

Projeto de Pesquisa

Iniciação Científica

Aprendizado de transformações de imagens via classificação de microrregiões

Estudante

Pedro Henrique Barbosa de Almeida

Bacharelado em Ciência da Computação

Orientadora

Nina S. T. Hirata

Departamento de Ciência da Computação

Instituto de Matemática e Estatística

Universidade de São Paulo

Resumo: Várias transformações de imagens podem ser realizadas por operadores de imagens que são modelados por funções que processam os *pixels* individualmente. Esta característica permite que a definição dessas funções seja inserida no contexto de aprendizado de máquina como um problema de aprendizado de classificadores de *pixels*. Dentre os processamentos de imagens comumente utilizados destaca-se a segmentação, que produz uma partição dos pontos da imagem tal que determinadas regiões correspondem aos componentes de interesse e outras a partes a serem ignoradas em uma posterior análise. Devido ao alto custo computacional de se processar cada *pixel* individualmente, *pixels* “similares” podem ser agrupados em microrregiões de forma a reduzir o número de elementos atômicos. O problema de classificar *pixels* pode então ser trocado pelo problema de classificar essas microrregiões. Em problemas de segmentação, contanto que a borda dessas microrregiões tenha boa aderência ao contorno das regiões de interesse, não há perda de precisão e poderá haver um ganho significativo em termos de custo computacional. O objetivo deste projeto de pesquisa é estender os métodos de aprendizado de operadores já estabelecidos, adicionando métodos de aprendizado de operadores que atuam sobre microrregiões. Os novos métodos deverão ser integrados à biblioteca TRIOSlib, mantida pelo grupo, aplicados em problemas de segmentação de imagens, e comparados aos operadores que atuam sobre *pixels*.

Palavras-chave: visão computacional; aprendizado profundo; microrregiões; operadores de imagens; transformações de imagens.

São Paulo, 26 de maio de 2018

1 Introdução

Sistemas de visão computacional dependem de informações extraídas das imagens [Pri12, Sze11]. Essa extração de informações é uma tarefa complexa, sendo altamente relacionada com o propósito da aplicação e dos tipos de imagens a serem analisadas. A maioria das análises depende, por sua vez, de um processamento adequado das imagens. Dentre os processamentos, a segmentação de regiões de interesse destaca-se por ser um processamento necessário em praticamente todas as análises. Por exemplo, em análise de imagens de documentos, um objetivo comum é o reconhecimento de caracteres, de forma a possibilitar a seleção de palavras-chaves presentes no documento para que as mesmas possam ser utilizadas para indexação. Antes do reconhecimento de caracteres propriamente, pode ser interessante detectar as regiões na imagem que correspondem a texto (e portanto são passíveis de serem passadas para um OCR, i.e., para um reconhecedor de caracteres). Outro exemplo é a segmentação de vasos sanguíneos em imagens da retina do olho. A delineação precisa dos vasos permite a quantificação da estrutura dos vasos, a qual pode ser utilizada para diagnóstico de possíveis enfermidades do olho.

O problema de segmentação de imagens [GW02] consiste no particionamento do domínio de imagens em segmentos disjuntos, de forma que cada segmento corresponda a um objeto ou parte de objeto que sejam componentes de interesse na análise em questão. Em geral, as imagens contém também um ou mais segmentos que correspondem ao *background* (i.e., regiões da imagem que não são de interesse).

A segmentação de imagens costuma ser abordada como um problema de classificação de pixels, ou de microrregiões (por exemplo, superpixels), ou ainda um problema de demarcação, por meio de *bounding boxes*, de uma área retangular que contém os objetos de interesse. A precisão das demarcações varia de muito preciso (caso de pixels) para pouco preciso (caso dos *bounding boxes*). A granularidade dos elementos atômicos (*pixel*, microrregião, retângulos) é determinante para a precisão da segmentação. Por exemplo, no caso de segmentação de vasos sanguíneos, a precisão em nível de pixels é desejada pois levará à quantificações mais precisas. Por outro lado, nos casos nos quais o objetivo é por exemplo a detecção da presença de pessoas em uma cena, a precisão de *bounding box* em geral é suficiente. A classificação no nível de microrregiões encontra-se em um nível intermediário entre essas duas. Dependendo da aderência das bordas dessas microrregiões ao contornos presentes na imagem, maior ou menor será a precisão de uma segmentação final.

Os problemas de segmentação discutidos acima podem ser abordados como um problema de aprendizado computacional. O aprendizado de operadores de imagens [MHH16] refere-se ao problema de, dados pares de imagens entrada-saída como as mostradas na figura 1, determinar a transformação que mapeia a imagem de entrada para a correspondente imagem de saída. Mais especificamente, os pares de imagens são usados como dados de treinamento. Espera-se que

a transformação obtida, isto é, o operador de imagens resultante do processo de aprendizado, caracterizado por meio de um classificador de *pixels* por exemplo, seja capaz de transformar igualmente bem outras imagens similares. Essa formulação inclui não apenas transformações do tipo segmentação mas outros como filtragem ou restauração.

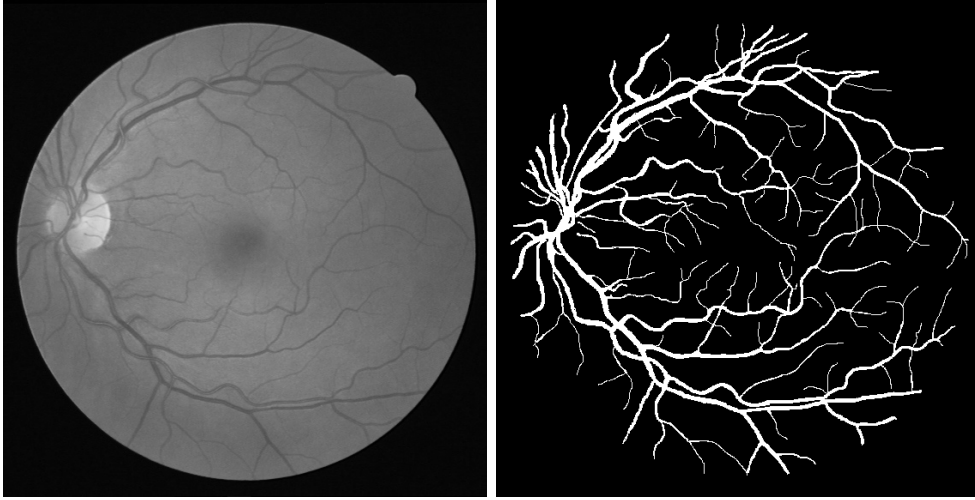


Figura 1: Um exemplo de par de imagens entrada-saída, ilustrando a transformação desejada.

No contexto de **aprendizado de operadores de imagens**, a granularidade mais explorada é a de *pixels* [MHH16]. Mais recentemente o grupo trabalhou também com métodos que atuam sobre componentes conexos em imagens binárias [JAMH17] e um método que realiza a detecção/segmentação em nível de *bounding boxes* [JAH18]. As diferentes granularidades de representação foram também exploradas pelo grupo no contexto de segmentação hierárquica de imagens [KH09].

Devido à quantidade de *pixels*, o custo computacional do processamento *pixel a pixel* é alto. A classificação de microrregiões pode ser uma opção interessante para se contornar o alto custo computacional associado à classificação de *pixels*, desde que o cálculo dessas microrregiões possa ser efetuada eficientemente. Microrregiões tais como *flat zones* [Soi03], bacias de captação do algoritmo *watershed* [Soi03], ou *superpixels* como os gerados pelo algoritmo SLIC e similares [ASS⁺12], possuem tempo de processamento aceitável e podem reduzir uma grande quantidade de *pixels* a umas centenas de microrregiões. Com isso, métodos que precisariam ser aplicados a cada *pixel* individualmente podem ser substituídos por métodos que processam microrregiões.

O **objetivo** deste projeto de pesquisa é a extensão do aprendizado de operadores de imagens [MHH16] de modo que sejam contemplados também os operadores que atuam em nível de componentes de maior granularidade que a dos *pixels*. O trabalho recentemente proposto pelo grupo, descrito em [JAMH17], alinha-se com esse objetivo. Neste projeto pretendemos ampliar

a diversidade de tipos de microregiões contemplando os citados acima e outros que se mostrarem interessantes. Além do desenvolvimento desses métodos, um objetivo secundário é a integração desses métodos ao TRIOSlib (<https://trioslib.github.io/>), uma biblioteca para aprendizado de transformações de imagens que vem sendo desenvolvido ao longo dos anos pelo grupo.

Seguindo esta introdução, o restante deste texto está organizado da seguinte forma. Na seção 2 fazemos uma breve introdução aos conceitos associados a transformações de imagens realizadas em nível de *pixel* e a formulação do problema de desenho de operadores de imagens como um problema de aprendizado de máquina. Na seção 3 descrevemos a proposta deste projeto de pesquisa, detalhando como pretendemos realizar a extensão do projeto de operadores para que eles atuem sobre regiões. Descrevemos também os materiais e métodos a serem empregados no desenvolvimento. Na seção 4 detalhamos as atividades a serem desenvolvidas, juntamente com um cronograma de execução. Por fim, na seção 5 discutimos as formas de avaliação dos resultados e apresentamos algumas considerações finais.

2 Background

Nesta seção introduzimos brevemente alguns conceitos, notações e resultados que fundamentam a proposta a ser apresentada na próxima seção.

Seja $E = \mathbb{Z}^2$ o domínio das imagens. Uma imagem definida sobre E pode ser representada por uma função $f : E \rightarrow \{0, 1, \dots, k-1\}^c$, na qual k é o número de intensidades possíveis em cada ponto da imagem e c é o número de canais (ou bandas) na imagem [Soi03]. Tipicamente temos $k = 256$ e, neste caso, se $c = 1$ temos as imagens em tons de cinza e se $c > 1$ temos as imagens multi-canais (imagens coloridas são tais que $c = 3$ e uma representação típica é o modelo RGB consistindo de um canal para R (vermelho), um para G (verde) e um terceiro para B (azul)). Quando $k = 2$ e $c = 1$ temos as imagens binárias.

O conjunto $\{0, 1, \dots, k-1\}$ será denotado por \mathbb{K} e o conjunto de todas as imagens de E em \mathbb{K} será denotado por \mathbb{K}^E . Transformações de imagens podem ser representadas por mapeamentos do tipo $\Psi : \mathbb{K}^E \rightarrow \mathbb{K}^E$, isto é, por operadores que levam imagens em imagens [Hei94].

Duas propriedades são importantes quando consideramos transformações de imagens, atuando na granularidade de *pixels*. A invariância à translação e a definição local. Dizemos que um operador de imagens Ψ é invariante à translação se $\Psi(f)_z = \Psi(f_z)$, para todo $z \in E$ e $f \in \mathbb{K}^E$. A notação f_z indica a imagem f transladada por z . O operador Ψ é localmente definido se existe um subconjunto não vazio $W \subseteq E$, tal que $[\Psi(f)](z) = [\Psi(f[W_z])](z)$, para todo $z \in E$ e $f \in \mathbb{K}^E$. A notação $f[W_z]$ denota a imagem f com o suporte restrito a W_z (o conjunto W transladado por z). Em outras palavras, o valor da imagem transformada $\Psi(f)$ em qualquer ponto z pode ser determinado considerando-se apenas a imagem restrita a uma pequena região

em torno de z .

Um operador de imagens que é invariante à translação e localmente definido pode ser caracterizado por uma função local [Hei94, MHH16], da seguinte forma:

$$[\Psi(f)](x) = \psi(f[W_x]_{-x}) \quad (1)$$

Nesta equação, ψ é uma função de W em \mathbb{K} e $f[W_x]_{-x}$ é a imagem f restrita ao suporte W_x e transladada por $-x$ para que seu suporte coincida com W . De forma informal, a equação acima significa que o valor da transformação da imagem f por Ψ no ponto x pode ser calculado aplicando-se uma função ψ sobre um recorte de tamanho W , da imagem f , centrado em x .

Os trabalhos existentes [MHH16, BDT97, Hir09] sobre aprendizado de operadores de imagens apoiam-se na formulação acima. A ideia central consiste em considerar a função ψ como sendo um classificador. A entrada desse classificador é um recorte da imagem com suporte igual a W , e a saída é a intensidade da imagem transformada no ponto no qual o recorte foi efetuado.

O treinamento de classificadores desse tipo poder ser realizado a partir de exemplos, isto é, pares (\mathbf{x}, y) de recorte e respectivo valor na saída, coletados de pares de imagens ilustrativos da transformação desejada (como o par mostrado na figura 1). Em princípio, não há restrições quanto ao tipo de algoritmos de classificação a serem usados. Já foram utilizados por exemplos as árvores de decisão [MHH14], SVM [MHHC16] e redes neurais convolucionais [JAH18]. Os métodos desenvolvidos encontram-se implementados na biblioteca TRIOSlib (<https://trioslib.github.io/>).

3 Descrição da proposta

Nesta seção detalhamos a abordagem proposta para o desenvolvimento do projeto de pesquisa.

Ao considerarmos a representação de imagens, a que considera como unidade elementar os pontos (ou *pixels*) é bastante conveniente pois uma imagem pode ser armazenada em uma estrutura do tipo matriz. Isto permite que pontos na imagem possam ser acessados individualmente. Além disso, a relação de vizinhança espacial entre pontos está implicitamente codificada na estrutura matricial. Porém, do ponto de vista de processamento computacional, transformações que operam sobre cada um dos *pixels* podem ser custosos uma vez que uma imagem pode conter milhares de *pixels*. Portanto, uma abordagem comum consiste em se alterar a granularidade de representação das imagens, passando-a de *pixels* para regiões. Com isso, as transformações de imagens podem ser modeladas para atuar sobre as regiões e não sobre os *pixels*, resultando em geral em uma enorme redução no tempo de processamento. Esta redução no tempo de processamento vem acompanhado, porém, de aumento no espaço utilizado para a representação

da imagem. Esse aumento deve-se a estruturas adicionais necessárias para se ter fácil acesso individual às regiões.

3.1 Pesquisa e desenvolvimento

A principal ideia a ser explorada neste projeto de pesquisa diz respeito à granularidade de representação das imagens no aprendizado de operadores de imagens. Pretendemos criar métodos de aprendizado de operadores de imagens que atuem em imagens representadas na granularidade de regiões. Em essência, quando consideramos uma representação em termos de regiões, podemos imaginar que o domínio de definição de imagens E é particionado, de acordo com alguma característica da imagem f , em múltiplas regiões. Por exemplo, SLIC [ASS⁺12] é um algoritmo que agrupa *pixels* e particiona o domínio da imagem em regiões denominados de *superpixels*. Um exemplo de resultados do SLIC e de outros algoritmos é mostrado na figura 2.

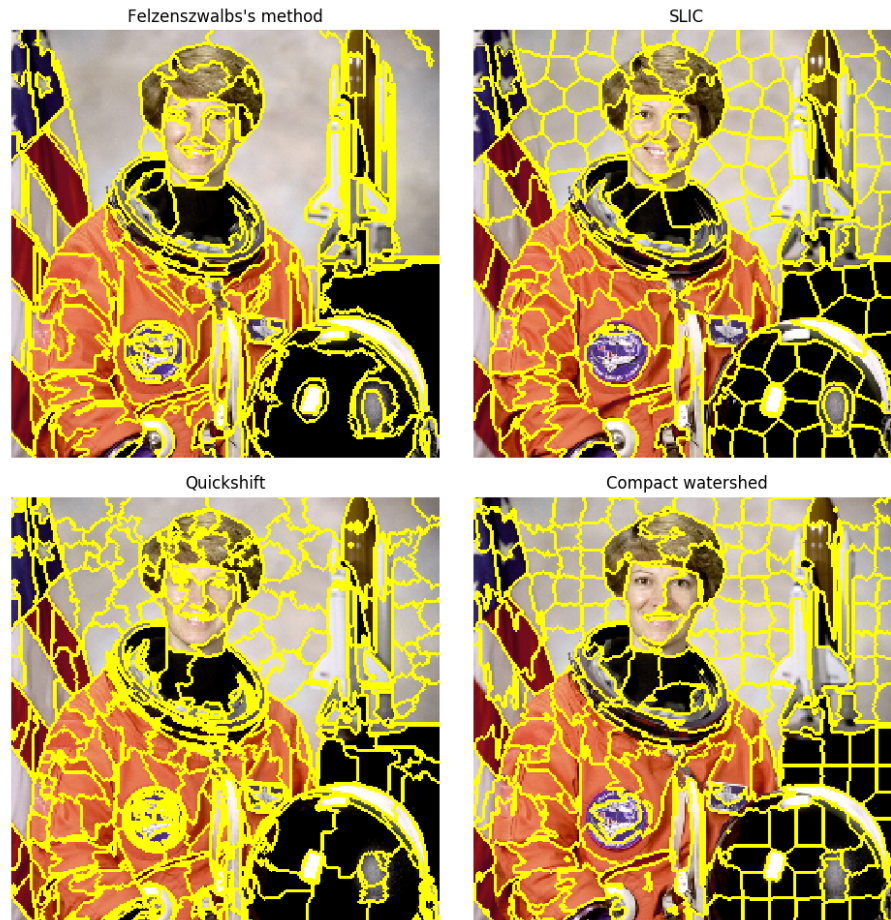


Figura 2: Exemplos de partição de imagens em microregiões (fonte: http://scikit-image.org/docs/stable/auto_examples/segmentation/plot_segmentations.html)

Mesmo considerando-se a representação de imagens em termos de componentes de maior

granularidade, podemos manter a possibilidade de se descrever a transformação entre imagens por uma função. A única diferença é que a função deverá ser aplicada sobre cada uma das regiões e não sobre cada *pixel*. Seja E_i uma região qualquer da imagem. Definimos:

$$[\Psi(f)](x) = \psi(f[W_{E_i}]), \forall x \in E_i \quad (2)$$

na qual $f[W_{E_i}]$ poderia ser um recorte da imagem f que engloba a região E_i . Neste caso, a função ψ calcula um valor de saída e esse valor é atribuído na imagem de saída a todos os pontos de E_i . Por exemplo, em [JAMH17] os componentes conexos foram considerados como tais regiões e os recortes retangulares realizados foram centrados no componente. O classificador utilizado foi um rede neural convolucional simples [Nie15, GBC16], para determinar se o componente tratava-se de um caractere (parte de região de texto) ou não.

Dada esta formulação, os pontos específicos que pretendemos investigar são:

1. **Diferentes formas para a partição de imagens em regiões:** dentre as formas a serem consideradas estão os componentes conexos [Soi03], os *flat zones* [Soi03], as bacias de captação resultantes da supersegmentação gerada pelo algoritmo de segmentação morfológica *watershed* [Soi03], algoritmos de cálculo de superpixels tais como o SLIC [ASS⁺12]. Nesta investigação deverão ser examinados o tempo de processamento, o número de componentes (regiões) gerados, e a qualidade das regiões quanto à aderência aos possíveis contornos de interesse.

A variedade de formas de se gerar as regiões é importante pois diferentes tipos de imagens possuem características distintas e não há garantias de que haja uma única forma de partição que seja adequada a todos os tipos de segmentações desejadas.

2. **Estrutura de dados para o armazenamento das regiões:** devemos considerar estruturas de dados que permitam rápido acesso à cada uma das regiões. Uma estrutura de dados geralmente usada quando há necessidade de processamento de regiões são os grafos. Será estudado a conveniência da utilização de grafos e possíveis estruturas alternativas ou complementares.
3. Dada uma partição da imagem, **diferentes formas de classificação de regiões:** adotando-se a abordagem tradicional de aprendizado de máquina [AMLM12], cada região pode ser caracterizada por um conjunto de características (*features*) e estas poderiam ser usadas pelo classificador. Adotando-se técnicas de aprendizado profundo [GBC16], um corte retangular centrado na região pode ser efetuado sobre cada região e este corte pode alimentar diretamente as redes neurais convolucionais, por exemplo. Neste caso, os principais aspectos a serem explorados seriam o tamanho do recorte, os diferentes tipos de normalização do recorte, ou diferentes combinações de canais da imagem.

4. **Implementação dos algoritmos e métodos na TRIOSlib:** a extensão desta biblioteca conferiria maior versatilidade à família de métodos implementados, bem como colaboraria positivamente para reprodutibilidade dos resultados e difusão de conhecimentos.

Uma vez que já existem várias bibliotecas de processamento de imagens de código aberto como o `scikit-image`, os códigos desses pacotes que implementam os algoritmos de partição de imagens em microrregiões poderão ser integrados diretamente à implementação por meio de chamada a código externo. Similarmente, o `scikit-learn` disponibiliza algoritmos de aprendizado de máquina que já vêm sendo utilizados junto ao TRIOSlib. Para a construção e manipulação de grafos, a opção considerada é `NetworkX`, uma biblioteca já utilizada pelo grupo. Com isso, espera-se que a maior parte do esforço de implementação seja direcionado ao *pipeline* de processamento e sua experimentação.

O *pipeline* de processamento deverá incluir a aplicação de um algoritmo de particionamento da imagem em microrregiões, a extração de amostras (pares (\mathbf{x}, y) a partir das imagens de treinamento), tarefa que pode incluir o cálculo e extração de *features* das regiões, organização desses dados em subconjuntos de treinamento e validação (a depender do algoritmo de treinamento a ser utilizado), o próprio treinamento em si, o armazenamento do resultado do treinamento (classificador), um módulo de aplicação de um classificador pré-treinado, e outro para cálculo de métricas de desempenho.

5. **Aplicação do método** em diferentes problemas de segmentação de imagens, acompanhada de avaliação da influência dos diferentes parâmetros (tipo de regiões, tamanho do recorte, algoritmo de aprendizado, *features* de região) sobre os resultados (qualidade da segmentação, tempo de processamento), e comparação dos resultados com os obtidos pelos métodos que atuam em nível de *pixels*.

3.2 Materiais e métodos

O projeto de pesquisa será desenvolvido juntamente ao laboratório de eScience do IME/USP. O laboratório disponibiliza a infra-estrutura computacional necessária para o andamento da pesquisa.

O desenvolvimento do projeto envolverá estudos de conceitos e fundamentos e uma parte prática que inclui implementação de algoritmos, desenho e execução de experimentos computacionais, e avaliação de resultados. Os estudos serão baseados em livros disponíveis na biblioteca do instituto ou materiais disponíveis publicamente na *web*. Todos os códigos desenvolvidos serão abertos (*open-source*) e as bibliotecas que pretendemos utilizar são todas públicas.

Os progressos deverão ser relatados e discutidos em reuniões de grupo que ocorrem semanalmente. Nessas reuniões, o candidato terá oportunidades de interagir e aprender com os outros

estudantes, de graduação e pós-graduação. Havendo necessidade, reuniões específicas entre estudante e orientadora poderão também ser realizadas.

Fase de estudos preliminares: O plano de desenvolvimento do projeto contempla uma fase inicial de estudos preliminares. O propósito desses estudos preliminares é a aquisição de conceitos e fundamentos básicos e familiarização com as práticas comuns nas áreas de processamento de imagens e de aprendizado de máquina. Esses estudos não visam uma formação completa, mas uma que seja suficiente para o estudante iniciar o desenvolvimento da proposta. Durante o andamento, estudos aprofundados sobre tópicos específicos serão realizados conforme a necessidade. Na fase de estudo preliminar, parte das seguintes referências serão estudadas:

- **Processamento de imagens**

- E. R. Dougherty and R. A. Lotufo, *Hands-on Morphological Image Processing*, 2003 [DL03]
- P. Soille, *Morphological Image Analysis*, 2003 [Soi03]

Estes dois livros introduzem a Morfologia Matemática, um ramo da área de processamento de imagens. A introdução aos conceitos e fundamentos de processamento de imagens por meio da Morfologia Matemática é interessante, pois ela explora fortemente as características geométricas e topológicas presentes nas imagens, favorecendo uma interpretação bastante intuitiva de seu efeito quando os processamentos são aplicados sobre uma imagem. Do ponto de vista formal, a Morfologia Matemática apoia-se na teoria de reticulados completos, que provê uma sólida fundamentação matemática e garante várias propriedades interessantes relacionadas à representação dos operadores de imagens [Hei94]. Conceitos como componentes conexos, *flat zones* e *watershed* que são elementos centrais nesta proposta estão presentes nessas referências.

- **Aprendizado de máquina**

- *Learning from data* [AMLMI12] juntamente com o material complementar disponível online em <https://work.caltech.edu/telecourse.html>.
- *Neural Networks and Deep Learning*, Michael Nielsen (<http://neuralnetworksanddeeplearning.com/>)

O primeiro livro será utilizado para se adquirir uma noção geral sobre o que é aprendizado de máquina, e um entendimento mais aprofundado sobre modelos lineares. O segundo livro será útil para iniciar a transição para modelos não lineares e, mais especificamente, para redes neurais, que posteriormente deverão evoluir até as redes convolucionais.

Aspectos práticos de processamento de imagens serão exercitados por meio da implementação de algoritmos básicos como erosão e dilatação e por meio de experimentação de bibliotecas públicas tais como `scikit-image` (<http://scikit-image.org/>) e `Pyhton morphology toolbox` (<https://github.com/robertoalotufo/ia870>). Com relação ao aprendizado de máquina, serão utilizados o material produzido na disciplina de aprendizado de máquina do IME/USP (<https://github.com/MLIME/MAC0460>) e a biblioteca `scikit-learn` (<http://scikit-learn.org/stable/>). Paralelamente será realizada a familiarização com o TRIOSlib. Para tal, o estudante poderá contar com a ajuda dos demais membros do grupo.

Fase de desenvolvimento: Nesta fase serão realizadas as atividades que visam a execução da pesquisa e desenvolvimento descritos acima. As principais atividades planejadas estão listadas na próxima seção, juntamente com um cronograma. As reuniões semanais serão úteis para o acompanhamento do progresso e para eventuais ajustes no plano de trabalho.

4 Plano de trabalho e cronograma

De acordo com a proposta descrita acima, as atividades podem ser agrupadas em três grupos principais:

1. Estudos preliminares

O estudante vem desenvolvendo atividades de IC desde o início de ano. Portanto, os estudos preliminares referentes a processamento de imagens já foram realizadas.

Além disso, ele vem também participando das reuniões semanais do grupo.

Em breve, o aluno deverá iniciar os estudos preliminares referentes a aprendizado de máquina e buscar familiarização com a biblioteca TRIOSlib.

2. Desenvolvimento dos métodos

- (a) Estudo (e eventual implementação) de algoritmos para particionamento de imagens, bem como a melhor forma de representação das regiões resultantes (considerando-se a necessidade de acesso eficiente posterior a essas regiões)
- (b) Desenvolvimento do método de treinamento sobre regiões. Esta atividade envolve desde a preparação de dados, o treinamento, e a geração de classificadores
- (c) Integração desses algoritmos e métodos na TRIOSlib. As técnicas de criação de regiões, assim como de classificação de regiões, deverão ser integradas ao TRIOSlib na forma de um *pipeline* completo e testado.

3. Aplicação e avaliação dos métodos. O *pipeline* desenvolvido deverá ser aplicado sobre diferentes conjuntos de imagens. Os resultados obtidos deverão ser analisados.
4. Documentação. Ao final deverá ser gerado um relatório científico. Além disso, códigos gerados serão documentados. Caso pertinente, poderão ser elaborados artigos descrevendo os métodos e os resultados.

Um cronograma proposto para essas atividades é exibido a seguir.

Atividade	Mês											
	1	2	3	4	5	6	7	8	9	10	11	12
1. Estudos preliminares	x	x	x									
2. Desenvolvimento dos métodos		x	x	x	x	x	x	x	x	x		
2.a Criação de regiões		x	x	x	x							
2.b Desenvolvimento do método de treinamento				x	x	x	x	x				
2.c Integração ao TRIOSlib						x	x	x	x	x		
4. Aplicação e avaliação									x	x	x	x
5. Relatório final / Artigos										x	x	x

5 Comentários finais

Forma de análise dos resultados: Os resultados deste projeto de pesquisa poderão ser quantificados em termos de número de algoritmos de partição de imagens estudados e/ou implementados e utilizados, ou por medidas relativas ao desenvolvimento de código (que estarão registrados no repositório `git`), ou número de aplicações efetivamente realizadas, e quantidade de registros documentais (manual, relatório científico, artigos) gerados. Além disso, qualitativamente, podemos avaliar se a integração dos métodos ao TRIOSlib ocorreu de forma satisfatória, se houve impacto sobre outros projetos realizados pelo grupo, e se os resultados das aplicações produziram resultados qualitativamente interessantes.

Observações finais: O estudante já realizou boa parte dos estudos preliminares relacionados ao processamento de imagens. Também está iniciando a familiarização com o TRIOSlib e deverá em breve iniciar os estudos referentes a aprendizado de máquina.

Referências

- [AMLM12] Yaser S. Abu-Mostafa, Hsuan-Tien Lin, and Malik Magdon-Ismail. *Learning From Data*. AMLBook, 2012. 6, 8

- [ASS⁺12] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, November 2012. 2, 5, 6
- [BDT97] J. Barrera, E. R. Dougherty, and N. S. Tomita. Automatic Programming of Binary Morphological Machines by Design of Statistically Optimal Operators in the Context of Computational Learning Theory. *Electronic Imaging*, 6(1):54–67, 1997. 4
- [DL03] E. R. Dougherty and R. A. Lotufo. *Hands-on Morphological Image Processing*. SPIE Press, 2003. 8
- [GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>. 6
- [GW02] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley Publishing Company, second edition, 2002. 1
- [Hei94] H. J. A. M. Heijmans. *Morphological Image Operators*. Academic Press, Boston, 1994. 3, 4, 8
- [Hir09] Nina S. T. Hirata. Multilevel training of binary morphological operators. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4):707–720, April 2009. 4
- [JAH18] Frank D. Julca-Aguilar and N. S. T. Hirata. Symbol detection in online handwritten graphics using Faster R-CNN. In *Proceedings of DAS*, 2018. 2, 4
- [JAMH17] F. D. Julca-Aguilar, Ana L. M. Maia, and N. S. T. Hirata. Text/non-text classification of connected components in document images. In *30th Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 450–455, 2017. 2, 6
- [KH09] B. Klava and N. S. T. Hirata. Interactive image segmentation with integrated use of the markers and the hierarchical watershed approaches. In *Proceedings of the Fourth International Conference on Computer Vision Theory and Applications*, pages 186–193, 2009. 2
- [MHH14] I. S. Montagner, R. Hirata Jr., and N. S. T. Hirata. A machine learning based method for staff removal. In *22nd International Conference on Pattern Recognition (ICPR)*, pages 3162 – 3167, 2014. 4
- [MHH16] Igor S. Montagner, Nina S. T. Hirata, and R. Hirata Jr. Image operator learning and applications. In *Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, pages 38–50, 2016. 1, 2, 4
- [MHHC16] Igor S. Montagner, Roberto Hirata Jr, Nina S. T. Hirata, and Stéphane Canu. Kernel approximations for w-operator learning. In *Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 386–393, 2016. 4
- [Nie15] Michael A. Nielsen. *Neural Networks and Deep Learning*. Determination Press, 2015. 6
- [Pri12] Simon D. J. Prince. *Computer Vision – Models, Learning and Inference*. Cambridge, 2012. 1
- [Soi03] P. Soille. *Morphological Image Analysis*. Springer-Verlag, Berlin, 2nd edition, 2003. 2, 3, 6, 8
- [Sze11] R. Szeliski. *Computer Vision – Algorithms and Applications*. Springer, 2011. 1