UNIVERSIDADE FEDERAL DE OURO PRETO

Segmentação de núcleos em células cervicais obtidas em exames de Papanicolaou

Débora Nasser Diniz
Universidade Federal de Ouro Preto

Orientador: Dr. Marcone Jamilson Freitas Souza

Coorientadora: Dra. Andrea Gomes Campos Bianchi

Dissertação de Mestrado submetida ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Ouro Preto como requisito para a obtenção do título de Mestre em Ciência da Computação.

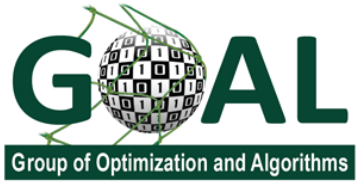
Ouro Preto, Agosto de 2019

Segmentação de núcleos em células cervicais obtidas em exames de Papanicolaou

Débora Nasser Diniz
Universidade Federal de Ouro Preto

Orientador: Dr. Marcone Jamilson Freitas Souza

Coorientadora: Dra. Andrea Gomes Campos Bianchi



D585s Diniz, Débora Nasser.
Segmentação de núcleos em células cervicais obtidas em exames de Papanicolaou [manuscrito] / Débora Nasser Diniz. - 2019.
61f.: il.: color; tabs..

Orientador: Prof. Dr. Marcone Jamilson Freitas Souza.
Coorientadora: Prof.ª Drª. Andrea Gomes Campos Bianchi.

Dissertação (Mestrado) - Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Biológicas. Departamento de Computação. Programa de Pós-Graduação em Ciência da Computação.
Área de Concentração: Ciência da Computação.
1. Segmentação de núcleos. 2. Células cervicais. 3. Árvore de decisão I. Souza, Marcone Jamilson Freitas. II. Bianchi, Andrea Gomes Campos. III. Universidade Federal de Ouro Preto. IV. Titulo.
CDU: 618.14-047.38

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

ATA DE DEFESA DE DISSERTAÇÃO

Aos 09 dias do mês de agosto do ano de 2019, às 14:00 horas, nas dependências do Departamento de Computação (Decom), foi instalada a sessão pública para a defesa de dissertação da mestranda **Debora Nasser Diniz**, sendo a banca examinadora composta pelo Prof. Dr. Marcone Jamilson Freitas Souza (Presidente - UFOP), pela Fatima Nelsizeuma Sombra de Medeiros (Membro - Externo), pelo Prof. Puca Huachi Vaz Pena (Membro - UFOP) e pela Profa. Dra. Andrea Gomes Campos Bianchi (Co-Orientadora - UFOP). Dando início aos trabalhos, o presidente, com base no regulamento do curso e nas normas que regem as sessões de defesa de dissertação, concedeu à mestranda 50 minutos para apresentação do seu trabalho intitulado "Segmentação de Núcleos em Células Cervicais Obtidas em Exames de Papanicolaou". Terminada a exposição, o presidente da banca examinadora concedeu, a cada membro, um tempo máximo de 60 minutos para perguntas e respostas à candidata sobre o conteúdo da dissertação, na seguinte ordem: Primeiro, Profa. Fatima Nelsizeuma Sombra de Medeiros; segundo, Prof. Puca Huachi Vaz Pena; terceiro, Profa. Andrea Gomes Campos Bianchi; quarto, Prof. Marcone Jamilson Freitas Souza. Dando continuidade, ainda de acordo com as normas que regem a sessão, o presidente solicitou aos presentes que se retirassem do recinto para que a banca examinadora procedesse à análise e decisão, anunciando, a seguir, publicamente, que a mestranda foi aprovada por unanimidade, sob a condição de que a versão definitiva da dissertação deva incorporar todas as exigências da banca, devendo o exemplar final ser entregue no prazo máximo de 60 (sessenta) dias à Coordenação do Programa. Para constar, foi lavrada a presente ata que, após aprovada, vai assinada pelos membros da banca examinadora e pela mestranda. Ouro Preto, 09 de agosto de 2019.

Prof. Dr. Marcone Jamilson Freitas Souza
Presidente

Profa. Dra. Andrea Gomes Campos Bianchi

Fatima Nelsizeuma Sombra de Medeiros
(Participação por Videoconferência)

Prof. Puca Huachi Vaz Pena

Mestranda

Certifico que a defesa realizou-se com a participação a distância do(s) membros(s) Fatima Nelsizeuma Sombra de Medeiros e que, depois das arguições e deliberações realizadas, cada participante a distância afirmou estar de acordo com o conteúdo do parecer da banca examinadora, redigido nesta ata.

Prof. Dr. Marcone Jamilson Freitas Souza
Presidente

Resumo

Este trabalho tem seu foco na detecção de núcleos em imagens sintéticas de células cervicais. Este é um passo importante na construção de uma ferramenta computacional para ajudar os citopatologistas a identificarem alterações celulares a partir de exames de Papanicolaou. Para detectar esses núcleos propomos duas abordagens, a primeira baseada em *Iterated Local Search* (ILS) e a segunda em *Árvore de Decisão* (DT). O objetivo é melhorar a assertividade do exame e reduzir a carga de trabalho do profissional. As duas abordagens utilizam características de uma região da imagem para identificar um núcleo. Para ambas, foi necessário fazer um pré-processamento das imagens para dividí-las em regiões a serem analisadas. Para isto, foram utilizados os algoritmos *Simple Linear Iterative Clustering* (SLIC) e *Density Based Spatial Clustering of Applications with Noise* (DBSCAN). No ILS, foi feita uma investigação para saber quais dessas características são relevantes para a identificação dos núcleos. O pacote *irace* foi utilizado para fazer a calibração automática dos parâmetros do ILS. Já para a DT proposta, foi construída uma base de dados com todas as características extraídas das regiões e feita uma seleção das mais importantes por meio de uma matriz de correlação. Com essas características selecionadas foi feito o treinamento. Por fim, as abordagens propostas foram comparadas entre si e com outros métodos da literatura segundo as métricas revocação, precisão e F1, usando-se o banco de dados *ISBI Overlapping Cytology Image Segmentation Challenge* (2014). Os resultados obtidos mostraram a superioridade da abordagem via DT sobre o ILS em todas as métricas, assim como sua superioridade sobre todos os outros métodos da literatura com relação às métricas F1 e revocação.

Palavras-chave: Segmentação de Núcleos, Células Cervicais, *Iterated Local Search*, Meta-Heurística, Superpixel, Árvore de Decisão, Método Estatístico, *Simple Linear Iterative Clustering*, *Density Based Spatial Clustering of Applications with Noise*.

Abstract

This work focuses on the detection of nuclei in synthetic images of cervical cells. Finding nuclei is an important step in building a computational tool to help cytopathologists identify cell changes from Pap smears. To detect these nuclei, we propose two approaches, the first based on Iterated Local Search (ILS) and the second on Decision Tree (DT). The tool aims to improve the assertiveness of the exam and reduce the workload of the professional. Both approaches used characteristics of a region to identify a nucleus. For both, it was necessary to preprocess the images to divide them into regions to be analyzed. For this, the Simple Linear Iterative Clustering (SLIC) and Density Based Spatial Clustering of Applications with Noise (DBSCAN) algorithms were used. In ILS, an investigation was made to know which of these characteristics are relevant for nucleus identification. The irace package was used to calibrate ILS parameters automatically. Besides, for the DT model, this work extracted several features from the regions in order to select the best characteristics that represent a nucleus region. The selection was made utilizing a correlation matrix, and the chosen features were used to train the model. Finally, the proposed approaches were compared with each other and with other methods in the literature according to recall, precision, and F1 metrics using the ISBI Overlapping Cytology Image Segmentation Challenge database (2014). The results showed the superiority of the DT approach over the ILS in all metrics. Also, the DT model presented better results for F1 and recall metrics over all other literature methods.

Keywords: *Nuclei Segmentation, Cervical Cells, Iterated Local Search, Metaheuristic, Superpixel, Decision Tree, Statistical Approach.*

Declaração

Esta dissertação é resultado de meu próprio trabalho, exceto onde uma referência explícita é feita ao trabalho de outros, e não foi submetida para outra defesa nesta nem em outra universidade.

Débora Nasser Diniz

Agradecimentos

Tive o apoio de pessoas muito importantes, dentre as quais agradeço especialmente:

À minha mãe, Norma, meu maior exemplo, por ser uma pessoa maravilhosa, dedicada e que sempre me acolheu com muito carinho e palavras de apoio.

Ao meu namorado, Rafael, por estar sempre presente me dando muito amor, incentivando meus estudos e por ter participado ativamente na DT proposta neste trabalho.

Aos meus orientadores, Marcone e Andrea, por todo o empenho, paciência, horas de dedicação e por estarem sempre dispostos a ajudar e contribuir com o meu aprendizado.

Aos meus amigos da Computação, em especial ao Breno, pela companhia, amizade e apoio para enfrentar os desafios acadêmicos.

À todos os meus amigos e familiares, em especial a minha prima, Alice, e as "novinhas", que torceram por mim e me acompanharam nessa jornada, sempre me apoiando.

À todos os integrantes do grupo de estudos CRIC, em especial à Cláudia e Mariana, por todo o conhecimento compartilhado, opiniões e apoio para a realização deste trabalho.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), pelo apoio financeiro dado à este trabalho (Código de Financiamento 001).

À Universidade Federal de Ouro Preto (UFOP), à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) por todo o suporte prestado.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Nomenclatura

AIDS	*Acquired Immunodeficiency Syndrome* / Síndrome da Imunodeficiência Adquirida
Area	Número de *pixels* de um *cluster*
AG	Algoritmo Genético
AM	Algoritmo Memético
BBoxArea	Número total de pixels de um *bounding box*
BMJ	*British Medical Journal*
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CIA	Circularidade, Intensidade e Área
CIE	Comissão Internacional de Iluminação
Circ	Indica o quão circular é um *cluster*
CNN	Rede Neural Convolucional / *Convolutional Neural Network*
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
ConvexArea	Número de *pixels* do menor polígono convexo que envolve um *cluster*
CRIC	*Center of Recognition and Inspection of Cells*
CSF	*Circular Shape Function*
DBSCAN	*Density Based Spatial Clustering of Applications with Noise*
Diameter	Diâmetro de um círculo que tem a mesma área que um *cluster*
DRLSE	*Distance Regularized Level Set Evolution*
DT	Árvore de Decisão / *Decision Tree*

EDbscan	Limite de valor/distância de tolerância correspondente no DBSCAN
Euler	Característica de Euler de um *cluster*
Excent	Excentricidade de uma elipse que engloba um *cluster*
Extent	Proporção do número de *pixels* de um *cluster* sobre os do *bounding box*
FAPEMIG	Fundação de Amparo à Pesquisa do Estado de Minas Gerais
FCM	*Fuzzy C-Means Clustering*
FilledArea	Número de *pixels* preenchidos de um *cluster*
FN	Falso Negativo
FP	Falso Positivo
GRASP	*Greedy Randomized Adaptive Search Procedure*
HIV	Vírus da Imunodeficiência Humana / *Human Immunodeficiency Virus*
ICESP	Instituto Nacional do Câncer do Estado de São Paulo
ILS	Busca Local Iterada / *Iterated Local Search*
ILSMax	Número máximo de iterações sem melhora do ILS
INCA	Instituto Nacional do Câncer
IntMax	Intensidade máxima dos *pixels* de um *cluster*
IntMed	Intensidade média dos *pixels* de um *cluster*
IntMin	Intensidade mínima dos *pixels* de um *cluster*
IRS	*Indian Resource Satellite*
ISBI	Simpósio Internacional de Imagem Biomédica
kSlic	Número de *superpixels* desejados no SLIC
LISS III	*Linear Imaging Self-Scanner*
MajorAxis	Comprimento do maior eixo da elipse que engloba um *cluster*
MinorAxis	Comprimento do menor eixo da elipse que engloba um *cluster*
MSER	*Maximally Stable Extremal Region*
mSlic	Fator de ponderação entre as diferenças de intensidade e distância no SLIC

nSolRand	Número de soluções iniciais geradas aleatoriamente no ILS
nStart	Número de reinícios do *Multi-Start* ILS
OMS	Organização Mundial de Saúde
prec	Precisão / *Precision*
r_{dec}	Passo definido para os parâmetros decimais no ILS
r_{int}	Passo definido para os parâmetros inteiros no ILS
rec	Revocação / *Recall*
RF	*Random Forest*
seRadiusSlic	Limite de tamanho considerado para regiões no SLIC
SLIC	*Simple Linear Iterative Clustering*
Solidity	Proporção do número de *pixels* de um *cluster* sobre os do menor polígono convexo que o envolve
SVM	Máquina de Vetores de Suporte / *Support Vector Machine*
TP	Verdadeiro Positivo
UFOP	Universidade Federal de Ouro Preto
VND	Descida em Vizinhança Variável / *Variable Neighbourhood Descent*

Capítulo 1

Introdução

Segundo a Organização Mundial de Saúde (OMS), 14.1 milhões de novos casos de câncer são diagnosticados a cada ano no mundo inteiro. Além disso, uma estimativa é que esta é a causa de 13% de todas as mortes no mundo por ano, o que corresponde a 8,2 milhões de pessoas. A OMS informa ainda que, segundo cientistas, a previsão é que o número de casos de câncer aumente 70% nas próximas décadas, chegando a 21,4 milhões em 2032.

No Brasil, segundo o Ministério da Saúde, o Instituto Nacional do Câncer (INCA) e o Instituto do Câncer do Estado de São Paulo (ICESP), o terceiro tipo de câncer que mais mata mulheres é o câncer do colo do útero, e este número aumentou 28,6% nos últimos 10 anos.

Uma forma de prevenção utilizada no Brasil é a realização periódica do exame de Papanicolaou a fim de rastrear alterações nas células do colo do útero. A necessidade desta periodicidade é apresentada em um estudo publicado na edição online do *British Medical Journal* (BMJ) o qual mostrou que pacientes que descobriram a doença através do exame de Papanicolaou tiveram uma taxa de sobrevivência de 92%, enquanto que as que descobriram através dos sintomas tiveram uma taxa de 66%. Isto ocorre porque os sintomas só começam a aparecer quando a doença já está em um estágio mais avançado, enquanto que o exame permite detectar as lesões precocemente, ainda no início da doença, facilitando o tratamento.

Existem dois modos de se realizar o exame de Papanicolaou. No primeiro deles, inicialmente é realizada a coleta do material. Para isto, é introduzido na vagina um instrumento chamado espéculo. O profissional realiza uma inspeção visual do interior

da vagina e coleta com uma espátula de madeira amostras celulares da superfície externa e com uma escova de nylon da superfície interna do colo do útero. As células colhidas são colocadas em uma lâmina de vidro, sendo denominado esfregaço citológico, e este é enviado para análise em laboratórios especializados em citopatologia. O segundo modo inicia-se de maneira análoga ao modo anterior, utilizando-se apenas a escova de nylon, mas antes de colocar as amostras celulares em uma lâmina, elas são colocadas em um meio líquido que é utilizado para remover outros elementos, tais como muco, bactérias e hemácias, a fim de diminuir elementos da flora e melhorar a visibilidade das células, uma vez que diminui a sobreposição. Apesar do uso do meio líquido concentrar as células em áreas menores e melhorar a qualidade das imagens (agilizando a leitura manual), este método possui um custo muito elevado, fato que determina a sua menor utilização.

Uma lâmina com as amostras celulares coletadas no exame possui em torno de 15000 imagens para serem analisadas. Uma vez que esta análise é realizada manualmente por profissionais capacitados, o volume de dados é extenso. Assim, surgem dificuldades para a realização do mesmo, como por exemplo a fadiga física e mental dos profissionais. Além disso, o procedimento requer grande conhecimento técnico por parte do profissional, o que reduz o número de pessoas aptas a realizá-lo e encarece o custo da mão de obra.

O exame de Papanicolaou é interpretativo e depende da experiência do citopatologista. Assim, busca-se auxiliar esses profissionais a realizarem a análise da lâmina. Consequentemente, busca-se também reduzir o número de falsos positivos (casos em que o patologista detecta uma lesão falsa) e os falsos negativos (casos em que uma lesão celular não é detectada), uma vez que isso interfere no desempenho físico e/ou na saúde psicológica de um paciente. Por estes motivos, surge a importância de buscar metodologias que proporcionem melhorias na qualidade do resultado de um exame.

Dessa forma, o primeiro passo para identificar se uma célula possui alterações malignas é a detecção e a segmentação de seus núcleos, uma vez que as características morfológicas e texturais do núcleo apresentam variações significativas quando estão alteradas (Moshavegh et al., 2012; Samsudin et al., 2016). Citopatologistas identificam uma lesão quando nota-se uma alteração na relação núcleo/citoplasma, acompanhada por alterações na distribuição da cromatina, hipercromasia e formato da membrana nuclear. Por outro lado, os cientistas da computação levantam a hipótese de que apenas a identificação da irregularidade nuclear, a diferença de textura e a hipercromasia, ou a condensação irregular da cromatina, seriam suficientes para identificar uma célula suspeita. Segundo Plissiti et al. (2011), o que pode acontecer, por exemplo, é um aumento no tamanho do núcleo, a irregularidade de sua forma de ácido nucleico, diferença de

textura e hipercromasia, ou condensação irregular da cromatina.

Sendo assim, a ideia deste trabalho é propor e estudar novos métodos para segmentação de núcleos em células cervicais, uma vez que ainda há como melhorar os métodos existentes pois não foi encontrada uma solução satisfatória para o problema. Como já dito anteriormente, este é apenas o primeiro passo para a automatização do exame. Caso uma boa solução seja encontrada, será necessário fazer um novo estudo para a classificação destas células.

Este Capítulo encontra-se organizado como se segue. A Seção 1.1 descreve os objetivos geral e específicos deste trabalho. A Seção 1.2 e 1.3 apresentam sua motivação e limitações, respectivamente. A Seção 1.4 aborda a metodologia da pesquisa adotada. E, por fim, a Seção 1.5 apresenta o delineamento do restante do trabalho.

1.1 Objetivos

Esta seção tem como finalidade apresentar os objetivos deste trabalho. A Subseção 1.1.1 aponta o seu objetivo geral e a Subseção 1.1.2 os objetivos específicos.

1.1.1 Objetivo Geral

Este trabalho tem como objetivo principal desenvolver e validar métodos de segmentação de núcleos em células cervicais obtidas em exames de Papanicolaou.

1.1.2 Objetivos Específicos

Os objetivos específicos deste trabalho a serem atingidos são:

- utilização de métodos de otimização e aprendizagem supervisionada para a identificação de regiões que contêm núcleo;
- realização de experimentos de validação dos métodos propostos;
- comparação com métodos da literatura que utilizaram a base do desafio *Overlapping Cervical Cytology Image Segmentation Challenge*, proposto pelo Simpósio

Internacional de Imagem Biomédica (ISBI, das iniciais em inglês *International Symposium on Biomedical Imaging*) em 2014.

1.2 Justificativa

A interpretação visual de uma lâmina coletada em um Exame de Papanicolaou é extensa, subjetiva e necessita de conhecimento técnico por parte do profissional. Zhong e Najarian (2001) retratam que a análise realizada manualmente sofre de uma taxa de falsos negativos, estando ela entre 5% e 55%. Este fato é preocupante, uma vez que pode afetar a saúde psicológica e/ou física da mulher, além de não indicar a necessidade do tratamento da doença, o que implica em seu avanço e agravamento.

Dessa forma, uma ferramenta que faça a triagem das células coletadas no exame pode minimizar esta taxa de falsos negativos, uma vez que este equívoco pode estar relacionado à fadiga devido ao volume extenso de dados a serem analisados.

Outra vantagem associada ao uso de uma ferramenta de auxílio à análise de uma lâmina seria a redução do tempo necessário gasto para a análise.

1.3 Limitação

A limitação desse trabalho está no fato de que ele não trata bases reais. O estudo aqui presente se limitou a analisar bases sintéticas que foram disponibilizadas no desafio *Overlapping Cervical Cytology Image Segmentation Challenge* ocorrido no ISBI de 2014.

1.4 Metodologia da Pesquisa

Aqui, serão apresentadas as etapas que foram realizadas em busca da obtenção dos objetivos propostos. São elas:

- Revisão de Literatura: tem como objetivo analisar os trabalhos existentes que abordam a segmentação de núcleos em imagens celulares e seus respectivos resultados. Aborda também conceitos essenciais para o entendimento deste trabalho;

- Proposta de Solução: para solução do problema apresentado se propõe um pré-processamento para as imagens e, posteriormente, dois algoritmos para determinar se uma região é um núcleo, sendo os algoritmos baseados em: (i) uma meta-heurística de busca local e (ii) em modelos estatísticos que utilizam um treinamento supervisionado para a classificação;

- Validação da Solução Proposta: as soluções encontradas pelos dois algoritmos propostos foram validadas por meio de suas comparações com o *ground truth* proveniente da base de dados utilizada para testá-los;

- Avaliação dos resultados experimentais: análise das soluções obtidas com uso de inferência estatística na comparação a ser feita com diferentes técnicas já conhecidas na literatura.

1.5 Organização do Texto

O restante deste documento encontra-se organizado como segue. Os Capítulos 2 e 3 apresentam o referencial teórico e os trabalhos diretamente relacionados ao tema de pesquisa, respectivamente. O Capítulo 4 contém a descrição do problema a ser resolvido. O Capítulo 5 aponta o pré-processamento realizado e as abordagens heurística e estatística propostas neste trabalho. O Capítulo 6 apresenta os resultados dos experimentos realizados, bem como faz uma análise e discussão dos resultados obtidos. Por fim, o Capítulo 7 conclui o trabalho e aponta perspectivas de trabalhos futuros.

Capítulo 2

Referencial Teórico

Durante o desenvolvimento da proposta de solução para o problema tratado nesta dissertação, várias abordagens foram utilizadas e testadas, e este capítulo apresenta o referencial teórico útil ao entendimento das abordagens utilizadas. Na Seção 2.1 detalha-se a realização do exame de Papanicolaou. Nas Seções 2.2 e 2.3 são apresentados os algoritmos de segmentação SLIC e DBSCAN, respectivamente. As Seções 2.4 e 2.5 abordam os algoritmos ILS e *Multi-Start*. A Seção 2.6 expõe o método de classificação via DT e, por fim, a Seção 2.7 apresenta as medidas de avaliação que foram utilizadas nas abordagens.

2.1 O Exame de Papanicolaou

O exame de Papanicolaou (Traut e Papanicolaou, 1943) é um procedimento simples, rápido e de baixo custo, fatos estes que estão relacionados ao seu uso generalizado. A Figura 2.1 apresenta como é a realização deste exame. Primeiramente, um instrumento (em cinza na figura) chamado espéculo é introduzido no canal vaginal. No passo 2, são coletadas amostras celulares do colo do útero com uma espátula de madeira. Por fim, no passo 3, estas células coletadas são colocadas em uma lâmina, que é enviada para um laboratório especializado em citopatologia para serem analisadas.

A importância da realização deste exame acontece porque ele permite um diagnóstico no estágio inicial da doença (quando ela ainda é assintomática), o que aumenta consideravelmente a chance de cura. Mammas e Spandidos (2012) apresentaram que o uso do exame de Papanicolaou reduziu em 70% a mortalidade causada pelo câncer de colo

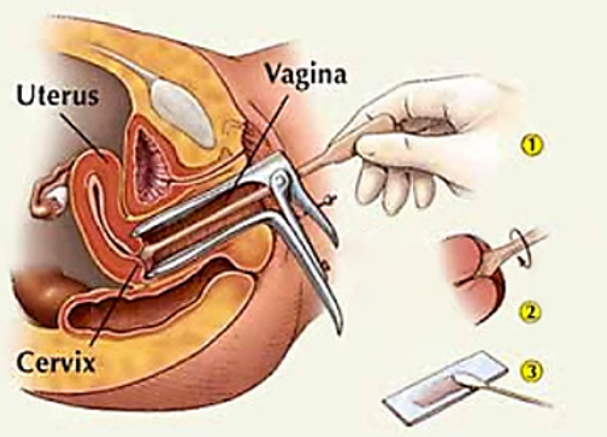


Figura 2.1: Como é feito o exame de Papanicolaou.

de útero nos últimos 60 anos.

A recomendação do Ministério da Saúde é de que mulheres de 25 a 64 anos e as que possuem uma vida sexual ativa devem fazer o exame periodicamente. Esta recomendação se dá pelo fato dessas mulheres estarem sujeitas à uma maior ocorrência de lesões de alto grau. Além disso, nos casos de mulheres de até 25 anos, a probabilidade de uma regressão natural das lesões é bastante alta.

2.2 Simple Linear Iterative Clustering (SLIC)

Simple Linear Iterative Clustering (SLIC) é um algoritmo de segmentação proposto por Kovesi (2000), cuja ideia é gerar *superpixels* por meio da clusterização dos *pixels* baseados na similaridade de suas cores e nas suas proximidades dentro da imagem. O algoritmo foi baseado no método *k-means* (Duda et al., 2000; MacQueen, 1967) e considera um espaço de cinco dimensões $[labxy]$, sendo que l, a e b são valores do espaço de cores CIELAB[1] e x e y são as coordenadas dos *pixels*.

Considerando uma imagem com N *pixels* e um parâmetro de entrada K indicando aproximadamente a quantidade de *superpixels* desejada, tem-se que o tamanho de cada um deles gerado pelo *SLIC* será de N/K *pixels*. Além disso, para que todos tenham aproximadamente o mesmo tamanho, é colocado um centro $C_k = [l_k, a_k, b_k, x_k, y_k]^T$, onde k varia de 1 até K, a cada distância $S = \sqrt{N/K}$ *pixels*.

[1]Espaço de cores definido pela Comissão Internacional de Iluminação (CIE) em 1976 (McGuire, 1992)

Uma vez que a extensão espacial de qualquer *superpixel* é aproximadamente $S \times S$, sabemos que os *pixels* associados ao centro estão dentro de uma área $2S \times 2S$ ao seu redor no plano xy, que é a área de pesquisa para o agrupamento.

O Algoritmo 2.1 mostra o pseudocódigo do processo de geração dos *superpixels* pelo SLIC.

Algoritmo 2.1: SLIC

```
1  Inicialize os centros de clusters C_k = [l_k, a_k, b_k, x_k, y_k]^T por amostragem de pixels a distancia S.
2  Mova os centros de cluster para a posição de gradiente mais baixa em uma vizinhança de 3 x 3.
3  Defina label l(i) = −1 para cada pixel i.
4  Defina a distância d(i) = ∞ para cada pixel i.
5  repita
6      para para cada centro de cluster C_k faça
7          para para cada pixel i dentro de uma área 2S × 2S ao redor do cluster C_k faça
8              Calcule a distância D_S entre C_k e i (Eq. (2.1)).
9              se D_S < d(i) então
10                 Defina d(i) = D_S
11                 Defina l(i) = k
12     Calcular novos centros de cluster e erro residual E
13 até E ≤ threshold ;
```

A distância D_S proposta por Kovesi (2000) no Algoritmo 2.1 (linha 8) para definir os melhores *pixels* de uma vizinhança ao redor de um *cluster* C_k é dada por:

$$\begin{aligned} D_S &= d_{lab} + \frac{m}{S} d_{xy}, \\ d_{lab} &= \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2}, \\ d_{xy} &= \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}, \end{aligned} \tag{2.1}$$

onde D_S é a soma da distância *lab* e da distância xy normalizada pelo intervalo S. A variável m é utilizada para controlar a compactação de um *superpixel*.

Como apresentado por Achanta et al. (2012), a vantagem do SLIC em relação ao método *k-means* é a redução no número de cálculos, limitando o espaço de busca para uma região proporcional ao *superpixel* e o controle sobre o tamanho e compactação dos *superpixels*, uma vez que a medida da distância ponderada utiliza cor e proximidade espacial.

2.3 Density Based Spatial Clustering of Applications with Noise (DBSCAN)

Ester et al. (1996) propuseram um método utilizado para realizar o agrupamento de regiões semelhantes baseado na densidade dos *clusters*. Em seguida serão apresentados os conceitos e definições necessárias para o seu entendimento, seguido da apresentação do método propriamente dito.

2.3.1 Conceitos e Definições

Para os conceitos e definições que se seguem, seja D um conjunto de pontos num espaço S.

Conceito 1: Dados dois pontos $p_1, p_2 \in D$, diz-se que p_1 é vizinho de p_2 se existir um número real *Eps*, chamada distância *Eps*, tal que a distância entre p_1 e p_2 seja menor ou igual a *Eps*, isto é, $|p_1 - p_2| \leq Eps$. Dito de outra maneira, a distância *Eps* é aquela que delimita o raio permitido para determinar se dois pontos $p_1, p_2 \in D$ são vizinhos.

Definição 1: Vizinhança-Eps de um ponto p, denotada por $N_{Eps}(p)$ representa o conjunto de pontos D que estão a uma distância *Eps* do ponto p, isto é:

$$N_{Eps}(p) = \{q \in S \,|\, dist(p, q) \leq Eps\} = D. \tag{2.2}$$

Conceito 2: A densidade de um ponto p, denotada por $\sigma(p)$, é o número de pontos que estão a uma distância *Eps* do ponto p, isto é, $\sigma(p) = |N_{Eps}(p)| = |D|$.

Conceito 3: $MinPts$ é número mínimo de pontos em uma vizinhança-Eps de um ponto p.

Definição 2: Existem duas condições para que um ponto p seja diretamente alcançável pela densidade de um ponto q, respeitando-se a distância *Eps* e o número mínimo de pontos $MinPts$. São elas:

$$p \in N_{Eps}(q), \tag{2.3}$$

$$|N_{Eps}(q)| \geq MinPts. \tag{2.4}$$

Definição 3: Um ponto p é dito densamente alcançável a partir de um ponto q se existir uma cadeia de pontos p_1,... ,p_n no qual p_1=q e p_n=p de tal modo que p_{i+1} é diretamente alcançável pela densidade do ponto p_i.

Conceito 4: Um ponto p é dito ponto central se a sua vizinhança-Eps, $N_{Eps}(p)$, contém pelo menos $MinPts$ pontos.

Conceito 5: Um ponto p é dito ponto de fronteira se não é um ponto central mas é um ponto densamente alcançável a partir de qualquer ponto central.

Definição 4: Um ponto p é dito conectado por densidade a um outro ponto q se existir um ponto central r de forma que p e q sejam ambos densamente alcançáveis a partir de r.

Definição 5: Um *cluster* C é um subconjunto não vazio de D se forem satisfeitas as seguintes condições:

$$\forall p, q \quad p \in C \wedge \text{q é densamente alcançável a partir de p} \rightarrow q \in C, \tag{2.5}$$

$$\forall p, q \in C \rightarrow \text{p é conectado por densidade com q.} \tag{2.6}$$

Definição 6: Ruído é o conjunto de pontos de D que não pertencem a nenhum *cluster*.

2.3.2 O Algoritmo DBSCAN

O algoritmo funciona buscando vizinhos similares a um ponto e agrupando-os. Desta forma, inicia-se de um ponto p aleatório pertencente a D, buscando-se a vizinhança-Eps, $N_{Eps}(p)$, deste ponto. Se o tamanho desta vizinhança for superior a $MinPts$, então estes pontos formarão um novo *cluster*. A seguir, calcula-se recursivamente todos os pontos densamente conectados a partir de qualquer ponto central já pertencentes a este novo *cluster*. Porém, caso o tamanho desta vizinhança seja inferior a $MinPts$, então o ponto p é considerado ruído e o algoritmo é reiniciado de um outro ponto que ainda não foi

visitado. É importante destacar que estes ruídos podem ser identificados posteriormente como pontos de fronteira de outro ponto central. O ponto de parada do algoritmo ocorre quando todos os pontos forem visitados e processados.

2.4 Iterated Local Search (ILS)

O algoritmo ILS (Lourenço et al., 2010; Stützle, 1998) consiste em explorar o espaço de soluções por meio da aplicação de buscas locais, seguidas de perturbações nesses ótimos locais com o objetivo de explorar novas regiões desse espaço. Essas perturbações devem ser fortes o suficiente para evitar que o algoritmo fique preso em ótimos locais e, assim, explore diferentes soluções, mas fracos o suficiente para evitar reinícios aleatórios.

O Algoritmo 2.2 mostra o pseudocódigo do método ILS. Inicialmente, parte-se de uma solução inicial (linha 1) e aplica-se nela uma busca local (linha 2). Para evitar ficar preso nessa solução (que é possivelmente um ótimo local), a solução corrente s é perturbada e uma nova busca local é feita (linhas 8 e 9), gerando-se uma solução auxiliar s''. Se s'' for melhor que s, então a nova solução corrente s passa a ser s'' e o nível de perturbação é reiniciado (linhas 4 à 13). Caso contrário, então o nível de perturbação é incrementado (linha 15). Conforme já explicado anteriormente, o nível de perturbação representa a intensidade da perturbação que será realizada. Todo este processo é repetido até que se atinja o número máximo de iterações ($ILSMax$) sem melhora na solução corrente.

2.5 Multi-Start

O algoritmo *Multi-Start* (Martí et al., 2013) é uma meta-heurística que consiste na repetição da geração de soluções aleatórias, seguida de seu refinamento por meio de uma heurística de busca local. A melhor solução encontrada durante o procedimento iterativo é a retornada pelo algoritmo.

No caso desta dissertação, o algoritmo *Multi-Start* foi aplicado ao ILS apresentado na Subseção 2.4, conforme Algoritmo 2.3. Na linha 1, uma solução é construída e refinada pelo ILS. Essa solução é atribuída ao $s^\star$ por ser a melhor encontrada até o momento. Na sequência, o algoritmo entra em laço de repetição no qual uma nova solução s é construída e refinada pelo ILS (linha 3). Se essa solução s for melhor que $s^\star$, então s passa a ser a melhor solução encontrada até o momento (linha 5). O critério de parada

Algoritmo 2.2: ILS

Entrada: $ILSMax$

```
1  s_0 ← Solução inicial
2  s ← BuscaLocal(s_0)
3  iter ← 0
4  melhorIter ← iter
5  nivel ← 1
6  repita
7      iter ← iter + 1
8      s' ← perturbacao(s, nivel)
9      s'' ← BuscaLocal(s')
10     se f(s'') > f(s) então
11         s ← s''
12         melhorIter ← iter
13         nivel ← 1
14     senão
15         nivel ← nivel + 1
16 até iter − melhorIter < ILSMax ;
17 retorna s
```

do laço de repetição foi definido como o número de reinícios ($nStart$) do método.

Algoritmo 2.3: Multi-Start ILS

Entrada: $ILSMax, nStart$

```
1  s* ← ILS(ILSMax)
2  para i ← 2 to nStart faça
3      s ← ILS(ILSMax);                          ▷ Algoritmo 2.2
4      se f(s) > f(s*) então
5          s* ← s
6  retorna s*
```

2.6 Árvore de Decisão (DT)

Árvore de decisão é um método supervisionado de classificação que é bastante utilizado na comunidade de mineração de dados e que vem apresentando um excelente desempenho para diferentes aplicações (Chao e Junzheng, 2018; Lindoff e Berry, 2011; Lussier et al., 2019)

Uma árvore de decisão é uma representação em estrutura de árvore (ver Figura 2.2) que apresenta uma sequência de decisões inter-relacionadas para guiar a classificação de

uma instância através de condições a fim de alcançar o resultado.

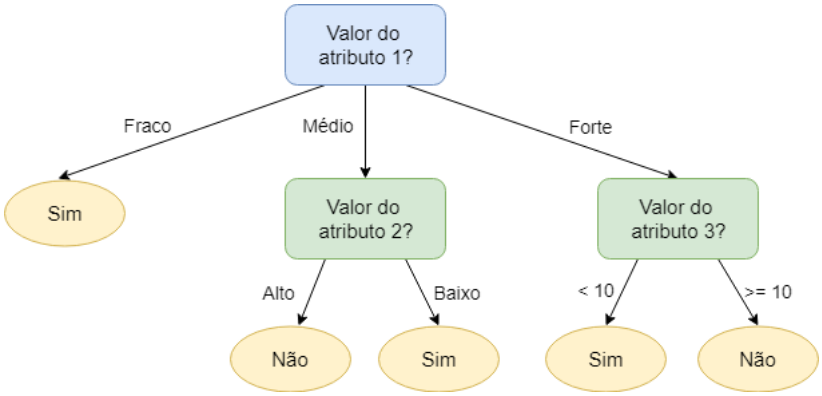


Figura 2.2: Árvore de decisão.

A composição da estrutura hierárquica de uma DT é dada por **nós** (retângulos arredondados) que especificam um teste a ser feito em um atributo, **ramos** (setas) que representam as possíveis respostas do teste e **folhas** (elipses) que indicam uma classe de resultado. Os nós são divididos em raiz (azul) que indica onde a árvore começa e nós regulares (verde) que são as etapas intermediárias entre a raiz e uma folha.

Dessa forma, para classificar uma nova instância nas classes *Sim* ou *Não*, no caso do exemplo, é necessário caminhar da raiz até alcançar um nó folha. Suponha que na Figura 2.2 o valor do Atributo 1 seja *Médio* e do Atributo 2 *Baixo*. Portanto, a nova instância é classificada como pertencente à classe *Sim*.

2.7 Medidas de Avaliação

Para avaliar uma metodologia de detecção de núcleos podemos utilizar as métricas de precisão e revocação. De acordo com Manning e Schütze (1999), precisão é definida pela medida da proporção de itens selecionados corretamente pelo sistema e pode ser calculada pela seguinte Equação:

$$prec = \frac{TP}{TP + FP}, \tag{2.7}$$

no qual TP é o número de núcleos detectados corretamente pela aplicação e FP é o

número de núcleos que foram detectados pela aplicação mas que não correspondiam de fato a um núcleo.

Manning e Schütze (1999) definiram também que revocação é definida pela proporção dos itens-alvo selecionados pelo sistema e pode ser calculada pela Equação (2.8):

$$rec = \frac{TP}{TP + FN}, \tag{2.8}$$

no qual TP é o número de núcleos detectados corretamente pela aplicação e FN é o número de núcleos que não foram encontrados.

Por fim, Manning e Schütze (1999) definiram também que a medida F1 é definida pela média harmônica entre a precisão e a revocação, conforme mostrado na Equação (2.9):

$$F1 = 2 \times \frac{prec \times rec}{prec + rec}, \tag{2.9}$$

no qual $prec$ e rec são calculados pelas Equações (2.7) e (2.8), já mostradas anteriormente.

Capítulo 3

Revisão de Literatura

A segmentação de células e núcleos permite diferentes abordagens, desde a segmentação baseada na região até uma Rede Neural Convolucional (CNN, das iniciais em inglês *Convolutional Neural Network*)(Krizhevsky et al., 2012). Nesta seção são analisados alguns destes trabalhos.

Ren e Malik (2003) propuseram o conceito de *superpixel* para a segmentação de imagens, que se tornou muito utilizado. *Superpixel* é um grupo de *pixels* que são agrupados por terem cores ou níveis de cinza semelhantes. Song et al. (2014) utilizam o conceito de *superpixel* como um estágio de agrupamento para gerar os *superpixels* que foram usados para treinar uma CNN, a fim de classificar o que era fundo na imagem, citoplasma ou núcleo. Eles obtiveram uma precisão de 94,50% para a detecção do núcleo e valores de 0,9143 $\pm$0,0202 e 0,8726 $\pm$0,0008 para precisão e revocação, respectivamente, relativos a segmentação do núcleo das células.

Ushizima et al. (2014) apresentaram a ideia de estimar a massa celular como pré-processamento inicial e utilizar *superpixels* para detectar núcleos. Além disso, propuseram a detecção do citoplasma utilizando uma faixa ao redor do núcleo, crescimento de regiões baseado em grafos e diagramas de Voronoi. Os autores concluíram que o método proposto é insensível a pequenas variações dos parâmetros de entrada das técnicas que foram utilizadas.

Nosrati e Hamarneh (2014) sugeriram uma abordagem de detecção de núcleos combinando as técnicas *Maximally Stable Extremal Region* (MSER)(Matas et al., 2004) e *Random Forest* (RF)(Breiman, 2001). Com base nos núcleos detectados, o citoplasma foi montado utilizando-se múltiplas funções de distância sinalizadas e segmentado pelo

emprego do método de Chan-Vese (Chan e Vese, 2001). Os experimentos realizados mostraram que o método foi eficiente quanto à precisão e à revocação. Os autores destacaram, ainda, que o algoritmo se sobressaiu no que se refere à velocidade, sendo cerca de 14x mais rápido que o algoritmo proposto por Lu et al. (2013).

Mariarputham e Stephen (2015) elaboraram uma estratégia que utiliza textura para classificação. Para isto, foram extraídas 24 características organizadas em 7 grupos, como por exemplo, um com características relativas ao tamanho do núcleo e do citoplasma e outro com características quanto ao deslocamento do núcleo dentro do citoplasma. Além disso, utilizaram alguns tipos de Máquina de Vetores de Suporte (SVM, das iniciais em inglês *Support Vector Machine*)(Joachims, 1998) e redes neurais para a classificação. Após experimentos, concluíram que o método também ajuda na seleção dos recursos que são mais adequados para todos os tipos de classes. Esses resultados mostraram que não há um conjunto exclusivo de recursos adequado para todas as classes. Os melhores resultados foram aqueles obtidos utilizando-se SVM.

Xing et al. (2015) apresentaram uma CNN para gerar um mapa de probabilidade com a finalidade de obter os contornos iniciais das células utilizando uma abordagem de mesclagem de região iterativa. Em seguida, núcleos individuais são separados combinando um modelo de forma esparso baseado em seleção robusta e um modelo local deformável repulsivo. Os experimentos comparativos com o estado atual da arte mostraram desempenho superior da abordagem proposta.

Saha et al. (2016) propuseram uma abordagem na qual criaram uma imagem binária usando o limiar baseado em histograma. Em seguida, introduziram uma *Circular Shape Function* (CSF) que impõe uma restrição de forma sobre as regiões consideradas, buscando melhorar a segmentação do núcleo utilizando o *Fuzzy C-Means Clustering* (FCM)(Bezdek et al., 1984). Regiões não nucleadas foram filtradas empregando operações morfológicas. Análises quantitativas e qualitativas indicaram que a proposta apresentou resultados satisfatórios quanto à detecção e segmentação de núcleos.

Lee e Kim (2016) sugeriram um método que também gera *superpixels* pelo algoritmo SLIC. São calculados os valores médios de intensidade para cada um dos *superpixels* a fim de gerar uma imagem de valor médio, do qual é extraída a massa celular por meio de um limiar adaptativo usando o método do triângulo. A partir dessas massas celulares, os núcleos das células são extraídos por um limiar local e é aplicada a remoção de *outliers*[1].

[1] *Outliers* são dados que se diferenciam drasticamente dos demais dados, ou seja, é um valor aberrante, que foge da normalidade.

Por fim, é feito um particionamento dos *superpixels* e um refinamento do contorno da célula para segmentação do citoplasma. O método se apresentou eficaz e competitivo com os trabalhos comparados por ele.

Tareef et al. (2017) desenvolveram um método baseado em características distintivas locais e deformação de forma guiada, as quais são incorporadas e classificadas por uma SVM a fim de segmentar a imagem em núcleos, células e fundo. Além disso, eles utilizaram uma estrutura baseada na teoria de codificação esparsa e orientada por características representativas da forma de construir o citoplasma de cada célula. Em seguida, a forma obtida é refinada pelo método *Distance Regularized Level Set Evolution* (DRLSE). Testes mostraram que essa abordagem superou as abordagens comparadas por ele quanto à segmentação de células e quanto à precisão da obtenção dos limites dos núcleos.

Braz e Lotufo (2017) propuseram uma abordagem que utiliza uma CNN para realizar a detecção de núcleos. A fim de não limitar o tamanho das imagens a serem submetidas ao tratamento, após o treinamento, as camadas da rede que estavam totalmente conectadas são convertidas em camadas convolucionais. O método se apresentou competitivo com aqueles utilizados para a detecção de núcleos e que eram estado da arte à época.

Oliveira et al. (2017) apresentaram um método que utiliza os algoritmos SLIC, DBSCAN e uma calibração feita por um Algoritmo Memético (AM) (Burke et al., 1995), isto é, um Algoritmo Genético (AG) com buscas locais para a segmentação do núcleo. Além disso, eles segmentam o núcleo com base em informações de sua circularidade, intensidade e área. O método apresentou bons resultados, mas não será utilizado neste trabalho para fins de comparação pois os autores utilizaram apenas a base de 45 imagens, sendo 40 para treino e 5 para teste, sendo que esta não é a divisão original da base. Além disso, o método é de otimização multiobjetivo e tem como desvantagem o fato de que o tempo de processamento é muito elevado.

A Tabela 3.1, a seguir, resume os trabalhos da literatura que tratam da segmentação. Na primeira coluna especificam-se os autores e o ano de publicação; na segunda, o veículo de publicação; na terceira, a estratégia utilizada para tratar o problema abordado e, na última, a base de dados utilizada para realizar os experimentos.

Assim como apresentado na revisão de literatura, não foi encontrada uma abordagem que utilize um algoritmo ILS como estratégia para a detecção de núcleos. Por outro lado, esse método tem sido aplicado com sucesso para a solução de vários problemas

Tabela 3.1: Trabalhos relacionados quanto a segmentação.

Trabalho	**Publicado em**	**Estratégia**	**Base de dados**
Song et al. (2014)	*36th Annual International Conference of the IEEE*	*Superpixel*, CNN	Dados de 200 mulheres com idade entre 22 e 64 anos, pacientes do Sixth People's Hospital of Shenzhen
Ushizima et al. (2014)	*International Symposium on Biomedical Imaging*	*Superpixel*	Desafio do ISBI 2014
Nosrati e Hamarneh (2014)	*ISBI Overlapping Cervical Cytology Image Segmentation Challenge*	MSER, RF, Chan-Vese	Desafio do ISBI 2014
Mariarputham e Stephen (2015)	*Computational and Mathematical Methods in Medicine*	SVM, Redes Neurais	Base do Hospital Universitário de Herlev, Dinamarca
Xing et al. (2015)	*IEEE Transactions on Medical Imaging*	CNN	Base com imagens de tumor cerebral, NET e câncer de mama
Saha et al. (2016)	*Digital Image Computing: Techniques and Applications*	CSF, FCM	Desafio do ISBI 2014
Lee e Kim (2016)	*IEEE Conference on Computer Vision and Pattern Recognition Workshops*	*Superpixel*	Desafio do ISBI 2014 e 2015
Tareef et al. (2017)	*Neurocomputing*	SVM, DRLSE	Desafio do ISBI 2014
Braz e Lotufo (2017)	XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Imagens	CNN	Desafio do ISBI 2014
Oliveira et al. (2017)	*IEEE Congress on Evolutionary Computation*	*Superpixel*, AM	Desafio do ISBI 2014

não-lineares como o aqui estudado, como em Coelho et al. (2016), Zhou e Hao (2017) e Song et al. (2018b). Portanto, por essas razões esse método foi um dos escolhidos para

tratar o problema de detecção de núcleos.

Coelho et al. (2016) apresentaram uma proposta que combina *Greedy Randomized Adaptive Search Procedure* (GRASP), ILS e *Variable Neighbourhood Descent* (VND) para o planejamento de distribuição de uma empresa. Segundo os autores, esse método obtém soluções competitivas e melhora as soluções da empresa, levando a economias significativas nos custos de transporte.

Zhou e Hao (2017) aplicaram um algoritmo ILS para resolver o problema da dispersão diferencial mínima[2]. Experimentos realizados mostraram que o algoritmo desenvolvido compete favoravelmente com os algoritmos do estado da arte à época, superando 131 resultados e obtendo 42 valores iguais ao que se tinha na época para as 190 instâncias utilizadas.

Song et al. (2018b) propuseram um algoritmo ILS para encontrar uma solução viável para o problema de alocação de horários. Testes mostraram que o algoritmo desenvolvido alcança resultados competitivos em comparação com os algoritmos existentes no momento do estudo. Os experimentos consideraram 60 instâncias, sendo que o método encontrou soluções viáveis em 58 delas em tempo razoável, dos quais três delas não tinham sido encontradas pelos algoritmos do estado da arte da época.

Além disso, também não foi encontrada uma abordagem que utiliza DT para a detecção de núcleos. Contudo, o método tem apresentado bons resultados quando aplicado a outros problemas, como em Dantas et al. (2015) e Medeiros et al. (2016).

Dantas et al. (2015) aplicaram uma DT juntamente com o cálculo da distância Euclidiana para reconhecimento de emoções em expressões faciais. Elas foram classificadas em sete emoções básicas, sendo elas: alegria, surpresa, raiva, medo, desgosto, tristeza e neutra. A abordagem foi integrada ao ambiente virtual de aprendizagem (*Moodle*) visando reconhecer as emoções dos estudantes para analisar seus comportamentos. O método obteve 86,4% de acurácia geral, sendo satisfatório para todas as emoções.

Medeiros et al. (2016) apresentaram um modelo utilizando DT para auxiliar profissionais da saúde a identificar os padrões de comportamento no uso dos serviços da Estratégia Saúde da Família das pessoas que possuem a Síndrome da Imunodeficiência Adquirida (AIDS) resultante da infecção pelo Vírus da Imunodeficiência Humana (HIV) e são atendidas no ambulatório. O banco de dados utilizado possui informações de 141

[2]Dado um conjunto de n elementos separados por uma matriz de distância em pares, o problema busca identificar um subconjunto de m elementos (m < n) de modo que a minimizar a diferença entre a soma máxima e a soma mínima das distâncias entre quaisquer dois elementos escolhidos.

clientes aidéticos de um ambulatório especializado. Com isso, construiu-se 23 regras, que inferem 80,1% de acerto.

Justino et al. (2017) utilizaram uma DT para fazer o mapeamento do uso de terra e cobertura vegetal da bacia do Rio São Tomé, na região de Alfenas, Minas Gerais. Foram utilizadas imagens multiespectrais do sensor *Linear Imaging Self-Scanner* (LISS III) localizado no sensor *Indian Resource Satellite* (IRS). Os autores destacaram que a classificação gerada foi consistente e com uma interpretação simplificada das regras.

Como já dito anteriormente, este trabalho tem como finalidade a segmentação de núcleos de células cervicais obtidas em imagens de Papanicolaou. O objetivo é maximizar o número de resultados positivos verdadeiros encontrados e minimizar o número de resultados falso positivos. Em outras palavras, o objetivo é maximizar o número de núcleos encontrados corretamente e minimizar o número de núcleos encontrados que, na realidade, não representam núcleos, e sim artefatos que mimetizam núcleos. No método proposto, portanto, escolheu-se utilizar o algoritmo de clusterização de *superpixels*, denominado SLIC, o algoritmo DBSCAN para agrupar alguns *superpixels* gerados em *clusters*. Além disso, é testada uma abordagem heurística baseada no ILS e uma abordagem via DT para identificar se um *superpixel* é ou não um núcleo.

Capítulo 4

Descrição do Problema

O problema tratado neste trabalho é o do desafio *Overlapping Cervical Cytology Image Segmentation Challenge*, ocorrido no ISBI de 2014.

Nesta competição foi disponibilizada uma base de dados contendo 945 imagens de células cervicais sintéticas, geradas a partir de imagens reais. Todas as imagens são de tamanho 512×512, em escala de cinza, com um número diferente de células (variando de dois a dez) e com diferentes níveis de sobreposição das células. Todas as imagens possuem um *ground truth* indicando a localização de seus núcleos.

Essas imagens são divididas em dois grupos: 45 para treinamento e 900 para teste.

Um exemplo dessas imagens é mostrado na Figura 4.1.

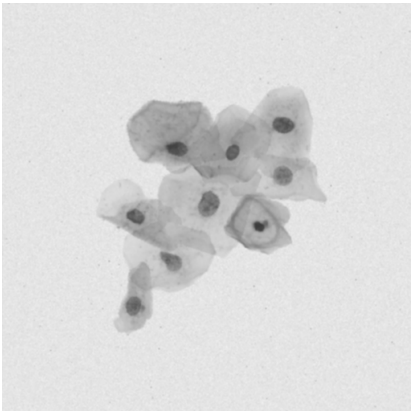


Figura 4.1: Exemplo de imagem sintética da base de dados.

O objetivo desse desafio, portanto, é segmentar núcleos e citoplasmas de forma in-

dependente. Assim, a nossa abordagem será apenas para a detecção e segmentação do núcleo seguindo a ideia defendida por muitos pesquisadores, como Moshavegh et al. (2012); Samsudin et al. (2016), de que apenas esta informação é suficiente para detectar de lesões, uma vez que as características morfológicas e de textura do núcleo apresentam variações significativas quando estão alteradas.

Para exemplificar, imagine que o algoritmo usado para detectar os núcleos da Figura 4.1 tenha obtido o resultado apresentado na Figura 4.2.

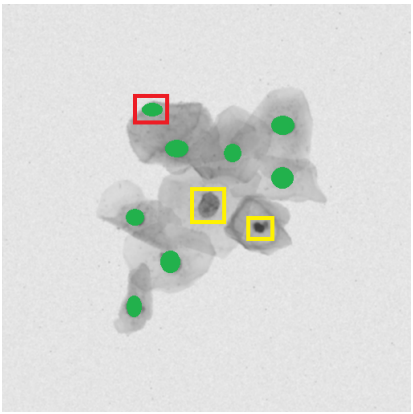


Figura 4.2: Exemplo de resultado obtido.

Supondo que os núcleos em verde foram os detectados pelo algoritmo, podemos verificar que ele:

- encontrou 1 núcleo incorreto (marcado de vermelho);
- encontrou 7 núcleos corretos (aqueles que não possuem nenhuma marcação);
- deixou de encontrar 2 núcleos (marcados em amarelo).

Assim, utilizando as Equações (2.7) e (2.8) podemos verificar que, para esse exemplo, teríamos uma precisão igual a 0,875 = 7/(7 + 1) e revocação igual a 0,778 = 7/(7 + 2). Desta forma, esses valores de precisão e revocação serão utilizados para mensurar a assertividade da solução encontrada. A revocação mensura o número de núcleos detectados, enquanto que a precisão avalia a quantidades de núcleos detectados que são efetivamente núcleos.

Como um computador não pode dar um diagnóstico, um citopatologista terá que analisar o resultado do método proposto. Assim, o objetivo de utilizar um computador

é minimizar o número de imagens que precisam ser analisadas, identificando se a célula possui núcleos anormais (informações relevantes) ou normais (descartando-os). Além disso, é crucial que todos os núcleos possíveis sejam detectados, uma vez que a não detecção de um núcleo pode levar a um diagnóstico errado.

O cenário perfeito seria aquele em que são detectados todos os núcleos presentes na imagem, sem encontrar outros núcleos incorretamente (precisão e revocação de 100%). No entanto, este caso é raro devido a dificuldades como sobreposição e ruídos.

Com isso, fica evidente que ter uma revocação de 100% é um cenário perfeito, pois garante a detecção de todos os núcleos, mesmo que também detecte falsos positivos. No entanto, o citopatologista ainda teria todas as informações necessárias para o diagnóstico. Ao mesmo tempo, é importante que a precisão seja a mais alta possível, pois determina a quantidade de trabalho reduzida.

Este trabalho propõe métodos que encontrem núcleos em imagens de células cervicais obtidas em exames de Papanicolaou. O foco deste trabalho, portanto, é encontrar uma alta taxa de revocação para seus métodos.

Capítulo 5

Desenvolvimento

Para a segmentação de núcleos em imagens de células cervicais obtidas em exames de Papanicolaou são propostas duas etapas, apresentadas no fluxograma da Figura 5.1. Propõe-se inicialmente um pré-processamento (Seção 5.1), seguido de duas opções de abordagem, uma heurística (Seção 5.2) e outra estatística (Seção 5.3). A heurística engloba a definição dos parâmetros e um método baseado na metaheurística de busca local ILS. A estatística compreende a construção de uma base de dados com atributos específicos, a seleção destes atributos e um método de treinamento supervisionado utilizando-se árvore de decisão para realizar a classificação. Todas as etapas serão apresentadas nas próximas seções.

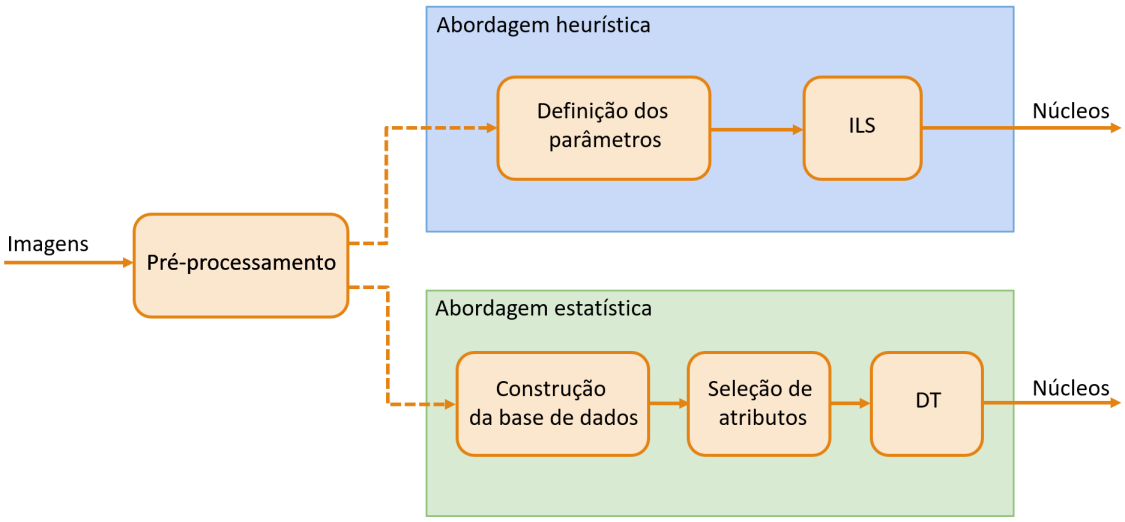


Figura 5.1: Fluxograma do desenvolvimento proposto.

5.1 Pré-Processamento

Inicialmente, o método proposto por este trabalho sugere um pré-processamento (Figura 5.2) em duas etapas.

Na primeira etapa, a imagem original da base de dados mostrada na Seção 4 (ver Figura 5.2 (a)) é usada como entrada do algoritmo SLIC (apresentado na Subseção 2.2), resultando em uma imagem com os *superpixels* (ver Figura 5.2 (b)).

Já na segunda etapa, esta imagem com os *superpixels* gerados é usada como entrada para o algoritmo DBSCAN (apresentado na Subseção 2.3), resultando em uma imagem agrupada dos *superpixels* semelhantes (ver Figura 5.2 (c)).

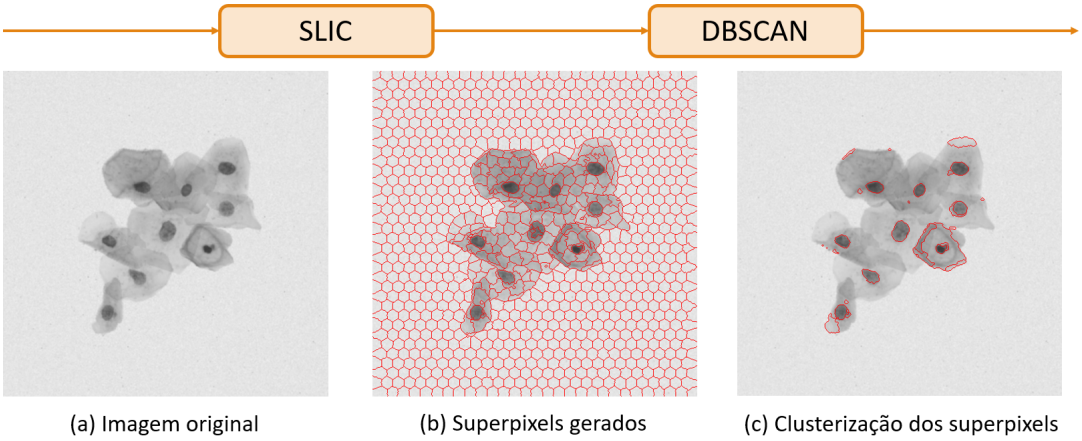


Figura 5.2: Fluxograma de pré-processamento de imagens.

Desta forma, encerra-se o pré-processamento e tem-se a separação da imagem original em *clusters* candidatos a núcleos, denominado C_k, no qual k varia de 1 até o número de *clusters*. Assim, cada *cluster* C_k será avaliado.

5.2 Abordagem Heurística via ILS

Nesta seção, a abordagem heurística proposta, baseada no algoritmo ILS (Lourenço et al., 2010), é descrita. Este método foi escolhido em vista de seu bom desempenho para resolver vários outros problemas complexos de otimização, como em Coelho et al. (2016), Zhou e Hao (2017) e Song et al. (2018b).

A Subseção 5.2.1 apresenta a definição dos parâmetros utilizados na abordagem

heurística proposta. A Subseção 5.2.2 aponta a representação de uma solução. A Subseção 5.2.3 contém a construção de uma solução inicial e define a estrutura de vizinhança adotada. A Subseção 5.2.4 apresenta a avaliação da solução. E, por fim, a Subseção 5.2.5 apresenta o algoritmo do *Iterated Local Search*.

5.2.1 Definição dos Parâmetros

Inicialmente, a avaliação dos *clusters* C_k candidatos a núcleos gerados no pré-processamento (apresentado na Seção 5.1) tem por base cinco parâmetros, denominados CIA, propostos por Oliveira et al. (2017): circularidade mínima, circularidade máxima, intensidade máxima, área mínima e área máxima.

Porém, percebeu-se (conforme apresentado na Subseção 6.2.2) que os valores máximos da circularidade e da área da base de treino não eram representativos da base de teste, limitando a heurística proposta. Dessa forma, foram selecionados apenas três parâmetros (circularidade mínima, intensidade máxima e área mínima) para definir se um *cluster* C_k é ou não um núcleo.

Para obter-se o valor da intensidade de um *cluster* C_k é calculada a média aritmética das intensidades dos *pixels* da região, conforme Equação (5.1).

$$intensidade(C_k) = \frac{\sum_{i=1}^{npixels(C_k)} intensidade(pixel(C_k[i]))}{npixels(C_k)}, \tag{5.1}$$

em que $npixels(C_k)$ é o número de *pixels* do *cluster* C_k e $pixel(C_k[i])$ é o i-ésimo *pixel* do *cluster* C_k.

Por sua vez, dado que um *pixel* é a menor unidade de uma imagem digital, é bidimensional e possui um tamanho associado a ele, podemos calcular o valor da área de um *cluster* C_k, denotado por $area(C_k)$ de acordo com o número de *pixels* nele contido, isto é:

$$area(C_k) = npixels(C_k). \tag{5.2}$$

Já a circularidade é calculada utilizando-se a Equação (5.3):

$$circularidade(C_k) = \frac{4 \cdot \pi \cdot area(C_k)}{(perimetro(C_k))^2}, \tag{5.3}$$

em que $perimetro(C_k)$ é o perímetro do *cluster* C_k, determinado pela Equação (5.4):

$$perimetro(C_k) = \sum_{i=1}^{npixels(C_k)} eBorda(pixel(C_k[i])), \tag{5.4}$$

na qual $eBorda(pixel(C_k[i]))$ é uma função que retorna 1 caso o i-ésimo pixel seja um *pixel* que está na borda do *cluster* C_k e 0, caso contrário.

Desta forma, é conhecido cada um dos valores dos parâmetros considerados para identificar se um *cluster* C_k é ou não um núcleo.

A Subseção 5.2.1.1 apresenta o objetivo da abordagem heurística proposta e a Subseção 5.2.1.2 aponta os limites definidos para os parâmetros utilizados.

5.2.1.1 Objetivo da Abordagem Heurística via ILS

A abordagem heurística via ILS tem por objetivo encontrar qual é a melhor combinação de todos os parâmetros CIA que produzem a melhor detecção dos núcleos da base de dados.

5.2.1.2 Limites dos Parâmetros

Para atingir o objetivo da abordagem heurística é necessário definir quais são os intervalos de valores que serão analisados em cada um dos parâmetros. Assim, definiu-se que seria utilizada a base de dados de treino (ver Seção 4) para estabelecer esses limites.

Dessa forma, foram analisados os valores correspondentes aos parâmetros de cada núcleo presente na base de treino. Por exemplo, coletou-se o valor da área de todos os núcleos. Assim, o menor valor obtido ficaria correspondente à menor área mínima permitida. Além disso, seria necessário determinar o valor da maior área mínima. Estipulou-se,

então, que tal valor seria estabelecido acrescentando 20% do seu valor de referência (área mínima).

Sendo assim, valores utilizados para todos os parâmetros CIA na abordagem heurística foram os apresentados na Tabela 5.1.

Tabela 5.1: Limites usados para os valores dos parâmetros.

	Valor	
Parâmetros	**Mínimo**	**Máximo**
Circularidade Mínima	0,48	0,71
Intensidade Máxima	57	178
Área Mínima	100	284

5.2.2 Representação da Solução

Uma solução s do problema é representada por um vetor de três posições, sendo que cada posição indica o valor de um dos parâmetros CIA dentro dos limites apresentados na Tabela 5.1.

Um exemplo de solução é mostrado a seguir. Nesta solução, o primeiro parâmetro, que mede a circularidade mínima, tem valor 0,5 e o terceiro parâmetro, que mensura a área mínima, tem valor 120, por exemplo.

$$s = \quad \langle\ 0{,}50,\ 70,\ 120\ \rangle.$$

5.2.3 Solução Inicial e Vizinhança

Uma solução inicial para o problema é obtida escolhendo-se randomicamente valores para cada um dos parâmetros CIA, respeitando-se os limites definidos na Tabela 5.1.

A seguir é ilustrado um exemplo em que a terceira posição da solução s é um valor inteiro e, portanto, possui o passo r_{int}. Supondo que $r_{int} = 5$ e que foi escolhido que a posição seria decrementada de um passo de tamanho $r_3 = 2$, então, gera-se um vizinho s' da solução s.

$$s = \quad \langle\ 0.50,\ 70,\ \mathbf{120} \rangle,$$

$$s' = \quad \langle\ 0.50,\ 70,\ \mathbf{118}\ \rangle.$$

Esta vizinhança foi a única utilizada, porque com apenas este movimento é possível percorrer todo o espaço de soluções do problema por métodos de buscas locais.

5.2.4 Avaliação da Solução

Uma vez determinados os valores dos parâmetros de cada *cluster* C_k (ver Subseção 5.2.1), então utiliza-se o Algoritmo 5.1 para verificar se o referido *cluster* respeita os limites definidos por uma solução s, indicando ser ou não um núcleo.

Algoritmo 5.1: Análise de um *cluster* C_k

Entrada: $parametros, cluster$

```
1 se circularidade(cluster) ≥ parametros[0] então
2     se intensidade(cluster) ≤ parametros[1] então
3         se area(cluster) ≥ parametros[2] então
4             retorna É um núcleo!
5 retorna Não é um núcleo!
```

Assim, utilizando-se o Algoritmo 5.1, para cada imagem I, uma máscara resultante X é gerada contendo apenas os *clusters* C_k dentro dos intervalos delimitados por todos os parâmetros CIA da solução s. Dessa forma, a imagem I pode ter menos ou mais núcleos do que o *ground truth*.

O processo de construção de uma máscara X é mostrado na Figura 5.3. Como podemos ver, na imagem I são cinco *clusters* candidatos a núcleos, mas apenas dois (em cor verde) foram identificados como núcleos. Assim, a máscara X gerada possui apenas esses dois *clusters*.

Se a máscara X contiver mais de um núcleo, então ela será decomposta em q imagens distintas, sendo q o número de núcleos da máscara X, isto é, $X = \cup_{i=1}^{q} X_i$, com $\cap_{i=1}^{q} X_i = \emptyset$, e cada imagem contém apenas um dos núcleos. Por exemplo, como apresentado na Figura 5.4, a máscara de dois núcleos é decomposta em duas novas máscaras X_1 e X_2, cada uma contendo apenas um de seus núcleos. O mesmo procedimento é realizado para

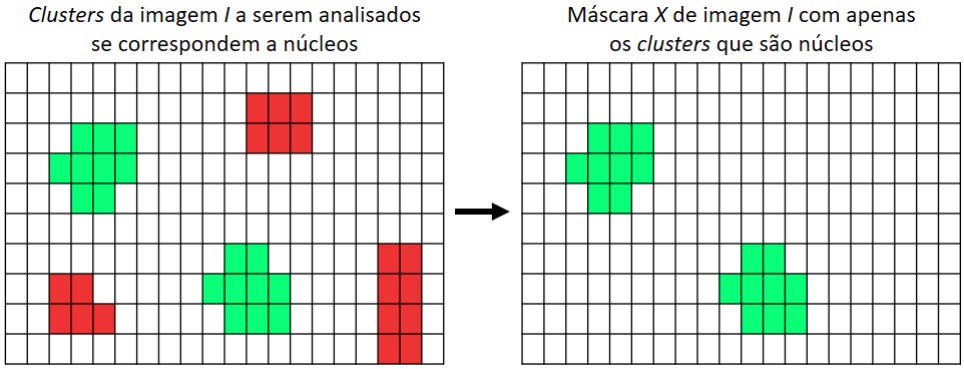


Figura 5.3: Exemplo de construção da máscara X.

cada modelo Y na imagem do *ground truth*. Observa-se que o *ground truth* de uma imagem I pode conter um número diferente de núcleos.

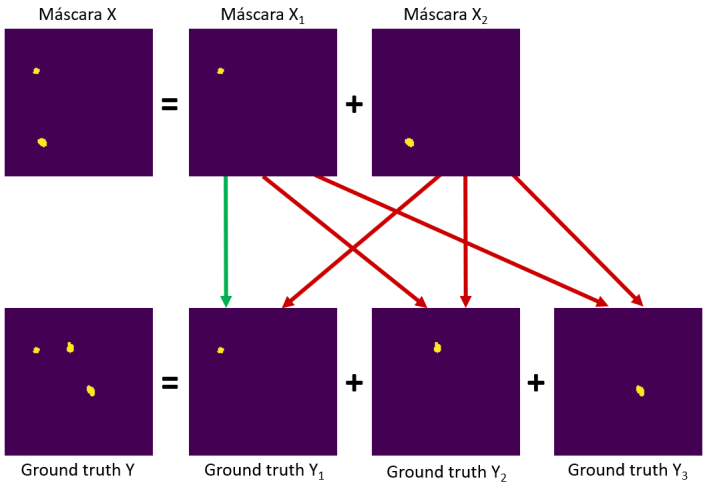


Figura 5.4: Comparação entre uma máscara X e seu *ground truth* Y.

Cada máscara X_i relativa a uma imagem I é comparada com todos os *ground truths* Y_j da mesma imagem sob avaliação para determinar o nível de assertividade do método.

Para tal, é utilizado o Coeficiente de Similaridade Dice (Dice, 1945), também conhecido por Coeficiente Sørensen–Dice. Esse coeficiente é calculado por meio da Equação (5.5), que é uma métrica estatística usada para comparar a similaridade entre duas amostras X_i e Y_j:

$$Dice(X_i, Y_j) = \frac{2|X_i \cap Y_j|}{|X_i| + |Y_j|}. \tag{5.5}$$

O valor do coeficiente dado pela Equação (5.5) está contido no intervalo real (0,1). Consideramos que um coeficiente maior que 0,6 indica que a similaridade entre eles é maior que 60% (Bradley et al., 2014). Consequentemente, o *cluster* C_k satisfazendo a essas condições é considerado um núcleo. Assim, se existir um *ground truth* Y_j tal que $Dice(X_i, Y_j) \geq 0,6$, então o algoritmo detectou corretamente que X_i é um núcleo (verdadeiro positivo). Caso contrário, diz-se que o algoritmo detectou erroneamente que X_i é um núcleo (falso positivo).

Na Figura 5.4 temos uma máscara X com dois núcleos que foram decompostos em duas máscaras X_1 e X_2, cada uma com um único núcleo. Além disso, temos um *ground truth* Y da imagem I que possui três núcleos. Esse *ground truth* foi decomposto em três outras máscaras: Y_1, Y_2 e Y_3, cada um com um único núcleo. Aplicando-se o Coeficiente de Similaridade, observa-se que X_1 e Y_1 foram considerados equivalentes (veja a seta verde). Como o valor $Dice(X_1, Y_1)$ foi maior que 0,6, podemos dizer que o núcleo em X_1 foi detectado corretamente (verdadeiro positivo). Ainda no mesmo exemplo, já que a máscara X_2 não foi considerada equivalente com nenhum *ground truth* Y_j, com $j = 1, 2, 3$ (veja setas vermelhas), podemos afirmar que o núcleo em X_2 foi detectado incorretamente (falso positivo).

Apresentados esses conceitos, uma solução s é avaliada pela função $F_1(s)$, já apresentada na Subseção 2.7, dada pela Equação (5.6), e que deve ser maximizada:

$$F_1(s) = 2 \times \frac{prec(s) \times rec(s)}{prec(s) + rec(s)}, \tag{5.6}$$

no qual as medidas de precisão $prec$ e revocação rec da solução s são calculadas de acordo com as Equações (5.7) e (5.8), respectivamente:

$$prec(s) = \frac{\sum_{I \in DataBase} TP(I, s)}{\sum_{I \in DataBase} [TP(I, s) + FP(I, s)]}, \tag{5.7}$$

$$rec(s) = \frac{\sum_{I \in DataBase} TP(I,s)}{\sum_{I \in DataBase}[TP(I,s) + FN(I,s)]}, \tag{5.8}$$

sendo TP(I,s), FP(I,s) e $FN(I,s)$ o número de resultados verdadeiros positivos, falsos positivos e falsos negativos, respectivamente, em cada imagem I da base de dados detectados pela aplicação da Equação (5.5) a todas as máscaras X_i dessa imagem I decomposta, que foram geradas a partir da solução s.

Desta forma, procura-se maximizar o número de resultados verdadeiros positivos (TP) e minimizar o número de resultados falsos positivos (FP) encontrados nas imagens da base de dados.

5.2.5 Iterated Local Search

Como já foi dito anteriormente, o algoritmo heurístico proposto neste trabalho é baseado na metaheurística de busca local *Iterated Local Search* e segue o *framework* do Algoritmo 2.2, descrito na Seção 2.4. O procedimento de perturbação, descrito pelo Algoritmo 5.2, consiste em aplicar $p + 1$ movimentos consecutivos na solução corrente, sendo p o nível de perturbação realizada.

Algoritmo 5.2: Perturbação

Entrada: $s, nivel$

```
1 s' ← s
2 nModificacoes ← nivel + 1
3 cont ← 1
4 repita
5 |   Aplique movimento aleatório em s'
6 |   cont ← cont + 1
7 até cont ≤ nModificacoes ;
8 retorna s'
```

Além disso, neste trabalho utilizou-se o método da Descida, com a estratégia *Best Improvement* (Hansen e Mladenović, 2006), como busca local do Algoritmo 2.2, linhas 2 e 9. Este método parte de uma solução inicial, conforme descrito na subseção seguinte, e a cada passo analisa todos os vizinhos da solução corrente, movendo-se para o melhor deles ao final de cada análise. A busca é encerrada quando encontra-se um ótimo local com relação à vizinhança considerada.

5.2.6 Multi-Start ILS x Múltiplas Soluções Iniciais

Algoritmos baseados em ILS normalmente dependem de uma boa solução inicial para obter bons resultados na detecção de núcleos. Porém, como apresentado na Subseção 5.2.3, a solução inicial do problema é obtida de forma completamente aleatória, não sendo possível, desta forma, garantir se a solução gerada é de boa qualidade.

Pensando nisso, foram testadas duas possibilidades para melhorar a qualidade da solução inicial, ambas apresentadas na Seção 6.2. A primeira delas consiste na implementação do algoritmo *Multi-Start ILS*, apresentado na Subseção 2.5. A segunda possibilidade analisada foi a construção de múltiplas soluções iniciais, sendo que apenas a melhor delas seria escolhida como solução inicial para o algoritmo ILS. Neste último caso, a linha 1 do Algoritmo 2.2, é substituída por um procedimento que consiste em gerar *nSolRand* soluções aleatórias e que retorne aquela com o melhor valor de avaliação. Assim, é necessário calibrar dois parâmetros:

- *nStart*, que é o número de reinícios do ILS;
- *nSolRand*, que corresponde ao número de soluções iniciais geradas aleatoriamente para a escolha da melhor dentre elas.

5.3 Abordagem via DT

A abordagem via DT foi escolhida pois apresenta bons resultados para resolver vários outros problemas correlatos ao deste trabalho, como de Dantas et al. (2015) e Medeiros et al. (2016).

A Subseção 5.3.1 apresenta a construção das bases de treinamento e teste necessárias. A Subseção 5.3.2 expõe a seleção de atributos que foi feita nas bases. Por fim, a Subseção 5.3.3 aponta a DT que foi utilizada neste trabalho.

5.3.1 Construção das Bases de Treinamento e Teste

Assim como apresentado na Seção 5.1, ao final do pré-processamento tem-se uma imagem de *clusters* que são candidatos a núcleos. Dessa forma, para a classificação através da árvore de decisão é necessário construir uma base de dados com as informações destes *clusters*.

Para isso, foi utilizada uma função do *Python* denominada *Regionprops*[1] para extrair características morfológicas dos *clusters*, sendo elas:

- **Área (*Area*):** número de *pixels* do *cluster*;
- **Área do Bounding Box (*BBoxArea*)**: número de total de *pixels* do *bounding box* (menor caixa possível que englobe todo o *cluster*);
- **Área do menor polígono convexo (*ConvexArea*):** Número de *pixels* do menor polígono convexo que envolve o *cluster*;
- **Área preenchida (*FilledArea*):** número de *pixels* preenchidos do *cluster*;
- **Circularidade (*Circ*):** indica o quão circular é o *cluster*;
- **Diâmetro (*Diameter*):** o diâmetro de um círculo com a mesma área que o *cluster*;
- **Excentricidade (*Excent*):** excentricidade de uma elipse que engloba o *cluster*;
- **Eixos (*MinorAxis*, *MajorAxis*):** comprimento do menor e do maior eixo de uma elipse que engloba o *cluster*;
- **Extensão (*Extent*):** proporção do número de *pixels* do *cluster* sobre os do *bounding box*;
- **Intensidade mínima, média e máxima (*IntMin*, *IntMed*, *IntMax*):** valores correspondentes à intensidade dos *pixels*;
- **Número de Euler (*Euler*):** característica de Euler do *cluster*;
- **Solidez (*Solidity*):** proporção do número de *pixels* do *cluster* sobre os do menor polígono convexo que o envolve.

Todas as características apresentadas são numéricas, sendo atribuídas às bases de treinamento e teste como atributos contínuos.

Além disso, como a DT é um método supervisionado de classificação, então é necessário que a base de treinamento possua um atributo-meta, que é aquele que se deseja inferir. Para tal, a imagem clusterizada é comparada com o seu gabarito (de maneira análoga ao que foi mostrado na Subseção 5.2.4) e o atributo-meta recebe 1 caso o *cluster* seja um núcleo ou 0, caso contrário.

[1] https://scikit-image.org/docs/0.15.x/api/skimage.measure.html#regionprops

5.3.2 Seleção de Atributos

Foi realizado um método de seleção de atributos utilizando uma matriz de correlação para reduzir o número de atributos e melhorar o desempenho da DT. A Figura 5.5 apresenta a matriz de correlação entre os atributos, com uma escala de -1 (vermelho escuro) a 1 (azul escuro), no qual -1 é a máxima relação inversa, 1 a máxima relação direta e 0 sem relação. A ordem dos atributos é para facilitar a visualização agrupando os correlacionados. Foram consideradas apenas as correlações acima de $|0,5|$. Essa abordagem foi aplicada porque qualquer atributo de um grupo correlacionado pode ser utilizado como um representativo dele. Dessa maneira, apenas um atributo pode representá-lo.

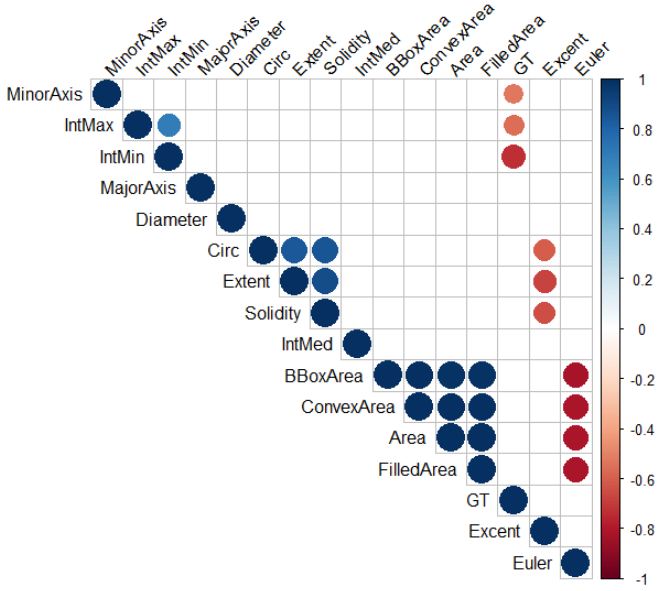
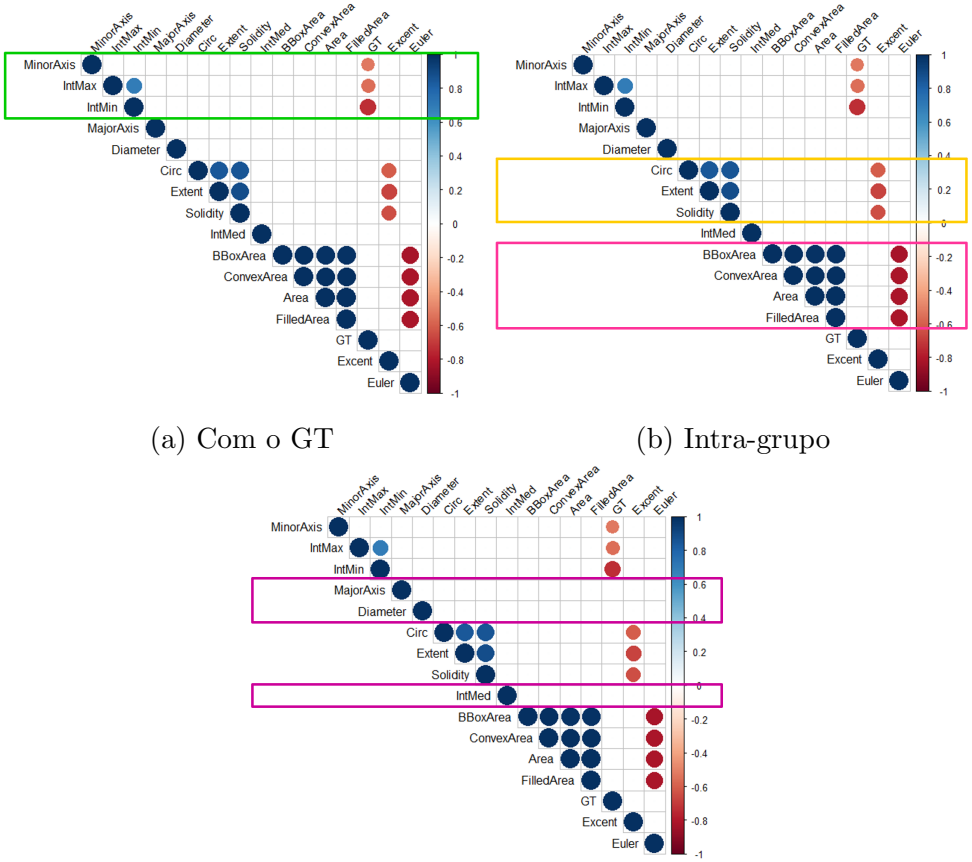


Figura 5.5: Matriz de correlação.

Inicialmente, os atributos foram separados em grupos. Para isso, foram analisadas três situações:

- atributos que possuem correlação com o GT (ver Figura 5.6a), obtendo o agrupamento em verde;
- atributos que possuem correlação entre si (ver Figura 5.6b), formando um grupo em amarelo (possuem correlação com o *Excent*) e outro em rosa (possuem correlação com o *Euler*);

- demais atributos que não têm correlação com nenhum outro atributo (ver Figura 5.6c), obtendo o grupo em roxo.



(a) Com o GT (b) Intra-grupo

(c) Sem relação

Figura 5.6: Separação em grupos de acordo com a correlação.

Dessa forma, foram obtidos quatro grupos. O grupo verde possui os atributos *MinorAxis*, *IntMax* e *IntMin*, que são correlacionados com GT. Os atributos do grupo amarelo são *Circ*, *Extent*, *Solidity* e *Excent*, enquanto o grupo rosa possui *BBoxArea*, *ConvexArea*, *Area*, *FilledArea* e *Euler*. Por fim, o grupo roxo possui os atributos *MajorAxis*, *Diameter* e *IntMed*.

Em seguida, é selecionado um ou nenhum atributo representativo de cada grupo, o modelo DT é treinado e avaliado. Todas as combinações são avaliadas para se obter a melhor delas.

Após o término desta avaliação, os atributos *IntMax*, *Circ* e *Area* formaram a melhor combinação. Assim, os outros atributos foram excluídos da análise.

5.3.3 Árvore de Decisão

A DT utilizada neste trabalho foi a fornecida pelo pacote denominado *rpart*[2], disponibilizado na linguagem R. A maioria de suas funcionalidades foram implementadas por Breiman et al. (1984).

[2]https://CRAN.R-project.org/package=rpart

Capítulo 6

Experimentos e Resultados

Neste capítulo são descritos, analisados e discutidos os experimentos relativos aos métodos propostos de segmentação de núcleos em células cervicais.

O pré-processamento proposto (Seção 5.1) foi desenvolvido na linguagem MATLAB, o algoritmo heurístico ILS (Seção 5.2) em Python e a abordagem estatística pela Árvore de Decisão (Seção 5.3) em R. Todos os experimentos foram realizados em um processador Intel Core i7-8700 com processador de 3.20 GHz, 16 GB de RAM, Windows 10 de 64 bits.

Após a execução dos algoritmos propostos foram utilizadas as medidas de precisão (*prec*) e revocação (*rec*) para determinar a qualidade da detecção de núcleos retornada, conforme as Equações (5.7) e (5.8), já apresentadas na Subseção 5.2.4.

A Seção 6.1 descreve a calibração dos parâmetros do pré-processamento e apresenta os melhores valores encontrados para esta etapa. A Seção 6.2 apresenta os experimentos realizados, com calibração automática de parâmetros, e discute hipóteses levantadas a fim de tentar melhorar os resultados. A Seção 6.3 apresenta os resultados obtidos nas etapas de treinamento e teste da DT. Por fim, a Seção 6.4 apresenta e discute os resultados obtidos nos experimentos, além de compará-los com a literatura.

6.1 Pré-Processamento

Para realizar o pré-processamento foi necessário definir o valor de quatro parâmetros, sendo três do SLIC e um do DBSCAN, são eles:

- *kSlic*, que é o número de *superpixels* desejados;
- *mSlic*, como o fator de ponderação entre as diferenças de intensidade e distância;
- *seRadiusSlic*, limite de tamanho considerado para regiões, onde aquelas morfologicamente menores que o limiar unem-se com regiões adjacentes;
- *EDbscan*, limite de valor/distância de tolerância correspondente, que controla quais *superpixels* são agrupados em um único conjunto.

A calibração destes parâmetros usados no pré-processamento foi feita utilizando-se a base de treinamento. Para tal, foi realizada uma busca de parâmetros por força bruta para verificar quais seriam as combinações de parâmetros que implicariam na maior quantidade de detecção de núcleos nas imagens. A força bruta foi escolhida porque o método retorna o mesmo resultado, se executado com os mesmos parâmetros, além de o tempo para testar todas as combinações ser factível. Foram considerados os valores apresentados na Tabela 6.1 para realizar este procedimento.

Tabela 6.1: Valores considerados na força bruta para cada parâmetro.

Parâmetro	**Valores**
kSlic	{500, 1000, 1500, 2000, 2500, 3000}
mSlic	{5, 10, 15, 20, 25, 30, 35, 40}
seRadiusSlic	{0, 1, 1,5}
EDbscan	{5, 6, 7, 8, 9, 10}

Ao todo, a base de treinamento possui 270 núcleos, sendo que foram encontradas 6 melhores combinações de parâmetros que detectaram 269 destes. Estas combinações são apresentadas na Tabela 6.2.

Como todas as combinações mostradas na Tabela 6.2 são equiparáveis, escolheu-se aleatoriamente que os valores utilizados nos demais testes seriam $kSlic = 2000$, $mSlic = 10$, $seRadiusSlic = 1,5$ e $EDbscan = 7$.

Tabela 6.2: Melhores combinações de parâmetros retornadas pela força bruta.

kSlic	*mSlic*	*seRadiusSlic*	*EDbscan*
2000	10	1,5	7
2000	25	1	6
2000	25	1	7
2500	5	1,5	7
2500	10	1,5	6
2500	10	1,5	7

6.2 ILS

Nesta subseção são apresentados os experimentos referentes à abordagem heurística via ILS. Inicialmente, são apresentados os testes feitos com cinco parâmetros CIA na Subseção 6.2.1. Na sequência, foram analisadas três hipóteses: (i) apenas três parâmetros são suficientes para a classificação (Subseção 6.2.2); (ii) apenas dois parâmetros são suficientes para a classificação (Subseção 6.2.3); (iii) a adição de um novo parâmetro pode melhorar a classificação (Subseção 6.2.4). Por fim, a Subseção 6.2.5 apresenta uma comparação resumida do que foi encontrado nos experimentos anteriores.

As calibrações de parâmetros apresentadas nesta seção foram feitas utilizando-se o pacote *irace* (López-Ibáñez et al., 2016). O *irace* é um método que determina, dado um conjunto de instâncias do problema, a melhor combinação de valores para os parâmetros de um algoritmo de otimização.

Além disso, todos os experimentos realizados foram executados 30 vezes, sendo registrados os melhores valores e os valores médios de cada um.

6.2.1 Cinco Parâmetros CIA

Como já apresentado na Seção 5.2.1, inicialmente considerou-se cinco parâmetros CIA na solução, sendo eles: circularidade mínima, circularidade máxima, intensidade máxima, área mínima e área máxima.

Inicialmente, para a realização dos experimentos, seria necessário definir o valor de cinco outros parâmetros: o número máximo de iterações do ILS (*ILSMax* - apresentado

no Algoritmo 2.2), os valores do salto para parâmetros decimais e inteiros (r_{dec} e r_{int}, definidos na Subseção 5.2.3), o número de soluções iniciais e de reinícios do ILS (*nSolRand* e *nStart* - apresentadas na Subseção 5.2.6). Como já dito anteriormente, esses parâmetros foram determinados utilizando-se a ferramenta *irace*.

A Tabela 6.3 apresenta os valores que o *irace* considerou ao definir cada um dos parâmetros.

Tabela 6.3: Valores considerados pelo *irace* para cada parâmetro a ser calibrado.

ILSMax	3, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 100, 150, 200, 250, 300
r_{dec}	0,001, 0,003, 0,005, 0,01, 0,03, 0,05, 0,1, 0,2, 0,3
r_{int}	1, 3, 5, 7, 9
nSolRand	1, 3, 5, 7, 9
nStart	1, 3, 5, 7, 9, 11, 13, 15, 17

Os melhores valores encontrados pelo *irace* foram: $ILSMax = 20$, $r_{dec} = 0,05$, $r_{int} = 5$, $nSolRand = 1$, $nStart = 5$. Estes valores foram os utilizados nos testes, sendo que nas 30 execuções foram obtidos os seguintes resultados:

- precisão média: 0,955 (variando entre 0,949 e 0,967);
- revocação média: 0,801 (variando entre 0,740 e 0,895);
- F1 médio: 0,870 (variando entre 0,832 e 0,926).

Como já explicado no Capítulo 4, a medida de revocação é a mais importante para o problema abordado. Dessa forma, a melhor solução encontrada obteve **precisão de 0,959, revocação de 0,895 e F1 de 0,926**, sendo ela:

$$s^* = \quad \langle\ 0{,}66,\ 1{,}62,\ 150,\ 111,\ 681\ \rangle$$

Além disso, foi possível constatar que, para este caso, a utilização do *Multi-Start* ILS foi melhor do que aplicar o ILS partindo-se da melhor dentre múltiplas soluções iniciais.

6.2.2 Três Parâmetros CIA

Na sequência, levantou-se a hipótese de que os valores máximos da circularidade e da área da base de treino não eram representativos da base de teste, o que limitava o método ILS proposto.

Dessa maneira, utilizou-se novamente o *irace*, com os mesmos valores apresentados na Tabela 6.3, para calibrar os parâmetros do ILS a fim de verificar a hipótese levantada.

Os melhores valores encontrados pelo o *irace* foram: $ILSMax = 50$, $r_{dec} = 0,05$, $r_{int} = 1$, $nSolRand = 1$, $nStart = 15$. Nas 30 execuções de teste obteve-se:

- precisão média: 0,961 (variando entre 0,959 e 0,961);
- revocação média: 0,939;
- F1 médio: 0,946.

A melhor solução encontrada obteve **precisão de 0,961, revocação de 0,939 e F1 de 0,946**, sendo dada por:

$$s^* = \quad \langle\ 0{,}66,\ 152,\ 111\ \rangle.$$

Com isso, foi possível constatar que a hipótese levantada, de que os valores máximos da circularidade e da área da base de treino não eram representativos da base de teste, estava correta. Além disso, também foi possível averiguar que, para esta hipótese, o uso do *Multi-Start* ILS teve melhor desempenho.

6.2.3 Combinação dos Parâmetros CIA Dois a Dois

Outra hipótese levantada é a de que apenas dois dos três parâmetros seriam suficientes para a classificação. Com isso, foi analisada a combinação dois a dois dos parâmetros CIA utilizando-se o pacote *irace*, com os valores da Tabela 6.3, para calibrar os parâmetros do ILS. Os melhores valores encontrados pelo *irace* estão descritos a seguir, de acordo com a combinação realizada:

- **Circularidade e Intensidade (CI):**

- $ILSMax = 40$;
- $r_{dec} = 0,05$;
- $r_{int} = 9$;
- $nSolRand = 5$;
- $nStart = 9$.

- **Circularidade e Área (CA):**
  - $ILSMax = 40$;
  - $r_{dec} = 0,05$;
  - $r_{int} = 9$;
  - $nSolRand = 5$;
  - $nStart = 9$.

- **Intensidade e Área (IA):**
  - $ILSMax = 20$;
  - $r_{dec} = 0,01$;
  - $r_{int} = 3$;
  - $nSolRand = 1$;
  - $nStart = 7$.

A Tabela 6.4 apresenta os resultados médios obtidos nas 30 execuções de cada uma das combinações.

Tabela 6.4: Resultados da combinação dois a dois.

Combinação	Precisão	Revocação	F1	Melhores valores
CI	0,897	0,776	0,832	C = 120, I = 0,71
CA	0,943	0,908	0,925	C = 114.5, A = 0,71
IA	0,933	0,920	0,926	I = 133, A = 111

Portanto, como nenhuma das soluções encontradas obteve melhores resultados do que já foi obtido anteriormente, então foi possível constatar que a hipótese levantada

no início desta seção não é verdadeira. Além disso, para as combinações CI e CA foi possível perceber que a variante de melhor desempenho foi combinar o *Multi-Start* ILS com a estratégia de inicializar o ILS com a melhor dentre múltiplas soluções iniciais. Por outro lado, para a combinação IA, o *Multi-Start* ILS teve o melhor desempenho.

6.2.4 Adição do Parâmetro Excentricidade

Por fim, a última hipótese analisada é a de que a adição de mais um parâmetro, no caso a excentricidade, poderia favorecer a classificação. Novamente foi utilizado o *irace*, com os valores da Tabela 6.3, para calibrar os parâmetros do ILS, sendo que os melhores valores encontrados pelo o *irace* foram: $ILSMax = 300$, $r_{dec} = 0,01$, $r_{int} = 3$, $nSolRand = 7$, $nStart = 15$. Nas 30 execuções de teste obteve-se:

- precisão média: 0,943 (variando entre 0,941 e 0,959);
- revocação média: 0,902 (entre 0,896 e 0,905);
- F1 médio: 0,923 (entre 0,921 e 0,927).

A melhor solução encontrada obteve **precisão de 0,941, revocação de 0,905 e F1 de 0,922**, sendo dada por:

$$s^* = \quad \langle\ 0{,}66,\ 132,\ 110,\ 0{,}83\ \rangle.$$

Portanto, como a solução encontrada também não obteve melhores resultados do que já foi obtido anteriormente, então pode-se afirmar que adicionar o parâmetro de excentricidade não trouxe benefícios, invalidando a hipótese levantada. Além disso, pôde-se perceber que a estratégia de melhor desempenho é aquela que combina o *Multi-Start* ILS com a inicialização do ILS com a melhor de um conjunto de soluções iniciais.

6.2.5 Comparação dos Experimentos Baseados em ILS

De forma resumida, portanto, a Tabela 6.5 apresenta os melhores resultados obtidos em todos os experimentos baseados no método ILS.

Tabela 6.5: Resultados obtidos nos experimentos do ILS.

Parâmetros	Precisão	Revocação	F1
CIA (5)	0,959	0,895	0,926
CIA (3)	**0,961**	**0,939**	**0,946**
CI	0,897	0,776	0,832
CA	0,943	0,908	0,925
IA	0,933	0,920	0,926
CIA + excentricidade	0,941	0,905	0,922

É possível perceber que a estratégia de utilizar apenas três parâmetros CIA (circularidade mínima, intensidade máxima e área mínima) foi a melhor considerando as métricas precisão, revocação e F1.

Além disso, foi possível perceber pelos experimentos que em alguns casos apenas o *Multi-Start* resolveria o problema da necessidade de uma boa solução inicial (ver Subseção 5.2.6), mas em outros casos a combinação do *Multi-Start* com as múltiplas soluções iniciais foi a melhor estratégia. Dessa forma, não é possível escolher a melhor estratégia geral. Contudo, o melhor resultado encontrado utilizou a estratégia *Multi-Start* ILS.

6.3 DT

A DT obtida na etapa de treinamento é apresentada na Figura 6.1. Observamos que a árvore gerada é composta de uma raiz (azul), três nós intermediários (verde) e cinco folhas (amarelo) que correspondem à classificação final: núcleo ou não núcleo, assim como explicado na Seção 2.6.

A variável de destino (mostrada nos números em negrito nos nós folha) foi considerada como um número contínuo, ao invés de um número categórico, fazendo com que a DT forneça classificação entre 0 e 1. Esta estratégia foi utilizada para permitir a escolha de um limiar que distingue melhor as classes.

Como podemos observar na Figura 6.1, as folhas que correspondem a não núcleo possuem valores muito pequenos, próximos de zero. Por outro lado, as correspondentes a núcleo são valores mais altos. Dessa forma, o limiar escolhido e utilizado neste trabalho

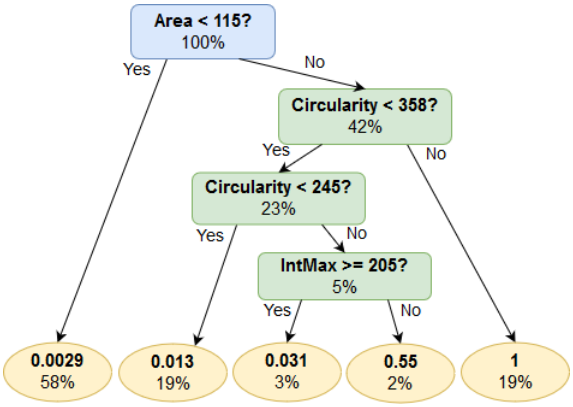


Figura 6.1: DT obtida no treinamento.

foi de 0,5.

Supondo que se deseja classificar um *cluster* com área = 245, circularidade = 364 e intensidade máxima = 292 na DT resultante da etapa de treinamento exposta na Figura 6.1, então, inicia-se caminhando na árvore a partir do nó raiz. Como o *cluster* tem uma área superior a 115, a ramificação indica analisar o nó intermediário da circularidade. Em seguida, como a sua circularidade é superior a 358, então a ramificação indica ir em direção à folha. Como a variável de destino (1) é maior do que o limiar escolhido (0,5), então o *cluster* é classificado como núcleo.

Ao classificar toda a base de teste utilizando-se a DT, obteve-se uma precisão de 0,985, uma revocação de 0,993 e um F1 de 0,989. Neste caso, não foi necessário repetir o experimento 30 vezes porque como as bases de treino e teste sempre serão as mesmas, então a árvore retornada e seu resultado também será igual.

6.4 Discussão e Comparação dos Resultados

O pré-processamento realizado utilizando-se o SLIC e o DBSCAN encontrou 269 dos 270 núcleos presentes na base de treinamento.

No ILS foi feita uma investigação de quais características (atributos) seriam relevantes para a classificação, sendo analisadas quatro combinações: (i) contendo 5 atributos CIA; (ii) contendo 3 atributos CIA; (iii) combinando dois a dois dos 3 atributos CIA; (iv) acrescentando a excentricidade aos 3 atributos CIA. A melhor estratégia encontrada para a base de dados utilizada foi a (ii), sendo que a melhor solução encontrada é

apresentada a seguir:

$$s^* = \quad \langle\ 0{,}66,\ 152,\ 111\ \rangle.$$

Como podemos ver, a circularidade mínima de um *cluster* deveria ser 0,66, a intensidade máxima 152 e a área mínima 111, para que ele fosse considerado um núcleo. Esta solução apresentou **precisão de 0,961, revocação de 0,939 e F1 de 0,946**.

Com os testes da Subseção 6.2 foi possível perceber que não existe apenas uma solução que prevaleça em todos os experimentos para resolver o problema da necessidade de uma boa solução inicial (apresentado na Subseção 2.5). Em alguns casos, apenas o *Multi-Start* foi suficiente, mas em outros a combinação das duas estratégias teve melhor desempenho. Porém, a melhor solução encontrada foi utilizando-se apenas o *Multi-Start* ILS.

Por outro lado, a melhor solução da DT apresentou **precisão de 0,985, revocação de 0,993 e F1 de 0,989**. Com a DT apresentada na Figura 6.1 é possível concluir que a classificação de uma nova instância é feita com, no máximo, 4 passos para definir se um *cluster* é ou não um núcleo.

A Tabela 6.6 apresenta os valores de precisão e revocação obtidos pelas abordagens propostas e por outros métodos da literatura. Além disso, também foi apresentado um resultado parcial obtido do ILS, Diniz et al. (2019), publicado no *21th International Conference on Enterprise Information Systems (ICEIS, 2019)*. Nesse artigo, o método não havia sido calibrado pelo pacote *irace*. Os valores estão ordenados de acordo com o valor de F1.

Como pode ser visto pela Tabela 6.6, a abordagem via DT obteve as melhores medidas de revocação e F1, além de ter a segunda melhor precisão dentre todos os métodos com os quais ela foi comparada. Quanto à abordagem via ILS, é possível constatar que a utilização do *irace* para realizar a calibração automática dos parâmetros melhorou o resultado. Porém, apesar de o *Multi-Start* ILS gerar soluções que superam vários outros métodos da literatura, ele não se destacou em nenhuma das métricas analisadas.

Como o pré-processamento foi desenvolvido em uma linguagem diferente do algoritmo ILS, ele foi executado de maneira isolada e seus resultados foram armazenados para uma futura utilização. Esta é a parte mais demorada de nossa aplicação, pois a metodologia de agrupamento (DBSCAN) é baseada em agrupamentos hierárquicos. Sendo assim, o

Tabela 6.6: Comparação entre métodos para detecção de núcleos.

Estudo	Estratégia	F1	Precisão	Revocação
DT Proposta	Superpixel, DT	**0,989**	0,985	**0,993**
Zhang et al. (2019)	BTTFA	0,980	**0,990**	0,971
Song et al. (2018a)	CSRCM	0,971	0,983	0,959
Tareef et al. (2017)	Superpixel, SVM	0,964	**0,990**	0,940
Multi-Start ILS Proposto	Superpixel, ILS	0,946	0,961	0,939
Diniz et al. (2019)	Superpixel, ILS	0,929	0,985	0,879
Lu et al. (2015)	MSER	0,928	0,977	0,883
Ushizima et al. (2014)	Superpixel	0,926	0,959	0,895
Braz e Lotufo (2017)	CNN	0,923	0,929	0,917
Saha et al. (2016)	CSF, FCM	0,916	0,918	0,915
Nosrati e Hamarneh (2014)	MSER, RF	0,898	0,903	0,893

pré-processamento de cada imagem demora em média 25 segundos, totalizando cerca de 6 horas para processamento de todas as imagens da base de dados utilizada.

Porém, apesar de ser uma fase demorada, sua execução foi realizada somente uma vez, dado que seus resultados foram armazenados. A partir daí, tendo lido esses dados armazenados, a DT e o ILS propostos retornam resultados em menos de um segundo.

Capítulo 7

Considerações Finais

7.1 Conclusões

Este trabalho introduz um método baseado em ILS e outro em DT para detectar núcleos em imagens de células cervicais obtidas em exames de Papanicolaou. O objetivo principal é simular a análise de citopatologistas, uma vez que o exame Papanicolaou utiliza características morfológicas e de distribuição da cromatina no núcleo para detectar anormalidades.

Os dois métodos foram comparados com relação às métricas F1, precisão e revocação, usando-se a base de dados do *Overlapping Cervical Cytology Image Segmentation Challenge*.

Na abordagem heurística baseada em ILS foi feita uma investigação sobre a utilização de atributos considerando circularidade, intensidade, área e excentricidade. Com a realização dos experimentos foi possível constatar que a estratégia de vários reinícios do ILS, denominada *Multi-Start ILS*, tem melhor desempenho que a abordagem ILS convencional. Além disso, também foi evidenciada a importância de utilizar o pacote *irace* para realizar a calibração automática de parâmetros do método, visto que os resultados obtidos com seu uso melhoraram em relação àqueles obtidos sem a aplicação dessa ferramenta.

A melhor solução encontrada pelo *Multi-Start ILS* analisa cada imagem em relação aos valores dos parâmetros de circularidade mínima, intensidade máxima e área mínima. Porém, quando comparado com outros métodos da literatura e a própria abordagem via

DT, ela não se destacou em nenhuma das métricas que foram analisadas.

Por outro lado, a abordagem baseada em DT considerou 15 características morfológicas dos *clusters* candidatos a núcleo. Porém, após uma seleção de atributos, apenas três (circularidade, intensidade máxima e área) foram utilizadas para a classificação. A abordagem produziu resultados que superam os de outros métodos da literatura com relação às medidas de revocação e F1. Seu desempenho em relação à precisão, apesar de não ter superado o melhor resultado da literatura, foi apenas 0,51% abaixo deste.

Sabe-se que a revocação está relacionada ao número de núcleos que o algoritmo encontrou. Portanto, é importante que a revocação dos exames de Papanicolaou seja a mais próxima de um possível, pois a falha em detectar uma lesão pode influenciar o prognóstico.

Por outro lado, como um computador não pode diagnosticar, as imagens devem ser analisadas posteriormente por um patologista. Assim, não é primordial que o método tenha a precisão perfeita, ou seja, não é necessário que todos os *clusters* detectados como núcleos sejam realmente núcleos. Porém, é importante perceber que, quanto maior a precisão, maior é a redução do trabalho do profissional.

Por fim, como trabalho futuro sugere-se a realização de um estudo para analisar a influência de outros parâmetros no método baseado em ILS, a fim de melhorá-lo. Também sugere-se a realização de experimentos utilizando-se imagens reais.

7.2 Publicação Gerada

Como fruto deste trabalho, o seguinte artigo foi publicado:

Título: *An Iterated Local Search Algorithm for Cell Nuclei Detection from Pap Smear Images*

Autores: Débora N. Diniz, Marcone J. F. Souza, Claudia M. Carneiro, Daniela M. Ushizima, Fátima N. Sombra de Medeiros, Paulo H. C. Oliveira e Andrea G. C. Bianchi

Evento: 21th International Conference on Enterprise Information Systems (ICEIS 2019)

Local: Heraklion, Creta, Grécia

Data: 3 a 5 de Maio de 2019

Qualis em Ciência da Computação: B2

Esse artigo recebeu o prêmio de melhor trabalho de estudante na área de Inteligência Artificial e Sistemas de Suporte à Decisão no ICEIS 2019. Além disso, os autores foram convidados a enviar uma versão estendida para o livro da série *Lectures Notes in Business Information Processing* (LNBIP)[1] ou para o *Journal of Information*[2].

[1] https://www.springer.com/series/7911
[2] https://www.mdpi.com/journal/information

Referências Bibliográficas

Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P. e Süsstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 34, p. 2274 – 2282.

Bezdek, J. C.; Ehrlich, R. e Full, W. (1984). Fcm: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, v. 10, n. 2-3, p. 191–203.

Bradley, A. P.; Carneiro, G. e Lu, Z. Evaluation metric, (2014). URL `https://cs.adelaide.edu.au/~carneiro/isbi14_challenge/evaluation.html`. Acessado em 01/04/2019.

Braz, E. F. e Lotufo, R. A. (2017). Nuclei detection using deep learning. *Anais do XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*, p. 1059–1063, São Pedro - Brasil.

Breiman, L. (2001). Random forests. *Machine learning*, v. 45, n. 1, p. 5–32.

Breiman, L.; Friedman, J. H.; Olshen, R. A. e Stone, C. J. (1984). *Classification and regression trees*. Wadsworth Brooks/Cole Advanced Books Software, Monterey - CA.

Burke, E. K.; Newall, J. P. e Weare, R. F. (1995). A memetic algorithm for university exam timetabling. *International Conference on the Practice and Theory of Automated Timetabling*, p. 241–250. Springer, (1995).

Chan, T. F. e Vese, L. A. (2001). Active contours without edges. *IEEE Transactions on image processing*, v. 10, n. 2, p. 266–277.

Chao, W. e Junzheng, W. (2018). Cloud-service decision tree classification for education platform. *Cognitive Systems Research*, v. 52, p. 234 – 239.

Coelho, V. N.; Grasas, A.; Ramalhinho, H.; Coelho, I. M.; Souza, M. J. F. e Cruz, R. C. (2016). An ils-based algorithm to solve a large-scale real heterogeneous fleet vrp with multi-trips and docking constraints. *European Journal of Operational Research*, v. 250, n. 2, p. 367 – 376.

Dantas, A. C.; Melo, S.; Moura, F. e Fernandes, M. (2015). Reconhecimento dinâmico de emoções através de expressões faciais utilizando árvore de decisão. *Simpósio Brasileiro de Informática na Educação*, p. 1102–1111, Maceió - Brasil.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, v. 26, n. 3, p. 297–302.

Diniz, D. N.; Souza, M. J. F.; Carneiro, C. M.; Ushizima, D. M.; Sombra, F. N. S. M.; Oliveira, P. H. C. e Bianchi, A. G. C. (2019). An iterated local search algorithm for cell nuclei detection from pap smear images. *Proceedings of the 21st International Conference on Enterprise Information Systems*, p. 319–327, Setubal. INSTICC.

Duda, R. O.; Hart, P. E. e Stork, D. G. (2000). *Pattern Classification*. Wiley-Interscience, New York - USA, 2ª edição.

Ester, M.; Kriegel, H.; Sander, J. e Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, p. 226–231, Portland - Oregon. AAAI Press.

Hansen, P. e Mladenović, N. (2006). First vs. best improvement: An empirical study. *Discrete Applied Mathematics*, v. 154, n. 5, p. 802 – 817.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. *European conference on machine learning*, p. 137–142, Berlin - Alemanha. Springer.

Justino, R.; Martines, M. e Kawakubo, F. (2017). Classificação do uso da terra e cobertura vegetal utilizando técnicas de mineração de dados. *Revista do Departamento de Geografia*, v. 33, p. 36–46.

Kovesi, P. D. Matlab and octave functions for computer vision and image processing, (2000). URL `https://www.peterkovesi.com/matlabfns/`. Acessado em 20/09/2018.

Krizhevsky, A.; Sutskever, I. e Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, p. 1097–1105, Lake Tahoe - USA. Curran Associates Inc.

Lee, H. e Kim, J. (2016). Segmentation of overlapping cervical cells in microscopic images with superpixel partitioning and cell-wise contour refinement. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, p. 63–69, Las Vegas - USA. IEEE.

Lindoff, G. S. e Berry, M. J. A. (2011). *Data Mining Techniques: For Marketing, Sales, and Costumer Relationship Management*. Wiley Publishing, Inc., 3ª edição.

Lourenço, H. R.; Martin, O. C. e Stützle, T. (2010). *Iterated Local Search: Framework and Applications*, volume 146 of *International Series in Operations Research Management Science*, p. 363–397. Kluwer Academic Publishers, 2ª edição.

Lu, Z.; Carneiro, G. e Bradley, A. P. (2013). Automated nucleus and cytoplasm segmentation of overlapping cervical cells. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, p. 452–460. Springer, (2013).

Lu, Z.; Carneiro, G. e Bradley, A. P. (2015). An improved joint optimization of multiple level set functions for the segmentation of overlapping cervical cells. *IEEE Transactions on Image Processing*, v. 24, n. 4, p. 1261–1272.

Lussier, P.; Deslauriers-Varin, N.; Collin-Santerre, J. e Bélanger, R. (2019). Using decision tree algorithms to screen individuals at risk of entry into sexual recidivism. *Journal of Criminal Justice*, v. 63, p. 12 – 24.

López-Ibáñez, M.; Dubois-Lacoste, J.; Cáceres, L. P.; Birattari, M. e Stützle, T. (2016). The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, v. 3, p. 43 – 58.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, p. 281–297, Berkeley - California. University of California Press.

Mammas, I. N. e Spandidos, D. A. (2012). George n. papanicolaou (1883-1962): Fifty years after the death of a great doctor, scientist and humanitarian. *Balkan Union of Oncology*, v. 17, n. 1, p. 180–184.

Manning, C. D. e Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT press, Cambridge - USA.

Mariarputham, E. J. e Stephen, A. (2015). Nominated texture based cervical cancer classification. *Computational and Mathematical Methods in Medicine*, v. 2015.

Martí, R.; Resende, M. G. C. e Ribeiro, C. C. (2013). Multi-start methods for combinatorial optimization. *European Journal of Operational Research*, v. 226, n. 1, p. 1 – 8.

Matas, J.; Chum, O.; Urban, M. e Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, v. 22, n. 10, p. 761–767.

McGuire, R. G. (1992). Reporting of objective color measurements. *HortScience*, v. 27, n. 12, p. 1254–1255.

Medeiros, L. B.; Trigueiro, D. R. S. G.; Silva, D. M.; Nascimento, J. A.; Monroe, A. A.; Nogueira, J. A. e Leadebal, O. D. C. P. (2016). Integração entre serviços de saúde no cuidado às pessoas vivendo com aids: uma abordagem utilizando árvore de decisão. *Ciência Saúde Coletiva*, v. 21, p. 543 – 552.

Moshavegh, R.; Bejnordi, B. E.; Mehnert, A.; Sujathan, K.; Malm, P. e Bengtsson, E. (2012). Automated segmentation of free-lying cell nuclei in pap smears for malignancy-associated change analysis. *Engineering in Medicine and Biology Society*, p. 5372–5375, San Diego - USA. IEEE.

Nosrati, M. S. e Hamarneh, G. (2014). A variational approach for overlapping cell segmentation. *ISBI Overlapping Cervical Cytology Image Segmentation Challenge*, p. 1–2.

Oliveira, P. H. C.; Moreira, G.; Sabino, D. M. U.; Carneiro, C. M.; Medeiros, F. N. S.; Araújo, F. H. D.; Silva, R. R. V. e Bianchi, A. G. C. (2017). A multi-objective approach for calibration and detection of cervical cells nuclei. *2017 IEEE Congress on Evolutionary Computation (CEC)*, p. 2321–2327.

Plissiti, M. E.; Nikou, C. e Charchanti, A. (2011). Automated detection of cell nuclei in pap smear images using morphological reconstruction and clustering. *IEEE Transactions on Information Technology in Biomedicine*, v. 15, n. 2, p. 233–241.

Ren, X. e Malik, J. (2003). Learning a classification model for segmentation. *Proceedings Ninth IEEE International Conference on Computer Vision*, p. 10–17, Nice - França. IEEE.

Saha, R.; Bajger, M. e Lee, G. (2016). Spatial shape constrained fuzzy c-means (fcm) clustering for nucleus segmentation in pap smear images. *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, p. 1–8, Gold Coast - Australia. IEEE.

Samsudin, N. A.; Mustapha, A.; Arbaiy, N. e Hamid, I. R. A. (2016). Extended local mean-based nonparametric classifier for cervical cancer screening. *Proceedings of the International Conference on Soft Computing and Data Mining*, p. 386–395, Bandung - Indonesia. Springer.

Song, J.; Xiao, L. e Lian, Z. (2018)a. Contour-seed pairs learning-based framework for simultaneously detecting and segmenting various overlapping cells/nuclei in microscopy images. *IEEE Transactions on Image Processing*, v. 27, n. 12, p. 5759–5774.

Song, T.; Liu, S.; Tang, X.; Peng, X. e Chen, M. (2018)b. An iterated local search algorithm for the university course timetabling problem. *Applied Soft Computing*, v. 68, p. 597–608.

Song, Y.; Zhang, L.; Chen, S.; Ni, D.; Li, B.; Zhou, Y.; Lei, B. e Wang, T. (2014). A deep learning based framework for accurate segmentation of cervical cytoplasm and nuclei. *Engineering in Medicine and Biology Society (EMBC)*, p. 2903–2906, Chicago - USA. IEEE.

Stützle, T. *Local search algorithms for combinatorial problems: Analysis, Improvements, and New Applications*. Tese de doutorado, Darmstadt University of Technology, Germany, (1998).

Tareef, A.; Song, Y.; Cai, W.; Huang, H.; Chang, H.; Wang, Y.; Fulham, M.; Feng, D. e Chen, M. (2017). Automatic segmentation of overlapping cervical smear cells based on local distinctive features and guided shape deformation. *Neurocomputing*, v. 221, p. 94–107.

Traut, H. F. e Papanicolaou, G. N. (1943). Cancer of the uterus: the vaginal smear in its diagnosis. *California and western medicine*, v. 59, n. 2, p. 121.

Ushizima, D.; Bianchi, A. e Carneiro, C. (2014). Segmentation of subcellular compartments combining superpixel representation with voronoi diagrams. *Proceedings of the International Symposium on Biomedical Imaging*, Beijing - China. Elsevier.

Xing, F.; Xie, Y. e Yang, L. (2015). An automatic learning-based framework for robust nucleus segmentation. *IEEE Transactions on Medical Imaging*, v. 35, n. 2, p. 550–566.

Zhang, J.; Liu, Z.; Du, B.; He, J.; Li, G. e Chen, D. (2019). Binary tree-like network with two-path fusion attention feature for cervical cell nucleus segmentation. *Computers in Biology and Medicine*, v. 108, p. 223—233.

Zhong, L. e Najarian, K. (2001). Automated classification of pap smear tests using neural networks. *Proceedings of the International Joint Conference on Neural Networks*, p. 2899–2901, Washington - USA. IEEE.

Zhou, Y. e Hao, J. (2017). An iterated local search algorithm for the minimum differential dispersion problem. *Knowledge-Based Systems*, v. 125, p. 26 – 38.