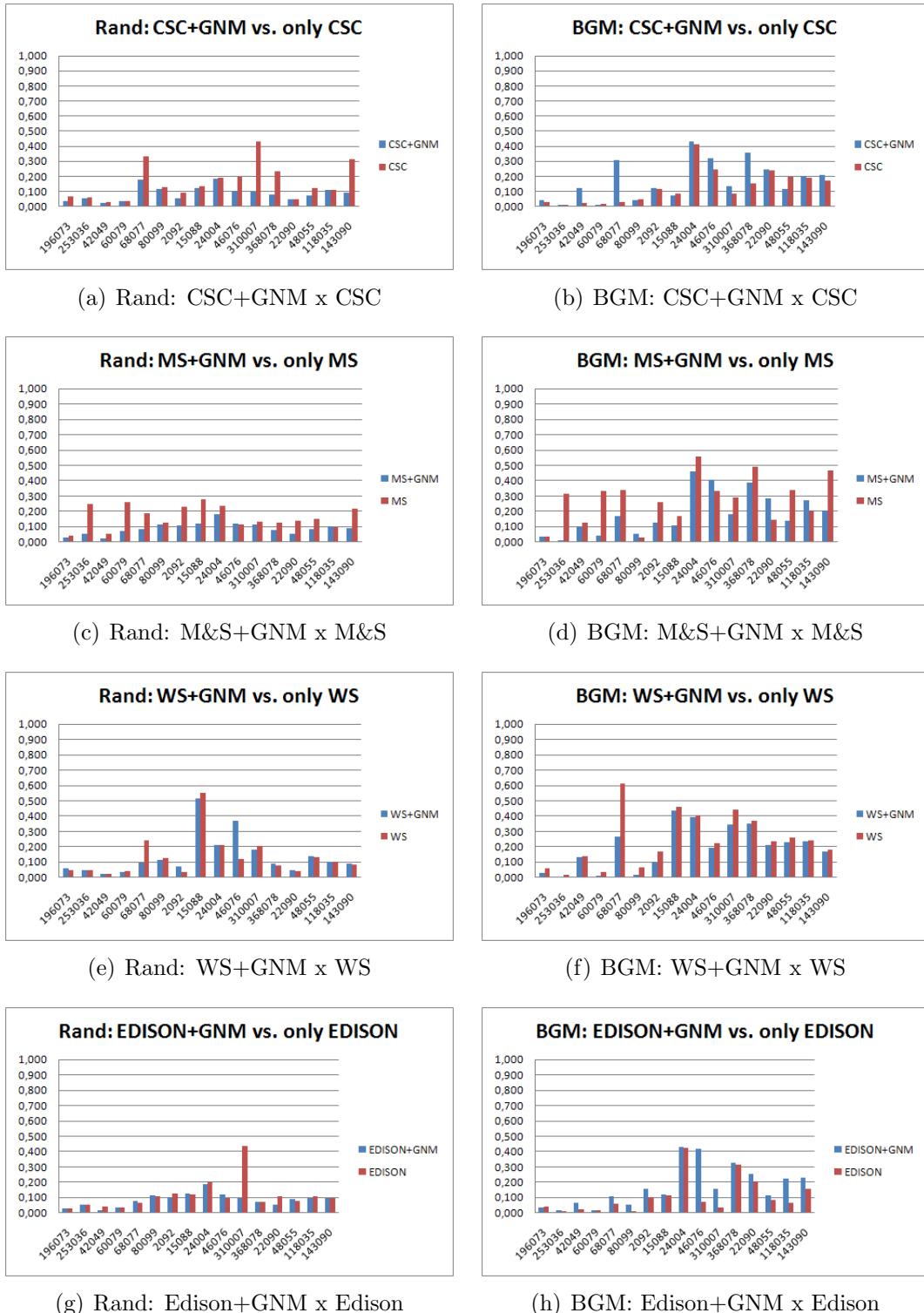


Figura 26: Comparação entre os valores médios de Rand e BGM entre as segmentações CSC, M&S, WS e Edison combinadas com GNM contra a aplicação individual destas.

dade (respectivamente, 0.165 contra 0.089 e 0.157 contra 0.087). Os valores resultantes da avaliação de BGM mostraram uma divisão, com CSC/CSC+GNM (0.129 contra 0.171) e Edison (0.108 contra 0.171) mostrando melhores resultados quando aplicados individualmente enquanto M&S/M&S+GNM (0.277 contra 0.187) e WS/WS+GNM (0.245 contra 0.196) foram aperfeiçoados com a combinação com o GNM. Novamente, GNM mostra melhor performance quando avaliado em termos das condições de Rand do que de BGM, apesar que os resultados com BGM ainda são melhores que algumas outras técnicas e praticamente tão boas a outras melhor avaliadas.

Considerando as abordagens das medidas de avaliação utilizadas, GNM mostra ser mais apto a produzir resultados com melhor qualidade em termos de pareamento de regiões com as imagens de *ground truth* testadas. A avaliação de BGM beneficia abordagens mais agressivas, mesmo que a segmentação avaliada seja de alguma forma sub-segmentada, como, por exemplo, apresentando alguns vazamentos de segmentos. Rand fornece melhores valores para segmentações que preservam detalhes melhor. GNM prova ser uma técnica confiável por ambas as medidas, especialmente em relação a medida a Rand, e para ambas as medidas os resultados apresentaram alguns dos melhores valores em termos de desvio de padrão, também mostrando que o GNM é uma técnica estável. Outro algoritmo que se mostrou bastante confiável por ambas medidas foi Edison, mostrando que a abordagem por *mean-shift* também é interessante quando buscam-se objetos formados por longos e progressivos gradientes de cores. Blobworld teve os menores graus de similaridade avaliados, o que de certa forma era esperado, dada suas características inerentes.

5.2 Resultados dos testes de desempenho de velocidade

Para validar empiricamente a abordagem, várias imagens ao ar livre foram comparativamente processadas com um conjunto diferente de algoritmos. Para permitir comparações, todos testes foram realizados em um mesmo computador com os algoritmos

apresentados abaixo.

A adoção da base de dados de imagens de Berkeley (MARTIN et al., 2001) é feita novamente, pois é uma necessária e desejável escolha já é uma conjunto de dados bem conhecido com a característica de possuir imagens de referência, geradas a partir de segmentações manuais, para cada conjunto. A comparação com outras técnicas atualmente desenvolvidas é facilitada aos leitores já que diversos outros autores e publicações fazem uso desta mesma base de imagens.

As técnicas de segmentação usadas nos seguintes testes foram: a) GNM aplicado sobre imagens pré-segmentadas por CSC, com o limiar igual a 30 (tempo total será a soma do tempo de execução de ambos os algoritmos); b) GNM aplicado sobre imagens pré-segmentadas pela segmentação baseada na funcional de Mumford&Shah, com *lambda* igual a 600 (tempo total será a soma do tempo de execução de ambos os algoritmos); c) CSC, com um limiar selecionado para produzir o melhor resultado; d) segmentação baseada na funcional de Mumford&Shah, com um *lambda* selecionado para produzir o melhor resultado; e) JSEG (DENG; MANJUNATH, 2001), uma técnica não supervisionada bem conhecida.

Tempo total de execução para cada conjunto com cada algoritmo selecionado é mostrado na Tabela 4. O tempo de execução para é obtido através da diferença de dois marcadores de tempo, um no início e outro no fim do processo de execução de cada algoritmo. Média e desvio padrão também são apresentados.

Os testes foram executados em um AMD Athlon 64 2.2 GHz e a unidade de tempo é segundos.

Tabela 4: O tempo de execução total para o conjunto de imagens do conjunto de dados de Berkeley processados usando um conjunto de técnicas selecionadas. O tempo médio e desvio padrão são mostrados nas últimas linhas da tabela.

Imagen	CSC+GNM	M&S+GNM	CSC	M&S	JSEG
Média:	0.680s	5.622s	0.080s	5.056s	11.972s
Desvio padrão:	0.059s	0.941s	0.010s	0.829s	4.149s

Como a tabela 4 mostra, a combinação de CSC e GNM mostra resultados com um valor por volta de um segundo e meio, sendo mais lento somente do que a aplicação individual do CSC. Isto era esperado, considerando os tempos cumulativos de CSC+GNM. O tempo médio é várias vezes menor que o do Mumford&Shah e JSEG. Há pouca variação entre os tempos obtidos para o CSC+GNM, enquanto novamente de acordo com a exceção do CSC individualmente, todas técnicas mostram variações mais altas.

Comparando CSC+GNM com M&S+GNM, sem considerar a aplicação do pós-processamento, observa-se que o GNM leva mais tempo em média no caso CSC+GNM do que no caso M&S+GNM. Isto é consequência da existência de vários pequenos fragmentos que são produzidos pelo CSC, algo que não ocorre no Mumford&Shah. O resultado é um número maior de vértices do grafo que precisam ser avaliados no caso do CSC+GNM. É importante perceber que, porém, o GNM tem uma performance estável em ambos os casos, com pequena variação tanto no caso do CSC+GNM quanto no M&S+GNM.

O pós-processamento, em média, consumiu 0.06s nos testes realizados aqui. A implementação atual desse passo de pós-processamento foi feita de forma a se integrar ao restante do processo do algoritmo e por isso age de forma otimizada. Essa pequena perda de desempenho é facilmente compensada por uma maior qualidade nos resultados em termos de imagem.

A comparação realizada em termos de velocidade é simples e não pode ser vista como um *benchmark* definitivo, mas sim como um indicativo da performance do GNM e como esta é uma técnica razoavelmente rápida enquanto também é capaz de produzir segmentações com qualidade comparável com técnicas do estado da arte.

6 Discussão

Simplicidade foi uma das metas buscadas quando este método foi concebido. Abordagens similares foram realizadas anteriormente por (TRÉMEAU; COLANTONI, 2000) e (WU; LEAHY, 1993), onde o problema era definido com uma otimização do fluxo de grafos em sub-grafos. Outra abordagem de particionamento de grafos introduz uma medida global obtida através de uma análise do principal componente de um conjunto de diferentes parâmetros associadas com cada elemento da imagem (DUPUIS; VASSEUR, 2006). Nossa abordagem também busca sub-grafos na rede original mas é muito mais simples e age somente localmente em cada aresta, sem tentar encontrar um valor ótimo para todo o sub-grafo que representa uma região e sem usar qualquer informação global da imagem ou função de custo. São aplicadas somente simples medidas de similaridade não-adaptativas e produzem resultados de qualidade em imagens de razoável dificuldade, comparável a abordagens mais complexas. O método é baseia-se em que padrões locais de variação de características da imagem são tão importantes quanto as características estáticas. Para descobrir ordem nestes padrões é essencial a um algoritmo de segmentação. O processo de descobrir os padrões de variação da imagem é descrito como um processo de local baseado em similaridade por um simples e consistente *framework*.

A abordagem do GNM pode ser vista como um procedimento de *labeling*. O objetivo de técnicas como essa é rotular unicamente todos objetos em uma imagem. Geralmente é um processo aplicado depois do uso de algum outro método de pré-processamento de imagens, muitas vezes uma segmentação. Porém, o resultado de uma segmentação de crescimento por regiões se comporta como um método de *labeling*, delegando pixels para regiões

e associando valores de cor média com eles. Deste ponto de vista, o pré-processamento em que nossa abordagem se baseia é uma pré-segmentação com um método que destaca os óbvios pedaços de regiões onde um análise de padrão de gradientes não é necessária, assim evitando processamento desnecessário.

A principal diferença entre este método e outros métodos de *labeling* de pós-processamento de imagens desenvolvidos para buscar melhores resultados sobre segmentações é que o método aqui apresentado não é baseado em conhecimento ou em qualquer modelo ou domínio, isto é, não necessita de conhecimento adicional sobre o conteúdo ou o domínio de aplicação das imagens, dependendo somente de informação presente na imagem.

7 Conclusão

Foi mostrado empiricamente e através de medidas de avaliação reconhecidas que a qualidade das segmentações geradas por esta abordagem de dois passos é promissora e comparável a métodos do estado da arte da área de segmentação de imagens. O algoritmo de segmentação aqui apresentado é um método de segmentação de pós-processamento que é independente do método aplicado para gerar a imagem de entrada super-segmentada, como mostrado pelas comparações entre os resultados produzidos por quatro diferentes métodos de segmentação de imagens contra a combinação destes com o GNM.

É importante notar ainda que qualquer medida de avaliação que busca definir o “melhor” resultado possível em uma área complexa, como a segmentação de imagens, deve ter seus valores de avaliação considerados com cuidado. Estas medidas não provêm uma forma definitiva de determinar qual é a melhor segmentação, mas fornecem um interessante e objetivo recurso. Com essas medidas é possível avaliar diferentes segmentações sob os diferentes contextos e propriedades que a abordagem que as medidas de avaliação utilizam. Ambas as medidas de avaliação aqui selecionadas, Rand e BGM, por exemplo, aplicam abordagens bastante distintas para determinar aquilo que consideram como uma melhor segmentação e, tal como é difícil definir quão correta uma segmentação é, é difícil argumentar qual prova ser mais correta. Estas medidas provêm, afinal, formas de encontrar se, sob as condições esperadas pela abordagem destas, uma segmentação é capaz de agir em um desejado nível de qualidade. Observando os resultados apresentados em seções anteriores, é possível ver que o GNM, sob as propriedades e condições de ambas as técnicas selecionadas, pode obter segmentações com um nível de qualidade no mínimo

similar a várias técnicas reconhecidas. Os resultados obtidos com GNM ainda apresentam bons resultados em termos de velocidade quando comparado com outras técnicas.

O modo de atuação como um segundo passo ou um passo seqüencial sobre outras técnicas pode parecer diferir do convencional mas esta escolha tem somente o efeito que a segmentação grosseira inicial será benéfica ao GNM para passos futuros contanto que poucos vazamentos ocorram nessa identificação dos objetos. Como o GNM atualmente está voltado a busca de propriedades mais específicas a combinação de propriedades oferecidas por outras técnicas somente faz o algoritmo apresentado atingir melhores resultados.

O algoritmo do GNM, apesar de ter mostrado que sua abordagem é válida, ainda pode receber aperfeiçoamentos. Como futuros trabalhos existem: fortalecer o componente estrutural para reconhecimento dos gradientes longos e contínuos, utilizando mais conhecimento sobre as informações topológicas conhecidas; tornar a função de avaliação menos rígida, especialmente em relação a utilização de limiares; e aperfeiçoar ainda mais a capacidade perceptiva da função de similaridade do algoritmo, para tornar o algoritmo ainda mais robusto em relação a efeitos como ofuscamento e sombreamento. Um aperfeiçoamento possível em termos da implementação é a utilização da tecnologia de processadores gráficos de uso geral (*GP-GPU*), de modo a tornar a sua execução muito mais veloz.

ANEXO A - Anexos

A.1 Parâmetros dos algoritmos testados com as imagens selecionadas e avaliadas

Tabela 5: Os parâmetros específicos de cada algoritmo testado que foram selecionados como aqueles que produzem uma melhor segmentação seguindo critérios apresentados em seções anteriores.

	CSCGNM		MSGNM		EDISONGNM		WSGNM	
	T_{cp}	T_{rp}	T_{cp}	T_{rp}	T_{cp}	T_{rp}	T_{cp}	T_{rp}
118035	0.065	0.035	0.055	0.060	0.060	0.050	0.010	0.010
143090	0.035	0.015	0.035	0.045	0.030	0.035	0.070	0.015
15088	0.045	0.060	0.040	0.045	0.035	0.060	0.010	0.010
196073	0.015	0.065	0.020	0.060	0.020	0.055	0.050	0.015
2092	0.025	0.055	0.040	0.035	0.030	0.045	0.060	0.020
22090	0.025	0.065	0.020	0.070	0.025	0.065	0.010	0.010
24004	0.035	0.025	0.035	0.045	0.060	0.035	0.030	0.020
253036	0.065	0.020	0.055	0.045	0.060	0.060	0.010	0.015
310007	0.030	0.065	0.040	0.055	0.035	0.045	0.030	0.010
368078	0.050	0.065	0.055	0.070	0.065	0.070	0.010	0.015
42049	0.065	0.045	0.065	0.060	0.050	0.070	0.040	0.020
46076	0.035	0.060	0.040	0.065	0.045	0.045	0.065	0.035
48055	0.035	0.055	0.045	0.065	0.050	0.055	0.4	0.55
60079	0.015	0.060	0.020	0.060	0.025	0.030	0.010	0.020
68077	0.040	0.060	0.045	0.060	0.040	0.065	0.040	0.010
80099	0.015	0.055	0.055	0.040	0.050	0.030	0.070	0.015
	CSC	MS	Edison		WS	RHSEG		
	Limiar	Lambda	SR	SS	Nível	Nível		
118035	60	10000	20	8	0	18		
143090	70	10500	13	8	0.01	13		
15088	80	15000	14	8	0.01	17		
196073	50	3000	5	8	0.01	12		
2092	60	14000	16	8	0.01	27		
22090	40	11000	10	8	0.01	11		
24004	50	12000	12	8	0.01	22		
253036	60	11500	19	8	0.01	44		
310007	60	8000	19	8	0.01	13		
368078	90	11000	14	8	0.01	30		
42049	100	14500	20	8	0.01	29		
46076	70	8000	19	8	0.01	28		
48055	70	11500	19	8	0.01	16		
60079	40	10500	9	8	0.01	45		
68077	90	11000	21	8	0.01	16		
80099	40	10000	15	8	0.01	19		

A.2 Resultados obtidos com cada conjunto de imagens para cada um de seus respectivos ground truths

A.2.1 Rand

Tabela 6: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por CSC+GNM.

196073	253036	42049	60079	68077	80099
0.037	0.012	0.022	0.038	0.180	0.016
0.032	0.254	0.015	0.021	0.163	0.520
0.032	0.010	0.028	0.010	0.174	0.019
0.031	0.016	0.023	0.075	0.178	0.019
0.034	0.011	0.021	0.041	0.189	0.020
0.011				0.176	
Média	Média	Média	Média	Média	Média
0.033	0.052	0.022	0.037	0.177	0.119
D. Padrão					
0.003	0.099	0.005	0.025	0.009	0.224
2092	15088	24004	46076	310007	368078
0.057	0.117	0.293	0.117	0.096	0.075
0.059	0.120	0.129	0.098	0.095	0.138
0.061	0.121	0.309	0.117	0.095	0.044
0.057	0.116	0.132	0.093	0.096	0.068
0.055	0.123	0.054	0.098	0.101	0.045
0.044					0.128
0.056					0.057
Média	Média	Média	Média	Média	Média
0.056	0.119	0.183	0.105	0.097	0.079
D. Padrão					
0.005	0.003	0.112	0.012	0.003	0.039
22090	48055	118035	143090		
0.049	0.060	0.102	0.067		
0.050	0.129	0.136	0.125		
0.054	0.064	0.083	0.051		
0.069	0.061	0.155	0.154		
0.024	0.106	0.067	0.060		
0.049	0.063				
Média	Média	Média	Média		
0.049	0.080	0.109	0.091		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.014	0.029	0.036	0.045		

Tabela 7: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por M&S+GNM.

196073	253036	42049	60079	68077	80099
0.031	0.014	0.020	0.076	0.077	0.012
0.026	0.256	0.018	0.057	0.065	0.513
0.026	0.013	0.025	0.050	0.087	0.013
0.024	0.019	0.021	0.110	0.085	0.012
0.027	0.014	0.019	0.077	0.108	0.015
	0.014			0.088	
Média	Média	Média	Média	Média	Média
0.027	0.055	0.021	0.074	0.085	0.113
D. Padrão					
0.003	0.098	0.003	0.023	0.014	0.224
2092	15088	24004	46076	310007	368078
0.108	0.114	0.296	0.135	0.111	0.076
0.109	0.118	0.131	0.117	0.111	0.139
0.112	0.118	0.311	0.135	0.111	0.042
0.108	0.127	0.132	0.110	0.112	0.066
0.106	0.119	0.047	0.116	0.114	0.043
0.096					0.129
0.106					0.055
Média	Média	Média	Média	Média	Média
0.106	0.119	0.184	0.123	0.112	0.079
D. Padrão					
0.005	0.005	0.115	0.012	0.001	0.040
22090	48055	118035	143090		
0.053	0.078	0.086	0.065		
0.052	0.111	0.126	0.120		
0.059	0.080	0.061	0.051		
0.066	0.078	0.184	0.152		
0.025	0.088	0.051	0.059		
0.053	0.080				
Média	Média	Média	Média		
0.051	0.086	0.102	0.090		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.014	0.013	0.055	0.044		

Tabela 8: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por Edison+GNM.

196073	253036	42049	60079	68077	80099
0.024	0.253	0.017	0.039	0.070	0.513
0.032	0.011	0.019	0.035	0.080	0.014
0.030	0.011	0.015	0.021	0.081	0.014
0.023	0.011	0.027	0.008	0.081	0.013
0.027	0.018	0.017	0.074	0.087	0.015
0.012				0.081	
Média	Média	Média	Média	Média	Média
0.027	0.053	0.019	0.035	0.080	0.114
D. Padrão					
0.004	0.098	0.004	0.025	0.005	0.223
2092	15088	24004	46076	310007	368078
0.103	0.138	0.149	0.132	0.093	0.062
0.102	0.118	0.131	0.113	0.095	0.068
0.099	0.123	0.299	0.106	0.095	0.036
0.102	0.120	0.060	0.132	0.099	0.040
0.105	0.126	0.284	0.112	0.094	0.056
0.100					0.121
0.105					0.131
Média	Média	Média	Média	Média	Média
0.102	0.125	0.185	0.119	0.095	0.073
D. Padrão					
0.002	0.008	0.103	0.012	0.002	0.038
22090	48055	118035	143090		
0.052	0.046	0.174	0.133		
0.051	0.150	0.123	0.057		
0.068	0.050	0.049	0.076		
0.054	0.046	0.061	0.154		
0.056	0.132	0.085	0.064		
0.024	0.049				
Média	Média	Média	Média		
0.051	0.079	0.099	0.097		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.015	0.049	0.051	0.044		

Tabela 9: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por WS+GNM.

196073	253036	42049	60079	68077	80099
0.060	0.005	0.024	0.036	0.100	0.012
0.063	0.251	0.022	0.040	0.125	0.014
0.059	0.013	0.024	0.076	0.085	0.013
0.058	0.006	0.018	0.022	0.099	0.515
0.058	0.007	0.031	0.009	0.089	0.010
	0.007			0.081	
Média	Média	Média	Média	Média	Média
0.060	0.048	0.024	0.037	0.097	0.113
D. Padrão					
0.002	0.100	0.005	0.025	0.016	0.225
2092	15088	24004	46076	310007	368078
0.072	0.509	0.344	0.358	0.178	0.089
0.073	0.515	0.181	0.377	0.179	0.146
0.076	0.510	0.359	0.358	0.178	0.056
0.071	0.528	0.080	0.378	0.179	0.079
0.070	0.500	0.103	0.377	0.180	0.059
0.057					0.136
0.069					0.071
Média	Média	Média	Média	Média	Média
0.070	0.512	0.213	0.369	0.179	0.091
D. Padrão					
0.006	0.010	0.131	0.011	0.001	0.036
22090	48055	118035	143090		
0.041	0.102	0.087	0.040		
0.043	0.224	0.122	0.098		
0.045	0.103	0.062	0.062		
0.064	0.105	0.176	0.169		
0.041	0.199	0.051	0.072		
0.040	0.102				
Média	Média	Média	Média		
0.046	0.139	0.099	0.088		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.009	0.057	0.051	0.050		

Tabela 10: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por *Color Structure Code*.

196073	253036	42049	60079	68077	80099
0.067	0.017	0.027	0.038	0.340	0.026
0.063	0.261	0.031	0.021	0.358	0.528
0.065	0.017	0.023	0.010	0.325	0.029
0.064	0.023	0.029	0.074	0.338	0.029
0.062	0.017	0.026	0.041	0.309	0.030
	0.017			0.324	
Média	Média	Média	Média	Média	Média
0.064	0.059	0.027	0.037	0.332	0.128
D. Padrão					
0.002	0.099	0.003	0.024	0.017	0.223
2092	15088	24004	46076	310007	368078
0.093	0.126	0.300	0.205	0.429	0.241
0.094	0.134	0.137	0.186	0.428	0.220
0.097	0.129	0.315	0.206	0.427	0.261
0.093	0.146	0.126	0.187	0.429	0.243
0.091	0.129	0.064	0.187	0.435	0.229
0.080					0.212
0.092					0.242
Média	Média	Média	Média	Média	Média
0.091	0.133	0.188	0.194	0.429	0.235
D. Padrão					
0.005	0.008	0.113	0.010	0.003	0.016
22090	48055	118035	143090		
0.051	0.106	0.102	0.311		
0.051	0.148	0.136	0.371		
0.055	0.112	0.083	0.291		
0.070	0.108	0.154	0.302		
0.026	0.160	0.067	0.305		
0.050	0.110				
Média	Média	Média	Média		
0.051	0.124	0.108	0.316		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.014	0.023	0.036	0.032		

Tabela 11: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por Mumford & Shah.

196073	253036	42049	60079	68077	80099
0.048	0.259	0.051	0.249	0.189	0.024
0.043	0.189	0.045	0.266	0.175	0.530
0.043	0.259	0.055	0.251	0.179	0.027
0.042	0.263	0.052	0.290	0.196	0.027
0.043	0.260	0.050	0.249	0.199	0.027
	0.259			0.181	
Média	Média	Média	Média	Média	Média
0.044	0.248	0.051	0.261	0.186	0.127
D. Padrão					
0.002	0.029	0.004	0.018	0.010	0.226
2092	15088	24004	46076	310007	368078
0.233	0.277	0.335	0.123	0.133	0.124
0.233	0.279	0.178	0.106	0.133	0.183
0.234	0.281	0.351	0.123	0.133	0.095
0.233	0.275	0.211	0.102	0.134	0.113
0.233	0.280	0.110	0.104	0.136	0.100
0.217					0.174
0.234					0.099
Média	Média	Média	Média	Média	Média
0.231	0.278	0.237	0.112	0.134	0.127
D. Padrão					
0.006	0.002	0.103	0.010	0.001	0.037
22090	48055	118035	143090		
0.140	0.153	0.083	0.244		
0.147	0.159	0.116	0.303		
0.144	0.157	0.063	0.219		
0.166	0.154	0.170	0.115		
0.113	0.135	0.047	0.210		
0.138	0.156				
Média	Média	Média	Média		
0.141	0.152	0.096	0.218		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.017	0.009	0.049	0.068		

Tabela 12: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por Edison.

196073	253036	42049	60079	68077	80099
0.026	0.255	0.043	0.040	0.079	0.512
0.031	0.010	0.045	0.037	0.061	0.008
0.028	0.010	0.047	0.018	0.061	0.007
0.025	0.012	0.039	0.009	0.060	0.008
0.026	0.014	0.042	0.073	0.063	0.010
	0.011			0.067	
Média	Média	Média	Média	Média	Média
0.027	0.052	0.043	0.036	0.065	0.109
D. Padrão					
0.002	0.100	0.003	0.024	0.007	0.225
2092	15088	24004	46076	310007	368078
0.127	0.118	0.106	0.082	0.435	0.062
0.127	0.124	0.157	0.105	0.436	0.070
0.124	0.122	0.337	0.108	0.437	0.035
0.126	0.118	0.080	0.083	0.444	0.042
0.111	0.119	0.321	0.105	0.434	0.047
0.125					0.123
0.130					0.133
Média	Média	Média	Média	Média	Média
0.124	0.120	0.200	0.096	0.437	0.073
D. Padrão					
0.006	0.003	0.121	0.013	0.004	0.039
22090	48055	118035	143090		
0.106	0.031	0.029	0.123		
0.106	0.161	0.104	0.054		
0.126	0.029	0.124	0.067		
0.104	0.029	0.168	0.159		
0.110	0.139	0.103	0.065		
0.094	0.031				
Média	Média	Média	Média		
0.108	0.070	0.105	0.094		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.010	0.063	0.050	0.045		

Tabela 13: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por *Watershed*.

196073	253036	42049	60079	68077	80099
0.046	0.006	0.022	0.048	0.248	0.026
0.050	0.251	0.020	0.051	0.267	0.028
0.052	0.014	0.022	0.068	0.233	0.026
0.051	0.006	0.016	0.017	0.247	0.513
0.047	0.007	0.029	0.020	0.212	0.025
0.007				0.230	
Média	Média	Média	Média	Média	Média
0.049	0.048	0.022	0.041	0.240	0.124
D. Padrão					
0.003	0.099	0.005	0.022	0.019	0.218
2092	15088	24004	46076	310007	368078
0.035	0.546	0.340	0.113	0.202	0.081
0.038	0.552	0.177	0.128	0.205	0.143
0.039	0.546	0.355	0.113	0.203	0.047
0.036	0.566	0.083	0.127	0.204	0.071
0.033	0.536	0.097	0.127	0.203	0.050
0.039					
0.033					
Média	Média	Média	Média	Média	Média
0.036	0.549	0.210	0.122	0.203	0.079
D. Padrão					
0.003	0.011	0.130	0.008	0.001	0.039
22090	48055	118035	143090		
0.038	0.095	0.089	0.039		
0.040	0.212	0.124	0.100		
0.043	0.100	0.061	0.059		
0.061	0.099	0.178	0.163		
0.038	0.186	0.053	0.069		
0.038	0.095				
Média	Média	Média	Média		
0.043	0.131	0.101	0.086		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.009	0.053	0.051	0.048		

Tabela 14: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por JSEG.

196073	253036	42049	60079	68077	80099
0.079	0.120	0.052	0.231	0.195	0.511
0.088	0.273	0.053	0.231	0.217	0.008
0.088	0.274	0.047	0.238	0.227	0.007
0.079	0.272	0.057	0.223	0.226	0.008
0.081	0.276	0.053	0.271	0.247	0.010
0.274				0.212	
Média	Média	Média	Média	Média	Média
0.083	0.248	0.052	0.239	0.221	0.108
D. Padrão					
0.005	0.063	0.004	0.019	0.017	0.225
2092	15088	24004	46076	310007	368078
0.154	0.495	0.207	0.090	0.227	0.123
0.154	0.473	0.175	0.072	0.228	0.136
0.152	0.478	0.343	0.067	0.226	0.102
0.153	0.476	0.106	0.092	0.228	0.115
0.140	0.482	0.328	0.073	0.228	0.114
0.152					0.185
0.157					0.191
Média	Média	Média	Média	Média	Média
0.152	0.481	0.232	0.079	0.227	0.138
D. Padrão					
0.005	0.009	0.102	0.011	0.001	0.036
22090	48055	118035	143090		
0.059	0.193	0.194	0.316		
0.054	0.099	0.109	0.232		
0.065	0.194	0.065	0.255		
0.057	0.194	0.082	0.150		
0.061	0.083	0.086	0.221		
0.030	0.197				
Média	Média	Média	Média		
0.054	0.160	0.107	0.235		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.012	0.054	0.051	0.060		

Tabela 15: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por RHSEG.

196073	253036	42049	60079	68077	80099
0.083	0.277	0.312	0.276	0.231	0.577
0.084	0.036	0.310	0.290	0.235	0.073
0.082	0.032	0.306	0.294	0.242	0.073
0.088	0.031	0.311	0.294	0.241	0.071
0.084	0.033	0.314	0.293	0.214	0.075
	0.030			0.260	
Média	Média	Média	Média	Média	Média
0.084	0.073	0.311	0.289	0.237	0.174
D. Padrão					
0.002	0.100	0.003	0.008	0.015	0.225
2092	15088	24004	46076	310007	368078
0.217	0.126	0.137	0.146	0.249	0.120
0.217	0.130	0.316	0.119	0.248	0.115
0.216	0.122	0.301	0.147	0.248	0.137
0.204	0.128	0.055	0.131	0.247	0.120
0.216	0.131	0.136	0.130	0.250	0.177
0.216					0.133
0.215					0.169
Média	Média	Média	Média	Média	Média
0.214	0.127	0.189	0.135	0.248	0.139
D. Padrão					
0.004	0.004	0.114	0.012	0.001	0.025
22090	48055	118035	143090		
0.053	0.241	0.146	0.201		
0.054	0.105	0.059	0.192		
0.071	0.243	0.129	0.189		
0.054	0.242	0.094	0.262		
0.059	0.129	0.077	0.115		
0.032	0.245				
Média	Média	Média	Média		
0.054	0.201	0.101	0.192		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.013	0.065	0.036	0.052		

Tabela 16: Valores da avaliação de Rand para todos os *ground-truths* comparados com os resultados produzidos por Blobworld.

196073	253036	42049	60079	68077	80099
0.422	0.312	0.281	0.276	0.297	0.374
0.421	0.328	0.282	0.277	0.312	0.373
0.424	0.313	0.282	0.267	0.301	0.373
0.422	0.312	0.278	0.266	0.295	0.457
0.418	0.312	0.290	0.273	0.293	0.378
0.310				0.299	
Média	Média	Média	Média	Média	Média
0.421	0.315	0.283	0.272	0.300	0.391
D. Padrão					
0.002	0.007	0.004	0.005	0.007	0.037
2092	15088	24004	46076	310007	368078
0.241	0.603	0.333	0.271	0.264	0.280
0.242	0.607	0.274	0.271	0.265	0.289
0.240	0.604	0.340	0.274	0.264	0.272
0.241	0.617	0.322	0.271	0.264	0.280
0.242	0.597	0.259	0.271	0.267	0.281
0.248					0.289
0.243					0.278
Média	Média	Média	Média	Média	Média
0.242	0.606	0.305	0.272	0.265	0.281
D. Padrão					
0.003	0.007	0.036	0.001	0.001	0.006
22090	48055	118035	143090		
0.233	0.255	0.326	0.265		
0.231	0.179	0.333	0.319		
0.234	0.258	0.298	0.255		
0.226	0.254	0.366	0.237		
0.221	0.189	0.318	0.255		
0.233	0.256				
Média	Média	Média	Média		
0.230	0.232	0.328	0.266		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.005	0.037	0.025	0.031		

A.2.2 BGM

Tabela 17: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por CSC+GNM.

196073	253036	42049	60079	68077	80099
0.040	0.012	0.121	0.015	0.291	0.034
0.036	0.013	0.114	0.010	0.284	0.045
0.040	0.011	0.137	0.011	0.312	0.046
0.040	0.015	0.121	0.015	0.311	0.045
0.039	0.011	0.120	0.016	0.342	0.045
	0.011			0.313	
Média	Média	Média	Média	Média	Média
0.039	0.012	0.123	0.014	0.309	0.043
D. Padrão					
0.001	0.002	0.009	0.003	0.020	0.005
2092	15088	24004	46076	310007	368078
0.124	0.071	0.529	0.352	0.133	0.375
0.127	0.074	0.412	0.300	0.130	0.423
0.129	0.074	0.529	0.350	0.132	0.282
0.123	0.071	0.367	0.288	0.134	0.371
0.121	0.075	0.315	0.302	0.158	0.296
0.127					0.405
0.120					0.331
Média	Média	Média	Média	Média	Média
0.124	0.073	0.430	0.318	0.137	0.355
D. Padrão					
0.003	0.002	0.096	0.030	0.012	0.054
22090	48055	118035	143090		
0.239	0.125	0.236	0.207		
0.227	0.085	0.256	0.237		
0.266	0.123	0.095	0.197		
0.263	0.123	0.244	0.194		
0.211	0.079	0.180	0.195		
0.255	0.122				
Média	Média	Média	Média		
0.243	0.109	0.202	0.206		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.022	0.021	0.067	0.018		

Tabela 18: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por M&S+GNM.

196073	253036	42049	60079	68077	80099
0.038	0.013	0.101	0.044	0.152	0.047
0.035	0.014	0.113	0.041	0.142	0.056
0.038	0.012	0.098	0.042	0.170	0.056
0.038	0.016	0.103	0.045	0.175	0.056
0.037	0.012	0.099	0.044	0.208	0.057
	0.012			0.171	
Média	Média	Média	Média	Média	Média
0.037	0.013	0.103	0.043	0.170	0.054
D. Padrão					
0.001	0.002	0.006	0.002	0.023	0.004
2092	15088	24004	46076	310007	368078
0.128	0.106	0.558	0.427	0.173	0.398
0.131	0.109	0.460	0.403	0.172	0.449
0.131	0.109	0.558	0.426	0.178	0.328
0.130	0.108	0.395	0.387	0.179	0.392
0.127	0.111	0.344	0.401	0.201	0.334
0.130					0.425
0.125					0.366
Média	Média	Média	Média	Média	Média
0.129	0.109	0.463	0.409	0.181	0.385
D. Padrão					
0.002	0.002	0.096	0.017	0.012	0.045
22090	48055	118035	143090		
0.275	0.155	0.293	0.210		
0.275	0.097	0.325	0.229		
0.303	0.151	0.132	0.198		
0.298	0.151	0.372	0.193		
0.260	0.087	0.251	0.197		
0.293	0.151				
Média	Média	Média	Média		
0.284	0.132	0.275	0.206		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.017	0.031	0.091	0.015		

Tabela 19: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por Edison+GNM.

196073	253036	42049	60079	68077	80099
0.038	0.017	0.059	0.017	0.090	0.053
0.039	0.016	0.061	0.016	0.106	0.053
0.039	0.015	0.073	0.010	0.114	0.047
0.039	0.017	0.090	0.011	0.113	0.053
0.036	0.020	0.058	0.016	0.136	0.053
	0.016			0.102	
Média	Média	Média	Média	Média	Média
0.038	0.017	0.068	0.014	0.110	0.052
D. Padrão					
0.001	0.002	0.013	0.003	0.015	0.003
2092	15088	24004	46076	310007	368078
0.165	0.121	0.336	0.442	0.156	0.352
0.157	0.123	0.414	0.407	0.157	0.341
0.152	0.121	0.547	0.389	0.153	0.260
0.163	0.119	0.302	0.444	0.175	0.254
0.164	0.120	0.547	0.405	0.150	0.310
0.153					0.376
0.158					0.404
Média	Média	Média	Média	Média	Média
0.159	0.121	0.429	0.417	0.158	0.328
D. Padrão					
0.005	0.002	0.115	0.024	0.010	0.057
22090	48055	118035	143090		
0.252	0.118	0.332	0.269		
0.248	0.116	0.283	0.209		
0.269	0.117	0.192	0.239		
0.276	0.114	0.084	0.207		
0.268	0.113	0.239	0.212		
0.217	0.117				
Média	Média	Média	Média		
0.255	0.116	0.226	0.227		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.022	0.002	0.095	0.027		

Tabela 20: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por WS+GNM.

196073	253036	42049	60079	68077	80099
0.027	0.005	0.129	0.012	0.285	0.017
0.027	0.007	0.127	0.013	0.310	0.018
0.027	0.010	0.130	0.012	0.252	0.017
0.027	0.005	0.131	0.010	0.284	0.017
0.025	0.006	0.140	0.011	0.243	0.014
0.006				0.237	
Média	Média	Média	Média	Média	Média
0.027	0.006	0.132	0.012	0.268	0.017
D. Padrão					
0.001	0.002	0.005	0.001	0.029	0.001
2092	15088	24004	46076	310007	368078
0.103	0.437	0.498	0.194	0.342	0.362
0.106	0.440	0.387	0.195	0.340	0.408
0.109	0.439	0.500	0.194	0.345	0.275
0.103	0.442	0.287	0.196	0.346	0.364
0.102	0.436	0.286	0.195	0.354	0.310
0.106					0.388
0.100					0.346
Média	Média	Média	Média	Média	Média
0.104	0.439	0.391	0.195	0.345	0.350
D. Padrão					
0.003	0.002	0.106	0.001	0.005	0.046
22090	48055	118035	143090		
0.204	0.231	0.252	0.151		
0.206	0.231	0.281	0.218		
0.225	0.230	0.099	0.162		
0.233	0.234	0.326	0.158		
0.193	0.215	0.208	0.159		
0.216	0.231				
Média	Média	Média	Média		
0.213	0.229	0.233	0.170		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.015	0.007	0.087	0.027		

Tabela 21: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por *Color Structured Code*.

196073	253036	42049	60079	68077	80099
0.032	0.013	0.023	0.017	0.030	0.038
0.028	0.015	0.026	0.013	0.031	0.048
0.032	0.013	0.022	0.013	0.032	0.048
0.032	0.017	0.025	0.017	0.034	0.048
0.032	0.013	0.020	0.018	0.034	0.048
	0.012			0.032	
Média	Média	Média	Média	Média	Média
0.031	0.014	0.023	0.016	0.032	0.046
D. Padrão					
0.002	0.002	0.002	0.002	0.002	0.004
2092	15088	24004	46076	310007	368078
0.114	0.080	0.510	0.285	0.081	0.158
0.114	0.085	0.396	0.218	0.079	0.171
0.117	0.083	0.512	0.285	0.080	0.164
0.114	0.088	0.351	0.215	0.082	0.167
0.113	0.083	0.293	0.221	0.106	0.112
0.115					0.146
0.112					0.157
Média	Média	Média	Média	Média	Média
0.114	0.084	0.412	0.245	0.086	0.154
D. Padrão					
0.002	0.003	0.097	0.037	0.012	0.020
22090	48055	118035	143090		
0.245	0.204	0.228	0.193		
0.225	0.167	0.255	0.271		
0.261	0.207	0.075	0.167		
0.255	0.203	0.243	0.067		
0.212	0.173	0.160	0.167		
0.252	0.205				
Média	Média	Média	Média		
0.242	0.193	0.192	0.173		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.019	0.018	0.075	0.073		

Tabela 22: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por Mumford & Shah.

196073	253036	42049	60079	68077	80099
0.034	0.340	0.129	0.320	0.321	0.020
0.030	0.175	0.126	0.344	0.310	0.035
0.034	0.339	0.130	0.342	0.339	0.035
0.034	0.343	0.131	0.322	0.351	0.035
0.034	0.339	0.127	0.320	0.383	0.035
	0.339			0.339	
Média	Média	Média	Média	Média	Média
0.033	0.313	0.129	0.330	0.341	0.032
D. Padrão					
0.002	0.067	0.002	0.012	0.025	0.006
2092	15088	24004	46076	310007	368078
0.261	0.168	0.667	0.347	0.284	0.491
0.261	0.170	0.529	0.324	0.283	0.575
0.263	0.171	0.667	0.345	0.291	0.421
0.261	0.167	0.498	0.316	0.291	0.489
0.261	0.172	0.424	0.322	0.309	0.446
0.263					0.548
0.261					0.451
Média	Média	Média	Média	Média	Média
0.262	0.170	0.557	0.331	0.292	0.489
D. Padrão					
0.001	0.002	0.107	0.014	0.010	0.056
22090	48055	118035	143090		
0.148	0.350	0.220	0.496		
0.163	0.291	0.259	0.574		
0.164	0.348	0.060	0.465		
0.167	0.346	0.319	0.339		
0.093	0.289	0.169	0.450		
0.149	0.346				
Média	Média	Média	Média		
0.147	0.328	0.205	0.465		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.028	0.030	0.098	0.085		

Tabela 23: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por Edison.

196073	253036	42049	60079	68077	80099
0.042	0.011	0.021	0.017	0.051	0.008
0.042	0.009	0.023	0.016	0.056	0.008
0.042	0.008	0.025	0.011	0.064	0.006
0.042	0.010	0.021	0.011	0.062	0.008
0.041	0.013	0.019	0.016	0.086	0.009
	0.009			0.051	
Média	Média	Média	Média	Média	Média
0.042	0.010	0.022	0.014	0.062	0.008
D. Padrão					
0.001	0.002	0.002	0.003	0.013	0.001
2092	15088	24004	46076	310007	368078
0.104	0.111	0.318	0.069	0.030	0.327
0.102	0.118	0.420	0.075	0.031	0.322
0.101	0.114	0.536	0.069	0.031	0.238
0.103	0.112	0.309	0.072	0.067	0.252
0.104	0.113	0.534	0.075	0.029	0.284
0.103					0.375
0.105					0.401
Média	Média	Média	Média	Média	Média
0.103	0.114	0.424	0.072	0.038	0.314
D. Padrão					
0.001	0.003	0.111	0.003	0.016	0.061
22090	48055	118035	143090		
0.205	0.094	0.076	0.218		
0.190	0.079	0.079	0.139		
0.217	0.073	0.060	0.153		
0.207	0.076	0.056	0.137		
0.220	0.084	0.059	0.137		
0.188	0.074				
Média	Média	Média	Média		
0.204	0.080	0.066	0.157		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.013	0.008	0.010	0.035		

Tabela 24: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por *Watershed*.

196073	253036	42049	60079	68077	80099
0.062	0.014	0.138	0.034	0.613	0.066
0.062	0.017	0.134	0.035	0.628	0.067
0.062	0.019	0.139	0.034	0.614	0.066
0.062	0.014	0.142	0.032	0.612	0.066
0.059	0.015	0.145	0.033	0.598	0.061
	0.016			0.604	
Média	Média	Média	Média	Média	Média
0.062	0.016	0.140	0.034	0.611	0.065
D. Padrão					
0.001	0.002	0.004	0.001	0.010	0.002
2092	15088	24004	46076	310007	368078
0.168	0.456	0.507	0.222	0.437	0.390
0.176	0.459	0.399	0.224	0.436	0.415
0.172	0.458	0.509	0.221	0.441	0.306
0.174	0.461	0.306	0.226	0.441	0.387
0.164	0.455	0.306	0.223	0.444	0.347
0.178					
0.163					
Média	Média	Média	Média	Média	Média
0.171	0.458	0.405	0.223	0.440	0.369
D. Padrão					
0.006	0.002	0.101	0.002	0.003	0.043
22090	48055	118035	143090		
0.226	0.262	0.260	0.165		
0.235	0.262	0.288	0.230		
0.249	0.262	0.110	0.172		
0.258	0.265	0.333	0.169		
0.219	0.256	0.216	0.171		
0.240	0.262				
Média	Média	Média	Média		
0.238	0.262	0.241	0.181		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.015	0.003	0.085	0.027		

Tabela 25: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por JSEG.

196073	253036	42049	60079	68077	80099
0.011	0.118	0.196	0.259	0.459	0.031
0.011	0.315	0.198	0.259	0.504	0.031
0.011	0.315	0.176	0.255	0.513	0.018
0.011	0.315	0.195	0.256	0.514	0.031
0.008	0.317	0.196	0.259	0.554	0.032
	0.315			0.463	
Média	Média	Média	Média	Média	Média
0.010	0.282	0.192	0.258	0.501	0.028
D. Padrão					
0.001	0.081	0.009	0.002	0.036	0.006
2092	15088	24004	46076	310007	368078
0.056	0.375	0.466	0.284	0.333	0.452
0.056	0.377	0.571	0.245	0.334	0.467
0.055	0.374	0.654	0.232	0.332	0.369
0.055	0.372	0.428	0.286	0.368	0.408
0.057	0.374	0.653	0.247	0.332	0.422
0.055					0.523
0.058					0.544
Média	Média	Média	Média	Média	Média
0.056	0.375	0.555	0.259	0.340	0.455
D. Padrão					
0.001	0.002	0.105	0.025	0.016	0.062
22090	48055	118035	143090		
0.227	0.478	0.469	0.614		
0.194	0.301	0.219	0.514		
0.219	0.468	0.192	0.546		
0.225	0.469	0.105	0.382		
0.236	0.294	0.206	0.503		
0.180	0.470				
Média	Média	Média	Média		
0.214	0.413	0.238	0.512		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.022	0.090	0.136	0.085		

Tabela 26: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por RHSEG.

196073	253036	42049	60079	68077	80099
0.020	0.010	0.081	0.444	0.559	0.024
0.017	0.013	0.077	0.493	0.600	0.024
0.020	0.008	0.064	0.505	0.575	0.024
0.020	0.008	0.079	0.512	0.574	0.017
0.020	0.009	0.079	0.505	0.550	0.024
	0.008			0.612	
Média	Média	Média	Média	Média	Média
0.019	0.010	0.076	0.492	0.578	0.022
D. Padrão					
0.002	0.002	0.007	0.027	0.023	0.003
2092	15088	24004	46076	310007	368078
0.323	0.076	0.477	0.397	0.444	0.383
0.326	0.079	0.582	0.325	0.444	0.341
0.322	0.076	0.582	0.398	0.464	0.405
0.323	0.078	0.387	0.339	0.443	0.394
0.322	0.080	0.406	0.338	0.443	0.406
0.322					0.405
0.322					0.392
Média	Média	Média	Média	Média	Média
0.323	0.078	0.487	0.360	0.448	0.389
D. Padrão					
0.001	0.002	0.093	0.035	0.009	0.023
22090	48055	118035	143090		
0.250	0.659	0.236	0.492		
0.267	0.472	0.153	0.464		
0.271	0.659	0.249	0.445		
0.242	0.658	0.219	0.554		
0.276	0.479	0.068	0.278		
0.237	0.656				
Média	Média	Média	Média		
0.257	0.597	0.185	0.447		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.016	0.094	0.075	0.103		

Tabela 27: Valores da avaliação de BGM para todos os *ground-truths* comparados com os resultados produzidos por Blobworld.

196073	253036	42049	60079	68077	80099
0.319	0.345	0.338	0.321	0.357	0.283
0.317	0.344	0.340	0.321	0.390	0.283
0.320	0.346	0.339	0.298	0.370	0.283
0.319	0.345	0.335	0.290	0.356	0.284
0.317	0.345	0.375	0.310	0.348	0.284
	0.344			0.352	
Média	Média	Média	Média	Média	Média
0.319	0.345	0.345	0.308	0.362	0.283
D. Padrão					
0.001	0.001	0.017	0.014	0.016	0.001
2092	15088	24004	46076	310007	368078
0.320	0.634	0.461	0.401	0.370	0.439
0.326	0.635	0.397	0.399	0.370	0.469
0.318	0.634	0.466	0.407	0.371	0.399
0.324	0.635	0.457	0.396	0.371	0.450
0.322	0.636	0.384	0.396	0.400	0.428
0.328					0.459
0.321					0.427
Média	Média	Média	Média	Média	Média
0.323	0.635	0.433	0.400	0.376	0.439
D. Padrão					
0.003	0.001	0.039	0.004	0.013	0.023
22090	48055	118035	143090		
0.360	0.425	0.451	0.385		
0.355	0.302	0.455	0.463		
0.361	0.426	0.413	0.358		
0.344	0.427	0.461	0.325		
0.343	0.294	0.443	0.359		
0.364	0.426				
Média	Média	Média	Média		
0.354	0.383	0.444	0.378		
D. Padrão	D. Padrão	D. Padrão	D. Padrão		
0.009	0.066	0.019	0.052		

A.3 Artigo aceito pela publicação Pattern Recognition Letters

Color image segmentation guided by a color gradient network

Aldo v. Wangenheim ^{a,*}, Rafael F. Bertoldi ^a, Daniel D. Abdala ^b, Michael M. Richter ^c

^a Image Processing and Computer Graphics Lab – LAPIX, CS Department, Federal University of Santa Catarina – UFSC, Florianópolis, SC, Brazil

^b Computer Vision and Pattern Recognition Group, University Münster, Münster, Germany

^c CS Department, University of Calgary, Calgary, Alberta, Canada

Received 20 December 2006

Available online 26 May 2007

Communicated by H.H.S. Ip

Abstract

Existing region-growing segmentation algorithms are mainly based on a static similarity concept, where only homogeneity of pixels or textures within a region plays a role. Typical natural scenes, however, show strong continuous variations of color, presenting a different, dynamic order that is not captured by existing algorithms which will segment a sky with different intensities and hues of blues or an irregularly illuminated surface as a set of different regions. We present and validate empirically a new, extremely simple approach that shows very satisfying results when applied on such scenes, while not showing poorer performance than traditional methods when applied to standard region-growing problems.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Region-growing segmentation; Natural color scenes; Color gradient networks

1. Introduction

Color image segmentation algorithms present a very important application field of both computational image analysis and computer vision. Their main goal is to identify homogeneous regions or regions of similar characteristics within an image according to a predefined aspect, especially with respect to their color intensities. This family of algorithms becomes very important when we intend to achieve a result that one could compare closely to the human vision.

An essential problem of current algorithms is that similarity identification is calculated over simple continuous pixel neighborhood similarity. This approach leads to errors when we consider, for instance, the blue sky on a

cloudless day. The sky presents a darker blue intensity on the upper parts and gradually and continuously decreases into lighter blues on the lower parts. This intensity variation will be identified as numerous regions on most of the current algorithms, while the human eye sees only one region, the sky itself, recognizing the order in the variation. The concept of an image segmentation algorithm based on the analysis of the relationships of continuous gradients aims to overcome this difficulty. We call this approach the gradient network segmentation method and this paper presents an overview of the method and results so far.

1.1. Color gradient network-based segmentation

Traditional region-growing segmentation algorithms, in order to produce the best possible segmentation of an image, *i.e.*, one that best divides the image, rely on the search for islands of high entropy in the image, *i.e.*, homogeneous regions where no changes occur. The Mumford and Shah segmentation algorithm (Mumford and Shah, 1989) is the best example of this methodology. The more

* Corresponding author. Present address: LAPIX/LISHA, Departamento de Informática, CTC, UFSC, Campus Universitário, 88.049-900 Florianópolis, SC, Brazil. Tel.: +55 48 3331 9516; fax: +55 48 3331 9516 x16.

E-mail address: awangenh@inf.ufsc.br (A.v. Wangenheim).

common Watershed algorithm Vincent and Soille (1991) and its more elaborate implementations, such as (Mittelhoeuser and Kruggel, 1995), although they use image gradients to define region boundaries, are not much different. This leads to a situation where images that show a clear order, but where this order is represented by the change patterns of colors or light intensities, are poorly segmented, see Fig. 1.

On the other side, when we humans observe a scene with a blue sky, where the blue tones change gradually from the horizon on, the order is obvious and we do not even try to interpret the sky as a series of regions of different “blueness”. *We recognize the order in the dynamics of the colors* and identify the sky as one single structure in the image. This occurs also with similar images where we do not have background knowledge, such as colored curved shapes that have never been seen before, where an irregular illumination produces different shades of color and light intensity. But traditional segmentation algorithms are not able to perform this simple and foreknowledge-independent recognition operation. This is illustrated by the scene (b) in Fig. 1, segmented with the color space implementation of the Mumford and Shah algorithm found in the new release of the Megawave package (Megawave, 2006), where the result shows the lake over segmented. If we modify the parameters in such a radical way that the sky or the lake is segmented as one single region, this will produce segment leakages in other parts of the image, resulting in an unusable segmentation. If we, however, take the gradual change patterns in the image into account, we will be able to obtain a segmentation result as shown in Fig. 1c.

Several approaches have been developed in the last 15 years to enhance the performance of traditional region growing segmentation techniques on natural scenes, ranging from simple hierarchical approaches (Priese and Rehrmann, 1993; Rehrmann and Priese, 1998) to very elaborate ones (Dupuis and Vasseur, 2006). More recent methods focus not only on identifying very similar color regions, but also on additional features such as the texture of objects (Deng and Manjunath, 2001; Dupuis and Vasseur, 2006; Kato and Pong, 2006). But even with the use of techniques capable of identifying objects of similar color and even texturized objects, there is another characteristic that

troubles segmentation algorithms: luminosity and the existence of large but slow and gradual variations of the color of the same object in an image, resulting in high total gradients of low local steepness. Some approaches have tried to solve the problem using different evaluation methods for chromacity and luminosity (Dony and Wesolkowski, 1999; Schneider et al., 2000).

We propose a novel post-segmentation technique to process pre-segmented images through the analysis of gradient networks that describe color variation patterns between segments, a method especially robust when applied to large continuous color variations in the same object in an image, as occurs with typical outdoors images. The objective of this method is to perform reliable and accurate segmentations of objects that show strong continuous color variations, as are caused by varying light incidence on outdoors color images.

2. Methods

2.1. General principle

The goal of our approach is to describe the dynamics of color-changing and to use them in a segmentation algorithm, trying to *recognize a higher order of organization in the image*, namely the *organization of the changes* or the *structure of the tendencies* in the image. The general idea is to perform this task using a graph that represents the nature of modifications between regions of the image as a piecewise cost function. This graph is then processed through an algorithm that melts regions that pertain to the same image-dynamic process, *i.e.*, that lay within a tendency, such as a long upward or downward slope. The algorithm was developed to be able to represent the chromatic differences between regions as costs and to recognize order in the change patterns of the image, melting regions that pertain to the same “ordered structure” of the graph. The method we propose is a general framework, and some of the components cited below could be changed through other techniques.

Initial image representation: To represent each single pixel as a node in a graph with a set of parameters is a costly solution and may introduce local noise that does



Fig. 1. A well-organized image where the order is given by structured gradients rather than by abrupt changes and two different segmentations of the image: (a) original image, (b) Mumford and Shah segmentation of the image using (Megawave, 2006), (c) Gradient network segmentation of the same image.

not contribute to the regional estimation of the color variation tendency. Thus we use region-segmentation as a pre-processing step, which will represent each region that is statically, *i.e.*, value-wise homogeneous within a set of parameters as a single object. This set of parameters will be domain- and pre-segmentation method-dependent and is discussed below.

Each object generated by this pre-segmentation will be considered an *atomic region*. This segmentation should be performed with such parsimonious parameters that the resulting image is over segmented and no segment leakage occurs. The gradient network is then built upon this over segmented image, with each *atomic region* being represented as a node in the gradient graph. In our implementation we have chosen the Mumford and Shah functional (Mumford and Shah, 1989) for the pre-segmentation because of its robustness and quality of results, but one could use a faster method such as Split and Merge, Watershed (Vincent and Soille, 1991) or even the CSC (Priebe and Rehrmann, 1993), since one does not search for a final segmentation, but for a pre-classification of pixels in homogeneous structures. For our validation, we tried to devise a set of stable parameters for the Mumford and Shah algorithm that did not produce any segment leakages and to use them throughout the experiment.

Similarity measure: What is regarded as homogeneous will be defined by a threshold. In principle, the similarity measure compares two such regions considered currently as homogeneous, measuring how close the regions are with respect to the degree of homogeneity. These regions are represented in a graph and the role of a similarity measure is to provide a cost for each edge connecting adjacent regions. The details will be presented in Section 2.2.

As in the case of the initial segmentation, the similarity measure calculation is also a slot in the framework where many different similarity measures could be used. In the experiment described in this paper, we chose to differentiate regions of clear and rough color perception in the scene of the image, as originally discussed by (Huang et al., 2006), improving the robustness of the color similarity measure to specular lights and shadows. The idea was to enhance robustness in the presence of strong luminosity variations, something usual with real world images. Besides the similarity measures, the thresholds will play an important role. A more formal description of this algorithm will be given below.

2.2. Description of the gradient network method

- (1) A pre-segmented image is used as input data. It is produced with such conservative and oversensitive parameters as to avoid that a segment in this image leaks from one object into another in the image.
 - (i) Along with the input image, two thresholds are passed as parameters. These thresholds, t_{cp} and t_{rp} , will be used to verify if the result of a gradient of colors applied to the similarity function are

acceptably smooth according to the perception associated to the threshold used.

- (ii) All segments in this input image are labeled.
- (2) Regions and their neighborhood relationships are represented through a connected graph $G(V, E)$. The labeled regions from the pre-segmented input image will be the vertices V of the graph and the neighborhood relations among those objects will be represented as the edges E of the graph. V will be called the *Set of Atomic Regions*.
- (i) This graph will always be a connected graph because all regions represented by the vertices are contained in the same image, and through the edges that link neighbor regions it is possible to reach from any given vertex any of the other vertices as it is possible to find a path from any pixel p_1 to any other pixel p_2 in a same image.
- (ii) Then, given the graph $G(V, E)$ of connected regions, it can be defined as follows:

$$V = \{v \in V | \forall((x_1, y_1) \subset v) \wedge (\neg\exists((x_2, y_2) \subset v) \wedge \mu(x_1, y_1) \neq \mu(x_2, y_2))\}$$

where every vertex $v \in V$ represents an *Atomic Region* and contains a set of ordered pairs of coordinates (x, y) in the orthogonal axis X and Y . For all these contained ordered pairs of coordinates (x, y) , the positions they all represent in the labeled input image will all have a same and unique value, where μ is the function that gives the label value for a given pair of coordinates (x, y) in the labeled image.

$$E = \{e \in E | (v_1, v_2 \in V) \wedge (v_1 \neq v_2) \wedge \exists(((x_1, y_1) \subset v_1) \wedge ((x_2, y_2) \subset v_2) \wedge (N_8(x_1, y_1) \supset (x_2, y_2)))\}$$

where every edge $e \in E$ is formed by a pair of vertices $v_1, v_2 \in V$ and between these two vertices there is at least pair of positions $(x_1, y_1) \subset v_1$ and $(x_2, y_2) \subset v_2$ that show an 8-connected kind of neighborhood.

- (3) In the next stage of the algorithm, each vertex $v \in V$ is associated with a new and unique *Meta-region* $m \in M$. Meta-regions are logical containers that will be used to store groups of regions that have a path among them and that are linked through color gradients that are considered acceptable according to the similarity measure proposed. The goal of the gradient network method is to merge meta-regions.

- (i) All meta-regions will behave like connected sub graphs, meaning that any vertex contained by a given meta-region has at least a path to any other vertex also contained in the same meta-region.

$$M = \{m \in M | v_1, v_2 \in V \wedge v_1, v_2 \subset m \wedge f_c(v_1, v_2) \leq T_c\}$$

where every meta-region $m \in M$ is formed by a set of vertices $v \in V$ that have an acceptably smooth gradient between pairs of vertices $v_1, v_2 \in V$ according to the T_c threshold.

- (ii) Any vertex v_i can pertain to only one Meta-region.
- (4) The algorithm then runs through the graph $G(V, E)$ looking in the edges $e \in E$ that connect vertices $v_1, v_2 \in V$ for gradients smooth enough to be considered acceptable according to the perception for this edge, resulting in the union of those vertices in the same meta-region $m \in M$.
 - (i) The goal is verifying if these two regions can be placed in the same meta-region m , meaning that both are considered similar by the measure. If the vertices already are contained by the same meta-region $m \in M$ there is no reason to keep verifying and therefore the process goes on, proceeding to another unverified edge, if there is still one. The edge is marked as *verified*. Each edge is examined only once.
 - (ii) If both vertices are not in the same meta-region $m \in M$, then the kind of perception to apply to this link will be determined.

The perception idea was developed based on concepts found in (Huang et al., 2006), where a difference of color perception is used to enhance the contrast evaluation in images. With these perceptions and the features they provide, it is easier to make the method robust to change of lighting and presence of shadows in the scene. Also it is an interesting concept because the question of smooth variation of colors in objects is directly connected to illumination and this type of evaluation through perceptions offers a powerful tool for dealing with this type of characteristic.

In our algorithm we use two kinds of perceptions:

- (1) *Clear perception*, where there is good color saturation and average levels of luminosity.
- (2) *Rough perception*, where there is low color saturation, or very high or very low intensities. The color values of the vertices are converted to the HSL color model, composed by three orthogonal axes that represent, respectively, hue, saturation and luminance. Currently the acceptance levels for both perceptions are defined parametrically as ranges of acceptance for the hue, saturation and luminance values, accordingly to (Huang et al., 2006). These ranges must partition the range that HSL values can assume in a way that a region is either classified as having a clear or rough perception. These parameters will allow a better configuration of the segmentation for the lighting conditions of the scene of the image.

The equations need two vertices as input arguments. These will always be neighboring regions, as a consequence from the fact that the input vertices are extracted from an edge of the graph and inherits this property from this edge. Given both vertices and the kind of perception their gradient

fit, it is checked if the value found with the evaluation function f_c is acceptable according the threshold T_c defined parametrically for the perception used. The function definition is

$$f_c(v_1, v_2) = \begin{cases} \alpha_{cp} * \min \left(\left| \int_{H(v_1)}^{H(v_2)} d\theta \right|, 1.0 - \left| \int_{H(v_1)}^{H(v_2)} d\theta \right| \right) \\ + \beta_{cp} * |S(v_1) - S(v_2)| + \gamma_{cp} * |L(v_1) - L(v_2)|, & \text{perception}_{\text{clear}} \\ \beta_{rp} * |S(v_1) - S(v_2)| + \gamma_{rp} * |L(v_1) - L(v_2)|, & \text{perception}_{\text{rough}} \end{cases}$$

where f_c is the smooth gradients evaluation function and it takes two vertices $v_1, v_2 \in V$ as parameters. The functions H , S , L take a vertex $v \in V$ as parameter and give the mean value of, respectively, hue, saturation and luminance from the color values of the pixel positions in the image that the vertex v contains. The coefficients α , β , γ correspond to the desired relevance given by the perception to hue, saturation and luminance, respectively. These coefficients are also parameters of the algorithm.

- (iii) In case of success, meaning that the result is smaller than the threshold T_c , a merging is performed between the meta-regions $m_1, m_2 \in M$ associated with these vertices $v_1, v_2 \in V$, because the gradient between these regions was considered so smooth that these regions actually identify a single object in the scene of the image. No other changes are performed on the vertices or the edge.

$$\begin{aligned} f_c(v_1, v_2) < T_c \rightarrow m_1, m_2 \in M | v_1 \subset m_1 \wedge v_2 \\ &\subset m_2, m_1 \cup m_2 \end{aligned}$$

If it was not lower than the threshold, then nothing is done and the algorithm continues. Step 4 will be performed once for all the edges present in the connected graph. The resulting meta-regions will represent the objects of the scene, formed by regions of low entropy, found by the initial segmentation that created the input image, that are connected through smooth color gradients.

- (5) Finally, the output image will be built based on the meta-regions $m \in M$, where now the color of each resulting Meta-region will correspond to the average color of all vertices $v \in V$ contained in the Meta-region.

3. Results

The gradient network method was applied on reference images provided by (Martin et al., 2001) and a few others collected by the authors. Processing results are shown in Fig. 2. The results were also compared to well-known state-of-the-art region-growing segmentation algorithms,

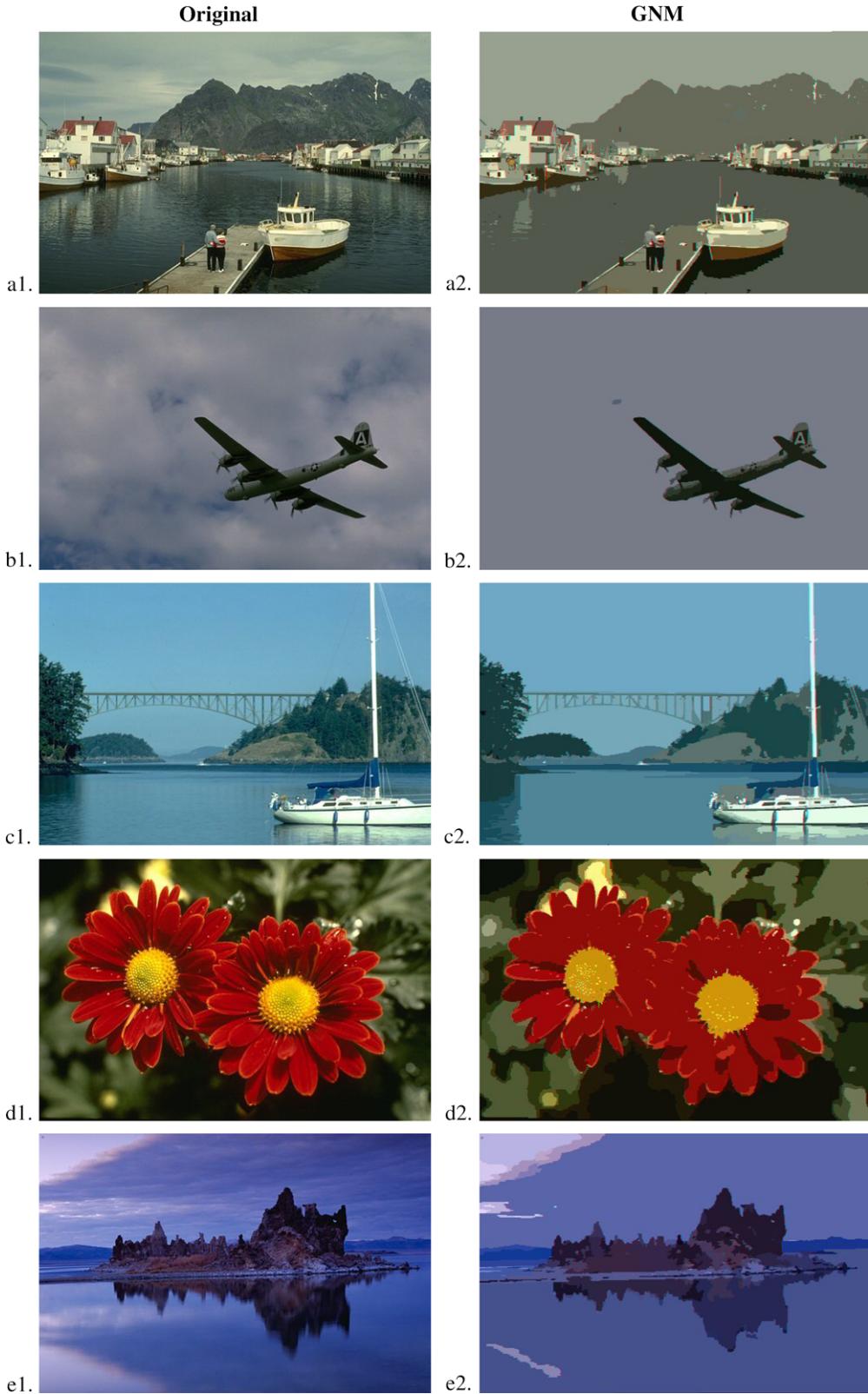


Fig. 2. Results obtained with the gradient network segmentation method on the (Martin et al., 2001) segmentation test images. All tests underwent a Mumford–Shah pre-segmentation with $\lambda = 600$.

namely CSC (Rehrmann and Pries, 1998), Mumford–Shah (Mumford and Shah, 1989), RHSEG (Tilton et al.,

2006) and JSEG (Deng and Manjunath, 2001), which were applied to the same images.

The methods used in the comparison shown in Fig. 3 were applied using the following parameters:

- All CSC images were produced with a threshold equal to 50 or 70.
- Each Mumford–Shah was generated with a different and specific lambda value. These lambda values are shown below the images.
- JSEG was used in an unsupervised way with all images.
- All RHSEG images were created with the similarity “entropy” (number 9 in the *params* file – segmentation parameters), with a factor of convergence equal to 1.75 and with a 0.1 importance to spectral clustering.

As stated in the algorithm description, the perception classification parameters must be defined manually. The parameters used for this work were the same for all images, using similar ranges to those found in (Huang et al., 2006). With all HSL values normalized to the [0, 1] range:

$$\begin{aligned} \text{Perception}_{\text{clear}} &= \{(S > 0.1) \wedge (0.2 < L < 0.95)\} \\ \text{Perception}_{\text{rough}} &= \{(S \leq 0.1) \vee (L \leq 0.2) \vee (L \geq 0.95)\} \end{aligned}$$

Depending on the classified perception, the coefficients α , β , γ have the following values, also normalized to the [0, 1] range, for all the tests presented:

$$\begin{aligned} \alpha_{\text{cp}} &= 0.7, \quad \beta_{\text{cp}} = 0.2, \quad \gamma_{\text{cp}} = 0.1 \\ \beta_{\text{rp}} &= 0.2, \quad \gamma_{\text{rp}} = 0.8 \end{aligned}$$

A larger set of processing results and comparisons is available at <http://www.lapix.ufsc.br/gnm/>. All tests shown here can be downloaded from <http://www.lapix.ufsc.br/gnm/Images.rar>.

The results depicted in Figs. 2 and 3 show that this simple algorithm is capable of organizing a pre-segmented image in meta-regions accordingly to the color variation patterns that a human observer would also see, producing coherent segmentations of objects in images, even when there is considerable variation within a meta-region, given that this variation is relatively smooth and continuous.

4. Discussion

Simplicity was one of the goals kept in mind when this method was conceived. Similar approaches have been used previously by (Tréneau and Colantoni, 2000) and (Wu and

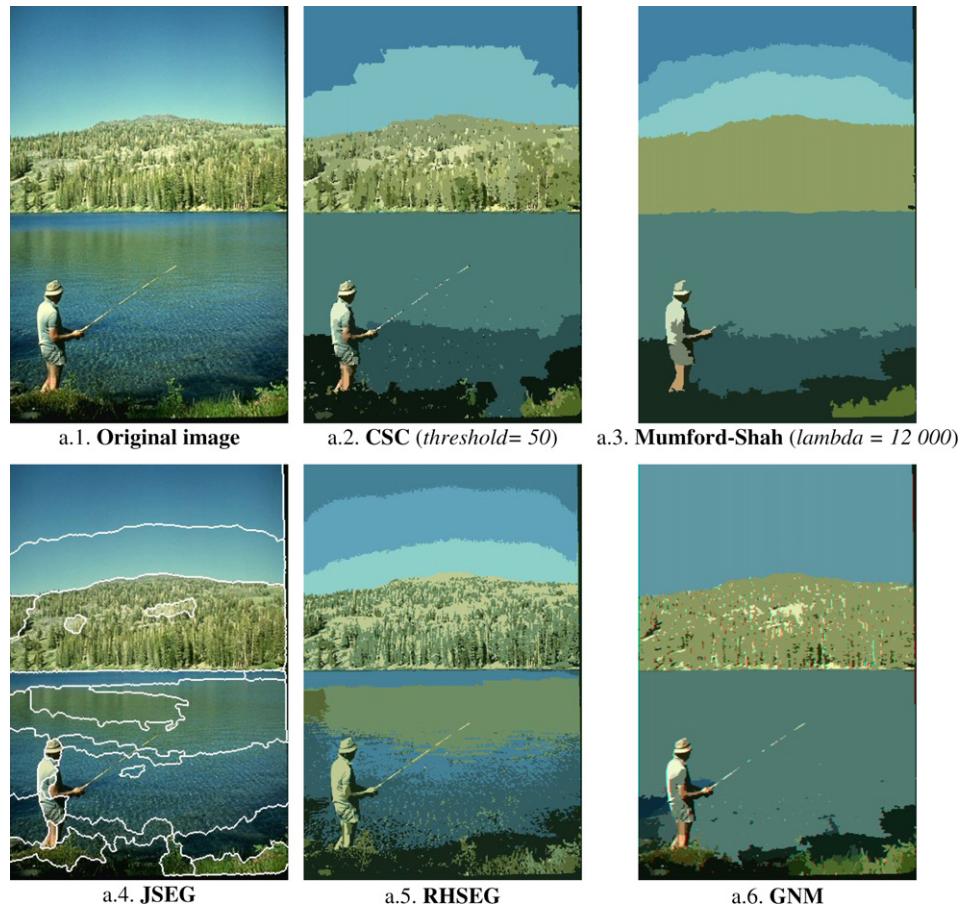


Fig. 3. Comparison of the GNM with other segmentation methods. (a) and (c) are images from the (Martin et al., 2001) segmentation test database. (b) is an image taken by the authors.

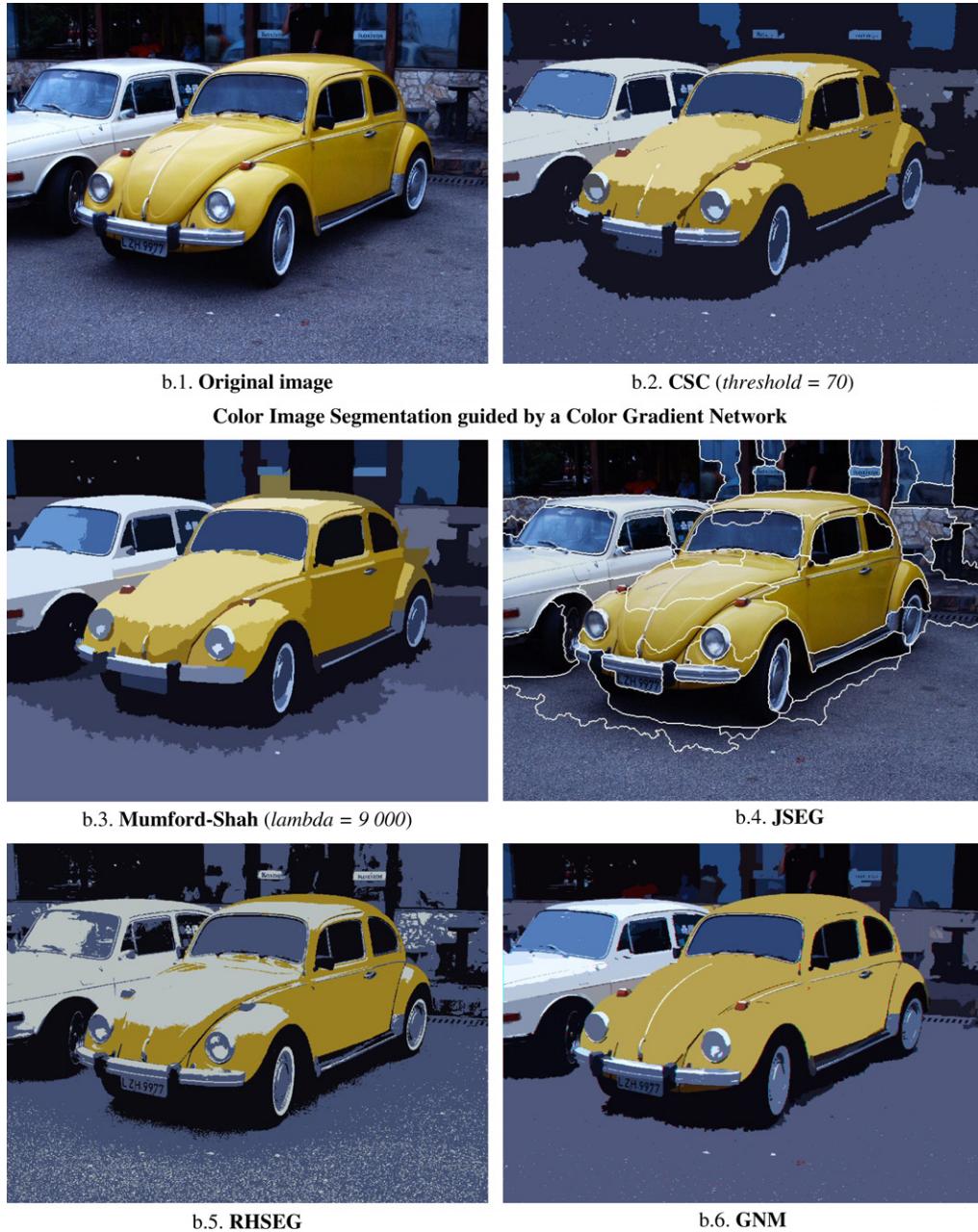


Fig. 3 (continued)

Leahy, 1993), where the problem was stated as a graph-flow optimization in sub graphs. Another graph-partitioning approach introduced a global measure obtained by a principal component analysis of a set of different parameters associated with each element of the image (Dupuis and Vasquez, 2006). Our approach also finds sub graphs in the original network but is much simpler and acts only locally on each edge, without trying to find an optimal value for the entire sub graph that represents a region and without using any global image data or cost function. It applies only simple non-adaptative local similarity measures and produces quality segmentation results on difficult images, comparable to the most complex approaches. The method is based on a simple assumption: *local variation*

patterns of the characteristics of the image are as important as the static characteristics. To discover order in these patterns is important for a segmentation algorithm. The process of discovering the variation patterns of the image is described as a simple local similarity-based process in one simple and consistent modular framework, with an elegant and integrated definition of stopping criteria.

Our approach can also be seen as a labeling procedure. The goal of labeling techniques is to label uniquely all the objects in an image. Usually it is a process applied after the use of some other image preprocessing method, often a segmentation. However, the result of a region-growing segmentation behaves like a labeling method, assigning pixels to regions and associating mean color values with



Fig. 3 (continued)

them. From this point of view, the pre-processing our approach relies on is a pre-segmentation with a method that points out the obvious region patches where a gradient-pattern analysis is not necessary, thus avoiding unnecessary processing.

The main difference between this method and other, image post-processing labeling methods developed to “correct” over segmentations is that it is not knowledge-based or application-domain-model-based, *i.e.*, it does not need additional knowledge on the content or the application domain of the images, depending only on image information.

References

- Deng, Y., Manjunath, B.S., 2001. Unsupervised segmentation of color-texture regions in images and video. *IEEE Trans. Pattern Anal. Machine Intell.* 23 (8), 800–810.
- Dony, R.D., Wesolkowski, S.B. 1999. Edge detection on color images using RGB vector angles. In: Proc. 1999 IEEE Canadian Conf. Electrical and Computer Engineering. Edmonton, Alberta, Canada May 9–12, pp. 687–692.
- Dupuis, A., Vasseur, P., 2006. Image segmentation by cue selection and integration. *Image Vision Comput.* 24 (10), 1053–1064, October.
- Huang, K., Wang, Q., Wu, Z., 2006. Natural color image enhancement and evaluation algorithm based on human visual system. *Computer Vision Image Understand.* 103 (1), 52–63, July.
- Kato, Z., Pong, T.C., 2006. A Markov random field image segmentation model for color textured images. *Image Vision Comput.* 24 (10), 1103–1114, October.
- Martin, D., Fowlkes, C., Tal, D., Malik, J. 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. 8th Internat. Conf. on Computer Vision, vol. 2. pp. 416–423.
- Megawave Homepage: <<http://www.cmla.ens-cachan.fr/Cmla/Megawave/>>, 2006 (accessed 14.9.06).
- Mittelheusser, Gangolf, Kruggel, Frithjof, 1995. Fast Segmentation of Brain Magnetic Resonance Tomograms. In: CVRMed'95: First International Conference on Computer Vision, Virtual reality and Robotics in Medicine, INRIA, Nizza. Springer Verlag. April.
- Mumford, D., Shah, J., 1989. Optimal approximations by piecewise smooth functions and associated variational problems. *Commun. Pure Appl. Math.* 42, 577–684.

- Priese, L., Rehrmann, V., 1993. A fast hybrid color segmentation method. DAGM-Symposium, 297–304.
- Rehrmann, V., Priese, L., 1998. Fast and robust segmentation of natural color scenes. ACCV 1, 598–606.
- Schneider, M.K., Fieguth, P.W., Karl, W.C., Willsky, A.S., 2000. Multiscale methods for the segmentation and reconstruction of signals and images. IEEE Trans. Image Process. 9 (3), 456–468, March.
- Tilton, J.C., D-dimensional formulation and implementation of recursive hierarchical segmentation, Disclosure of Invention and New Technology: NASA Case No. GSC 15199-1, May 26, 2006.
- Trémeau, A., Colantoni, P., 2000. Regions adjacency graph applied to color image segmentation. IEEE Trans. Image Process. 9 (4), 735–744.
- Vincent, L., Soille, P., 1991. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. IEEE Trans. Pattern Anal. Machine Intell. 13, 583–598.
- Wu, Z., Leahy, R., 1993. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. IEEE Trans. Pattern Anal. Machine Intell. 15 (11), 1101–1113, Nov.

A.4 Artigo submetido para a publicação Pattern Recognition

Improving Color Image Segmentation using the Gradient Network Method

R. Bertoldi¹, D. D. Abdala², A. v. Wangenheim¹, X. Jiang², M. M. Richter³, L. Priese⁴, F. Schmitt⁴

¹Image Proc. and Computer Graphics Lab – LAPIX,
Federal University of Santa Catarina, Florianópolis,
Brazil {fogo,
awangenh}@inf.ufsc.br

²CS Department, University
of Münster, Münster,
Germany
{abdalah,xjiang}@math.uni
-muenster.de

³CS Department.
University of Calgary,
Alberta, Canada
mrichter@cpsc.ucalgary.ca

⁴CS Department,
University of Koblenz,
Koblenz, Germany
{priese,fschmitt@uni-koblenz.de}

Abstract

The objective of this paper is to evaluate a new combined approach intended for reliable color image segmentation, in particular images presenting color structures with strong but continuous color or luminosity changes, such as commonly found in outdoors scenes. The approach combines the Gradient Network Method with common region-growing approaches used as pre-segmentation steps. The GNM is an efficient post-segmentation procedure based on graph analysis of global color and luminosity gradients altogether with a segmentation algorithm to produce a fast and reliable segmentation result. The approach was automatically evaluated using a close/open world approach. Different region-growing segmentation methods, namely CSC, Watershed, EDISON and Mumford & Shah with and without the GNM post-processing were compared against ground truth images using Rand and Bipartite Graph Matching indexes. These results were also confronted with other well established segmentation methods (RHSEG, JSEG and Blobworld). We further conclude that the CSC-GNM approach shows a good compromise in dealing with outdoors images.

Keywords: color image segmentation, region-growing, outdoor scenes, Gradient Network Method

1 Introduction

Natural color scenes, such as outdoors images composed by many colored objects that are acquired under uncontrolled conditions can show complex illumination patterns across one same object in the picture. Examples are variations in lightness and specular effects. State-of-the-art region-growing segmentation methods [1, 2, 3] present two main features that limit their applicability for dealing efficiently with natural scenes:

- (a) *A static region similarity concept*, where pixels or textures within a region are expected to be homogeneous. Typical natural scenes, however, show strong continuous variations of color, presenting a different, dynamic order that is not taken into account by such algorithms. They divide a sky region with different intensities of blue into segments or will represent an irregularly illuminated surface as a set of different regions. When the parameters of such algorithms are stressed in order to try to accomplish a correct segmentation of a large object showing a long continuous gradient of color, typically with a gradual but large color variation, a region leakage into other objects in the image is likely to occur. Then the algorithm will become unstable and even inapplicable;

- (b) *Increase in complexity to present more stable results*, that usually demands complex computations to detect segment-correlation clues, or that are built upon additional texture information. This slows down considerably the processing time without being much more stable when extreme color variations are present.

A number of segmentation approaches already have tried to deal with one or both problems: [1] proposes a hierarchical model to improve the segmentation of color scenes, and [2, 4, 5] address the problem by the additional analysis of object textures. Some authors have tried to overcome the problem using different methods of chromatic and luminescence evaluation [6, 7]. Finally, in [8, 9, 10] physical illumination models were proposed as basis for the formulation of segmentation algorithms.

The Gradient Network Method - GNM [11], intends to address the problem of luminance and reflection variations by searching for structured gradients along the various neighboring segments. It was devised to work together with a fast segmentation algorithm, which provides a pre-segmentation as a starting point. The general approach is to rely on fast segmentation algorithms such as CSC [12] or any other producing as output an over-segmentation with edge preservation and subsequently apply GNM, who will iteratively merge segments logically connected through organized gradient patterns improving the final segmentation.

2 Objectives

In this paper we are interested in evaluating the criteria to specify an optimal pre-segmentation algorithm to work with the GNM post-segmentation procedure and to compare the achieved results with other approaches. For this purpose we devised the following procedure, which included an evaluation using a close/open world approach:

- (a) First, different first step well-known segmentation methods to be used as pre-segmentation procedures were chosen: CSC [1], EDISON [13], Watershed [14] and Mumford & Shah [3].
- (b) Full segmentations performed with each of these algorithms were compared against ground-truth images using Rand [15] and Bipartite Graph Matching (BGM) [16] indexes. For each segmentation algorithm we selected a wide range of segmentation parameters, selected the segmentation considered to be the best one for every pair of image set and segmentation algorithm using visual inspection and generated Rand and BGM scores for the complete set of ground-truths for each image
- (c) These same segmentation algorithms were also selected to generate over-segmented images to be used in combination with the GNM post-processing algorithm. For each algorithm we selected a set of segmentation parameters that produced oversegmented images where no segment leakage with respect to any ground truth was allowed. Each of these results was used as an input for the GNM algorithm, which was also run with a set of different parameters. The resulting segmentations after post-processing with the GNM also underwent a selection of the segmentation considered to be the best one based on visual inspection. This result was also compared against the ground-truth images using Rand and BGM indexes.

(d) We compared these results to three other well-established segmentation methods: RHSEG [17], JSEG [2] and Blobworld [18] also using the ground-truth images and the Rand and BGM indexes.

The structure of this paper is as follows: In section 3 we review briefly the GNM main concepts. Section 4 describes the experiment and further discusses the objective evaluation of segmentation results compared against ground-truths. Section 5 reviews the evaluation indexes used, discussing the main characteristics and drawbacks of each of them. Finally, section 6 shows and discusses the results of our close/open world evaluation. All data and results are accessible in detail under <http://www.lapix.ufsc.br/gnm>.

3 The Gradient Network Method

The Gradient Network Method previously described in [11] and closely related to [19] was developed to deal with segmentation problems where objects in the scene will be represented by several different but similar and gradually varying color shades, as they often are found in outdoors scenes. The GNM looks for a higher degree of organization in the structure of the scene through search and identification of continuous and smooth color gradients.

For the GNM to be able to process an image and identify these characteristics, a pre-segmentation of the image must be performed. The goal is to obtain groups of segments with a high degree of similarity represented in a simple way, avoiding possible problems with local noise induced by high granularity (e.g. at pixel level). The clusters in eq. I are sets of pairs of coordinates in the Ω space of R^n values (in the case of our tests, inside the R^2 image spaces) and color ranges in the R^m , similar according to the segmentation approach used to realized the rough verification of the image and with dimensionality m determined by the color space selected (GNM currently uses HSI color space, so R^3 is its color range dimensionality).

$$C = \{(x_1, r_1), \dots, (x_i, r_j) | x_1, \dots, x_i \in \Omega \wedge \Omega \subseteq R^n \wedge r_1, \dots, r_j \in R^m\} \quad \text{I}$$

Every cluster $C = \{c_1, \dots, c_n\}$ is a discrete clustering meaning that each pixel is unequivocally assigned to a specific cluster, such that $X_k \cap X_l = \emptyset$ and $\bigcup_{i=1}^n X_i = \Omega$ for the every $X_k = \{x_1, \dots, x_p \subset C_1\}$ and $X_l = \{x_1, \dots, x_q \subset C_2\}$, $k \neq l$. Every cluster will also receive an unique label in a way that for any $L(c_k)$ and $L(c_l)$, $k \neq l$, are always different.

Given the clusters produced by the first segmentation step, GNM creates a graph $G(V, E)$ to structure this data, providing a simple way to describe the topology of these clusters of similarity. The graph is constructed from these clusters where vertices $V = \{v_1, \dots, v_n\}$ represent the roughly similar groups of pixels (see equation eq. II) and edges $E = \{e_1, \dots, e_m\}$ represent the existence of a neighborhood relationship between the given vertices. Each edge also is labeled by the existing gradient between the two vertices he connects, as described in equation eq. III.

$$V = \{v_i \in V | v_i = c_j \wedge c_j \in C\} \quad \text{II}$$

$$E = \{e_i \in E | e_i = (v_k, v_m, \nabla) \wedge v_k, v_m \in V \wedge \nabla \in R^+\} \quad \text{III}$$

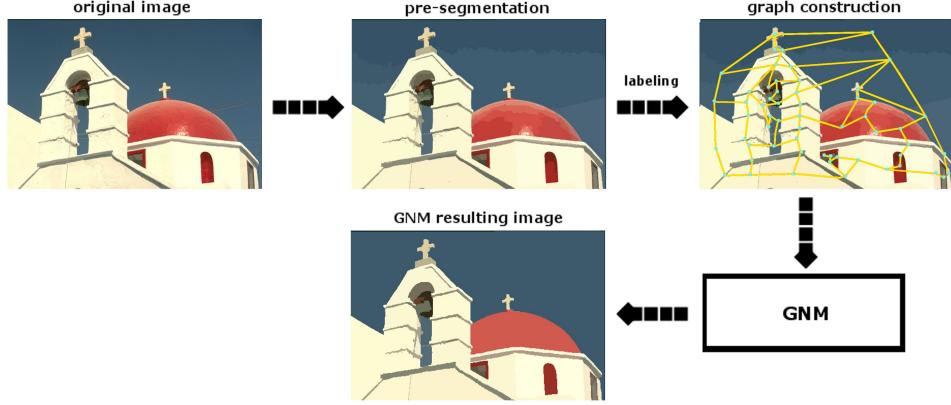


Figure 1. The GNM workflow, showing the steps to partition an image into similar objects using the GNM algorithm.

The objective of the algorithm is to minimize $\|G\|$ (vertices cardinality) by a merging procedure

that searches subgraphs $S_i \subseteq G$ which presents $\min(\sum_k^N \nabla_k) < \Phi$ where N is the edges' cardinality. GNM concentrates on regions of high similarity, specifically in the aspect of low color variation. So, after creating the graph $G(V, E)$, the next step is checking all the neighborhood relations if they comply to the similarity measure and provide continuous and smooth color gradients.

The evaluation of the continuity of the gradients along the paths found in the graph is performed by a function f_p that takes into account the perception variations, a concept similar to that found in [20]. This allows a better evaluation of the similarity among regions in different conditions of illumination. This is important because most long continuous gradients are the result of the presence of lighting effects in the scene of an image. With this additional feature, the algorithm becomes more robust in presence of such characteristics. Therefore, even when the neighborhood contains darker or more illuminated regions, it will search for the best possible gradient path in the graph [11]. An improvement of the implementation used in our tests over the algorithm described in [11] is that now all $e \in E$ will be ordered crescently by the resulting values from the perception function and then evaluated by the chosen similarity measure. Regions found acceptably similar will be grouped in meta-regions.

To generate the resulting subgraph, every vertex in the graph $G(V, E)$ is correlated to a meta-region. Meta-regions M are logical containers that will be used to represent the union among different vertices (eq. IV), as they are found necessary along the evaluation of the edges using the f_p function. Initially, each meta-region $m \in M$ will have a one-to-one relationship with a vertex $v \in V$.

$$M = \{m_i \in M \mid \bigcup_{v_j \in V} (v_j \in m_i) \wedge \forall j \neq k, v_j \neq v_k\} \quad \text{IV}$$

The resulting meta-regions at the end of the process will be the partitioned image output produced by the GNM segmentation. The total complexity for all these steps is $O(n(1+n) + m(1+\log m))$, where n is the number of vertices and m the number of edges.

Another feature was added to the GNM implementation used here, improving its segmentation results: After the segmentation is performed, a post-processing step is executed to remove

regions that are considered insignificant to be represented as partitions of the image. The process of running through this set eliminating unnecessary regions has a complexity of $O(k^2)$, where k is the number of meta-regions that should be checked by this step (the set can be built along the normal process of the algorithm). Every meta-region in this set is iterated, looking if this meta-region should be eliminated. The criterion for this elimination is if the area of a region is considered to be too small. This region is merged with the neighbor meta-region which is less affected by the union with it, considering the meta-region average color and area.

It is important to note, though it cannot be accounted in the GNM complexity, that the chosen algorithm for the pre-segmentation has an effect on the total time of processing. It can also affect the image segmentation results, depending on how properly the pre-segmentation performs the role that GNM expects from it. Therefore, a proper technique must be selected according to the application necessity, in a way that can ally quality and velocity performance as desired.

4 Experiment

In order to be able to objectively achieve the task we have proposed ourselves of choosing the optimal pre-segmentation algorithm to be used in conjunction with the GNM approach and then, to chose the optimal segmentation parameters for each algorithm, we accepted that ground-truths, or hand-made segmentations, representing the judgment of an human observer, should play the role of golden standards. The process of evaluation of image segmentation results was investigated by a number of researchers [21, 22, 23].

We adopted the Berkeley ground-truth image dataset [24]. This dataset is well known in the image processing community and each sample image contains various ground-truths generated by different subjects. The evaluation schema adopted in this paper works over a subset of 16 images selected from this dataset which present strong gradients in the form of continuous color variations in the same object.

In order to objectively evaluate the quality results of GNM we have envisioned an evaluation procedure. The first step was the generation of a series of segmentation results for CSC, Mumford & Shah (MS), Watershed (WS), RHSEG, JSEG, Blobworld and EDISON algorithms added of the tests with GNM combined with pre-segmentations generated with CSC, Mumford & Shah, EDISON and Watershed. Our purpose is to compare the GNM results with those obtained by several state-of-art algorithms, looking to validate our algorithm as a technique as capable as those selected for our tests.

Regarding the tests performed with the GNM, all pre-segmentations are obtained from a set of different parameter ranges. Though the parameters are specific to each pre-segmentation, the goal is only one: achieve an over-segmented image avoiding leakages and preserving smooth gradients to suit and allow a better segmentation by GNM. The parameter used for CSC was $threshold = 30$; Mumford & Shah images were generated with $lambda = 600$; EDISON was generated with a $SS = 3$ and $SR = 8$; Watershed images used $conductance = 2.0$, $number\ of\ interactions = 10$ and $threshold = 0.01$. Over each of the pre-segmented images produced by these algorithms was performed a series of GNM segmentations. Two parameters only were varied along all these tests, t_{cp} and t_{rp} , both in the $[0.01, 0.07]$ range, with an offset of 0.005. All other GNM parameters were kept with the standard values described in [11].

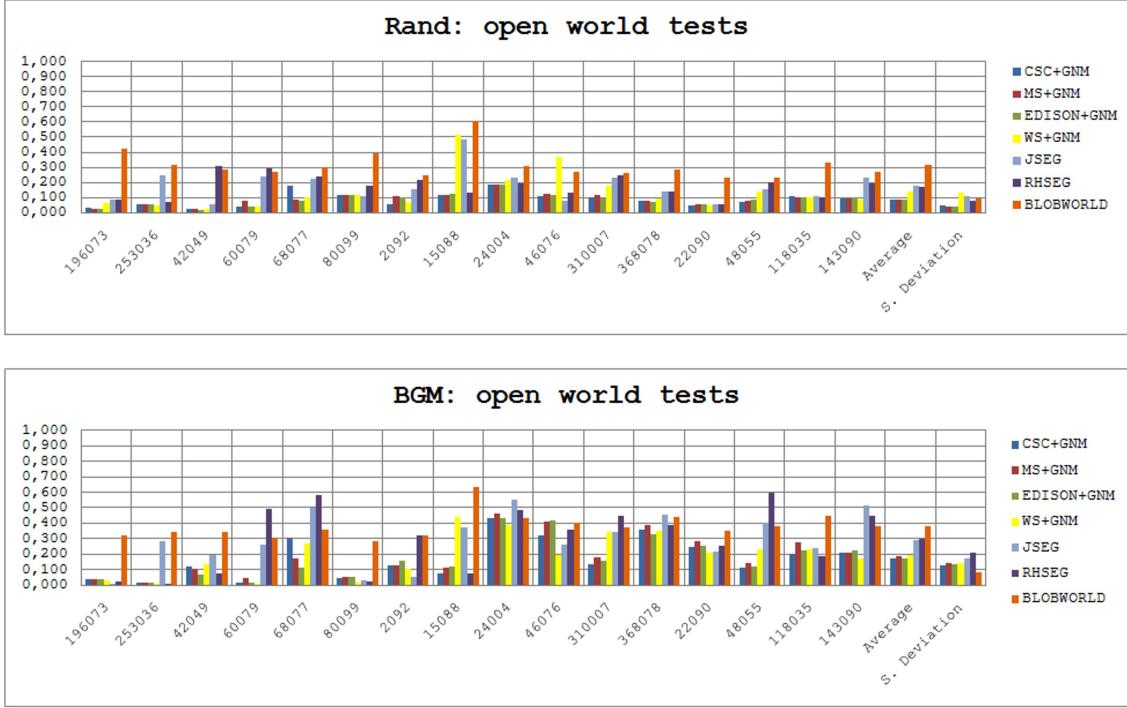


Figure 2. Graphs showing the average values obtained for all selected image sets with every tested algorithm. The two columns most to the right represent, respectively, average and standard deviation values for the algorithms.

A series of segmentation results was also produced using those segmentation techniques alone and used to be evaluated against the combined GNM approach. The parameter ranges and increment steps used for these segmentation methods were the following: 1) CSC: $20 \leq threshold \leq 100$, $step_{threshold} = 10$; 2) Mumford & Shah: $1000 \leq lambda \leq 15000$, $step_{lambda} = 500$; 3) EDISON: $3 \leq SS \leq 30$, $step_{SS} = 1$, $SR = 8$; 4) Watershed: $0 \leq threshold \leq 0.5$, $step_{threshold} = 0.05$. Blobworld and JSEG are unsupervised techniques and do not require parameters. RHSEG, however, is supervised but, as it is intrinsically a hierarchical segmentation, was not necessary to be tested over a range of values. All RHSEG segmentation results were created with the similarity “entropy” (segmentation parameter number 9 in the params file), with a factor of convergence equal to 1.75 and with a 0.1 importance to spectral clustering.

After all the image segmentations were produced by these techniques with the parameters listed, we adopted visual inspection to select the segmentation considered the best one for every pair of image set and segmentation algorithm. The selected results were those that seemed to better partition the images into relevant regions. This human-based selection is supposed to be a rough filtering of the segmentations and the group of image segmentations selected will be next evaluated by distance measures able to quantify the quality of image segmentations. The explanation about how this evaluation and which measures were used follows in the next chapter.

5 Evaluation

To better outline the quality of the results obtained with the different pre-segmentation + GNM combinations, the image segmentations were objectively evaluated with distance measures properly developed for this task. The goal of these measures is to compare an image segmentation to a given ground-truth image and quantify the quality of the object identification.



Figure 3. Comparison between the Rand and BGM average values for the GNM combined with several segmentations against these same selected segmentations used on their own:

The aim is to achieve the closest segmentation to the expected regions defined by the ground-truth. There are several approaches to these distance measures. According to [25], there are distances that evaluate through *counting of pairs* and *set matching*. In our tests, we used one measure of each kind: Rand [15] and Bipartite graph matching (BGM) [16], respectively a pairs-counting and a set-matching measure.

The ground-truth images provided by the Berkeley image dataset were tested with both distance measures as the quality standard for the GNM segmentations. Both Rand and BGM produce values between 0 and 1, 0 meaning a perfect match between segmentations and 1 meaning no

relation at all between them. The fact that both measures are in the same range facilitates comparing algorithms with different approaches and features. Details about these two evaluation measures can be found in the appendix A.

The GNM-combined segmentation results obtained a better average score with the Rand measure (see Figure 2.a) except for the WS+GNM, which showed comparable values. The mean Rand score for CSC+GMM was 0.087, MS+GNM was 0.089, EDISON+GNM was 0.085 and WS+GNM was 0.137. The next best result after the GNM results for Rand was EDISON with an average of 0.111, while the worst average was Blobworld's, 0.313. The GNM tests also averaged lesser values of standard deviation, all around 0.04, except for the WS+GNM that had a standard deviation value of 0.132. The smallest standard deviation after GNM was obtained with Mumford & Shah, being of 0.074.

The mean BGM scores for the GNM-combined segmentation results were not the best results, but were similar to the best BGM score (see Figure 2.b). The best BGM score was obtained with EDISON, that averaged 0.108, while the worst one was Blobworld with 0.383. The GNM results got the best scores after EDISON and CSC, which scored 0.129. BGM scores obtained with the GNM tests were 0.171 for CSC+GNM and EDISON+GNM, 0.187 for MS+GNM and 0.196 for WS+GNM. The standard deviation for all GNM segmentations were similar, around the 0.130 to 0.140 range. The smallest standard deviation was 0.082 for Blobworld, which had the worst average score. The next best standard deviation value were 0.111 and 0.117 for, respectively, CSC and EDISON. Considering both these segmentations got also the best average scores, in terms of the BGM evaluation, these are very stable and efficient segmentations.

Comparing the GNM-combined segmentation results with CSC, Mumford & Shah, EDISON and WS when run alone (Figure 3), GNM shows a sensible improvement in Rand scores for all techniques, except for WS, where the run-alone score of 0.130 was slightly better than the one obtained in combination with GNM. Still regarding Rand scores, MS and CSC showed the highest improvement (respectively, 0.165 versus 0.089 and 0.157 versus 0.087). The BGM scores showed a split, with CSC (0.129 versus 0.171) and EDISON (0.108 versus 0.171) showing better results when run alone while MS (0.277 versus 0.187) and WS (0.245 versus 0.196) were improved by a combination with GNM. As before, GNM shows a better performance when evaluated in terms of Rand conditions than BGM, though the BGM results are still better than several of the other tested segmentations.

Considering the approaches underlying each of these measures, GNM shows better results when evaluated in terms of region-pairing with the tested ground-truth images than when measured in terms of graph nodes matching. The BGM evaluation index rewards more aggressive approaches, even if the results are somewhat under-segmented, showing segment leakages. Rand scores better values for the segmentations that preserve details better. GNM proved to be a reliable technique by both measures, especially by Rand, and for both measures the results presented some of the best values in terms of deviation, also showing that it is a stable technique. The other algorithm that is very reliable by both measures was EDISON. Blobworld presented the smallest level of similarity measured. It was expected, given its inherent characteristics, where the focus is directed to capturing objects and providing a delimitation for them, and not the identification of illumination and slow progressive gradients.

The average execution time of the implementation of our GNM implementation was around 0.6 seconds. Some of the other segmentations tested averaged the following times: CSC average

time was around 0.085 seconds; MS tests averaged 5 seconds; JSEG average was around 12 seconds; and EDISON average was a little more than 20 seconds. All tests were performed on the same computer (AMD Athlon 64, 2.2 GHz and 512MB RAM memory) and all images selected from Berkely dataset have the same number of pixels. This comparison is simple and

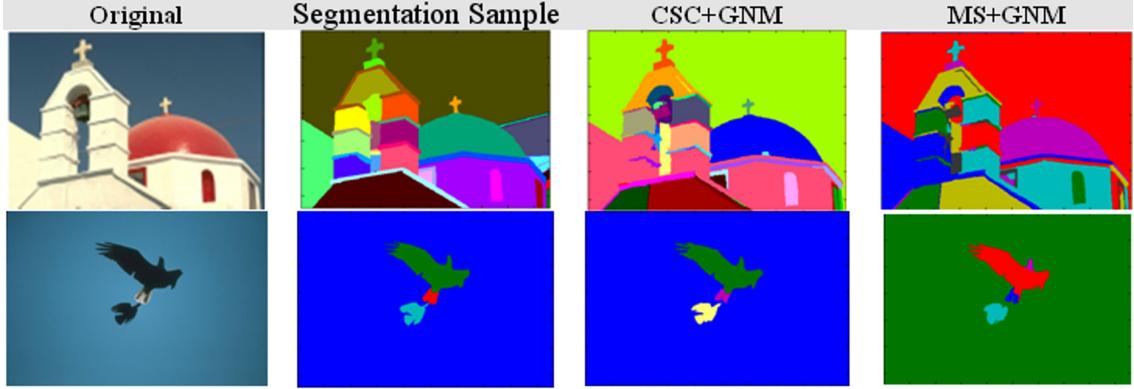


Figure 4. Examples of the image processing results obtained by the GNM, when combined with either CSC or Mumford-Shah. The first row corresponds to image number 118035 and the second row corresponds to image 135069 from the Berkeley image dataset. The segmentation sample image is also provided by the Berkeley dataset. More image sets, results comparing several techniques and higher resolution images can be found in www.lapix.ufsc.br/gnm. We also refer to [11] to view some image comparisons.

can't be viewed as definite benchmark, but indicate that the GNM can be a reasonably fast solution, as fast as CSC, for example, while producing also segmentation results of quality comparable to the most robust state-of-art techniques.

As the tests we realized resulted in more data than we would be able to properly display in this paper, we have made all the data tests available in a website. All the resulting data from the evaluation for all image sets is accessible at <http://www.lapix.ufsc.br/gnm>. There the graphs here displayed are also present, along individual graphs for every image set, tables showing all scores obtained and the images tested and the parameters used to obtain them.

6 Results and Discussion

We have shown empirically and through well known validation measures that the quality of the segmentations generated by our two-step approach is very promising and comparable to segmentations generated by state-of-the-art segmentation methods that were available for comparison when this paper was being written. The Gradient Network Method is a segmentation post-processing method that is independent of the region-growing method applied to generate the super-segmented input image, as shown by the comparison between the results produced by the four different segmentation methods used as the pre-processing step.

It is important to note that in any evaluation measure that intends to point the “best” possible result in such a complex field, as actually is color image segmentation, the values provided have to be taken with care. These measures may not provide an absolute way to determine which is actually the best segmentation, but they provide an interesting objective resource where you can compare and evaluate different segmentations under the optics of the specific approach the measure takes. Both selected measures, Rand and BGM, apply very distinct approaches to what is a better segmentation and, as it is tough to define how correct a segmentation is, it is hard to

argue which one proves to be more correct. What these measures actually provide are ways of finding if under expected conditions, reflected by the evaluation approach, a segmentation is able to perform at a desired level of quality. Observing the results presented here, we demonstrated that GNM, under the conditions of both selected measures, can obtain segmentation of at least similar quality as several state-of-art algorithms.

The processing time, however, is extremely shorter when a rapid approach like the CSC, which was originally developed for real-time color segmentation, is used, while also showing good results in terms of the validation scores. This shows that the processing step with the Gradient Network Method allows us to rely on very fast pre-segmentation methods that reduce the total processing time while producing end-segmentations of good quality. Further improvements could still be achieved in terms of efficiency with the use of a graphics processing unit for performing the necessary computations of the involved algorithms. This kind of technology, referred as General-Purpose Computing on Graphics Processing Units (GPGPU), would fulfill the requirements for real time applications. This could make the combination of CSC and GNM a feasible solution that deal with outdoor scenes, as robotics or traffic monitoring applications.

7. Acknowledgments

Daniel D. Abdala thanks the CNPq-Brazil for granting him a Ph.D scholarship under the process number 290101/2006-9.

8. References

- [1] V. Rehrmann and L. Pries. Fast and robust segmentation of natural color scenes. In ACCV, pages 598–606, 1998.
- [2] Y. Deng and B. S. Manjunath. Unsupervised segmentation of color texture regions in images and video. In Transactions on Pattern Analysis and Machine Intelligence, volume 23, pages 800–810. IEEE, 2001.
- [3] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. In Commun. Pure Applied Mathematics, volume 42, pages 577–684, 1989.
- [4] A. Dupuis and P. Vasseur. Image segmentation by cue selection and integration. Image and Video Computing, 24(10):1053–1064, 2006.
- [5] Z. Kato and T. C. Ong. A markov random field image segmentation model for color textured images. Image and Vision Computing, 24(10):1103–1114, 2006.
- [6] R. D. Dony and S. B. Wesolkowski. Edge detection on color image using rgb vector angles. In Proceedings of the 1999 IEEE Canadian Conference on Electrical and Computer Engineering, volume 9, pages 687–692, Alberta-Canada, May 1999.
- [7] M. K. Schneider, P. W. Fieguth, W. C. Karl, and A. S. Willsky. Multiscale methods for the segmentation and reconstruction of signal and images. In Transactions on Image Processing, volume 9, pages 456–468. IEEE, 2000.
- [8] Gudrun Klinker, Steven A. Shafer, and Takeo Kanade. A physical approach to color image understanding. International Journal of Computer Vision, 4:7–38, 1990.
- [9] P.W.M. Tsang and W.H. Tsang. Edge detection on object color. In IEEE International Conference on Image Processing, volume C, pages 1049–1052, 1996.
- [10] G. Healey. Segmenting images using normalized color. IEEE Transactions on Systems, Man and Cybernetics, SMC-22(1):64–73, January-February 1992.

- [11] A. Wangenheim, R. Bertoldi, D. D. Abdala, and M. M. Richter. Color image segmentation guided by a color gradient network. *Pattern Recognition Letters*, 28(13):1795–1803, October 2007.
- [12] L. Priese and P. Sturm. Introduction to the color structure code and its implementation, March 2003.
- [13] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.
- [14] L. Vincent and P. Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. In *Transactions on Pattern Analysis and Machine Intelligence*, volume 9, pages 735–744. IEEE, 1991.
- [15] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of American Statistical Association*, 66(846-850), 1971.
- [16] Xiaoyi Jiang, Cyril Marti, Christophe Irniger, and Horst Bunke. Distance measures for image segmentation evaluation. *EURASIP Journal on Applied Signal Processing*, pages 1–10, 2006.
- [17] J.C. Tilton. D-dimensional formulation and implementation of recursive hierarchical segmentation. Disclosure of Invention and New Technology: NASA Case No. GSC 15199-1, May 2006.
- [18] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, August 2002.
- [19] A. Tremeau and P. Colantoni. Regions adjacency graph applied to color image segmentation. In *Transactions on Image Processing*, volume 9, pages 735–744. IEEE, 2000.
- [20] K. Huang, Q. Wang, and Z. Wu. Natural color image enhancement and evaluation algorithm based on human visual system. *Computer Vision and Image Understanding*, 103(1):52–63, 2006.
- [21] Ranjith Unnikrishnan, Caroline Pantofaru, and Martial Herbert. Toward objective evaluation of image segmentation algorithms. In *Transactions on Pattern Analysis and Machine Intelligence*, volume 29, pages 929–943. IEEE, 2007.
- [22] Nivedita Sahasrabundhe, John E. West, Raghu Machiraju, and Mark Janus. Structured spatial domain image and comparison metrics. In *Proceedings of the Conference on Visualization*, pages 97–104, 1999.
- [23] Vito Di Gesu and Valery Starovoitov. Distance-based functions for image comparison. *Pattern Recognition Letters*, 20(2):207–214, 1999.
- [24] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of 8th International Conference on Computer Vision*, volume 2, pages 416–423, 2001.
- [25] H. D. Cheng, X. H. Jiang, Y. Sun, and J. Wang. Color image segmentation: advances and prospects. *Pattern Recognition*, 34:2259–2281, 2001.

Appendix

A Distance Measures

A.1 Rand Index

The *Rand index* first reported in [15] and reviewed in [16] is a similarity measure specially developed to evaluate the quality of clustering algorithms by comparison with other clustering results or with a golden standard (in our case, ground-truths). To compare two clustering results $C_1 = \{c_{11}, c_{12}, \dots, c_{1N}\}$ and $C_2 = \{c_{21}, c_{22}, \dots, c_{2M}\}$ over the same image $P = \{p_1, p_2, \dots, p_K\}$ where each element of C_1 or C_2 is a subset of P and $c_{1j} = \{p_{1j}, p_{2j}, \dots, p_{Lj}\}$, the following quantities are calculated:

- N_{11} - the number of pixels in the same cluster in both C_1 and C_2 .
- N_{00} - the number of pixels in different clusters both in C_1 and C_2 .

The rand index is so defined by eq. A.I

$$R(C_1, C_2) = 1 - \frac{N_{11} + N_{00}}{\frac{n(n-1)}{2}} \quad \text{A.I}$$

To compute the quantities N_{11} and N_{00} one must iterate over the entire image for each pixel in order to evaluate the conditions defined above given an $O(n^4)$ algorithm. A clever approach is to use the method described in [16] where a matching matrix is used to summarize the occurrences of pixels in the respective classes. The matching matrix is constructed allocating each cluster from the clustering C_1 to a row and each cluster from clustering C_2 to a column. The matrix cells are then defined as the intersection of the clusters specifying each row and column. If the matching matrix has $k \times l$ size each cell can be defined as $m_{ij} = |c_i \cap c_j|$, $c_i \in C_1$, $c_j \in C_2$.

The quantities N_{11} and N_{00} can be computed in terms of the matching matrix as follows:

$$N_{11} = \frac{1}{2} \left(\sum_{i=1}^k \sum_{j=1}^l m_{ij}^2 - n \right) \quad \text{A.II}$$

$$N_{00} = \frac{1}{2} \left(n^2 - \sum_{i=1}^k n_i^2 - \sum_{j=1}^l n_j^2 + \sum_{i=1}^k \sum_{j=1}^l m_{ij}^2 \right) \quad \text{A.III}$$

where n is the cardinality of P and n_i and n_j are the cardinality of the clusters c_{1i} and c_{2j} .

A.2 Bipartite Graph Matching

The *BGM index* [16] compute an one-to-one correlation between clusters at the same time trying to maximize the relationship. It considers each cluster of the C_1 and C_2 clustering as vertices of a bipartite graph. Edges are added between each vertex of the two partitions and they are valued as $|c_{1i} \cap c_{2j}|$, a value that can be directly extract from the matching matrix. Then the maximum-weight bipartite graph is defined as the subgraph $\{(c_{1i1}, c_{2j1}), \dots, (c_{1ir}, c_{2jr})\}$ where

only the edges from $c1_i$ to $c2_j$ with maximum weight are present. After all max-valued edges were found the overall graph weight is calculated by sum of all remaining edge weights.

$$BGM(C1, C2) = 1 - \frac{w}{n} \quad \text{A.VI}$$

Referências

- ABOUELELA, A. et al. Automated vision system for localizing structural defects in textile fabrics. *Pattern Recognition Letters*, v. 26, n. 10, p. 1435–1443, 2005.
- ANGULO, J.; SERRA, J. Modelling and segmentation of colour images in polar representations. *Image and Vision Computing*, v. 25, n. 4, p. 475–495, 2007.
- BOSCH, A.; MUÑOZ, X.; FREIXENET, J. Segmentation and description of natural outdoor scenes. *Image and Vision Computing*, v. 25, n. 5, p. 727–740, May 2007.
- CARSON, C. et al. Blobworld: image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 24, n. 8, p. 1026–1038, Agosto 2002.
- CASELLES, V.; KIMMEL, R.; SAPIRO, G. On geodesic active contours. *International Journal of Computer Vision*, v. 22, p. 61–79, 1997.
- CHAN, T.; VESE, L. A multiphase level set framework for image segmentation using mumford and shah model. *International Journal of Computer Vision*, v. 50, p. 271–293, 2002.
- CHENG, H. et al. Color image segmentation: advances and prospects. *Pattern Recognition*, v. 34, p. 2259–2281, 2001.
- COMANICIU, D.; MEER, P. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 24, n. 5, p. 603–619, 2002.
- DENG, Y.; MANJUNATH, B. Unsupervised segmentation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 23, n. 8, p. 800–810, 2001.
- DONY, R.; WESOLKOWSKI, S. Edge detection on color images using rgb vector angles. In: *1999 IEEE Canadian Conference on Electrical and Computer Engineering*. Edmonton, Alberta, Canada: [s.n.], 1999. p. 687–692.
- DOXYGEN. 2007. Disponível em: <<http://www.doxygen.org/>>. Acesso em: September 6 2007.
- DUPUIS, A.; VASSEUR, P. Image segmentation by cue selection and integration. *Image and Vision Computing*, v. 24, n. 10, p. 1053–1064, October 2006.
- FERNANDEZ-GARCIA, N. et al. Automatic generation of consensus ground truth for the comparison of edge detection techniques. *Image and Vision Computing*, v. 26, p. 496–511, 2008.

- FIGUEROA, P.; LEITE, N.; BARROS, R. Background recovering in outdoor image sequences: an example of soccer players segmentation. *Image and Vision Computing*, v. 24, n. 4, p. 363–374, 2006.
- FU, K.; MUI, J. A survey of image segmentation. *A survey of image segmentation*, v. 13, p. 3–16, 1981.
- GAUCH, J.; HSIA, C. Comparison of three-color image segmentation algorithms in four color spaces. In: *Proc. SPIE Vol. 1818, p. 1168-1181, Visual Communications and Image Processing '92, Petros Maragos; Ed. [S.l.: s.n.], 1992.* (Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, v. 1818), p. 1168–1181.
- GESU, V. D.; STAROVOITOV, V. Distance-based functions for image comparison. *Pattern Recognition Letters*, v. 20, n. 2, p. 207–214, 1999.
- GEVERS, T.; SMEULDERS, A. W. Color-based object recognition. *Pattern Recognition*, v. 32, p. 453–464, 1999.
- HARTMANN, G. Recognition of hierarchically encoded images by technical recognition of hierarchically encoded images by technical and biological systems. *Biological Cybernetics*, v. 57, n. 1-2, p. 73–84, 1987.
- HEALEY, G. Segmenting images using normalized color. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-22, n. 1, p. 64–73, January–February 1992.
- HUANG, K.; WANG, Q.; WU, Z. Natural color image enhancement and evaluation algorithm based on human visual system. *Computer Vision and Image Understanding*, v. 103, n. 1, p. 52–63, July 2006.
- ISO. *ISO IEC 14882:1998: Programming languages - C++*. Geneva, Switzerland: International Organization for Standardization, 1998. 732 p. Disponível em: <<http://www.iso.ch/cate/d25845.html>; <https://webstore.ansi.org/>; <http://webstore.ansi.org/ansidocstore/product.asp?sku=ISO%2FIEC+14882%2D1998>; <http://webstore.ansi.org/ansidocstore/product.asp?sku=ISO%2FIEC+14882%3A1998>>.
- JIANG, X. et al. Distance measures for image segmentation evaluation. *EURASIP Journal on Applied Signal Processing*, v. 2006, n. 1, p. 1–10, 2006.
- KATO, Z.; PONG, T. A markov random field image segmentation model for color textured images. *Image and Vision Computing*, v. 24, n. 10, p. 1103–1114, October 2006.
- KIM, D.; PARK, J. Connectivity-based local adaptive thresholding for carotid. *Image and Vision Computing*, v. 23, n. 14, p. 1277–1287, 2005.
- KLINKER, G.; SHAFER, S. A.; KANADE, T. A physical approach to color image understanding. *International Journal of Computer Vision*, v. 4, p. 7–38, 1990.
- LUCCHESE, L.; MITRA, S. Color image segmentation: A state-of-the-art survey. In: "Image Processing, Vision, and Pattern Recognition," Proc. of the Indian National Science Academy (INSA-A). [S.l.: s.n.], 2001. v. 67 A, n. 2, p. 207–221.

- MARTIN, D. et al. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: *8th International Conference on Computer Vision*. [S.l.: s.n.], 2001. v. 2, p. 416–423.
- MEGAWAVE. 2006. Disponível em: <http://www.cmla.ens-cachan.fr/Cmla/Megawave->. Acesso em: September 14 2006.
- MITTELHAEUSSER, G.; KRUGGEL, F. Fast segmentation of brain magnetic resonance tomograms. In: *CVRMed'95: First International Conference on Computer Vision, Virtual reality and Robotics in Medicine*. [S.l.: s.n.], 1995.
- MUMFORD, D.; SHAH, J. Optimal approximations by piecewise smooth functions and associated variational problems. *Commun. Pure Appl. Math.*, v. 42, p. 577–684, 1989.
- PAL, N.; PAL, S. A review on image segmentation techniques. *Pattern Recognition*, v. 26, n. 9, p. 1277–1294, 1993.
- PAN, Y.; BIRDWELL, J. D.; DJOUADI, S. Efficient implementation of the chan-vese models without solving pdes. In: *Multimedia Signal Processing 2006 IEEE 8th Workshop*. [S.l.: s.n.], 2006. p. 350–354.
- PRIESE, L.; REHRMANN, V. A fast hybrid color segmentation method. In: *DAGM-Symposium*. [S.l.: s.n.], 1993. p. 297–304.
- PRIESE, L.; STURM, P. *Introduction to the color structure code and its implementation*. March 2003. Disponível em: <http://www.uni-koblenz.de/lb/lb_downloads/download-/csc.pdf>. Acesso em: August 16 2007.
- RAND, W. M. Objective criteria for the evaluation of clustering methods. *Journal of American Statistical Association*, v. 66, p. 846–850, 1971.
- REHRMANN, V.; PRIESE, L. Fast and robust segmentation of natural color scenes. In: *ACCV*. [S.l.: s.n.], 1998. v. 1, p. 598–606.
- SAHASRABUNDHE, N. et al. Structured spatial domain image and comparison metrics. In: *In Proceedings of the Conference on Visualization*. [S.l.: s.n.], 1999. p. 97–104.
- SCHNEIDER, M. et al. Multiscale methods for the segmentation and reconstruction of signals and images. *IEEE Transactions on Image Processing*, v. 9, n. 3, p. 456–468, March 2000.
- SKARBEK, W.; KOSCHAN, A. *Colour Image Segmentation - A Survey*. [S.l.], October 1994.
- SUN, H.; YANG, J.; REN, M. A fast watershed algorithm based on chain code and its application in image segmentation. *Pattern Recognition Letters*, v. 26, n. 9, p. 1266–1274, 2005.
- TILTON, J. *D-dimensional formulation and implementation of recursive hierarchical segmentation*. May 2006. Disclosure of Invention and New Technology: NASA Case No. GSC 15199-1.

- TRÉMEAUX, A.; COLANTONI, P. Regions adjacency graph applied to color image segmentation. *IEEE Transactions on Image Processing*, v. 9, n. 4, p. 735–744, 2000.
- TSANG, P.; TSANG, W. Edge detection on object color. In: *IEEE International Conference on Image Processing*. [S.l.: s.n.], 1996. C, p. 1049–1052.
- UNNIKRISHNAN, R.; PANTOFARU, C.; HEBERT, M. Toward objective evaluation of image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 29, n. 6, p. 929–944, Junho 2007.
- VINCENT, L.; SOILLE, P. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Trans. Pattern Analysis and Machine Intell.*, v. 13, p. 583–598, 1991.
- WEICKERT, J. Applications of nonlinear diffusion in image processing and computer vision. *Acta Mathematica Universitatis Comenianae*, v. 70, n. 1, p. 33–50, 2001.
- WIT, T. de. Fast segmentation of solar extreme ultraviolet images. *Solar Physics*, v. 239, n. 1-2, p. 519–530, December 2006.
- WONG, K.; SPETSAKIS, M. Tracking based motion segmentation under relaxed statistical assumptions. *Computer Vision and Image Understanding*, v. 101, n. 1, p. 45–64, 2006.
- WU, Z.; LEAHY, R. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 15, n. 11, p. 1101–1113, November 1993.
- WXWIDGETS. 2007. Disponível em: <<http://www.wxwidgets.org/>>. Acesso em: September 6 2007.
- XU, C.; PRINCE, J. L. Snakes, shapes and gradient vector flow. *IEEE Trans. Image Processing*, v. 7, p. 359–369, 1998.