Universidade de São Paulo
Instituto de Matemática e Estatística
Bachalerado em Ciência da Computação

Pedro Ferreira Alexandre

Segmentação de imagens pela Transformada Imagem-Floresta em grafos de superpixels

São Paulo
Fevereiro de 2017

Segmentação de imagens pela Transformada Imagem-Floresta em grafos de superpixels

Monografia final da disciplina
MAC0499 – Trabalho de Formatura Supervisionado.

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

São Paulo
Fevereiro de 2017

Resumo

A técnica de superpixels consiste em agrupar pixels similares de uma imagem em regiões compactas e conexas. As regiões idealmente devem preservar as bordas de transição entre os diferentes objetos presentes na imagem . Essa técnica pode diminuir a complexidade de tarefas de visão computacional e processamento de imagens, pois no nível de superpixels temos um número de elementos da imagem significantemente menor, consequentemente elevando a eficiência. A Transformada Imagem-Floresta (Image Foresting Transform - IFT), Falcão et al. (2004), é um algoritmo que calcula uma floresta de caminhos ótimos em um grafo derivado da imagem. A segmentação interativa de objetos via IFT é obtida através da seleção de pixels sementes dentro e fora do objeto, sendo o objeto segmentado definido pela união das árvores enraizadas em suas sementes internas. Rauber et al. (2013) propuseram o uso da IFT sobre um grafo de regiões (superpixels) obtido via algoritmo de watershed. Segundo os autores, os resultados da segmentação via IFT no nível do grafo de regiões gerou melhores resultados quando comparado a IFT em um grafo no nível dos pixels. No entanto, existem algoritmos mais adequados para a geração de superpixels que o uso das bacias do watershed, tais como o SLIC. Esse trabalho visa avaliar o impacto do uso de um grafo de regiões obtido pelo método SLIC na segmentação via IFT. Os experimentos serão conduzidos usando sementes obtidas automaticamente de um gabarito de segmentação (ground truth), via técnica de erosão, gerando uma região de incerteza a ser segmentada entorno da borda.

Palavras-chave: superpixels, processamento de imagens, segmentação de imagens.

Abstract

The superpixels technique is to group similar pixels of an image in compact and related regions. The regions should ideally preserve the transition edges between different objects in the image. This technique can reduce the complexity of tasks of computer vision and image processing, because superpixels have a significantly lower number of picture elements, thus increasing efficiency. The Image Foresting Transform (IFT) by Falcão et al. (2004) is an algorithm that calculates a forest of optimal paths in a graph derived from the image. Interactive segmentation via IFT is achieved through the selection of seed pixels inside and outside the object, and the segmented object defined by the union of trees rooted in its internal seeds. Rauber et al. (2013) has proposed the use of IFT regions on a graph (superpixels) obtained via watershed algorithm. According to the authors, the results of segmentation via IFT in regions graph level generated better results when compared to IFT in a graph at the level of pixels. However, there are more suitable algorithms for generating superpixels that the use of watershed basins, such as the SLIC. This study aims to evaluate the impact of using a graph of regions obtained by SLIC method when segmented via IFT. The experiments will be conducted using seeds obtained automatically from a target template (ground truth), via erosion technique. generating an uncertainty region to be targeted around the edge.

Keywords: superpixels, image processing, image segmentation.

Sumário

Capítulo 1

Introdução

1.1 Descrição do problema

Quando trabalha-se com imagens nem sempre é possível imaginar quantos cálculos são necessários para realizar pequenas operações sobre elas. A maioria das fotos são compostas por mais de 1 milhão de pixels, e cada operação que envolva essa imagem, se não for devidamente otimizada, pode ter que ser repetida milhões de vezes.

No entanto, uma imagem, geralmente, contém diversas áreas redundantes, fazendo com que os mesmos cálculos sejam feitos repetidamente, causando o desperdício de tempo e computação.

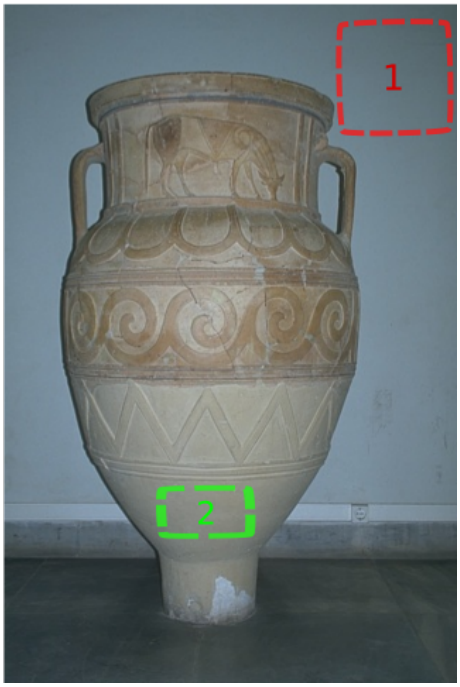

Figura 1.1: *Imagem com áreas redundantes.*

1.2 Agrupamento de pixels

Para tentar solucionar isso é possível que a imagem seja dividida, agrupando os pixels em regiões, e então serão economizadas operações que fossem ser aplicadas a pixels que estivessem na mesma região.

1.2.1 Abordagem trivial

Uma forma simples de representar as regiões é através de linhas paralelas às bordas da imagem, como a seguir:

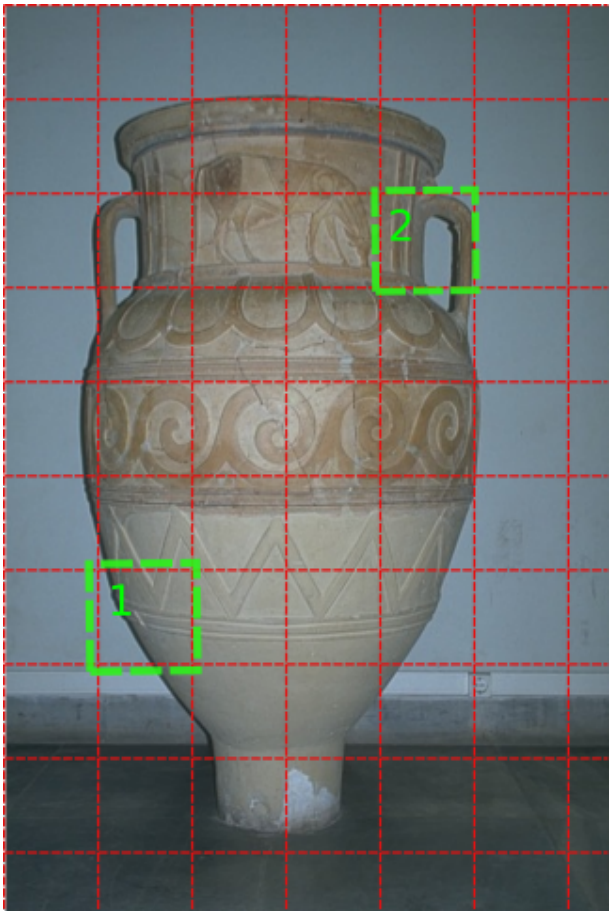

Figura 1.2: *Imagem superpixels em gride.*

Pode-se notar com essa solução, que uma região específica pode envolver um conjunto de pixels que contenha dois elementos P_1 e P_2 tais que as características destes são muito distintas entre si, e portanto não seria correto assumir que ambos possuem características iguais.

1.2.2 Abordagem elaborada

A técnica de superpixels consiste em agrupar pixels de uma imagem e classificá-los em regiões atomicamente significantes, podendo substituir a estrutura rígida dos pixels, sempre tentando preservar as características desta imagem, como bordas e cores.

Um superpixel é uma representação para uma região redundante da imagem. O uso dessa estrutura pode reduzir consideravelmente a complexidade de processamentos subsequentes que envolvam a imagem. A estrutura de superpixels se tornou parte importante de muitos algoritmos de computação visual, tais como: segmentação, estimativa de modelo corporal, e reconhecimento de objetos.

Figura 1.3: *Imagem superpixels SLIC.*

Pode-se notar a partir da imagem criada pelo SLIC que os contornos da imagem são preservados na maioria dos casos, e os pixels contidos em um superpixel podem ser considerados elementos com características iguais sem muita perda de informação.

1.3 Objetivos

Recentemente, superpixels tem atraído interesse por possibilitar rápida segmentação e uma descrição de características melhor do que janelas de tamanho fixo envolvendo os pixels.

A proposta deste trabalho é avaliar o desempenho em termos de acurácia para um algoritmo específico aplicado a uma imagem pré-processada e agrupada por regiões, superpixels, comparado com a aplicação do mesmo algoritmo na imagem a nível de pixels.

O algoritmo aplicado para a comparação dos resultados será a Transformada Imagem-Floresta (IFT - Image Foresting Transform) Falcão *et al.* (2004) . O Algoritmo da IFT será usado para separar objeto e fundo em uma imagem. Para aplicar a IFT sobre os superpixels é preciso realizar um pré-processamento da imagem.

No artigo Rauber *et al.* (2013) foi realizado um trabalho semelhante, da aplicação da IFT sobre imagens em pixels e superpixels. O estudo realizado por *Rauber* apresenta resultados satisfatórios embora o método de segmentação em superpixels utilizado não seja o mais atual.

Neste trabalho será verificado o comportamento do algoritmo da IFT ao utilizar o SLIC Achanta *et al.* (2012) , um dos métodos mais atuais para a criação dos superpixels, e comparar os resultados ao desempenho da IFT a nível de pixels.

Capítulo 2

Algoritmos

Nesta seção é explicado em mais detalhes o que é o SLIC e suas características. Logo em seguida, será descrito em mais detalhes o algoritmo da IFT, que será o algoritmo utilizado nos experimentos.

2.1 SLIC - Simple Linear Iterative Clustering

Simple linear iterative clustering Achanta *et al.* (2012) é um método para geração de superpixels baseado em gradientes ascendentes. Algoritmos dessa classe utilizam-se de múltiplas iterações para refinar um conjunto de pixels, até que um critério seja satisfeito, para a formação de um superpixel.

Esse algoritmo é mais rápido, utiliza menos memória, realiza ótima aderência com bordas e fronteiras de objetos na imagem e melhora a performance de algoritmos de segmentação. Simple linear iterative clustering é uma adaptação do k-means para a geração de superpixels, com duas importantes distinções:

- O número de cálculos de distância é drasticamente reduzido ao limitar o espaço de busca para uma região proporcional ao tamanho do superpixel. Isso faz com que a complexidade seja reduzida a linear no número de pixels N e independente do número de superpixels k.
- Uma medida ponderada de distância que combina cor e proximidade espacial, promovendo controle sobre o tamanho e a compacidade dos superpixels.

Neste trabalho os experimentos serão feitos a partir do gerador de superpixels conhecido como IFT-SLIC Alexandre *et al.* (2015) e não o SLIC propriamente dito. O IFT-SLIC é um método mais novo que possui melhores resultados que o SLIC.

2.2 IFT - Image Foresting Transform

A Transformada Imagem-Floresta (IFT - Image Foresting Transform) reduz problemas de processamento de imagem baseados em conexidade ao cálculo de uma floresta de caminhos ótimos no grafo derivado da imagem, seguido de um pós-processamento simples de atributos da floresta resultante, como visto em Falcão *et al.* (2004).

O Algoritmo da IFT pode ser usado para separar objetos do fundo em uma imagem. Nesse caso a IFT realiza uma busca através dos elementos da imagem, a partir de sementes iniciais que podem ser de objeto ou de fundo, por vizinhos similares. No decorrer da busca, elementos são conquistados virando objeto ou fundo e formando o conjunto final, que se trata de uma imagem binária.

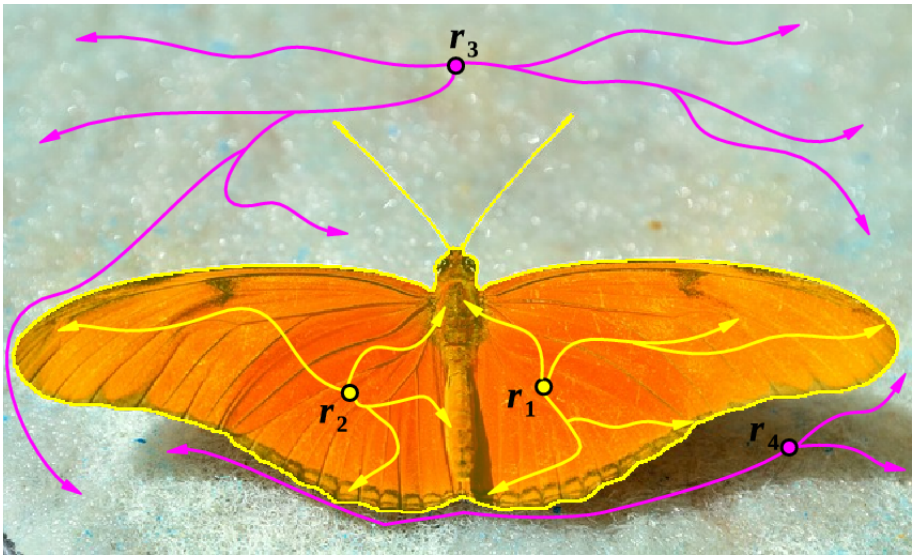

Figura 2.1: *Processo de conquista de regiões por sementes de objeto, r1 e r2, e sementes de fundo, r3 e r4*

Na figura (2.1) encontra-se uma mariposa e o comportamento de algumas sementes de fundo ou objeto. As sementes de objeto **r1** e **r2**, que se encontram dentro da mariposa, se deslocam conquistando o objeto e suas bordas. Na imagem essa conquista é indicada por uma curva de cor amarela. Enquanto isso as sementes **r3** e **r4** deslocam-se conquistando o fundo da imagem (cor roxa).

2.2.1 Definição

A abordagem utilizada para a IFT nesse trabalho é baseada em um grafo criado para a imagem. Como visto em Falcão *et al.* (2004), temos que uma imagem **I** pode ser vista como uma tupla $(\mathcal{I}, I)$ consistindo de um conjunto finito $\mathcal{I}$ composto por pixels (pontos no $\mathbb{Z}^2$) e uma função I que mapeia cada pixel t em $\mathcal{I}$ em algum valor $I(t)$ em algum espaço arbitrário.

No caso da IFT a nível de pixels, a partir de um pixel t a IFT se desloca através de t para seus pixels adjacentes do grafo gerado para a imagem. No caso da imagem segmentada em superpixels, sendo s um superpixel serão criados arcos entre superpixels pré-estabelecidos como vizinhos de s.

Caminhos no grafo da imagem

Dado um grafo da imagem, temos que um caminho simples π_t nesse grafo é uma sequência de pixels ou regiões, dependendo do tipo de segmentação da imagem. Todos os caminhos considerados nesse trabalho são caminhos simples, isto é, caminhos que passam por cada vértice no máximo uma vez. Assim, usamos a notação $\pi_t = \langle t \rangle$ para indicar um caminho composto por um único vértice, e $\pi_t = \langle s, t \rangle$ para indicar a extensão de um caminho π_s, com término em s, por uma aresta (s, t) presente no grafo da imagem.

Custos nos caminhos

Será assumida uma função f que associa a cada caminho π um custo $f(\pi)$, em algum conjunto totalmente ordenado de valores de custo. É comum que o custo do caminho esteja diretamente ligado a propriedades da imagem, tais como cor, gradiente, e posição do pixel ou superpixel ao longo do caminho.

A função de custo utilizada nesse trabalho é a f_{max}, definida por:

$$\begin{aligned} f_{max}(\langle t \rangle) &= h(t), \\ f_{max}(\pi.\langle s, t \rangle) &= max(f_{max}\{\pi), w(s, t)\} \end{aligned}$$

Onde $h(t)$ e $w(s, t)$ são as funções de custo inicial(handicap) e a função de peso dos arcos, respectivamente.

Usualmente o valor de custo inicial $h(t)$ é dado por 0, se o t é uma semente e $+\infty$ caso contrário, sendo t um pixel ou superpixel, dependendo do tipo de segmentação da imagem.

Caminho ótimo

Seja $\prod$ o conjunto de todos os caminhos possíveis no grafo de uma imagem $G = (\mathcal{I}, I)$ com destino t. Um caminho π_t é ótimo se $f(\pi_t) \leq f(\pi'_t)$ para qualquer outro caminho $\pi'_t \in \prod$. A transformada imagem-floresta para um grafo $G = (\mathcal{I}, I)$, e para a função de custo f_{max}, associa um caminho ótimo π_t para cada pixel ou superpixel $t \in \mathcal{I}$. No entanto, f deve ser suave, caso contrário os caminhos gerados podem não serem ótimos Falcão *et al.* (2004).

Capítulo 3

Conceitos

Essa seção explica em mais detalhes as ferramentas e alguns conceitos utilizados nesse trabalho. Será tratado o conceito de erosão de uma imagem e a geração de sementes para o algoritmo da IFT por erosão. Por fim, terá uma apresentação do Coeficiente de Dice, utilizado nesse trabalho para verificação da acurácia do método aplicado.

3.1 Erosão

A erosão (usualmente representada por $\ominus$) e a dilatação são consideradas as duas operações fundamentais do processamento morfológico de imagens, sendo assim, todas as outras operações são baseadas nelas.

Em morfologia binária, uma imagem é vista como um sub-conjunto de um espaço Euclideano $\mathbb{R}^d$ ou o conjunto de inteiros $\mathbb{Z}^d$, para alguma dimensão d.

A ideia básica em uma morfologia binária é examinar uma imagem utilizando um objeto simples com formato pré-definido, e decidir se esse objeto se enquadra nos objetos presentes na imagem. Esse objeto de examinação é chamado de elementro estruturante, e é uma imagem também.

Seja E um espaço Euclideano ou um gride de inteiros, e A uma imagem binária em E. A **erosão** de uma imagem binária A produzida pelo elemento estruturante B é definida por:

$$A \ominus B = \{z \in E | B_z \subseteq A\} \tag{3.1}$$

Onde B_z é a translação de B dada pelo vetor z, i.e.,

$$B_z = \{b + z | b \in B\}, \forall z \in E \tag{3.2}$$

Exemplo:

Dada a imagem binária A (3.1) a seguir e o elemento estruturante B (3.1)

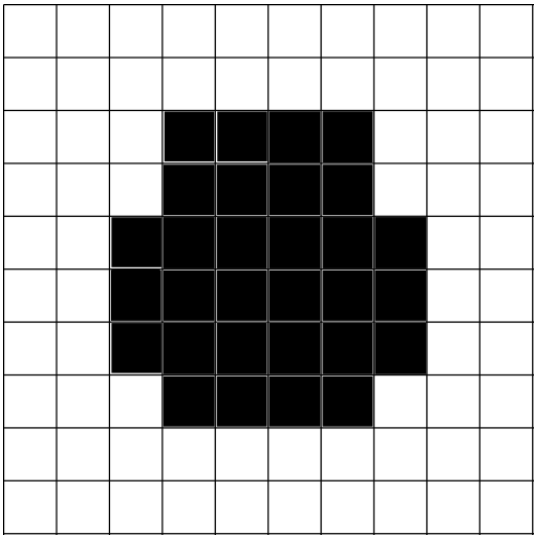

Figura 3.1: A

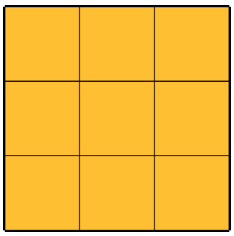

Figura 3.2: B

Aplicando-se o elemento estruturante B em cada elemento da imagem A, encontraremos elementos que não pertencem a $A \ominus B$, por exemplo o elemento centralizado no meio do elemento estruturante da imagem (3.3)

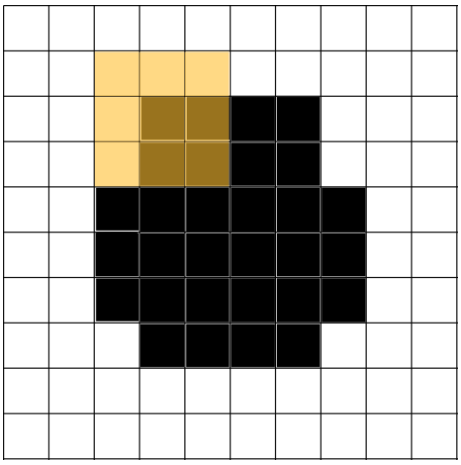

Figura 3.3

Também podem existir elementos que pertencem a $A \ominus B$, por exemplo o elemento centralizado no meio do elemento estruturante, como mostrado na imagem (3.4).

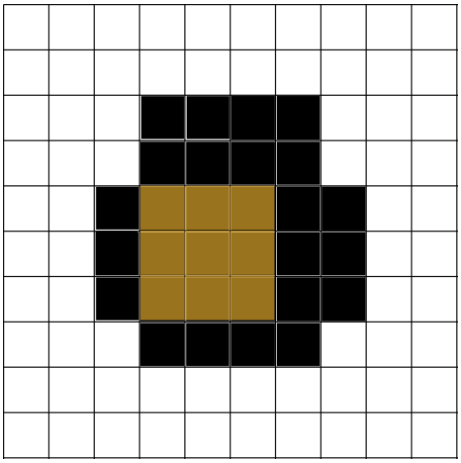

Figura 3.4

O resultado da operação de erosão, $A \ominus B$, será dado pela imagem (3.5).

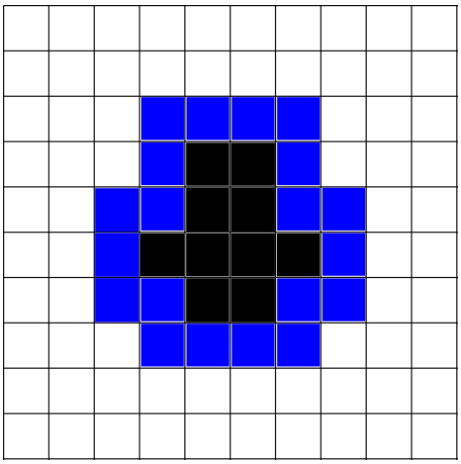

Figura 3.5

3.2 Gerando as sementes

3.2.1 Sementes na IFT

O algoritmo da IFT é inicializado a partir de sementes do fundo e do objeto. Através dessas sementes o algoritmo busca por regiões da imagem que pertençam ao mesmo objeto que a semente especificada. Por exemplo, se a IFT receber uma semente de fundo em uma iteração, ela vai percorrer os arredores daquela semente se guiando através de regiões semelhantes para clássificá-las como também sendo regiões pertencentes ao fundo da imagem.

3.2.2 Sementes por erosão

Uma forma de gerar as sementes é através da erosão da imagem do gabarito (imagem binária que separa o fundo do objeto). Aplicando a erosão no gabarito de uma imagem e utilizando raios diferentes para o elemento estruturante gera-se um conjunto de imagens para serem utilizadas como sementes da IFT. Se for realizado um processo de erosão, partindo do raio *0* até o raio máximo de erosão, serão geradas imagens distorcidas, gradualmente, sendo a imagem de raio zero uma semente extremamente significativa(gabarito sem erosão), até se tornar uma semente bem pouco significativa quando o raio é máximo.

Por exemplo, a partir da imagem a seguir:

Figura 3.6: *scissors, base de dados pública do grabcut*

Se for aplicada a erosão no respectivo gabarito, será obtida uma semente diferente para os diversos raios de erosão. Na imagem seguinte será apresentada uma imagem variando seu raio de erosão de 0 (imagem sem erosão) até 45.

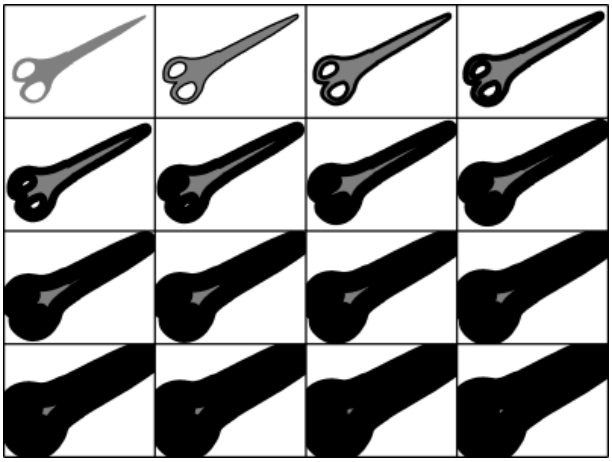

Figura 3.7: *raio 0 até 45, de 5 em 5.*

3.3 Coeficiente de Dice

Esse coeficiente também é conhecido por outros nomes, tais como: Índice de Sorensen ou Coeficiente de Sorensen-Dice. Essa medida é uma estatística usada para comparar a semelhança entre duas amostras. Também é conhecido como F1 score.

A aplicação da formula original era destinada para a verificação de presença ou ausência de uma informação específica.

$$CSD = \frac{2|X \cap Y|}{|X| + |Y|}$$

Neste trabalho a fórmula de Dice será utilizada para análisar a proximidade entre dois conjuntos X e Y, respectivamente, o conjunto de pixels do objeto segmentado e o conjunto

de pixels do gabarito. Onde X será o conjunto de pixels classificado pela IFT em objeto ou fundo. Quanto mais parecidos, segundo a fórmula de Dice, forem X e Y, então, melhores foram os resultados obtidos pela IFT na segmentação da imagem.

Capítulo 4

Experimentos

4.1 Resumo dos experimentos

Os experimentos consistiram em medir a acurácia da IFT ao separar objeto e fundo de uma imagem e foram rodados para o banco de imagens públicas do grabcut que contém 50 imagens. Foram feitas duas classes de rotinas principais:

- IFT sobre uma imagem bruta, sem pré-processamento em regioes (nível de pixels).
- IFT sobre regioes geradas pelo método SLIC, com superpixels com aproximadamente 100, 400, 900 e 1600 pixels.

4.2 Programas utilizados

Para rodar os experimentos foram utilizadas implementações do SLIC, da IFT e um programa para a produção de sementes com erosão. As bibliotecas externas utilizadas foram escritas em C++ e C.

- ift_sp.cpp: implementação da **IFT** com o **IFT_SLIC** da gft como gerador dos superpixels.
- ift.cpp: implementação da **IFT** sobre pixels.
- eroeval.cpp: realiza chamada para **ift_sp** e **ift** para varios raios de erosão da imagem.
- Também foi utilizada a biblioteca externa **gft**, onde encontram-se implementações de leitura , escrita e erosão de imagem.

4.3 Resultados obtidos

Sobre o processo de geração de sementes via erosão, o raio de erosão é maior para a parte do fundo (dobro do valor usado do raio interno), a fim de evitar a geração de sementes de objeto e fundo equidistantes da borda ideal do gabarito. Nos gráficos o eixo x mostra o valor da erosão do objeto, e a erosão do fundo é sempre o dobro desse valor. As curvas abaixo são as curvas médias encontradas para o banco de dados do grabcut contendo 50 imagens, a partir de cada uma das execuções da IFT. Para as curvas médias foram considerados somente os valores absolutos de raio de erosão com frequência maior do que 20 imagens.

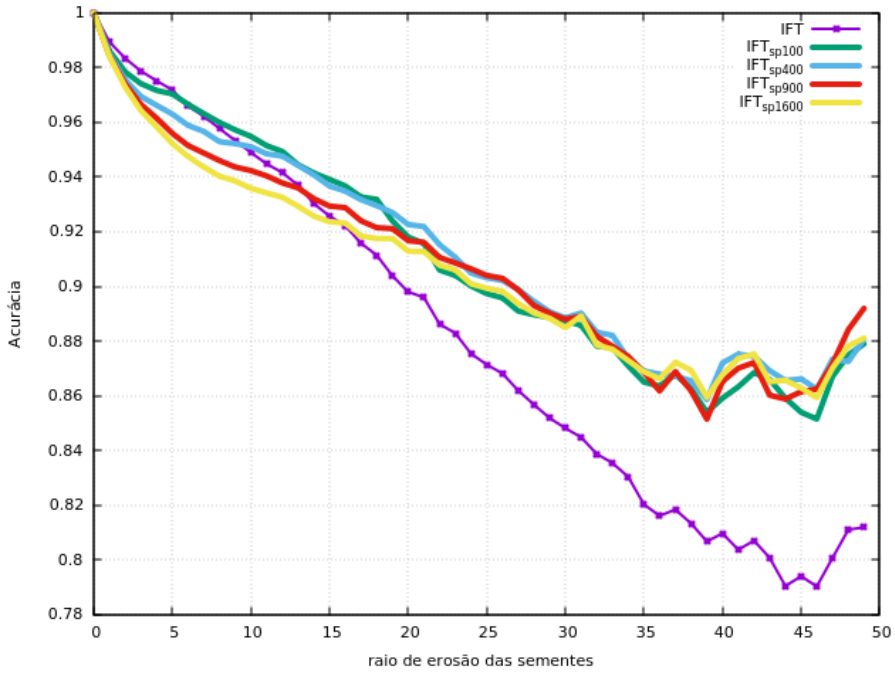

Figura 4.1: *Acurácia é o valor do Coeficiente de Dice, e as curvas apresentadas são as curvas médias de todas as imagens do grabcut*

4.3.1 IFT sobre pixels

Foi criado uma rotina para rodar o executável do **eroeval** para todas as imagens do grabcut, inicialmente passando como parâmetro a opção da IFT a nível de pixels. Representado na imagem dos resultados pela curva roxa traçejada, a IFT a nível de pixels obteve bons resultados quando as sementes eram mais significativas, raios de erosão menores do que 10 pixels, sendo a curva com maior acurácia entre as curvas geradas, no intervalo de raio 0 até 5. No entanto, ela piora sua acurácia ao longo da diminuição do raio da erosão.

4.3.2 IFT sobre superpixels

E em seguida foi executada a mesma rotina, mas passando a opção da IFT sobre superpixels, e a área aproximada dos superpixels a serem gerados pelo SLIC como parâmetros.

Um superpixel de área 100, por exemplo, indica que cada superpixel será composto por uma região de aproximadamente 10x10 pixels.

superpixels 10x10

A curva obteve um dos melhores comportamentos, e no início, quando todas as outras curvas de superpixels ficaram mais afastadas da curva da IFT sobre pixels, a curva verde se manteve bem próxima a curva da IFT sobre pixels. No final, assim como todas as curvas de superpixels, esta se manteve estável e bem melhor do que a curva tradicional, sobre pixels.

superpixels 20x20

Um pouco pior no início, quando comparado a IFT sobre a imagem de superpixels 10x10, no entanto, logo o comportamento da curva se mantém bem parecido com a curva anterior.

superpixels 30x30 e 40x40

Ambas obtiveram desempenho similar, e mostram que quanto maior o tamanho de um superpixel, de acordo com esse experimento, pior foi a acurácia para raios pequenos de erosão (caso onde as sementes fornecem muita informação do objeto e do fundo).

4.4 Análise final dos resultados

Comparando a IFT através do pré-processamento de uma imagem em superpixels através do SLIC com a IFT sobre pixels, para separar um objeto do fundo de uma imagem, os resultados mostraram que o método baseado em superpixels fornece uma estabilidade maior conforme ocorre aumento do raio da erosão.

Esse resultado é semelhante ao resultado encontrado em Rauber *et al.* (2013) onde o autor também utiliza a IFT, mas nesse caso os superpixels são gerados por watershed como apresentado em Lotufo *et al.* (2002). De acordo com os experimentos realizados nesse trabalho, o SLIC também obteve desempenho superior e confirmou-se uma técnica eficiente nesse caso de uso.

Lembrando-se que nesse trabalho não foi feita uma comparação direta com os resultados obtidos em Rauber *et al.* (2013). A ideia principal foi verificar o desempenho do SLIC como gerador de superpixels para a IFT, comparado a IFT em nível de pixels.

Capítulo 5

Conclusões

Neste trabalho foi apresentado um problema computacional relacionado a imagens e uma análise da aplicação de uma solução existente para a segmentação de imagens, o SLIC, ao algoritmo da IFT.

No **Capítulo 2** foi apresentada uma noção mais detalhada do SLIC e da IFT. Logo em seguida, no **Capítulo 3** foram introduzidos alguns conceitos para facilitar o entendimento da geração de sementes por erosão. Por fim também foi apresentado, brevemente, o Coeficiente de Dice.

Então, no **Capítulo 4**, foram realizados experimentos que visavam comparar o desempenho das execuções da IFT, sobre pixels e superpixels. A segmentação em superpixels utilizada no experimento reduziu bastante o número de elementos a serem computados pela IFT, pois cada imagem foi segmentada em regiões com superpixels de tamanho aproximado de 10x10, 20x20, 30x30 e 40x40 pixels, dependendo do experimento. Fazendo com que o número de regiões examinadas na IFT_sp fosse consideravelmente menor. No caso de superpixels 40x40, por exemplo, cada 1600 pixels da imagem se tornam um superpixels, e portanto o número de elementos é reduzido em aproximadamente 1600 vezes.

A idéia de realizar um pré-processamento da imagém parece bem interessante, pois por mais que esse pré-processamento custe tempo e computação, uma vez feito, essa imagem segmentada pode ser guardada e utilizada posteriormente por outros algoritmos que possam se aproveitar dessa estrutura de superpixels. No entanto, ao diminuir o número de pixels perdemos a informação individual de cada pixel.

Embora a segmentação em superpixels faça com que a imagem perca informação, os resultados mostram que a informação extraída da região quando é pré-computada pelo algoritmo do SLIC é valiosa, gerando resultados melhores do que a IFT sobre pixels quando a semente fornecida para separação do objeto é menos significativa.

Como o SLIC é considerado uma das ténicas mais atuais e os experimentos feitos em Rauber *et al.* (2013) para IFT-watershed já mostravam resultados positivos para superpixels, era esperado que os resultados encontrados para IFT-slic nesse trabalho fossem satisfatórios. Por isso os resultados obtidos estão dentro das expectativas imaginadas.

A princípio esse trabalho tinha a proposta de utilizar, adicionalmente à técnica de erosão para criação de sementes da IFT, a geração de sementes através de um usuário robô, assim como feito em Rauber *et al.* (2013) , para os experimentos. No entanto, algumas dificuldades na implementação fizeram com que somente a geração de sementes por erosão fosse implementada neste momento.

Capítulo 6

Agradecimentos

Agradeço ao meu orientador, Prof. Dr. Paulo A. V. de Miranda, por ser sempre acessível e auxiliar muito no desenvolvimento e entendimento dos algoritmos utilizados. Também gostaria de agradecer a Profa. Dra. Nina S. T. Hirata, por suas orientações e o acompanhamento feito, ajudando no bom andamento do trabalho.

Referências Bibliográficas

Achanta *et al.***(2012)** Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, Fellow e Sabine Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. 34(11). Citado na pág. 4, 5

Alexandre *et al.***(2015)** Eduardo B. Alexandre, Ananda Chowdhury, A. X. Falcão e Paulo A.V. Miranda. Ift-slic: A general framework for superpixel generation based on simple linear iterative clustering and image foresting transform. Em *XXVIII SIBGRAPI - Conference on Graphics, Patterns and Images.*, páginas 337–344. Citado na pág. 5

Falcão *et al.***(2004)** Alexandre X. Falcão, Jorge Stolfi e Roberto de Alencar Lotufo. The image foresting transform: Theory, algorithms, and applications. 26(1):19–28. Citado na pág. 4, 6, 7

Lotufo *et al.***(2002)** Roberto Lotufo, Alexandre Falcão e Francisco Zampirolli. Ift-watershed from gray-scale marker. Em *Computer Graphics and Image Processing. Proceedings. XV Brazilian Symposium*, páginas 146–152. Citado na pág. 17

Rauber *et al.***(2013)** Paulo E. Rauber, Alexandre X. Falcão, Thiago V. Spina e Pedro J. de Rezende. Interactive segmentation by image foresting transform on superpixel graphs. Em *XXVI Conference on Graphics, Patterns and Images*, páginas 131–138. Citado na pág. 4, 17, 20