



Detecção e Quantificação de Neurónios em *C. elegans*

DANIEL CANASTRO DIAS
Outubro de 2015

DETEÇÃO E QUANTIFICAÇÃO DE NEURÓNIOS EM C. ELEGANS

Daniel Canastro Dias



Departamento de Física

Mestrado em Engenharia Computação e Instrumentação Médica

2015

Relatório elaborado para satisfação parcial dos requisitos da Unidade Curricular de
Tese/Dissertação do Mestrado em Engenharia de Computação e Instrumentação Médica

Candidato: Daniel Canastro Dias, Nº 1130181, 1130182@isep.ipp.pt

Orientação científica:

Professor Doutor Jorge Manuel Fernandes dos Santos, jms@isep.ipp.pt;

Doutor Ricardo Gamelas Sousa, rsousa@rsousa.org



Departamento de Física

Mestrado em Engenharia de Computação e Instrumentação Médica

2015

Agradecimentos

Quero agradecer ao Professor Jorge Santos, por me ter dado a oportunidade de elaborar este projeto e pelo contínuo apoio durante o seu desenvolvimento.

Ao Doutor Ricardo Sousa, pelo constante apoio e paciência naquelas longas tardes na sala Octopus do INEB.

À Doutora Conceição Pereira por ter requisitado o tema em questão e por sempre se ter demonstrado disponível para esclarecer qualquer dúvida.

Resumo

Nesta dissertação é apresentado um estudo dos sistemas de processamento automático de imagem em contexto de um problema relacionado com a individualização de neurónios em imagens da nematoda *C. elegans* durante estudos relacionados com a doença de Parkinson.

Apresenta-se uma breve introdução à anatomia do verme, uma introdução à doença de Parkinson e uso do *C. elegans* em estudos relacionados e também é feita a análise de artigos em contexto de processamento de imagem para contextualizar a situação atual de soluções para o problema de extração de características e regiões específicas.

Neste projeto é desenvolvida uma *pipeline* com o auxílio do *software* CellProfiler para procurar uma resposta para o problema em questão.

Palavras-Chave

C. elegans; CellProfiler; Processamento de imagem, Extração de características.

Abstract

This thesis presents a study of an automatic image processing system in the context of a problem related to the individualization of neurons in the nematode *C. elegans* images for studies related to Parkinson's disease.

It presents a brief introduction to the worm's anatomy, an introduction to Parkinson's disease and use of *C. elegans* in related studies and is also made to articles analysis in image processing context to contextualize the current situation solutions to the problem extraction of features and specific regions.

In this project is developed a pipeline with the help of CellProfiler software to look for an answer to the problem at hand.

Keywords

C. elegans; CellProfiler; Image processing, Feature extraction.

Índice

AGRADECIMENTOS.....	III
RESUMO	IV
ABSTRACT	V
ÍNDICE	1
ÍNDICE DE FIGURAS	3
ÍNDICE DE TABELAS	7
ACRÓNIMOS.....	8
1. INTRODUÇÃO.....	10
1.1.MOTIVAÇÃO E OBJETIVO	11
1.2.ORGANIZAÇÃO DO RELATÓRIO	11
2. CONTEXTUALIZAÇÃO DO PROBLEMA.....	13
2.1.CAENORHABDITIS ELEGANS	13
2.2.DOENÇA DE PARKINSON	16
2.3.C. ELEGANS E PARKINSON	17
2.4.TRABALHOS RELACIONADOS	18
3. DETEÇÃO AUTOMÁTICA DOS NEURÓNIOS DO C.ELEGANS.....	22
3.1.CELLPROFILER E FUNCIONAMENTO DOS SEUS MÓDULOS	25
3.2.MÓDULOS DA PIPELINE.....	25
3.2.1. <i>Pré – Processamento</i>	27
3.2.2. <i>Deteção Dos Corpos Celulares</i>	31
3.2.3. <i>Filtragem</i>	32
3.3.ANÁLISE DOS DADOS	35
3.4.DETEÇÃO DOS AXÓNIOS	41
4. RESULTADOS E DISCUSSÃO	43
5. CONCLUSÃO.....	55
REFERÊNCIAS	56
ANEXO A. FUNCIONAMENTO DOS MÓDULOS DO CELLPROFILER	58

**ANEXO B. TABELA DAS CARACTERÍSTICAS DOS OBJETOS DETETADOS PELO
CELLPROFILER 95**

Índice de Figuras

Figura 1 Comparação de <i>C. elegans</i> com neurónios degradados (esquerda) e um saudável (direita), com Corpos celulares pintados a azul e axónios a vermelho.	10
Figura 2 Anatomia de <i>C. elegans</i> adulto [2].	14
Figura 3 Plano corporal da <i>C. elegans</i> com secções transversais da cabeça à cauda [2].	14
Figura 4 Esquema ilustrativo do sistema nervoso do <i>C. elegans</i> [13].	15
Figura 5 Método de Extração da Árvore Dendrítica [10].	20
Figura 6 Exemplo de uma Imagem de <i>C. elegans</i> retirada usando o microscópio <i>Axio Observer Z1 Feiss</i> .	23
Figura 7 Imagem com os Corpos Celulares (a Azul) e os Axónios (a Vermelho) identificados	23
Figura 8 Fluxograma do <i>Pipeline</i> elaborado.	24
Figura 9 Configuração do Módulo <i>Images</i> .	26
Figura 10 Configuração do Módulo <i>NamesAndTypes</i> .	26
Figura 11 Configuração do Módulo <i>Crop</i> .	27
Figura 12 Janela de seleção da área para realizar o <i>crop</i> da Imagem original	28
Figura 13 Configuração do Módulo <i>ApplyThreshold</i> .	29
Figura 14 Máscara Extraída do Módulo <i>ApplyThreshold</i> (à direita) em comparação com o corte feito no módulo <i>Crop</i> (à esquerda).	29
Figura 15 Configuração do Módulo <i>ImageMath</i> .	30
Figura 16 Configuração do Módulo <i>EnhanceandSurpressFeatures</i> .	30
Figura 17 Imagem após Pré-processamento.	31

Figura 18 Configuração do Módulo <i>IdentifyPrimaryObjects</i> .	31
Figura 19 Configuração do Módulo <i>FilterObjects</i> utilizando a área.	32
Figura 20 Configuração do Módulo <i>FilterObjects</i> utilizando a excentricidade.	33
Figura 21 Configuração do Módulo <i>EditObjectsManually</i> .	33
Figura 22 Janela de edição manual de objetos com exemplos de objetos eliminados (limites a tracejado) e objetos em fase de edição (pontos vermelhos como limite).	34
Figura 23 Configuração da janela <i>ExportToSpreadsheet</i> .	35
Figura 24 Janela de seleção de medições.	36
Figura 25 <i>Outlines</i> dos Corpos Celulares.	37
Figura 27 Exemplo de uma imagem com apenas os <i>outlines</i> de objetos à esquerda e respetiva imagem original à direita.	38
Figura 28 Exemplo de uma imagem de união dos <i>outlines</i> com a imagem original (imagem a verde para se notar melhor os <i>outlines</i>).	38
Figura 30 Exemplo de imagem com anotações do perito.	40
Figura 31 Extrato do algoritmo de <i>Matlab</i> para o cálculo do Coeficiente de Dice e a exportação dos dados para um ficheiro de texto.	41
Figura 32 Configuração da janela <i>IdentifySecondaryObjects</i> .	42
Figura 33 Configuração da janela <i>FilterObjects</i> pela área dos axónio	42
Figura 34 Exemplo de resultado não usando qualquer tipo de incremento de qualidade. À esquerda é possível ver o número de objetos que foram considerados como positivos e à direita as fronteiras desses objetos.	44
Figura 35 Exemplo de resultado de incremento de qualidade usando o módulo <i>EnhanceEdges</i> .	44

Figura 36 Exemplo de resultado de incremento de qualidade não usando o módulo <i>EnhanceEdges</i> .	45
Figura 37 Exemplo de resultado de incremento de qualidade melhorando apenas os <i>Speckles</i> da Imagem.	45
Figura 38 Exemplo de detecção de objetos com diâmetro mínimo de 30 unidades de <i>pixel</i> .	46
Figura 39 Exemplo de detecção de objetos com diâmetro mínimo de 10 unidades de <i>pixel</i> . A vermelho encontra-se assinalado um corpo celular que anteriormente não se encontrava detetado.	46
Figura 40 Exemplo de uma imagem com mais que um verme.	47
Figura 41 Exemplo de uma imagem com mais que um verme, mas após a aplicação de uma máscara com o local desejado à detecção de Corpos Celulares.	47
Figura 42 Exemplo do resultado de detecção de axónios antes de ocorrer qualquer tipo de filtragem.	54
Figura 43 Exemplo do resultado de detecção de axónios após a aplicação dos filtros.	54
Figura 44 Excentricidade de uma elipse [12].	82

Índice de Tabelas

Tabela 1	Regime de Crescimento e extração de imagens de <i>C. elegans</i> .	22
Tabela 2	Mínimos, Máximos e Média da Área, Excentricidade e Intensidade dos Corpos Celulares das dez primeiras imagens.	48
Tabela 3	Dados dos Corpos Celulares após aplicada filtragem com os valores da Tabela 2.	48
Tabela 4	Dados dos Corpos Celulares eliminando a filtragem por Intensidade Integrada.	49
Tabela 5	Dados dos Corpos Celulares após a introdução da Filtragem Manual.	51

Acrónimos

CR	–	C terminal de Ras
DP	–	Doença de Parkinson
DTE	–	Extração de Árvore Dendrítica
FN	–	Falso Negativo
FP	–	Falso Positivo
FPR	–	Taxa de Falsos Positivos
INEB	–	Instituto Nacional de Engenharia Biomédica
ISEP	–	Instituto Superior de Engenharia do Porto
LRRK2	–	Leucina Quinase 2
MER	–	Mínimo Retângulo Envolvente
MIP	–	Projeção Intensidade Máxima
MLE	–	Estimador de Máxima Verossimilhança
MR8	–	Resposta Máxima 8
FRS	–	Filtros Root
ROC	–	Receiver Operating Curve
TN	–	Verdadeiro Negativo
TP	–	Verdadeiro Positivo
TPR	–	Taxa de Verdadeira Positivos
TRAP	–	Tremor, Rigidez, Acinesia, instabilidade Postural
UPDRS	–	Unified Parkinson's Disease Rating Scale

1. INTRODUÇÃO

O *Caenorhabditis elegans* é uma nematoda pequeno e transparente, do qual se consegue adquirir imagens facilmente sem necessidade de recurso a técnicas de processamento de amostra. No caso do problema proposto para a tese, é estudada a doença de Parkinson em modelos de *C. elegans* através da alteração do seu genoma e posteriores consequências. Esta nematoda possui um total de 8 neurónios dopaminérgicos (subpopulação habitualmente atingida pela doença), dos quais 6 estão presentes na região da cabeça, a região em que se adquire a imagem (ver Figura 1). Na imagem à esquerda da Figura 1, observa-se os efeitos da degradação neuronal, exposto pelos pequenos inchaços ao longo do que resta dos axónios do verme e também por alguma disformidade nos Corpos Celulares. A integridade destes neurónios é uma medida direta da manifestação da doença. Neste tipo de imagem é de interesse a deteção e contagem dos corpos celulares dos neurónios, medição da área total ocupada por eles e avaliar a integridade das neurites, de maneira a que integrando estes dados se possa atribuir um nível de neurodegeneração.

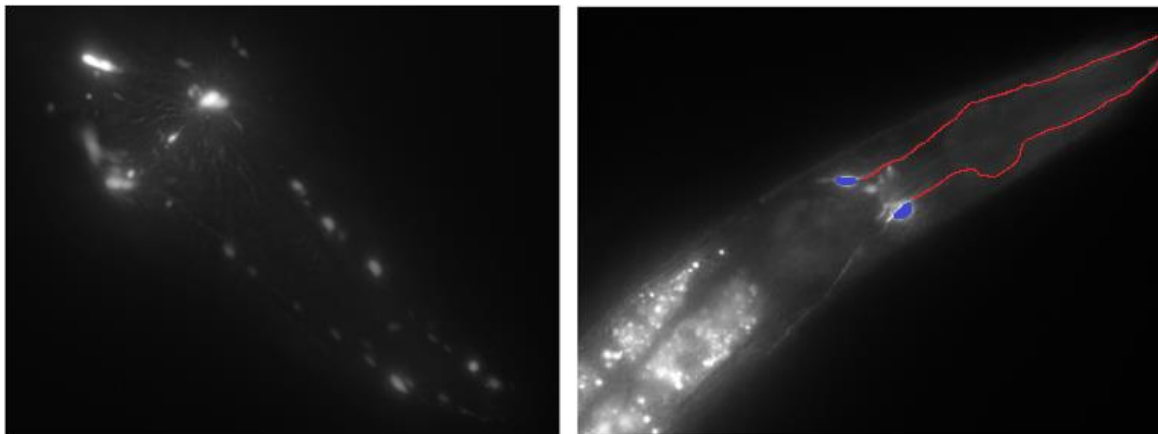


Figura 1 Comparação de *C. elegans* com neurónios degradados (esquerda) e um saudável (direita), com Corpos celulares pintados a azul e axónios a vermelho.

A análise feita às imagens nas presentes condições baseia-se muito no operador, o que sabemos estar sujeito à introdução de diversos erros que dificilmente são rastreáveis. Por isso, é importante automatizar este processo. O objetivo do trabalho proposto será pois o de

desenvolver um processo automático para análise de imagens de *C. elegans* e extração das respectivas características, como o número de corpos celulares visto na imagem a sua intensidade integrada, o número de axónios presentes e a sua área. Primeiramente é feita uma abordagem da anatomia do *C. elegans* e uma breve introdução à doença de Parkinson e sua relação com a nematoda para fazer uma ponte de conhecimento antes da passagem para a parte mais prática do trabalho. Também são mencionados dois artigos que apresentam um problema similar ao apresentado, ajudando a situar um pouco a situação atual da segmentação de *C.elegans*. Visto na metodologia desenvolvida ter sido utilizado um *software* específico, é importante referir que a explicação do funcionamento dos módulos utilizados encontra-se no Anexo A.

1.1. MOTIVAÇÃO E OBJETIVO

Na área da biologia molecular e celular, muitas vezes é necessário analisar imagens microscópicas de animais cujo código genético foi alterado e é necessário determinar alterações no seu fenótipo, ou seja, nas suas características observáveis como comportamento, morfologia e desenvolvimento, mas tal procedimento de análise de imagem é um processo moroso e com poucas ferramentas que possibilitem uma análise detalhada de elevadas quantidades de imagens. De forma a aliviar o fardo de analisar centenas de imagens e acelerar o processo seria conveniente a existência de uma ferramenta automática de análise e processamento de imagens biológicas.

O objetivo do trabalho proposto é então o de desenvolver um processo automatizado para análise de imagens da *C. elegans* e extração de características, para tal utiliza-se uma aplicação *open source* para tratamento de imagem que possa ser integrada no processo existente e que dê resposta aos objetivos do trabalho.

1.2. ORGANIZAÇÃO DO RELATÓRIO

O presente relatório encontra-se estruturado em cinco partes, sendo que, na primeira parte é feita a Introdução temática, definido o Objetivo a atingir, descrita a Metodologia de abordagem adotada e descrita a presente Estrutura. Na segunda parte é apresentada toda a Revisão Bibliográfica necessária à fundamentação do trabalho desenvolvido. Na terceira parte apresenta-se, de forma desenvolvida, todo o trabalho prático efetuado, na quarta parte,

apresentam-se os resultados obtidos durante o desenvolvimento do trabalho prático e na quinta parte são feitas as considerações finais sobre o trabalho realizado.

2. CONTEXTUALIZAÇÃO DO PROBLEMA

2.1. CAENORHABDITIS ELEGANS

O *Caenorhabditis elegans* (ver Figura 2) é uma pequena nematoda do solo (cerca de 1 mm de comprimento) que é utilizado como organismo modelo para vários estudos em laboratório. Nomeadamente para estudos de biologia do desenvolvimento, genética, envelhecimento e de ecotoxicologia. Este organismo possui uma série de características que o torna ideal para este tipo de estudos: possui um ciclo de vida curto, dimensões reduzidas, um pequeno genoma e é fácil de cultivar [1].

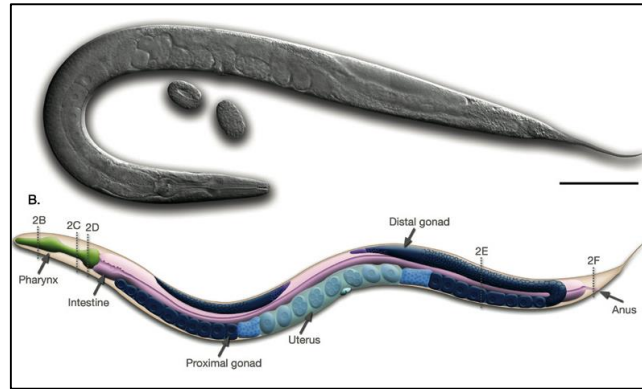


Figura 2 Anatomia de *C. elegans* adulto [2].

Semelhante a outras nematodas, tem um corpo não segmentado, cilíndrico, que é afunilado nas extremidades. Possui um plano de corpo da nematoda típico com um tubo exterior e um tubo interior separados um do outro (ver Figura 3). O tubo exterior (parede do corpo) consiste de cutícula, hipoderme, sistema excretor, neurónios e músculos, e o tubo interior, de faringe, do intestino e, no adulto, gônadas. Todos estes tecidos estão sob uma pressão hidrostática interna, regulada por um sistema osmorregulatório [2].

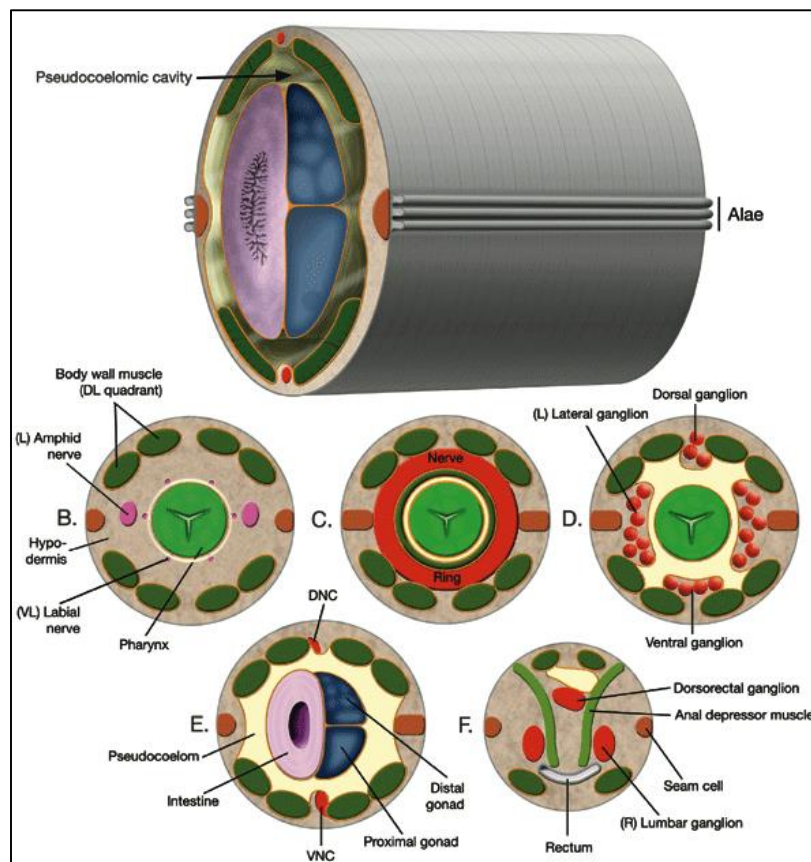


Figura 3 Plano corporal da *C. elegans* com secções transversais da cabeça à cauda [2].

Na Figura 3 o esquema A é a parede do corpo (tubo exterior) e é separada do tubo interior por um pseudoceloma, as linhas laranja indicam lâminas basais. B é a secção da cabeça anterior, C a secção no meio da cabeça, D a secção da cabeça posterior, E a secção através do corpo posterior (DNC significa cordão nervoso dorsal e VNC cordão nervoso ventral) e F é a secção da cauda [2].

A nematoda, tal como outros organismos, também tem os seus órgãos aglomerados em sistemas de funcionamento.

1) Sistema epitelial

A hipoderme, que segrega cutícula, é constituída pelo corpo principal de sincício, uma série de anéis concêntricos de cinco células sinciciais menores na cabeça, três células uninucleadas e uma célula sincicial na cauda. Nos lados, a hipoderme é interrompida pela linha sincicial de células que se formam na superfície da asa da cutícula durante determinados estágios de desenvolvimento [2].

2) Sistema Nervoso

As células do sistema nervoso estão organizadas em gânglios da cabeça e da cauda. A maioria dos neurónios do *C. elegans* está localizada na cabeça em torno da faringe (ver Figura 4). No corpo, uma linha contínua de corpos celulares dos neurónios reside na linha média, adjacente à hipoderme ventral. Além disso, há dois gânglios póstero-laterais sobre os lados, bem como alguns neurónios espalhados ao longo do corpo lateral. Os processos da maioria dos neurónios viajam no cordão nervoso ventral ou dorsal e projetam para o anel do nervo na cabeça, que constitui o maior neuropil do animal [2].

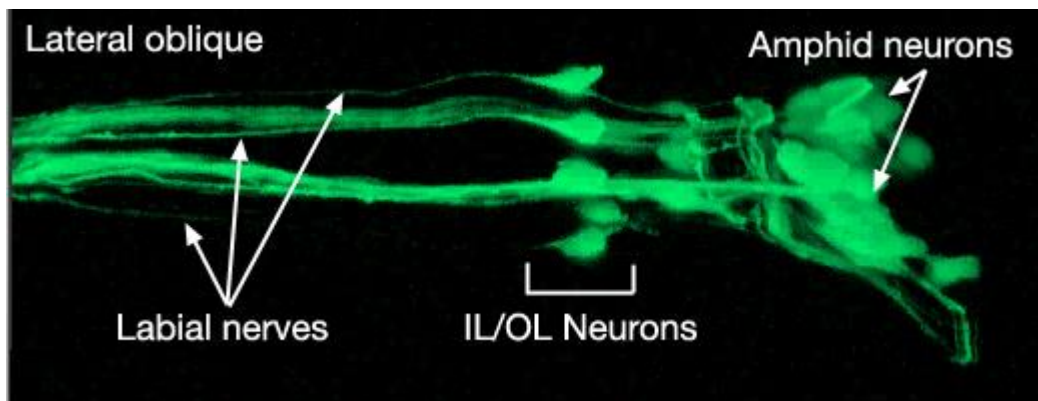


Figura 4 Esquema ilustrativo do sistema nervoso do *C. elegans* [13].

3) *Sistema muscular*

Neurónios e hipoderme são separados da musculatura por uma fina lâmina basal. Os músculos recebem a entrada dos neurónios através do envio de braços musculares para processos do neurónio motor que correm ao longo das cordas nervosas ou residem no anel nervoso. Os músculos estriados da parede do corpo estão dispostos obliquamente em tiras em quatro quadrantes, dois dorsais e dois ventrais, ao longo de todo o comprimento do animal [2].

4) *Sistema excretor*

Quatro células situadas no lado ventral da cabeça posterior compõem o sistema excretor, que participa na osmorregulação e eliminação de resíduos. Este abre-se para o exterior através do poro excretor [2].

2.2. DOENÇA DE PARKINSON

A doença de Parkinson inclui-se na família das doenças caracterizadas pelo distúrbio do movimento. O termo distúrbio do movimento refere-se a doenças neurológicas que causam distúrbios de movimento e não é atribuível a fraqueza motora ou espasticidade, perda sensorial, ou ataxia cerebelar. Distúrbios do movimento caracterizam-se pela pobreza e pela lentidão dos movimentos e o aumento do tónus ou por movimentos involuntários anormais. A maioria dos distúrbios de movimento ocorre com a disfunção das estruturas cerebrais subcorticais que compreendem os gânglios basais [3].

Existem quatro características fundamentais da Doença de Parkinson (DP) que podem ser agrupados sob a sigla TRAP: Tremor em repouso, Rigidez, Acinesia (ou bradicinesia) e instabilidade Postural. Devido aos diversos perfis e estilos de vida das pessoas afetadas pela DP, as deficiências motoras e não motoras devem ser avaliadas no contexto das necessidades de cada paciente. São utilizadas escalas de avaliação para avaliar a deficiência motora e incapacidade em pacientes com DP. A UPDRS (Unified Parkinson's Disease Rating Scale) é a escala mais bem estabelecida para a avaliação da incapacidade. Estudos que fazem uso de UPDRS para controlar a progressão da DP sugerem que o seu curso não é linear e que a taxa de degradação é variável e mais rápida na fase inicial da doença e em pacientes com dificuldade de marcha e instabilidade postural [4].

2.3. C. ELEGANS E PARKINSON

A Doença de Parkinson é o distúrbio de movimento neurodegenerativo mais comum. Pelo menos nove genes diferentes são conhecidos por causar DP familiar. Os estudos funcionais de genes relacionados com a DP implicam a disfunção das mitocôndrias, a agregação de proteínas, e a resposta ao *stress* na fisiopatologia da DP. As mutações autossômicas dominantes nas repetições ricas em leucina quinase 2 (LRRK2) estão associadas com a DP familiar e de início tardio. A mutação G2019S é a mutação mais comum e está presente em 40% das DP familiar e esporádica em amostras africanas, 30% das DP familiares em populações judaicas asquenazes e até 3% da DP esporádica na Europa e América do Norte. [5].

LRRK2 é um membro da família de proteínas ROCO e contém múltiplos domínios, incluindo um domínio de repetição rica em leucina, Roc GTPase, CR (C-terminal de Ras) de função desconhecida, domínio cinase e do domínio WD40-repetição no terminal C. LRRK2 está presente no citoplasma e também está associada a membranas (mitocôndria, retículo endoplasmático, e vesículas sinápticas). Estudos recentes demonstraram que as mutações, tais como G2019S, conduzem ao aumento da autofosforilação da LRRK2 e a atividade de cinase. A expressão transiente de formas mutantes de LRRK2 em neurónios provoca a morte celular. A expressão de LRRK2 mutante nos neurónios também diminui o comprimento da neurite em roedores. Os mamíferos têm um homólogo perto de LRRK2, denominado IRAK1 [5].

A evidência bioquímica indica que as mutações causadoras de doenças estão ligadas à GTPase e às atividades de quinase aberrante, e que estas modificações podem estar na base de toxicidade neuronal e patogênese de DP [6].

O *Caenorhabditis elegans* é um excelente organismo para modelar doenças degenerativas neuronais e tem sido usado para estudar o mecanismo de degeneração em modelos de DP e de outras doenças neurodegenerativas. O gene LRRK2 é altamente conservado entre espécies. O *C. elegans* tem um único ortólogo de LRRK2 humano. O ortólogo de LRRK1 e 2 em *Caenorhabditis elegans* é Irk-1. A disfunção das mitocôndrias está implicada na patofisiologia da DP. Por exemplo, α -sinucleína está associada à disfunção das mitocôndrias e do retículo endoplasmático. Parkin, PINK-1, e DJ-1, que são genes associados com DP hereditário, também estão associados com a função mitocondrial. Estas observações levantam a possibilidade de que LRRK2 também modula a função mitocondrial [5]. A

Expressão transgênica de LRRK2 e as mutações causadoras de doenças de LRRK2 em *C. elegans* resultam numa perda de neurónios dopaminérgicos e défices comportamentais. Estes modelos transgênicos reproduziram várias características-chave do parkinsonismo humano, indicando que a sobre-expressão de LRRK2 em *C. elegans* pode fornecer um modelo valioso para testes pré-clínicos de compostos para DP [6].

Atualmente, não há tratamento terapêutico para reduzir a velocidade ou melhorar a degeneração de neurónios dopaminérgicos na DP [6].

2.4. TRABALHOS RELACIONADOS

Apesar de não existir disponível um artigo antecessor que elabore uma segmentação de imagens como aquelas trabalhadas nesta dissertação é importante situar o leitor de como alguns trabalhos mais recentes executam as tarefas de segmentação de imagens para extrair características de *C. elegans*.

Em [9] explora-se o uso de técnicas de processamento de imagem em conjunto com os processos de análise de imagem para segmentar automaticamente gravações à escala de cinza de várias estirpes mutantes de *C. elegans*.

Antes de segmentação, cada *frame* do vídeo é pré-processado usando algumas técnicas de processamento de imagem para remover o ruído e outros artefactos que possam fazer com que o processo de segmentação falhe.

Para identificar os *pixels* que pertencem ao corpo do verme, a imagem sofre um *threshold* baseado em valores de intensidade, denotando *pixels* abaixo de um certo valor como pertencendo ao verme. O valor limite é identificado automaticamente através da procura de um valor que produz apenas um segmento contíguo (presumivelmente do verme), com uma área mínima especificada. A área de um segmento é calculada como o número total de *pixels* cuja intensidade encontra-se abaixo do *threshold*. A partir dos dados adquiridos de outras gravações, o número médio de *pixels* no verme é em torno de 240. A área mínima especificada, neste caso, é de 200 para explicar alguma variação. Devido à possibilidade de variação das condições de iluminação, é necessário ser capaz de determinar um valor de *threshold* ótimo automaticamente. Para conseguir isso, procura-se em todo o espaço de *thresholding* possível (1-255) utilizando um algoritmo de busca binária para alcançar um maior desempenho. O algoritmo funciona analisando a mediana do espaço observado. A

quantificação, na aplicação deste algoritmo é o número de componentes e a área de cada. Ao repetir este processo, o algoritmo garante uma conversão para uma solução, um *threshold* que produz um segmento de um tamanho mínimo (caso exista), num máximo de oito iterações.

A fim de determinar a área de um segmento, é necessária uma maneira de etiquetar *pixels* para determinar se estes fazem parte do mesmo componente. Isto é feito usando uma estrutura de dados *union-find* que inerentemente conta componentes como objetos. Uma vez que só é preciso o número de componentes e não a conectividade de pontos arbitrários, usa-se uma variação do algoritmo que otimiza a operação de união. Para encontrar componentes conectados (segmentos) numa imagem binária de *threshold*, é dado um identificador único a todos os *pixels*. Inicialmente, cada *pixel* é considerado como sendo a sua própria matriz. Os *pixels* são então conectados a todos os seus vizinhos se compartilharem o mesmo valor binário. O número de componentes pode ser determinado subtraindo 1 do número total de *pixels* de cada vez que uma união é feita (dois componentes tornam-se um componente). Depois de fazer todas as conexões, o valor remanescente é o número de componentes conectados.

A imagem binária resultante identifica *pixels* do verme como 1 (branco) e *pixels* não pertencentes ao verme como 0 (preto). No caso de existirem "buracos" na segmentação do corpo do verme, apesar das etapas de pré-processamento, uma operação morfológica de fechamento é aplicada à imagem de segmentação.

Em [10] o método de extração da árvore dendrítica (DTE) usa pequenas quantidades de dados de treino rotulados em projeções de intensidade máxima (MIP) para aprender os modelos de ruído dos recursos baseados em textura a partir das respostas das estruturas da árvore e do fundo da imagem. A estratégia reside na avaliação de modelos estatísticos de ruído que representam tanto a variabilidade gerada a partir do processamento de imagem e da agregação de informações nas imagens MIP. Estes modelos ruidosos são então utilizados dentro de uma estrutura probabilística para fornecer uma segmentação 2D da árvore dendrítica. Finalmente é feito um pós-processamento para refinar as segmentações e fornecer árvores "esqueletizadas", utilizando um processo de desbaste morfológico.

Para começar, considera-se que cada *pixel* q de uma imagem tem um rótulo binário, $L_q \in \{0,1\}$, indicando que o *pixel* pertence à árvore dendrítica ($L_q = 1$) ou em vez disso, ao fundo

($L_q = 0$). Dada a incerteza em torno do valor de uma etiqueta de *pixel* de uma imagem em que uma estrutura de árvore deve ser extraída, considera-se que L_q é uma variável aleatória binária com distribuição de probabilidade $P(L_q)$.

Na altura dos testes, é de interesse extrair a árvore dendrítica de uma nova imagem dada. Em primeiro lugar, calcula-se as características MR8 da imagem em cada localização de *pixel* da imagem. Em seguida, avalia-se a equação (1) para cada *pixel* na MIP usando as características da imagem e os modelos aprendidos com os dados de treino. Em particular, assume-se que $P(L_q = 1) = P(L_q = 0) = 1/2$:

$$P(L_p = 1|X_q = x) = \frac{P(X_q = x|L_p = 1)}{P(X_q = x|L_p = 0) + P(X_q = x|L_p = 1)}. \quad (1)$$

Enquanto a equação (1) prevê a probabilidade de um *pixel* fazer parte de uma árvore em cada local de imagem, é de interesse uma segmentação binária da árvore. Para este fim, é necessário selecionar um *threshold* de sensibilidade adequada para classificar cada *pixel* como fundo ou árvore usando o resultado da equação (1). O valor mais alto de otimização é alcançado quando a distância a partir do canto superior esquerdo do gráfico de ROC é mínima.

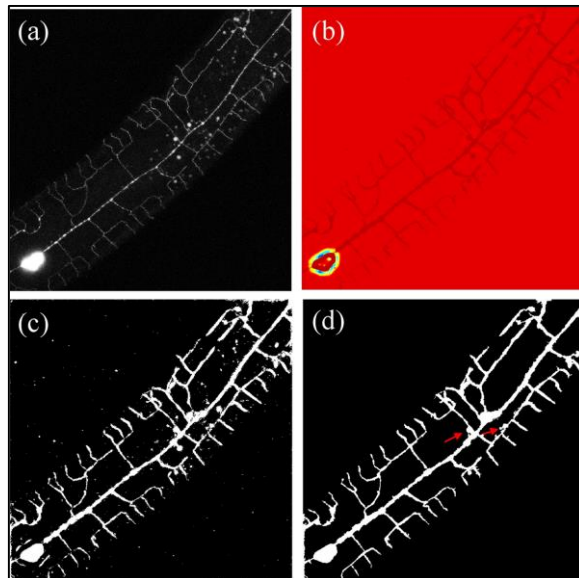


Figura 5 Método de Extração da Árvore Dendrítica [10].

Na Figura 5 em (a), mostra-se o resultado após calcular a probabilidade à posterior, Eq. (3). Em (b), valores baixos são vermelhos enquanto os valores mais elevados aparecem mais escuros. (c) representa o resultado após o *threshold* de sensibilidade otimizado ser aplicado a (b) para se obter uma segmentação grosseira. (d) mostra os resultados após o refinamento e pós-processamento de etapas automáticas. As setas vermelhas indicam exemplos de artefactos restantes, como um resultado da presença de grânulos, em torno do corpo celular e árvore [10].

Devido a diferenças nas imagens de treino e de teste, a etapa de extração tipicamente produz respostas espúrias (Ver Figura 13c). Por esta razão, procede-se a um pós-processamento com dois passos. No primeiro, suaviza-se automaticamente as respostas com a aplicação de operações morfológicas de abertura e fecho para remover *pixels* isolados e preencher regiões em falta.

Por fim, realiza-se uma etapa de pós-processamento semiautomática onde é dada a oportunidade ao usuário para tentar melhorar partes da árvore dendrítica segmentada que não são resolvidos a contento.

3. DETEÇÃO AUTOMÁTICA DOS NEURÓNIOS DO C.ELEGANS

As nematodas de *C. elegans* seguem um método de crescimento rigoroso, durante 11 dias, sendo que as imagens a serem tratadas neste trabalho são retiradas em três dias diferentes da sua vida adulta. O esquema do crescimento pode ser visto na tabela 1.

Tabela 1 Regime de Crescimento e extração de imagens de *C. elegans*.

Dias	Descrição
0	Sincronização
1	<i>Seeding</i> das Nematodas
2	Crescimento a 20°C
3	Crescimento a 20°C
4	<i>Adult Day</i> 1
5	Crescimento a 20°C
6	Crescimento a 20°C
7	<i>Adult Day</i> 4 – extração de imagem
8	<i>Adult Day</i> 5 – extração de imagem
9	Crescimento a 20°C
10	Crescimento a 20°C
11	<i>Adult Day</i> 8 – extração de imagem

Para a extração das imagens, as nematodas são colocados numa lâmina de microscópio e paralisados utilizando Sódio Azide. Depois é retirada uma imagem com ampliação de 630x (63x * 10x) utilizando um microscópio *Axio Observer Z1 Feiss*. As imagens retiradas são de 16bit (ver Figura 6).

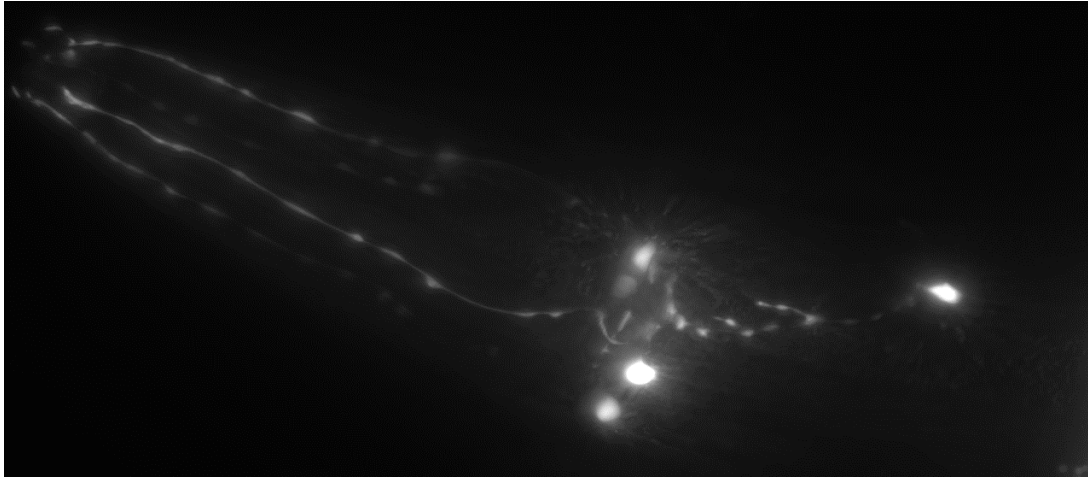


Figura 6 Exemplo de uma Imagem de *C. elegans* retirada usando o microscópio Axio Observer Z1 Zeiss .

A análise feita às imagens nas presentes condições baseia-se muito no operador, o que sabemos estar sujeito à introdução de diversos erros que dificilmente são rastreáveis. Por isso, é importante automatizar este processo. O objetivo do trabalho é a detecção dos Corpos Celulares e dos Axónios para extração de características (ver Figura 7).

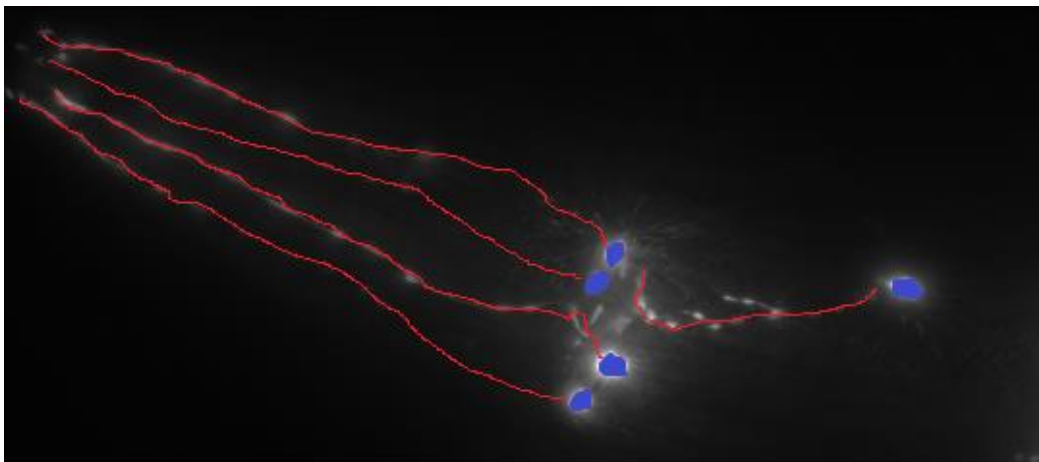


Figura 7 Imagem com os Corpos Celulares (a Azul) e os Axónios (a Vermelho) identificados.
O *pipeline* desenvolvido irá seguir o fluxograma da Figura 8.

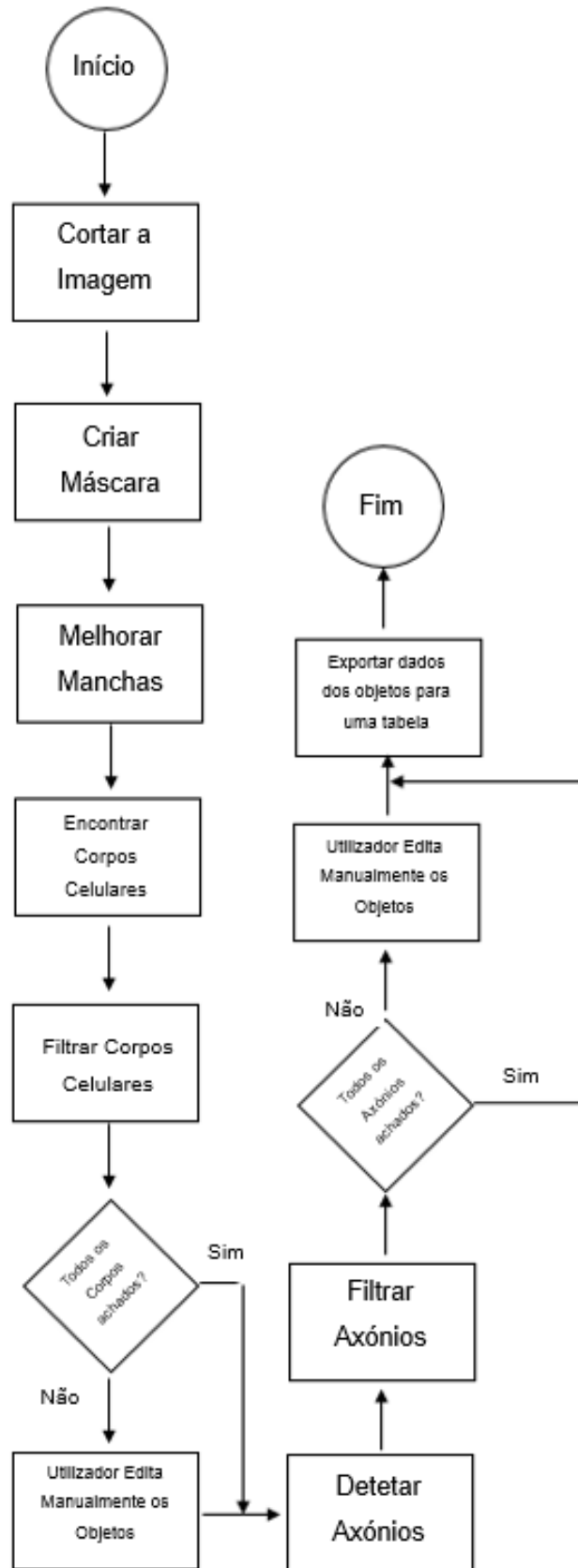


Figura 8 Fluxograma do Pipeline elaborado.

3.1. CELLPROFILER E FUNCIONAMENTO DOS SEUS MÓDULOS

O CellProfiler é um *software open-source* bastante versátil para quantificar dados de imagens biológicas. O CellProfiler é concebido para modular e analisar imagens de alta resolução, medindo o tamanho, forma, intensidade e textura de cada célula (ou outro objeto) em cada imagem. Usando uma interface *point-and-click*, os usuários podem construir um fluxo de módulos, ou seja, uma série sequencial de módulos em que cada um realiza uma função de processamento de imagem, como a correção de iluminação, identificação de objeto (segmentação) ou medição de objetos. Os usuários podem misturar e combinar módulos e ajustar as suas configurações para medir o fenótipo de interesse. Embora originalmente concebido para imagens de alta resolução, é igualmente apropriado para ensaios de baixa resolução.

Para uma descrição detalhada do funcionamento dos módulos que foram utilizados na *pipeline* final deste trabalho deve-se consultar o anexo A.

3.2. MÓDULOS DA PIPELINE

Antes de explicar os módulos principais do funcionamento da *pipeline* proposta, é necessário explicar alguns passos que o utilizador deverá ter em atenção antes de iniciar o processo.

Em primeiro lugar o utilizador deverá arrastar para o módulo *Images* (ver Figura 9) todas as imagens que deseja tratar, no caso de ser um elevado número de imagens, é preferível arrastar a pasta que as contem.



Figura 9 Configuração do Módulo *Images*.

No caso de o utilizador já possuir cortes (*crops*) das secções das imagens que deseja tratar, então também os deverá adicionar ao módulo *Images* e mais tarde deverá definir no módulo *NamesAndTypes* os subgrupos em que se dividem as imagens originais e os *crops* dessas imagens, separando-os de preferência pelo seu diretório, como é mostrado na Figura 10.

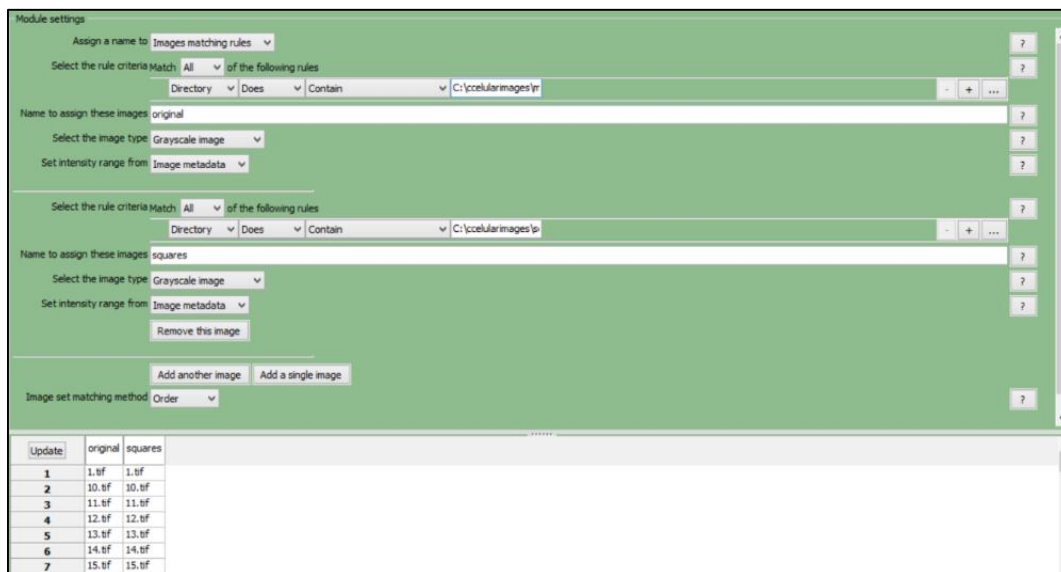


Figura 10 Configuração do Módulo *NamesAndTypes*.

Uma nota que é importante mencionar é que o CellProfiler ordena as imagens por ordem alfabética, por isso, quando as imagens apenas possuem um número em vez de um nome ou são separadas por um número de série, o CellProfiler irá ordenar da seguinte ordem: 1, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 2, 20, 21, etc. É importante ter em atenção esta ordenação pois irá facilitar depois na análise de dados que o CellProfiler irá mais tarde exportar para uma tabela (*datasheet*). Esta ordenação pode ser vista no módulo *NamesAndTypes*.

3.2.1. PRÉ – PROCESSAMENTO

Por forma a facilitar todo o processo inicial de deteção dos neurónios da nematoda *C.elegans* é necessário realizar um pré-processamento às imagens que foram fornecidas para o estudo em questão.

O primeiro problema visível em várias imagens é a presença de mais de que um verme no campo de visão. Para eliminar este problema a solução encontrada foi a realização de um *crop* (corte) manual da secção da imagem que o utilizador deseja analisar, ou seja, o utilizador deverá seleccionar qual o verme mais proeminente na imagem. Para tal, é utilizado o módulo *Crop* com a configuração apresentada na Figura 11.

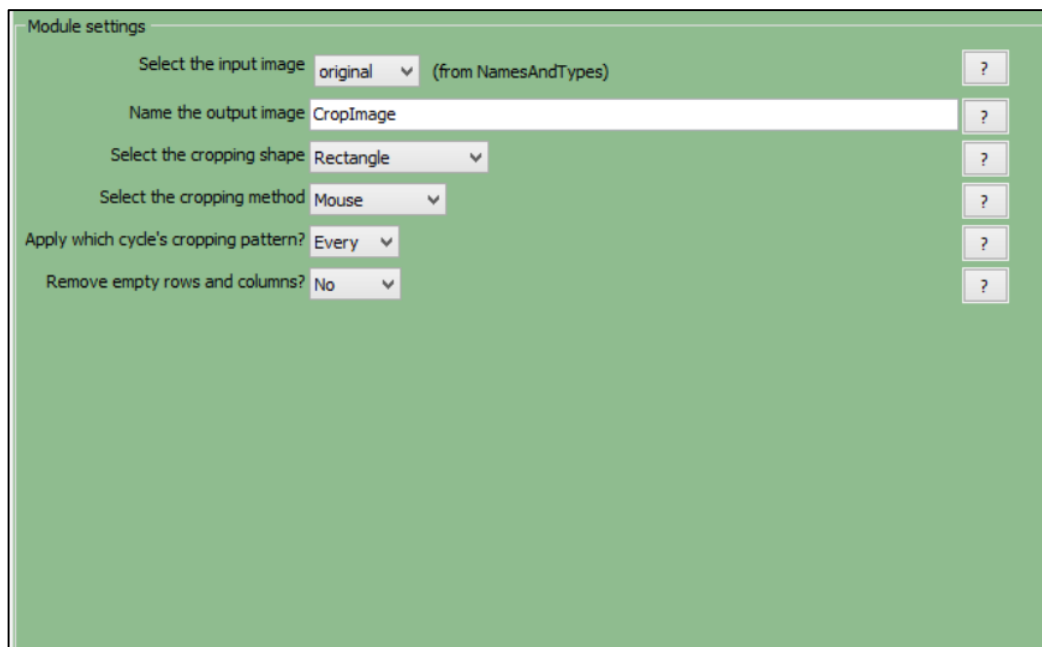


Figura 11 Configuração do Módulo *Crop*.

Durante a execução do módulo é apresentada ao utilizador a janela similar à apresentada na Figura 12, sendo que este deverá então arrastar os dois quadrados mais pequenos, para dimensionar e posicionar o retângulo / quadrado maior no verme a analisar.

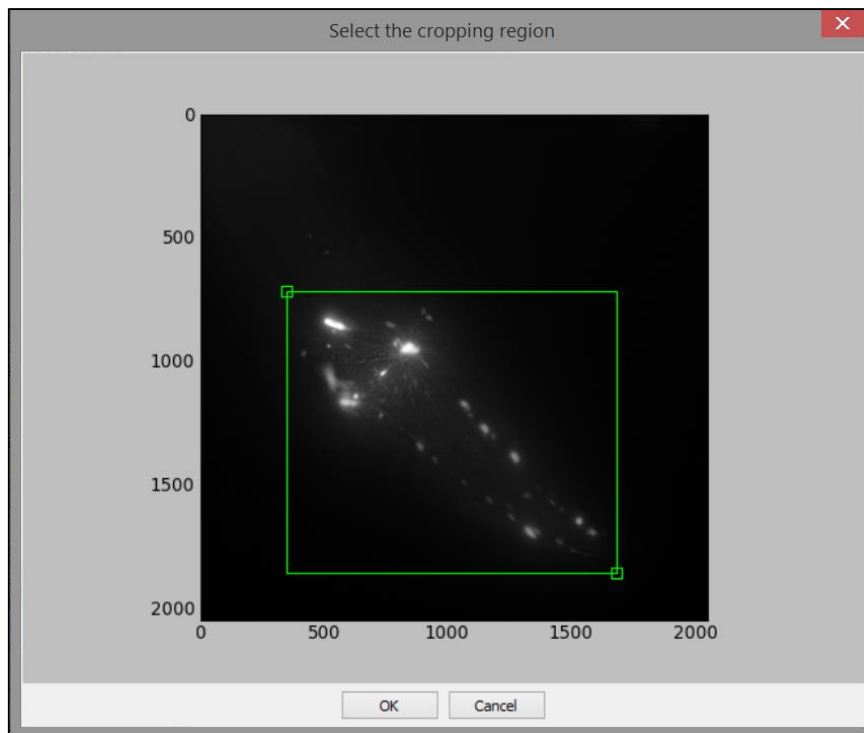


Figura 12 Janela de seleção da área para realizar o *crop* da Imagem original

O módulo *Crop* e respetivo módulo de gravação das imagens deverão ser desativados em casos em que os cortes das imagens já existam. Para desativar módulos basta clicar no visto a verde junto ao nome do módulo e este ficará esbranquiçado.

Apesar de a seleção da zona para analisar ser um passo bastante importante, é importante notar que a imagem que sai do módulo *Crop* é sujeita a um processo de normalização, o que altera os valores de medição da imagem, prejudicando os processos a elaborar mais tarde. Para anular esta transformação e obter uma máscara da imagem, utiliza-se o módulo *ApplyThreshold* que como o nome indica, aplica um *threshold* à imagem, passando esta a ser uma imagem binária, facilitando a posterior identificação da região de interesse (ver Figura 13). É importante referir que o valor de *threshold* é extremamente baixo, logo não alterará significativamente os valores finais que se pretendem medir, pois a máscara obtida trata-se de imagem binária (ver Figura 14).

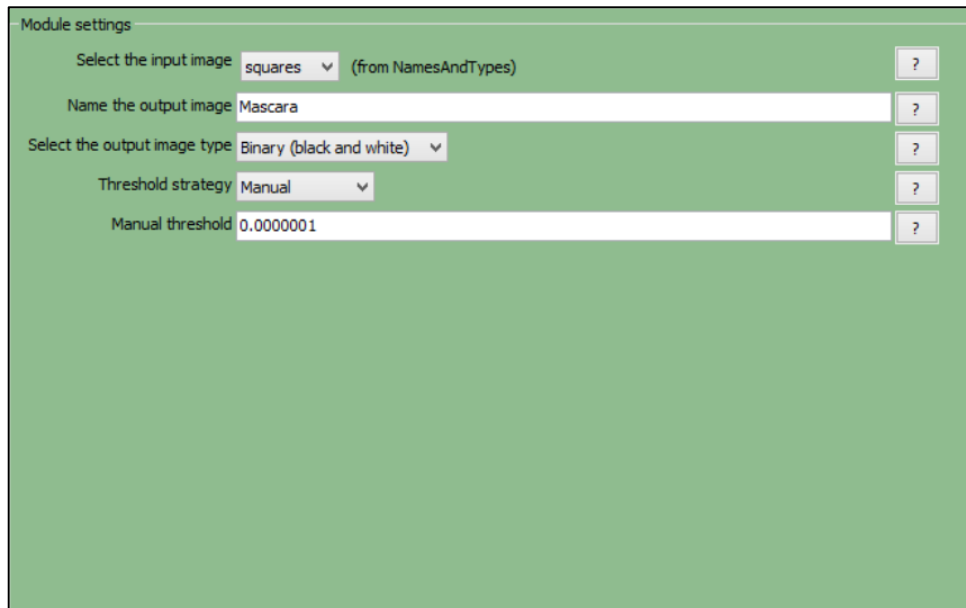


Figura 13 Configuração do Módulo *ApplyThreshold*.

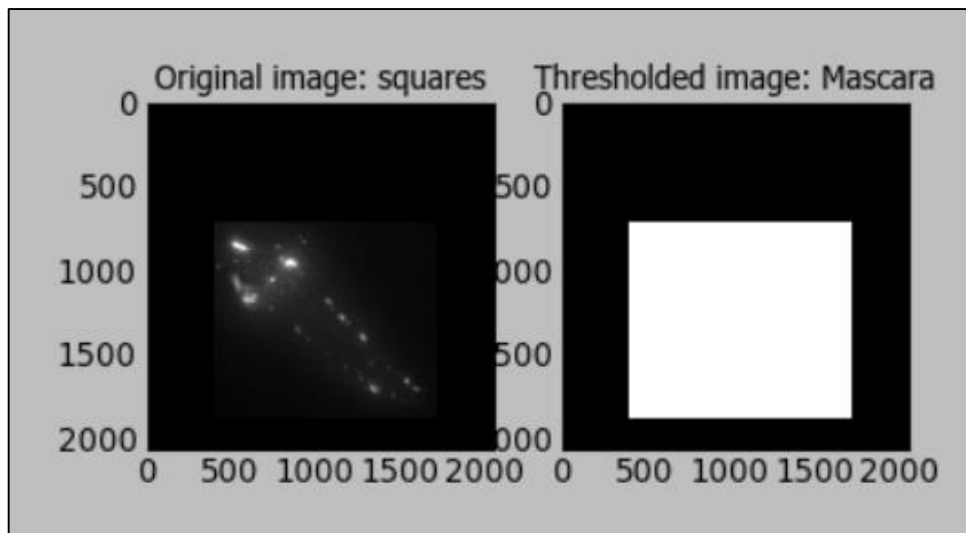


Figura 14 Máscara Extraída do Módulo *ApplyThreshold* (à direita) em comparação com o corte feito no módulo *Crop* (à esquerda).

Agora que se possui uma máscara da imagem com a qual se pretende trabalhar, é necessário realizar a multiplicação dessa máscara pela imagem original, mantendo-se assim apenas a secção desejada e anulando as regiões que não interessam tratar. Para tal utiliza-se o módulo *ImageMath* (ver Figura 15).

Figura 15 Configuração do Módulo *ImageMath*.

A nova imagem obtida estará assim concentrada na secção onde se encontram os corpos celulares dos neurónios que se pretendem analisar e cuja deteção é o primeiro passo para a extração do neurónio completo. Mas primeiro é preciso incrementar a qualidade de certas características da imagem para que o módulo de deteção que irá funcionar mais à frente na *pipeline* possa ter o seu trabalho facilitado. Para tal é utilizado o módulo *EnhanceandSurpressFeatures* (ver Figura 16) que neste caso irá fazer um melhoramento dos *speckls* (manchas) presentes na imagem, pois os corpos celulares são as áreas onde se verifica uma maior intensidade na imagem.

Figura 16 Configuração do Módulo *EnhanceandSurpressFeatures*.

3.2.2. DETEÇÃO DOS CORPOS CELULARES

Agora que a imagem se encontra otimizada (ver Figura 17) procede-se à detecção dos corpos celulares utilizando-se o módulo *IdentifyPrimaryObjects* (ver Figura 18). Este módulo é normalmente utilizado na detecção de núcleos em células, mas neste caso também se pode aplicar à detecção dos corpos celulares, visto estes terem características semelhantes.

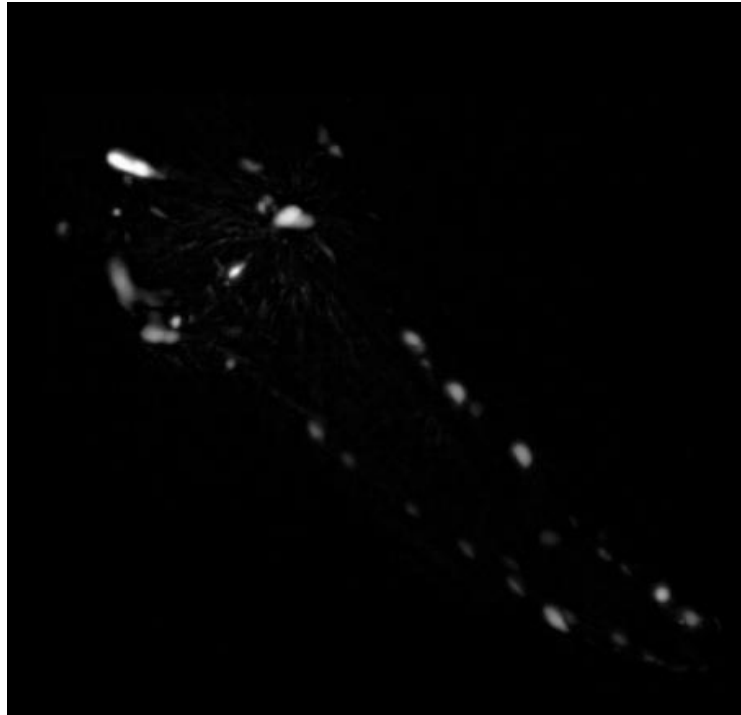
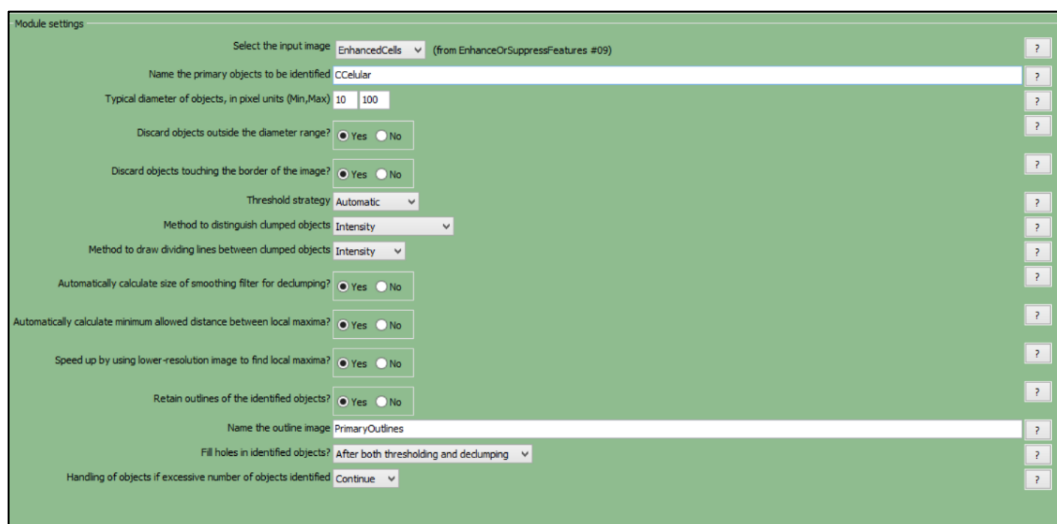


Figura 17 Imagem após Pré-processamento.



Module settings

Select the input image: EnhancedCells (from EnhanceOrSuppressFeatures #09) ?

Name the primary objects to be identified: CCellular ?

Typical diameter of objects, in pixel units (Min,Max): 10 100 ?

Discard objects outside the diameter range? Yes No ?

Discard objects touching the border of the image? Yes No ?

Threshold strategy: Automatic ?

Method to distinguish clumped objects: Intensity ?

Method to draw dividing lines between clumped objects: Intensity ?

Automatically calculate size of smoothing filter for declumping? Yes No ?

Automatically calculate minimum allowed distance between local maxima? Yes No ?

Speed up by using lower-resolution image to find local maxima? Yes No ?

Retain outlines of the identified objects? Yes No ?

Name the outline image: PrimaryOutlines ?

Fill holes in identified objects: After both thresholding and declumping ?

Handling of objects if excessive number of objects identified: Continue ?

Figura 18 Configuração do Módulo *IdentifyPrimaryObjects*.

Apesar de quase todos os processos deste módulo serem automáticos, existe um parâmetro que é necessário realçar, que é a definição do tamanho dos objetos a encontrar. Como se pode ver pela Figura 18 o intervalo de tamanho escolhido é entre 10 e 100 *pixels* porque este tamanho permite uma deteção dos corpos celulares em quase todas as imagens de teste que foram fornecidas pelo perito (em 86% das imagens, para ser exato). Apesar disso, por ser um intervalo um pouco abrangente, também categoriza vários falsos positivos como corpos celulares. Visto este intervalo ser o mais eficiente no que toca à deteção dos corpos, para eliminar os falsos positivos será necessário filtrar os objetos.

3.2.3. FILTRAGEM

A filtragem dos objetos é feita de duas formas diferentes: automática e manual.

A filtragem manual é feita através do módulo *FilterObjects*. Este módulo utiliza medições feitas anteriormente à imagem através do módulo *MeasureObjectSizeShape*. A filtragem é feita de forma sequencial, ou seja, é aplicado um primeiro filtro e depois é aplicado um outro filtro aos objetos que cumpriram os requisitos do filtro anterior. Após alguns testes, conclui-se que as melhores características de todas as que são medidas pelo CellProfiler são a Área e a Excentricidade do objeto. Assim, após a obtenção dos valores mínimos e máximos de todos os corpos celulares dentro do conjunto constituído por 10 imagens aleatórias, são utilizados os módulos *FilterObjects*, um com a filtragem por área (ver Figura 19), seguido de outro com filtragem por excentricidade (ver Figura 20).

Module settings

Name the output objects: FilteredCCelularArea ?

Select the object to filter: CCelular (from IdentifyPrimaryObjects #10) ?

Select the filtering mode: Measurements ?

Select the filtering method: Limits ?

Select the measurement to filter by: Category: AreaShape ?
Measurement: Area ?

Filter using a minimum measurement value? Yes No ?
Minimum value: 328

Filter using a maximum measurement value? Yes No ?
Maximum value: 2371

Add another measurement: Add

Retain outlines of the identified objects? Yes No ?

Name the outline image: FilteredOutlinesArea ?

Relabel additional objects to match the filtered object? Add an additional object ?

Figura 19 Configuração do Módulo *FilterObjects* utilizando a área.

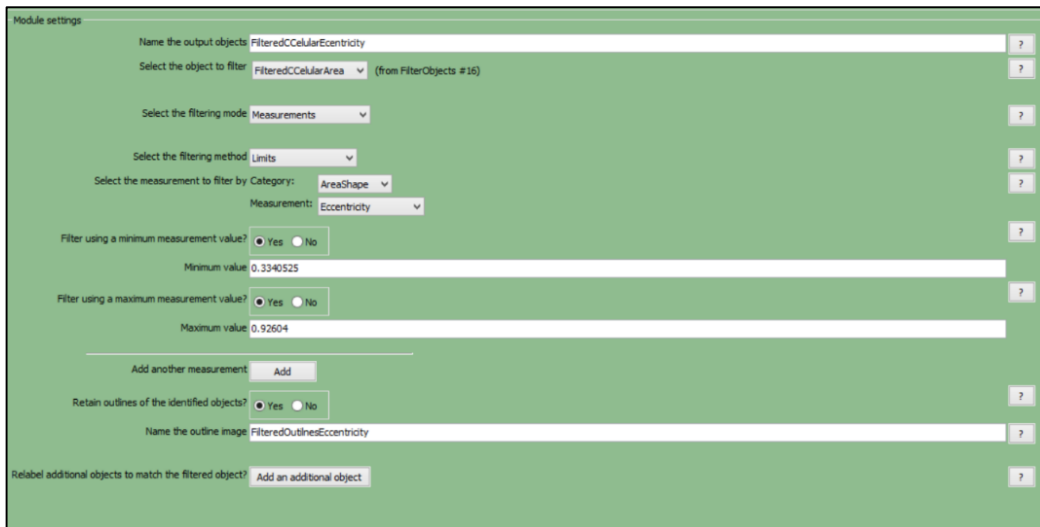


Figura 20 Configuração do Módulo *FilterObjects* utilizando a excentricidade.

Apesar de estas duas filtragens eliminarem bastantes falsos positivos, não é suficiente para os remover todos e para se obter uma deteção perfeita onde todos os corpos celulares foram detetados, por isso é dada a oportunidade ao utilizador para editar os objetos restantes manualmente, através do módulo *EditObjectsManually* (ver Figura 21).

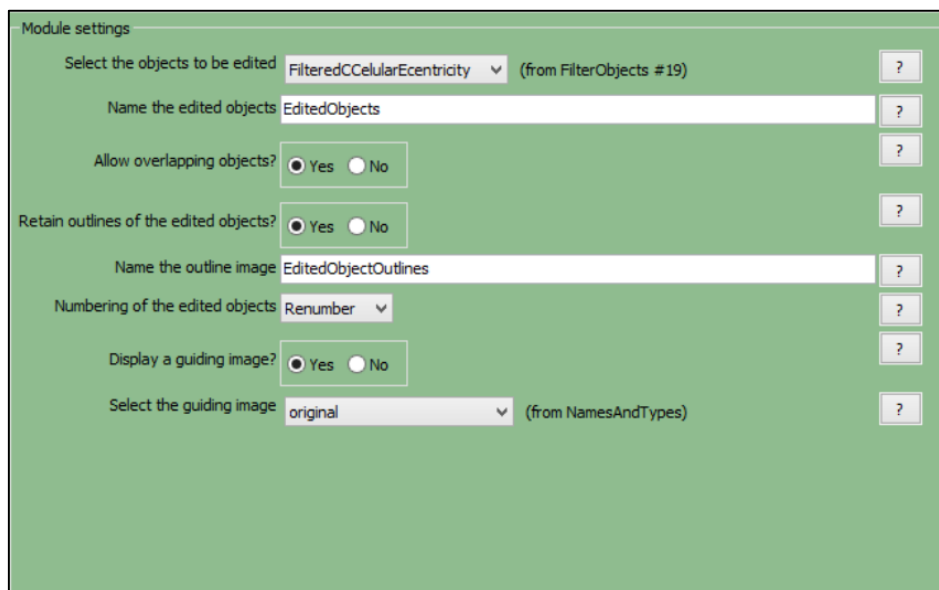


Figura 21 Configuração do Módulo *EditObjectsManually*.

Durante a análise de imagens, este módulo apresenta uma janela ao utilizador semelhante à da Figura 22. Aqui o utilizador deverá eliminar os objetos que considera incorretos, para isso deve clicar com o botão esquerdo do rato dentro dos limites (*outlines*) dos objetos apresentados para que estes fiquem com as linhas que os envolvem a tracejado, indicando

que serão eliminados no final do processo. Se o utilizador desejar adicionar um objeto que não foi identificado pelo *software* então deverá colocar o cursor do rato em cima do objeto que se deseja identificar e pressionar a tecla “N” do teclado, sendo que será apresentada ao utilizador uma circunferência composta por vários círculos vermelhos, os quais o utilizador deverá arrastar para poder alterar a forma dos limites do novo objeto. Para terminar esta tarefa basta clicar com o botão do lado direito do rato dentro dos limites deste novo objeto.

Caso seja necessário editar os limites de um ou vários objetos já existentes o utilizador deve pressionar o botão do lado direito do rato para que apareçam os pontos vermelhos para a edição dos limites. Para unir vários objetos num só, basta selecionar todos os objetos desejados com o botão do lado direito do rato e depois pressionar o botão “J” do teclado, formando assim um novo objeto.

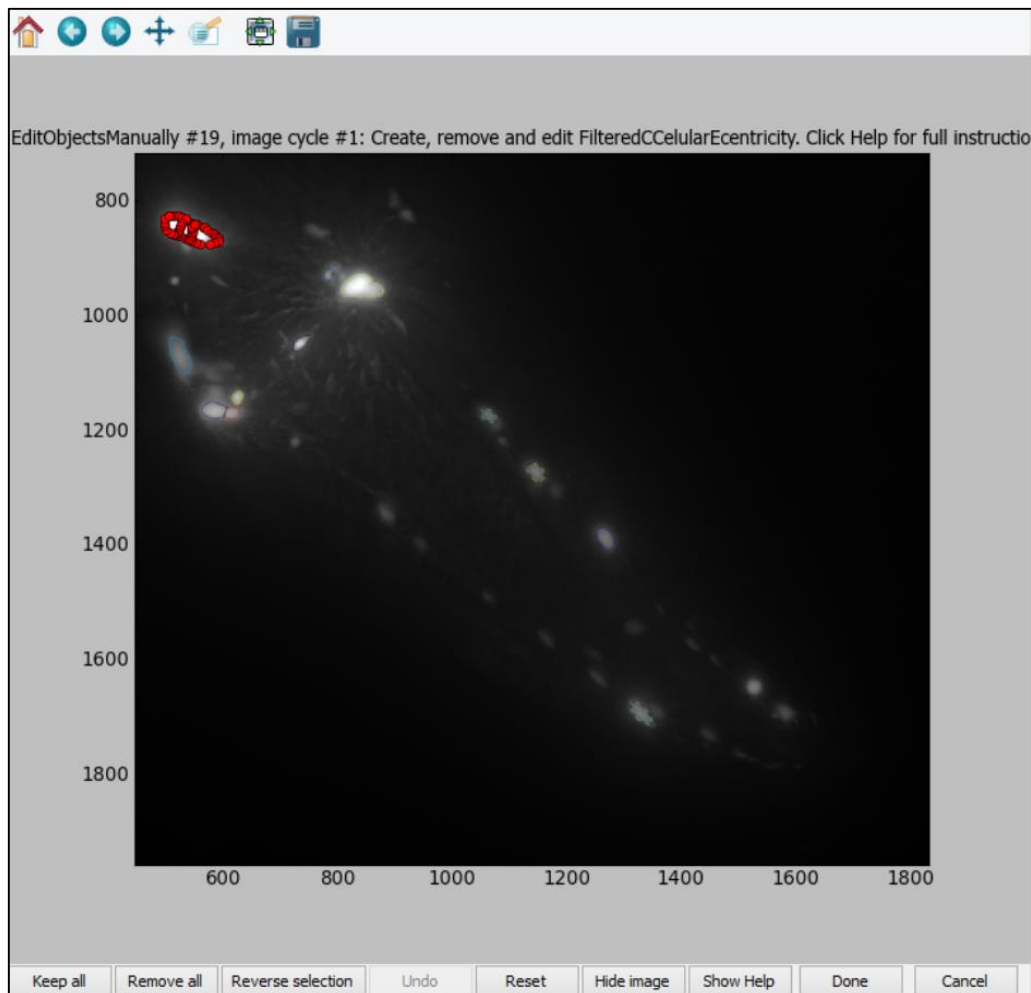


Figura 22 Janela de edição manual de objetos com exemplos de objetos eliminados (limites a tracejado) e objetos em fase de edição (pontos vermelhos como limite).

Este módulo possui várias outras ferramentas de edição que não são mencionadas, mas se o utilizar desejar conhecê-las e utilizar, basta clicar no botão “Help” da janela da Figura 26, sendo-lhe apresentada uma lista com as várias ferramentas do módulo e respetivas descrições. Quando o utilizador desejar terminar a edição basta clicar no botão “Done” para passar a uma próxima imagem.

3.3. ANÁLISE DOS DADOS

Agora que todos os objetos corretos foram obtidos, é possível ao utilizador analisar as suas características, desde a área até à intensidade. Para isso o CellProfiler cria uma tabela com todas essas informações e grava esses dados num ficheiro de texto que o utilizador poderá depois exportar para uma ferramenta de análise de dados como o *Excel*. Caso o utilizador deseje mudar o diretório do ficheiro de texto ou seleccionar outros dados para serem apresentados na tabela, este terá que alterar as configurações do módulo *ExportToSpreadsheet* (ver Figura 23). Para alterar as medições a apresentar deve-se clicar no botão “Press to select measurements” e será apresentada uma janela como a da Figura 24.

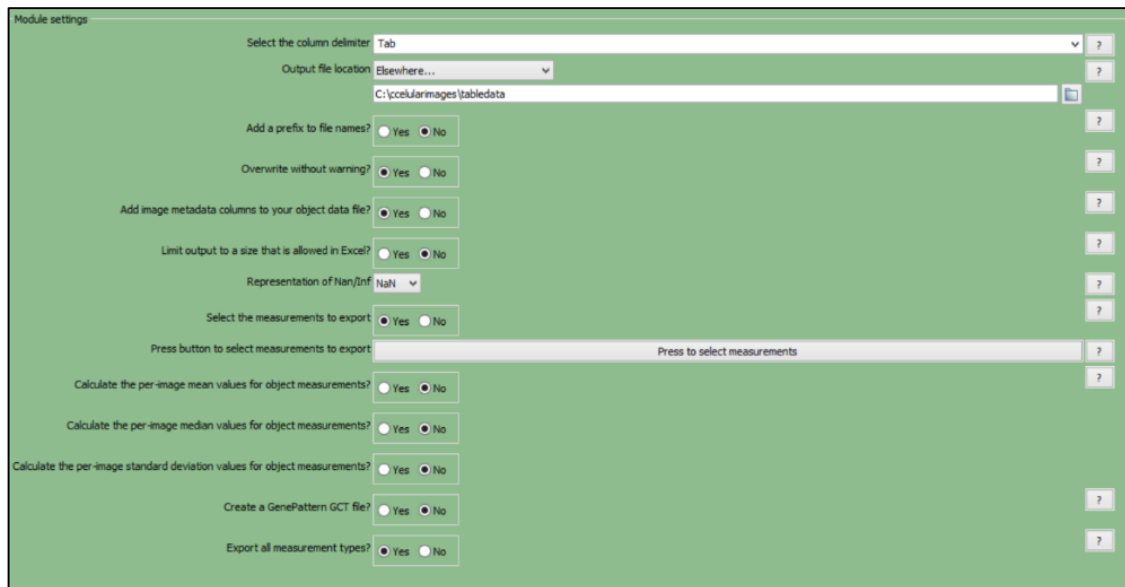


Figura 23 Configuração da janela *ExportToSpreadsheet*.

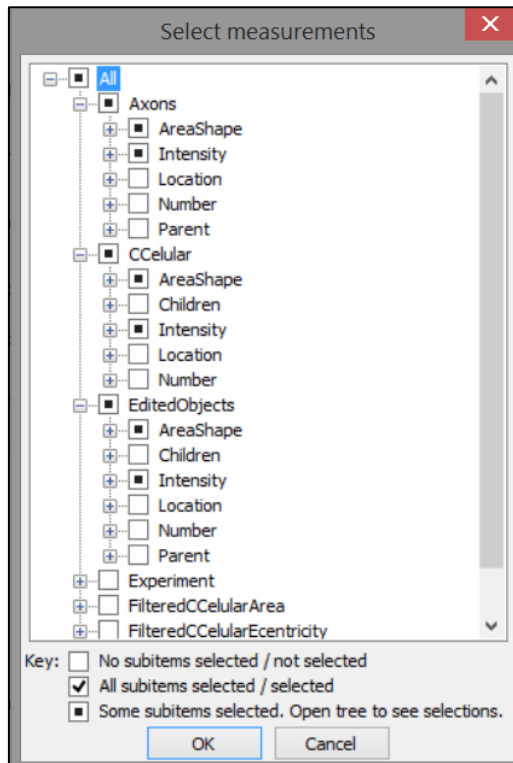


Figura 24 Janela de seleção de medições.

Este é o último módulo utilizado pelo CellProfiler, mas para continuar com a análise dos resultados e facilitar a avaliação da sua qualidade, foi criado um algoritmo em *Matlab* que faz a contagem automática de todos os objetos encontrados e também calcula o coeficiente de Dice de cada imagem. Inicialmente o algoritmo (ver Figura 26) lê as imagens originais, sem qualquer tipo de edição e converte-as para um tipo *uint8* pois para poder realizar as próximas tarefas as imagens devem ter uma matriz com essa natureza. Após obter os dados das imagens originais, o código passa à análise das imagens com os *outlines* (ver Figura 25). Visto essa ser uma imagem onde apenas aparecem os *outlines* dos objetos sobre um fundo preto (ver Figura 27), basta ao algoritmo procurar as formas que possuam uma cor branca e proceder ao enchimento das formas que eles formam, procedendo à contagem de quantos objetos são formados por esse enchimento. Depois o código origina uma nova imagem que é o resultado da união dos *outlines* com a imagem original (ver Figura 28).

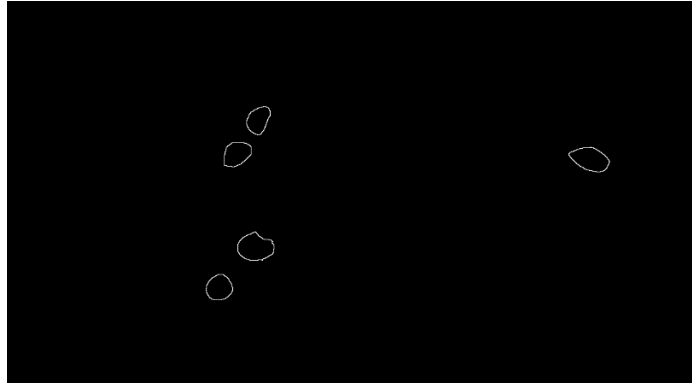


Figura 25 *Outlines dos Corpos Celulares.*

```

% le a imagem original!
originalFileName = strcat('C:\ccelularimagens\maxproj\', num2str(i), '.tif');
% le a imagem original, convertendo para uint8
im_original = im2uint8(imread(originalFileName));
[w h] = size(im_original);
% imagem para resultados de display
im_final = im_original;

% le a imagem com os outlines, convertendo para uint8
outlinesFileName = strcat('C:\ccelularimagens\objectoutlines\', num2str(i),
'.tif');
im_outlines = im2uint8(imread(outlinesFileName));
size(im_outlines);
%whos im_annotation
% figure, imshow(im_outlines)
% resize para os tamanhos da imagem original
im_outlines = imresize(im_outlines,[w h]);

% procura as anotações.. a branco
ind_White = find(im_outlines(:,:,1) >= 255);

% cria a mascara preenchendo os outlines
im_segm_cellprof = logical( zeros(w,h) );
im_segm_cellprof(ind_White) = 1;
im_segm_cellprof = imfill(im_segm_cellprof,'holes');

% contornos dos objectos
[C,L,N,A] = bwboundaries(im_segm_cellprof);
for k=1:length(C),
    boundary = C{k};

    ind = sub2ind(size(im_final),boundary(:,1),boundary(:,2));
    % e grava na imagem original (para efeitos de display)
    im_final(ind) = 255;

end

% cria a imagem final dos outlines sobreposta sobre a imagem original
im_final = cat(3,im_final,im_final,im_original);

finalFileName = strcat('C:\ccelularimagens\outlinevoriginal\', num2str(i),
'.png');
imwrite(im_final,finalFileName,'png');

```

Figura 26 Extrato do algoritmo de *Matlab* para a leitura dos dados das imagens originais e das imagens com os *outlines*.

É importante referir que o CellProfiler já executa este processo sozinho, mas para facilitar a avaliação de todas os *pipelines* que se foram testando, foi deixada esta função para o *Matlab*.

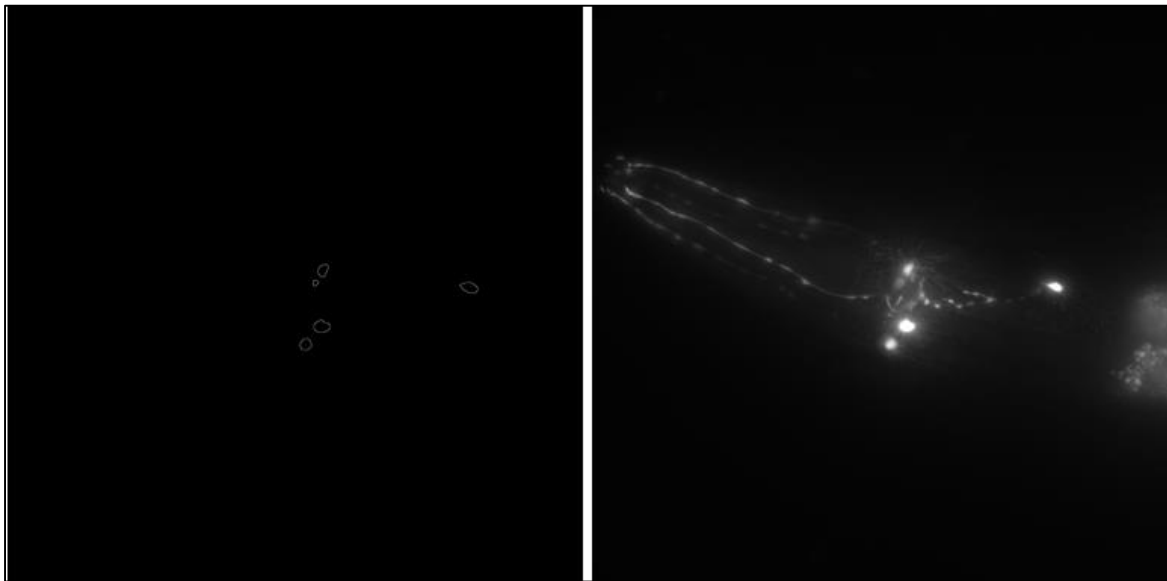


Figura 27 Exemplo de uma imagem com apenas os *outlines* de objetos à esquerda e respetiva imagem original à direita.

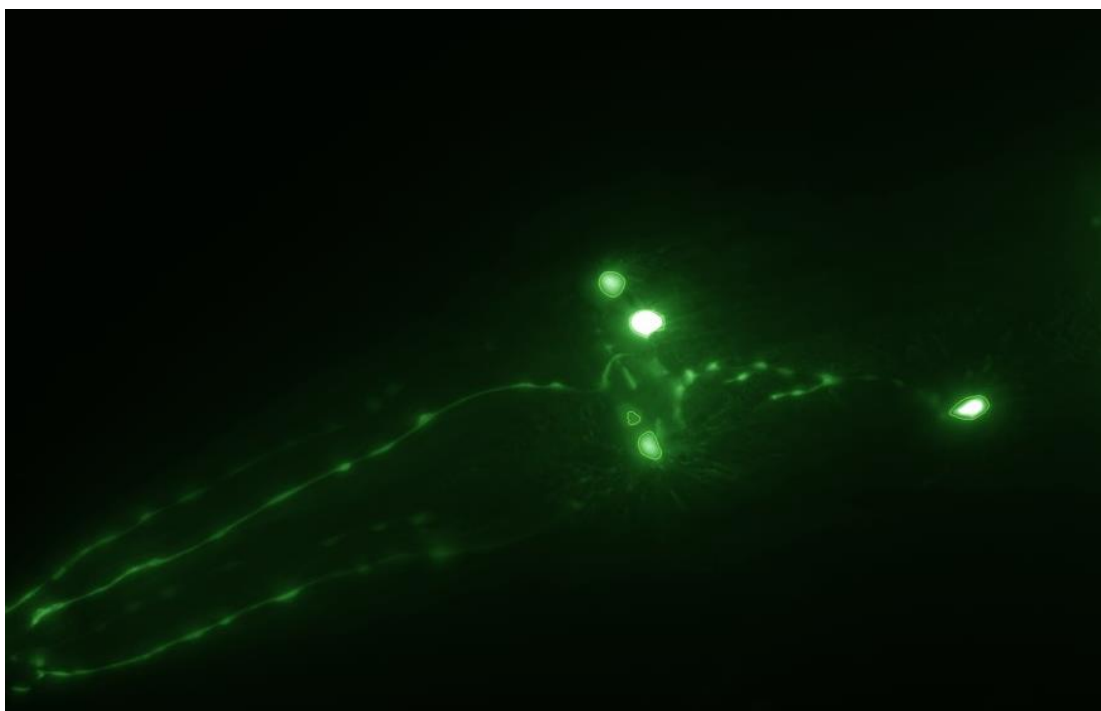


Figura 28 Exemplo de uma imagem de união dos *outlines* com a imagem original (imagem a verde para se notar melhor os *outlines*).

Depois deste passo, o algoritmo executa um procedimento semelhante para a análise das anotações feitas pelo perito (ver Figura 29). Neste caso, o perito anotou as localizações dos corpos celulares circundando-os em cor amarela (ver Figura 30), por isso o código desta vez procura essa cor em vez da branca.

```

% le a imagem com as anotações do Perito.
anotFileName = strcat('C:\ccelularimages\imageanotadas\', num2str(i), '.png');
im_annotation = imread(anotFileName);
size(im_annotation);
im_annotation_bw = im2bw(im_annotation);

% extrai apenas a regioao de interesse
[I,J] = find(im_annotation_bw == 0);
xtopleft = I(1);
ytopleft = J(1);
xbottomright = I(end);
ybottomright = J(end);

%ind = sub2ind([size(im_annotation,1), size(im_annotation,2)],I,J);
R = im_annotation(xtopleft:xbottomright,ytopleft:ybottomright,1);
G = im_annotation(xtopleft:xbottomright,ytopleft:ybottomright,2);
B = im_annotation(xtopleft:xbottomright,ytopleft:ybottomright,3);

im_annotation = cat(3,R,G,B);
%whos im_annotation
im_annotation = imresize(im_annotation,[w h]);

% procura as anotações.. a amarelo na imagens anotadas pelo Perito
ind_Red_Green = find(im_annotation(:,:,1) >= 230 & im_annotation(:,:,2) >=
230 & im_annotation(:,:,3) == 0);

% cria a mascara
im_segm_expert = logical( zeros(w,h) );
im_segm_expert(ind_Red_Green) = 1;
se = strel('disk',3);
im_segm_expert = imdilate(im_segm_expert,se);
im_segm_expert = imfill(im_segm_expert,'holes');

% contornos dos objectos
[O,D,M,X] = bwboundaries(im_segm_expert);

for k=1:length(O),
    boundary = O{k};

    ind = sub2ind(size(im_final),boundary(:,1),boundary(:,2));

    im_annotation(ind) = 255;
end

```

Figura 29 Extrato do algoritmo de *Matlab* aonde é feita a leitura dos dados das imagens anotadas pelo perito.

É importante referir, que neste caso foi necessário fazer um ajuste ao tamanho das imagens pois estas não se encontravam com a mesma configuração que as anteriores.

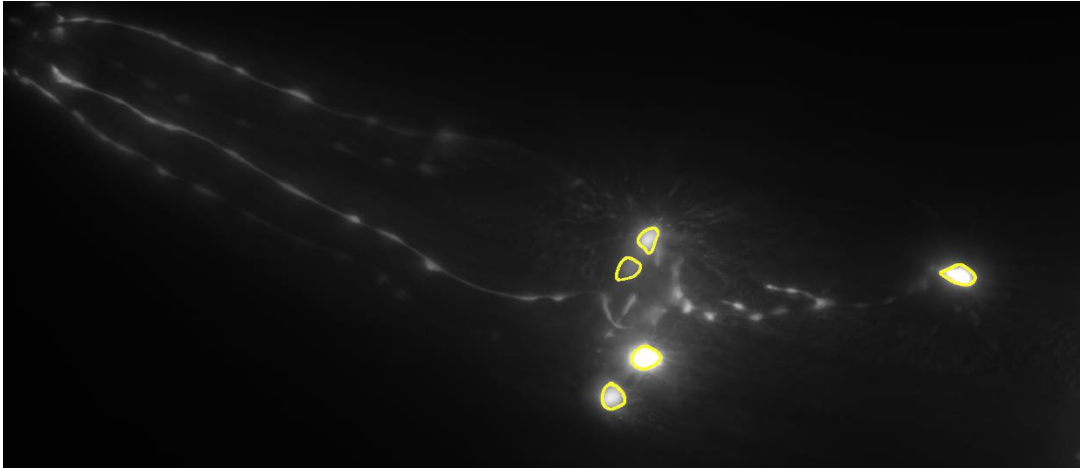


Figura 30 Exemplo de imagem com anotações do perito.

Por último, o algoritmo calcula o Coeficiente de Dice das duas imagens antes e após ocorrer a filtragem, para fins avaliativos e grava todos os dados num ficheiro de texto para ser analisado mais tarde por uma ferramenta de tratamento de dados (ver Figura 31). O coeficiente de Dice é um de uma série de medidas da extensão de sobreposição espacial entre duas imagens binárias [11]. É comumente utilizado em relatórios de desempenho de segmentação e dá mais peso aos casos em que as duas imagens concordam. Os seus valores variam entre 0 (ausência de sobreposições) e 1 (concordância perfeita). Neste trabalho, os valores dos dados são obtidos através da Equação 2, em que I corresponde à imagem gerada pela união dos *outlines* com a imagem original e A corresponde à imagem com as anotações do perito.

$$D = \frac{2(I \cap A)}{(I \cap A + I \cup A)} \quad (2)$$

```

%Calculo do Coeficiente de Dice
common=sum(im_seg_m_expert & im_seg_m_cellprof);
cm=sum(im_seg_m_expert); % the number of voxels in im_seg_m_expert
co=sum(im_seg_m_cellprof); % the number of voxels in im_seg_m_cellprof

DiceCoef=(2*common)/(cm+co);

%Calculo do Coeficiente de Dice para os objetos editados
common2=sum(im_seg_m_expert & im_seg_m_cellprof_filter4);
cm2=sum(im_seg_m_expert); % the number of voxels in im_seg_m_expert
co2=sum(im_seg_m_cellprof_filter4); % the number of voxels in
im_seg_m_cellprof_filter

DiceCoefFilter5=(2*common2)/(cm2+co2);
%-----
-----
%Criar Tabela com todos os dados pretendidos
%i---->Numero da Imagem
anot=length(O);%---->Numero de Corpos Celulares anotados pelo Perito
found=length(C);%---->Numero de Corpos Celulares encontrados pelo
CellProfiler
foundfilter1=length(Q);%---->Numero de Corpos Celulares Filtrados por area
encontrados pelo CellProfiler
foundfilter3=length(U);%---->Numero de Corpos Celulares Filtrados por
excentricidade encontrados pelo CellProfiler
foundfilter4=length(T);%---->Numero de Corpos Celulares Editados

data=[i;anot;found;DiceCoef;foundfilter1;foundfilter3;foundfilter4;DiceCoefF
ilter5];
fileID = fopen('corposcimagem.txt','a+');
fprintf(fileID,'%6.1f %6.1f %6.1f %12.8f %6.1f %6.1f %6.1f %12.8f
\r\n',data);
fclose(fileID);

```

Figura 31 Extrato do algoritmo de *Matlab* para o cálculo do Coeficiente de Dice e a exportação dos dados para um ficheiro de texto.

3.4. DETEÇÃO DOS AXÓNIOS

A deteção dos axónios é feita de forma bastante similar à dos corpos celulares. Em primeiro lugar, volta-se a utilizar o módulo *EnhanceOrSupressFeatures* na imagem original, mas desta vez é feita a melhoria das estruturas em linha denominadas de Neurites, depois é feito um melhoramento dessa nova imagem, melhorando os *Speckles*, como foi feito anteriormente para a deteção dos corpos celulares.

Agora que se possui esta imagem, irá ser feita a deteção dos axónios utilizando o módulo *IdentifySecondaryObjects* (ver Figura 32), que utiliza os objetos primários, ou seja, os corpos celulares como ponto-chave para a deteção dos axónios, através de um método de propagação.



Figura 32 Configuração da janela *IdentifySecondaryObjects*.

Agora que se possui os objetos secundários, estes também serão filtrados, para a eliminação de falsos positivos, utilizando o módulo *FilterObjects* (ver Figura 33), mas desta vez a filtragem é apenas feita utilizando a Área como medida, pois os axónios não são objetos redondos, logo não é recomendado utilizar a excentricidade. Os valores de filtragem foram obtidos da mesma maneira que foi utilizada para a obtenção dos valores de filtragem dos corpos celulares. Após a filtragem automática, passa-se à filtragem manual, cujo processo é idêntico ao utilizado para a edição manual dos corpos celulares.

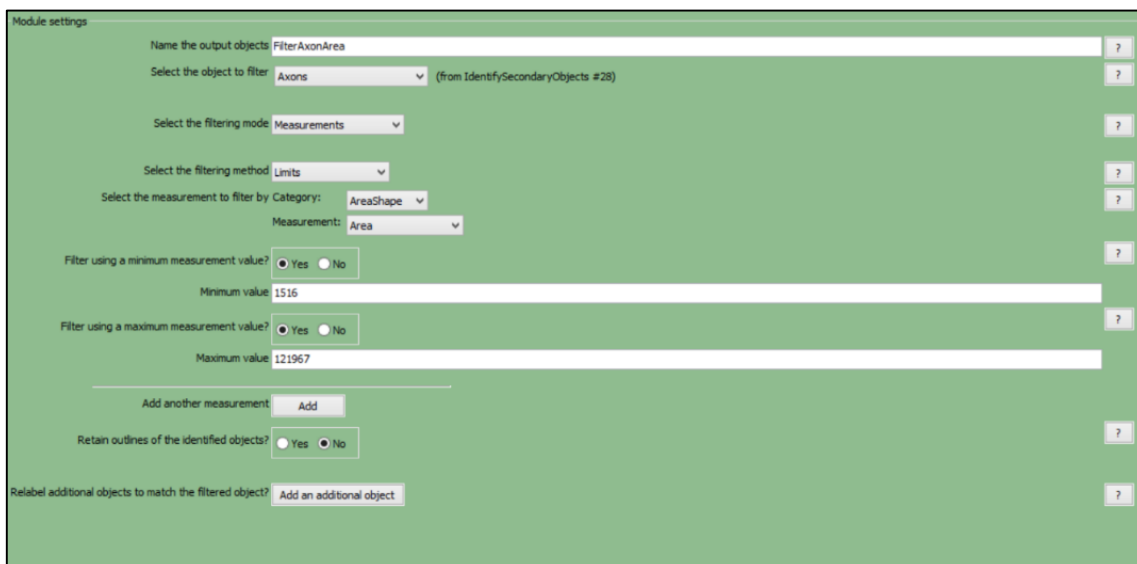


Figura 33 Configuração da janela *FilterObjects* pela área dos axónio

4. RESULTADOS E DISCUSSÃO

Ao longo do desenvolvimento da *pipeline* foram-se testando várias metodologias para a detecção de objetos e ao longo desses testes obtiveram-se várias imagens que justificam alguns passos e alterações que se encontram presentes na *pipeline* final.

Antes de existir o módulo de corte (*crop*), o primeiro passo a ter em conta era o melhoramento da qualidade de imagem, tendo em vista facilitar a detecção dos corpos celulares. Este passo é necessário pois a detecção dos objetos primários iria atribuir essa designação a demasiados componentes da imagem e falharia os corpos celulares por completo (ver Figura 34).

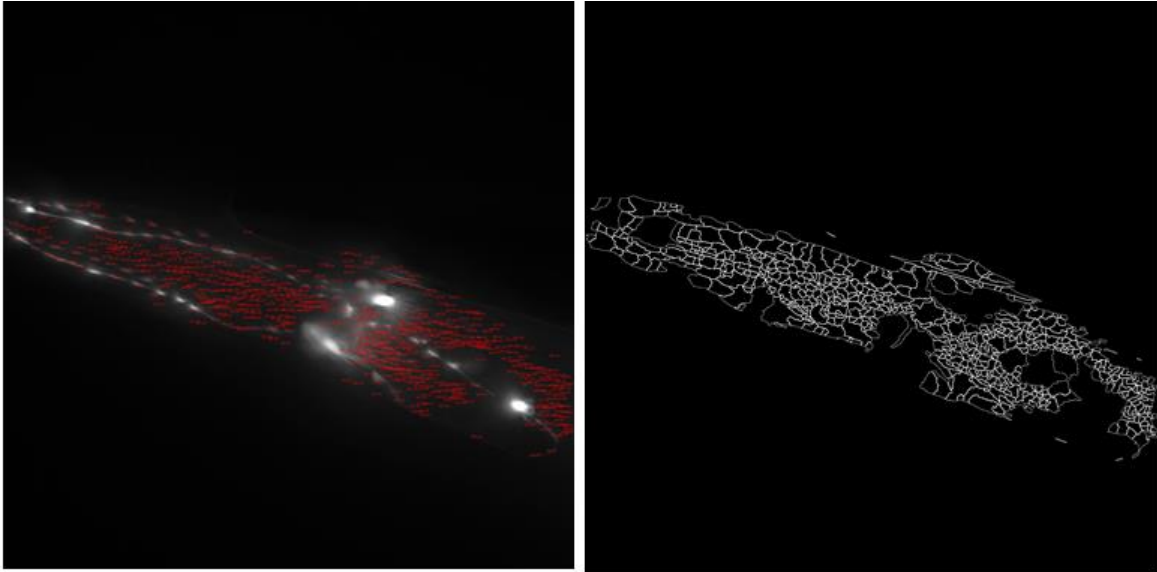


Figura 34 Exemplo de resultado não usando qualquer tipo de incremento de qualidade. À esquerda é possível ver o número de objetos que foram considerados como positivos e à direita as fronteiras desses objetos.

Como os corpos celulares apresentam uma forma circular, ponderou-se a utilização do módulo *EnhanceOrSupressFeatures* para incrementar a qualidade dos objetos circulares juntamente com o módulo complementar *EnhanceEdges*, que se focaria na melhoria das fronteiras dos objetos circulares, mas os resultados não foram satisfatórios como se pode ver na Figura 35.

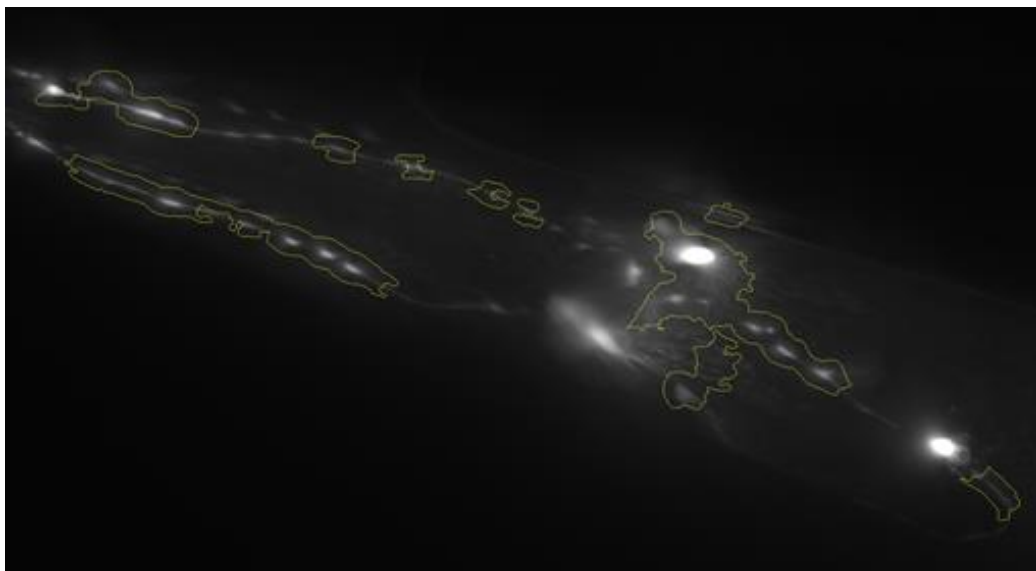


Figura 35 Exemplo de resultado de incremento de qualidade usando o módulo *EnhanceEdges*.

Nesta Figura, pode-se verificar que vários objetos sofreram uma união e alguns corpos celulares não foram selecionados, logo descartou-se o módulo *EnhanceEdges* (ver Figura 36).

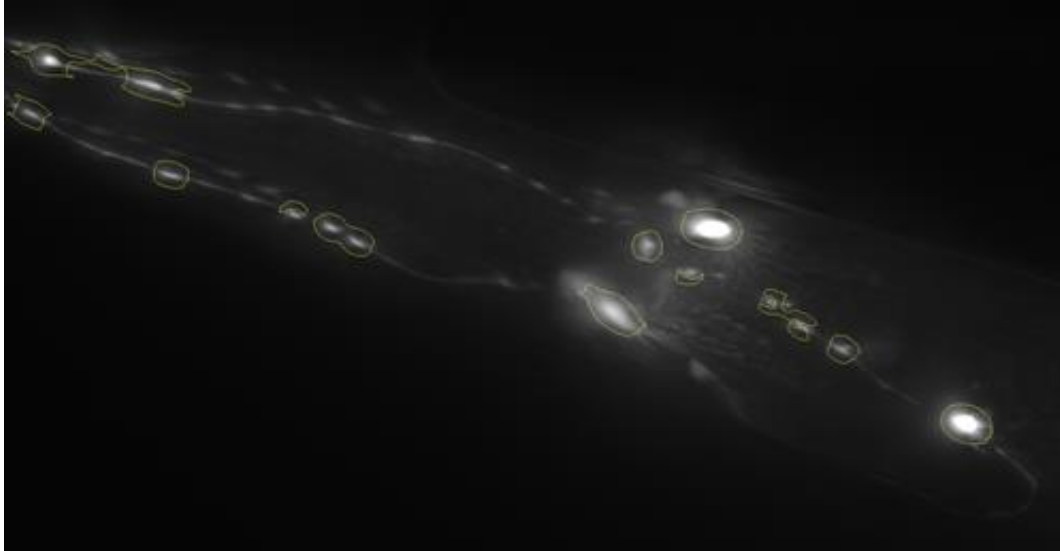


Figura 36 Exemplo de resultado de incremento de qualidade não usando o módulo *EnhanceEdges*.

Na Figura 35 é possível ver que existe uma melhoria relativamente ao exemplo da Figura 34, mas os valores dos objetos nas imagens é demasiado alterado para tomar como positivo, sendo perceptível pelo facto de as fronteiras dos objetos circundarem mais do que os objetos em si. Assim ponderou-se se bastaria a melhoria dos *speckles* para obter resultados mais favoráveis (ver Figura 37).

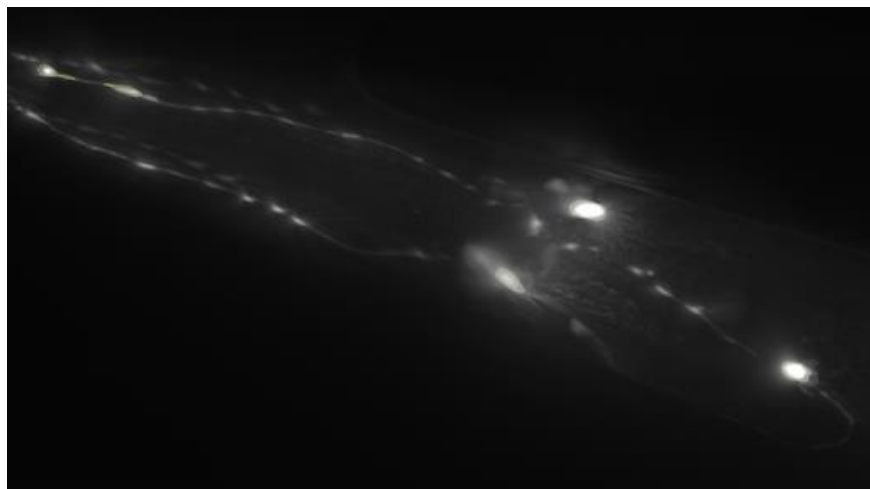


Figura 37 Exemplo de resultado de incremento de qualidade melhorando apenas os *Speckles* da Imagem.

Agora levantou-se outro problema: nem todos os corpos celulares estavam a ser encontrados. Assim decidiu-se reduzir o diâmetro dos objetos primários a serem detetados pelo módulo *IdentifyPrimaryObjects* de 30 para 10 unidades de *pixel* (Ver Figura 38 e 39).

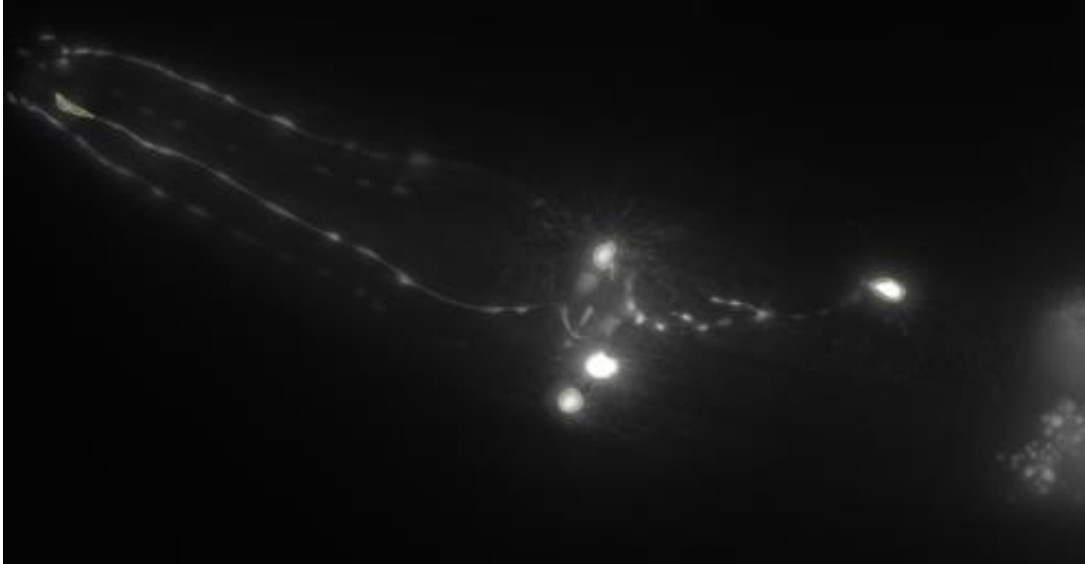


Figura 38 Exemplo de detecção de objetos com diâmetro mínimo de 30 unidades de *pixel*.

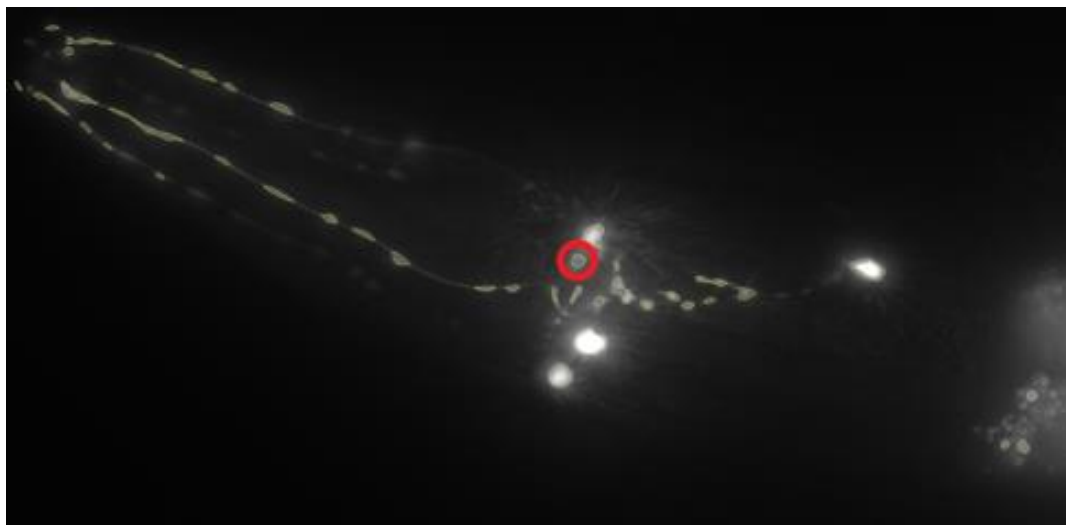


Figura 39 Exemplo de detecção de objetos com diâmetro mínimo de 10 unidades de *pixel*. A vermelho encontra-se assinalado um corpo celular que anteriormente não se encontrava detetado.

Com esta alteração de valor, todos os corpos celulares passaram a ser corretamente assinalados na grande maioria das imagens de teste mas, como a amplitude de valores de diâmetro de objetos primários é maior, é necessário fazer uma filtragem destes valores para apenas se ficar com os corretos assinalados.

Antes de ocorrer a filtração, ainda é necessário utilizar uma ferramenta para casos em que está presente mais que uma nematoda na imagem e apenas interessa a informação de um (ver Figura 40). Para tal é utilizado o módulo *Crop* que permite ao utilizador seleccionar a secção da imagem desejada (ver Figura 41).

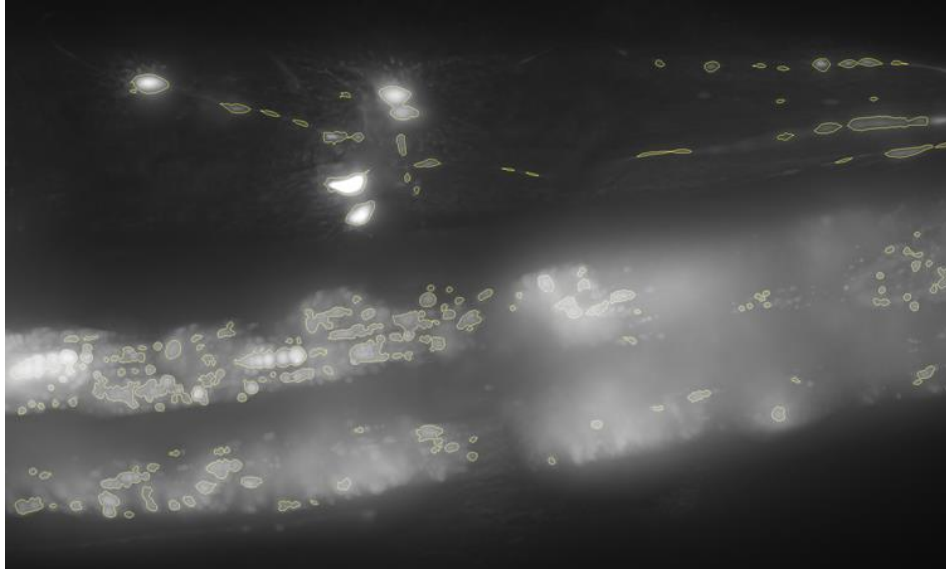


Figura 40 Exemplo de uma imagem com mais que um verme.

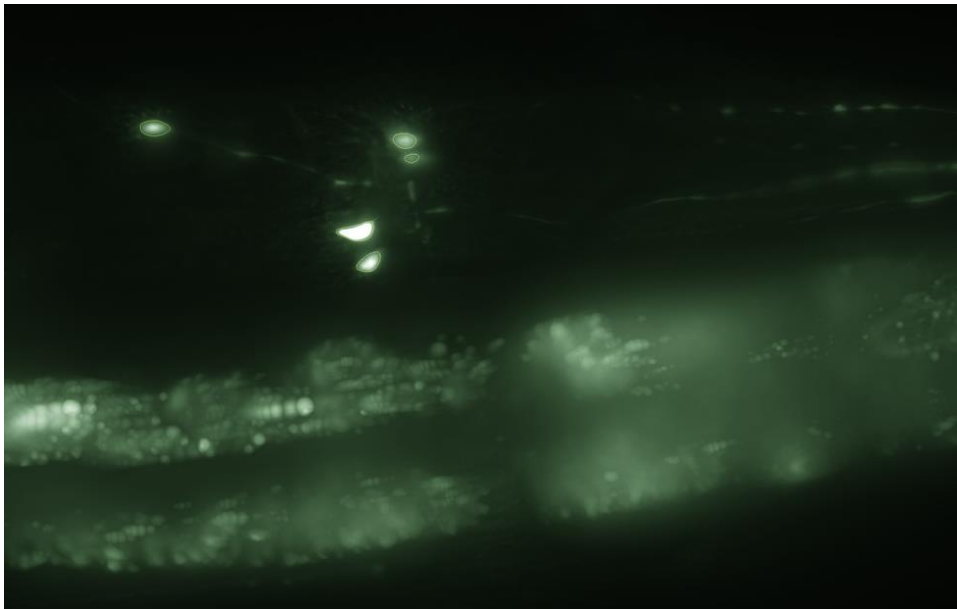


Figura 41 Exemplo de uma imagem com mais que um verme, mas após a aplicação de uma máscara com o local desejado à deteção de **Corpos Celulares**.

Como se pode verificar na Figura 41, vários dos falsos positivos foram eliminados, mas ainda se mantêm alguns dentro na zona de seleção. Para eliminar esses falsos positivos é

utilizado um processo de filtração. Com os dados fornecidos pela imagem e após a análise das ferramentas do CellProfiler conclui que se utilizaria inicialmente a intensidade dos objetos, a sua área e a sua forma (como os corpos celulares são normalmente circulares, utiliza-se a sua excentricidade). Para obter os valores dos mínimos e máximos a utilizar na filtração, foram obtidos os corpos das dez primeiras imagens e utilizados apenas os dados corretos destas (Tabela 2).

Tabela 2 Mínimos, Máximos e Média da Área, Excentricidade e Intensidade dos Corpos Celulares das dez primeiras imagens.

	Área	Excentricidade	Intensidade
Min	328	0,334	190,855
Max	2371	0,926	1737,154
Média	1278,375	0,680	966,715

Agora, com estes valores, são aplicados os filtros aos objetos, sendo os resultados obtidos descritos na tabela 3.

Tabela 3 Dados dos Corpos Celulares após aplicada filtração com os valores da Tabela 2.

Imagem	Corpos anotados pelo Perito	Corpos encontrados pelo CellProfiler	Filtração área	Filtração Excentricidade	Filtração Intensidade	Coefficiente de Dice
1	3	18	12	11	8	0,48
2	5	32	11	9	5	0,83
3	3	26	9	7	4	0,82
4	2	16	8	8	6	0,83
5	1	12	6	5	5	0,50
6	4	20	5	3	3	0,67
7	2	27	14	14	13	0,34
8	2	9	5	5	5	0,82
9	5	10	6	5	5	0,84
10	4	24	12	7	7	0,77
11	4	24	10	7	4	0,88
12	5	17	6	5	3	0,74
13	4	21	6	4	3	0,74
14	4	25	8	5	5	0,86
15	5	13	6	5	4	0,78
16	5	37	15	9	7	0,82
17	5	29	12	7	5	0,87
18	3	18	6	5	4	0,76
19	3	20	8	5	3	0,56
20	3	33	15	7	5	0,74
21	3	20	7	4	4	0,74
22	3	19	7	5	4	0,73
23	5	12	4	4	4	0,81
24	4	30	11	5	5	0,88
25	3	39	20	12	8	0,79
26	3	25	16	9	4	0,36
27	5	11	5	4	4	0,85
28	3	3	3	3	3	0,75
29	3	12	5	3	3	0,84
30	5	36	19	17	0	0,00
31	4	35	14	13	12	0,52
32	3	3	3	3	1	0,61
33	3	15	4	3	0	0,00
34	4	3	3	3	0	0,00
35	3	4	4	3	0	0,00
36	3	4	3	3	0	0,00

37	4	4	4	4	0	0,00
38	4	4	4	4	0	0,00
39	3	3	3	3	1	0,53
40	3	7	4	4	1	0,52
41	5	6	5	5	0	0,00
42	4	3	3	3	0	0,00
43	3	3	3	3	0	0,00
44	4	6	4	4	0	0,00
45	4	6	4	4	1	0,48
46	7	10	8	8	2	0,33
47	4	7	3	2	1	0,50
48	4	3	3	3	1	0,52
49	5	9	4	4	0	0,00
50	5	5	4	4	1	0,30
51	3	3	3	3	1	0,38
52	3	3	3	3	2	0,71
53	3	5	3	2	0	0,00
54	3	7	3	3	1	0,61
55	3	11	4	3	0	0,00
56	5	5	2	2	0	0,00
57	4	8	4	4	0	0,00
58	3	15	3	3	1	0,42
59	3	9	4	4	0	0,00
60	8	29	9	9	1	0,26
61	4	27	5	5	1	0,36
62	3	9	4	4	1	0,32
63	5	5	3	3	0	0,00
64	3	13	6	4	0	0,00
65	4	6	4	4	0	0,00
66	4	7	4	4	0	0,00
67	5	65	17	16	1	0,56
68	4	30	10	9	1	0,40
69	5	7	4	4	1	0,53
70	5	4	3	3	1	0,34
71	5	6	4	4	1	0,32
72	4	6	3	3	1	0,41
Total	277	1058	469	381	178	
Média						0,44

Após a análise dos resultados da tabela, verificou-se que a intensidade integrada das objetos não era um bom fator discriminatório, além de prejudicar bastante alguns dos resultados finais, como se pode ver pela existência de vários coeficientes de Dice nulos. Assim, este fator não foi considerado, tendo-se a Tabela 4 para a verificação dos resultados sem esse fator.

Tabela 4 Dados dos Corpos Celulares eliminando a filtragem por Intensidade Integrada.

Imagem	Corpos anotados pelo Perito	Corpos encontrados pelo CellProfiler	Filtragem área	Filtragem Excentricidade	Coefficiente de Dice
1	3	18	12	11	0,58
2	5	32	11	9	0,84
3	3	26	9	7	0,78
4	2	16	8	8	0,79
5	1	12	6	5	0,50
6	4	20	5	3	0,67
7	2	27	14	14	0,32
8	2	9	5	5	0,82
9	5	10	6	5	0,84
10	4	24	12	7	0,77
11	4	24	10	7	0,86
12	5	17	6	5	0,75
13	4	21	6	4	0,74

14	4	25	8	5	0,86
15	5	13	6	5	0,78
16	5	37	15	9	0,82
17	5	29	12	7	0,86
18	3	18	6	5	0,78
19	3	20	8	5	0,51
20	3	33	15	7	0,83
21	3	20	7	4	0,74
22	3	19	7	5	0,72
23	5	12	4	4	0,81
24	4	30	11	5	0,88
25	3	39	20	12	0,75
26	3	25	16	9	0,32
27	5	11	5	4	0,85
28	3	3	3	3	0,75
29	3	12	5	3	0,84
30	5	36	19	17	0,56
31	4	35	14	13	0,50
32	3	3	3	3	0,81
33	3	15	4	3	0,79
34	4	3	3	3	0,67
35	3	4	4	3	0,60
36	3	4	3	3	0,75
37	4	4	4	4	0,77
38	4	4	4	4	0,76
39	3	3	3	3	0,79
40	3	7	4	4	0,85
41	5	6	5	5	0,77
42	4	3	3	3	0,73
43	3	3	3	3	0,73
44	4	6	4	4	0,79
45	4	6	4	4	0,80
46	7	10	8	8	0,79
47	4	7	3	2	0,67
48	4	3	3	3	0,75
49	5	9	4	4	0,73
50	5	5	4	4	0,78
51	3	3	3	3	0,75
52	3	3	3	3	0,79
53	3	5	3	2	0,75
54	3	7	3	3	0,84
55	3	11	4	3	0,82
56	5	5	2	2	0,47
57	4	8	4	4	0,08
58	3	15	3	3	0,74
59	3	9	4	4	0,82
60	8	29	9	9	0,70
61	4	27	5	5	0,82
62	3	9	4	4	0,75
63	5	5	3	3	0,49
64	3	13	6	4	0,74
65	4	6	4	4	0,86
66	4	7	4	4	0,84
67	5	65	17	16	0,64
68	4	30	10	9	0,76
69	5	7	4	4	0,78
70	5	4	3	3	0,64
71	5	6	4	4	0,62
72	4	6	3	3	0,78
Total	277	1058	469	381	
Média					0,73

Como é possível observar, houve uma melhoria significativa dos resultados. Os coeficientes nulos foram completamente eliminados e a média é agora de 72%, estando as detecções quase

idênticas às anotações do perito, mas ainda existem imagens em que certos Corpos Celulares não foram identificados e ainda existem 104 falsos positivos.

Para eliminar os falsos positivos e melhorar os resultados finais, determinou-se que a melhor maneira seria através de uma filtragem manual, em que o utilizador eliminaria os falsos positivos restantes e identificaria os poucos Corpos Celulares que faltam. Estes serão então os resultados finais no que toca à deteção dos Corpos Celulares e podem ser visualizados na Tabela 5.

Tabela 5 Dados dos Corpos Celulares após a introdução da Filtragem Manual.

Imagem	Corpos anotados pelo Perito	Corpos encontrados pelo CellProfiler	Filtragem Área	Filtragem Excentricidade	Filtragem Manual	Coefficiente de Dice
1	3	18	12	11	3	0,81
2	5	32	11	9	5	0,87
3	3	26	9	7	3	0,83
4	2	16	8	8	2	0,88
5	1	12	6	5	1	0,92
6	4	20	5	3	4	0,78
7	2	27	14	14	2	0,82
8	2	9	5	5	2	0,90
9	5	10	6	5	5	0,84
10	4	24	12	7	4	0,87
11	4	24	10	7	4	0,88
12	5	17	6	5	5	0,82
13	4	21	6	4	4	0,85
14	4	25	8	5	4	0,88
15	5	13	6	5	5	0,84
16	5	37	15	9	5	0,83
17	5	29	12	7	5	0,87
18	3	18	6	5	3	0,80
19	3	20	8	5	3	0,76
20	3	33	15	7	3	0,91
21	3	20	7	4	3	0,77
22	3	19	7	5	3	0,82
23	5	12	4	4	5	0,81
24	4	30	11	5	4	0,89
25	3	39	20	12	3	0,89
26	3	25	16	9	3	0,81
27	5	11	5	4	5	0,85
28	3	3	3	3	3	0,75
29	3	12	5	3	3	0,84
30	5	36	19	17	5	0,74
31	4	35	14	13	4	0,82
32	3	3	3	3	3	0,81
33	3	15	4	3	3	0,79
34	4	3	3	3	4	0,67
35	3	4	4	3	3	0,71
36	3	4	3	3	3	0,75
37	4	4	4	4	4	0,77
38	4	4	4	4	4	0,76
39	3	3	3	3	3	0,79
40	3	7	4	4	3	0,86
41	5	6	5	5	5	0,77
42	4	3	3	3	4	0,83
43	3	3	3	3	3	0,79
44	4	6	4	4	4	0,79
45	4	6	4	4	4	0,80
46	7	10	8	8	7	0,79
47	4	7	3	2	4	0,86
48	4	3	3	3	4	0,84
49	5	9	4	4	5	0,77
50	5	5	4	4	5	0,85

51	3	3	3	3	3	0,75
52	3	3	3	3	3	0,79
53	3	5	3	2	3	0,80
54	3	7	3	3	3	0,84
55	3	11	4	3	3	0,82
56	5	5	2	2	5	0,83
57	4	8	4	4	4	0,08
58	3	15	3	3	3	0,81
59	3	9	4	4	3	0,83
60	8	29	9	9	8	0,78
61	4	27	5	5	4	0,82
62	3	9	4	4	3	0,75
63	5	5	3	3	5	0,73
64	3	13	6	4	3	0,80
65	4	6	4	4	4	0,86
66	4	7	4	4	4	0,84
67	5	65	17	16	5	0,85
68	4	30	10	9	4	0,84
69	5	7	4	4	5	0,78
70	5	4	3	3	5	0,80
71	5	6	4	4	5	0,74
72	4	6	3	3	4	0,78
Total	277	1058	469	381	277	
Média						0,80

Agora que os falsos positivos se encontram eliminados, o valor da média do Coeficiente de Dice é de aproximadamente 80%, sendo que as imagens das anotações do perito e os resultados do CellProfiler encontram-se com um nível de similaridade bastante elevado. Com os corpos celulares propriamente identificados, é possível obter a tabela com os dados das características de todos os objetos que o perito/utilizador deseje analisar, sendo que neste caso, apenas se mostra os valores de área, excentricidade e intensidade integrada (Anexo B). É importante voltar a referir que a ordem das imagens que o CellProfiler utiliza é alfabética, logo a imagem 2 na tabela 5 irá corresponder à imagem com o número 10 em seu nome e que a imagem 12 na tabela 5 irá corresponder à imagem 2 e de seguida virá a imagem 20 e por assim adiante. Esta organização pode ser visualizada num dos módulos iniciais do CellProfiler anteriormente referidos.

Após a deteção dos corpos celulares estar concluída, o objetivo seria a deteção dos axónios. Como foi dito, o pós-processamento é bastante idêntico ao utilizado com os Corpos Celulares, sendo que apenas se utiliza a área como filtro. Mais uma vez são então obtidos os valores para os axónios presentes nas dez primeiras imagens (Mínimo: 1516; Máximo: 121967).

A deteção final dos axónios acabou por ser um passo mal sucedido, pois em muitas imagens essa deteção foi errada (ver Figura 42), sendo então eliminada na filtração manual, obtendo-se imagens com zero axónios detetados ou então a área envolvendo os axónios era maior do que seria, englobando mais objetos que aqueles necessários (ver Figura 43). A edição manual, apesar de ser um ótimo meio para eliminar falsos positivos, no caso dos axónios não

é muito prática para a edição de áreas, pois devido aos *outlines* serem bastante compridos, irão existir muitos pontos de edição, impossibilitando a edição de zonas de deteção pré-existentes. É possível, no entanto, a criação de um novo objeto, mas existe um certo nível de destreza e esforço por parte do utilizador para delinear de forma correta os axónios. Desta forma, e porque o tempo disponível era reduzido, decidiu-se não continuar a investir nesta deteção, ficando esta parte para trabalho futuro.

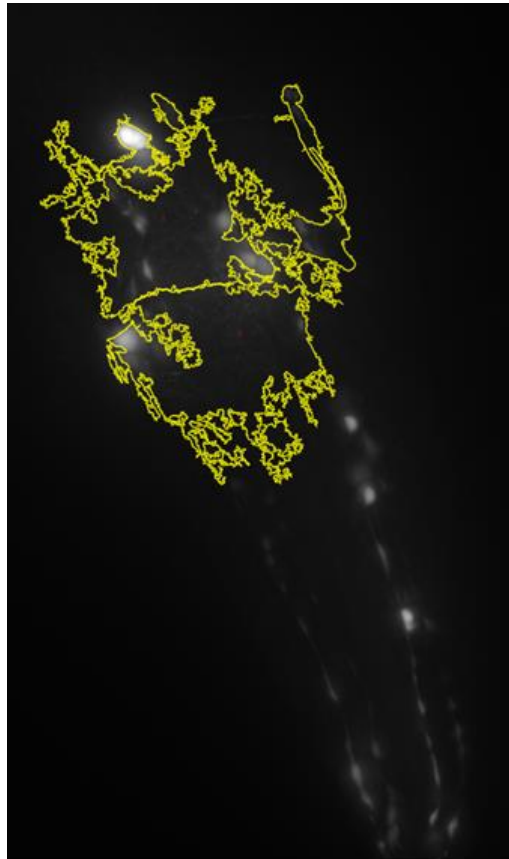


Figura 42 Exemplo do resultado de deteção de axónios antes de ocorrer qualquer tipo de filtragem.

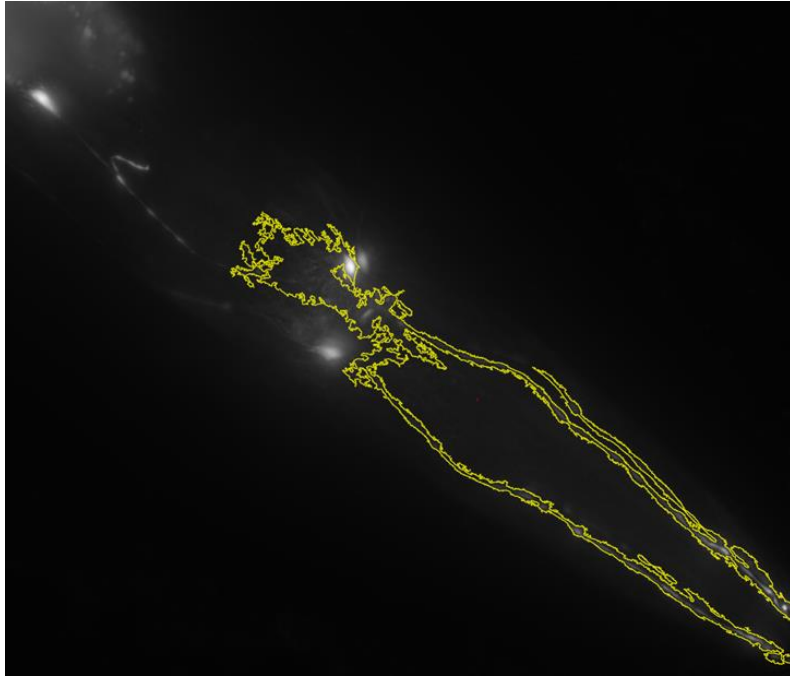


Figura 43 Exemplo do resultado de detecção de axónios após a aplicação dos filtros.

5. CONCLUSÃO

A análise de imagens manual é uma tarefa complicada, pois cada imagem possui dados diferentes de outra e é necessário uma capacidade de observação detalhada, que está sujeita a regulares erros humanos pelo que uma análise automática de imagens é uma tarefa ambiciosa e útil num ambiente de trabalho intensivo e sujeito a prazos apertados.

O *software* CellProfiler é uma ferramenta bastante maleável, mas não deixa de ser uma ferramenta que requer estudo por parte do utilizador, sendo que este *software* não é muito mencionado em muitos artigos de análise de imagens biológicas, logo não foi possível encontrar qualquer artigo recente que o mencione. Por esta razão, muitas vezes o CellProfiler não chega a ser considerado como uma ferramenta de desenvolvimento válida e acaba por ser substituído por outro tipo de *software* mais usado e popular, como por exemplo o *Matlab*.

Apesar disso e tendo em conta o desafio que foi apresentado no âmbito deste relatório, é seguro dizer que o CellProfiler é útil e permitiu uma resposta ao problema colocado, sendo possível um resultado mais rápido sem a necessidade de um estudo aprofundado e demorado de um outro *software open source*.

Dividindo a deteção dos neurónios em duas das suas partes constituintes, Corpo Celular e Axónio, a *pipeline* final conseguiu criar uma resposta para a deteção dos Corpos Celulares, mas ainda tem espaço para melhoria no que toca à deteção dos Axónios. Ao contrário dos Corpos Celulares, o Axónio é a parte que se encontra mais oculta nas imagens fornecidas dificultando mesmo a identificação a olho nu por parte de um utilizador. Além disso, é a zona que sofre uma maior degradação à medida que o verme envelhece.

Numa perspetiva para trabalhos futuros e tendo em conta [10], será interessante a incorporação da utilização de imagens de treino que auxiliarão na deteção dos componentes da imagem, possibilitando talvez a exclusão da filtração manual dos falsos positivos, tornando assim o processo completamente automático e melhorando os resultados da deteção dos axónios.

Referências

- [1] A. Diogo, M. Mota, “*Caenorhabditis elegans: Modelo Biológico para o século XXI*”, Laboratório de Nematologia/ICAM, Universidade de Évora, 2001.
- [2] A. Hartley, C. Marks, “Atlas of *C. elegans* Anatomy”, WormAtlas.org. [Em linha].Disponívelem:
<http://www.wormatlas.org/ver1/handbook/anatomyintro/anatomyintro.htm>. Consulta em 16 de Fevereiro 2015.
- [3] E. A. C. Pereira, T. Z. Aziz, “Parkinson’s disease and primate research: past, present, and future” History of Medicine, 2006.
- [4] J. Jankovic, "Parkinson's disease: clinical features and diagnosis". Journal of Neurology, Neurosurgery, and Psychiatry, 2008.
- [5] S. Saha, et al.,“ LRRK2 Modulates Vulnerability to Mitochondrial Dysfunction in *Caenorhabditis elegans*” The Journal of Neuroscience, 2009.
- [6] Z. Liu, et al., “Inhibitors of LRRK2 kinase attenuate neurodegeneration and Parkinson-like phenotypes in *Caenorhabditis elegans* and *Drosophila* Parkinson’s disease models”, Human Molecular Genetics, 2011.
- [7] Geng, W., Cosman, P., “Automatic Tracking, Feature Extraction and Classification of *C. elegans* Phenotypes”, UC San Diego Previously Published Works”, 2004.
- [8] Lei Q., Fuhui L., Hanchuan P., “3-D Registration of Biological Images and Modules : Registration of microscopic images and it’s uses in segmentation and anotation” , IEEE Signal Processing Magazine, 2015.
- [9] Moy, K. L. "C. elegans Segmentation." 2014.
- [10] Greenblum, Ayala, et al. "Dendritic tree extraction from noisy maximum intensity projection images in *C. elegans*." Biomedical Engineering Online, 2014.
- [11] Babalola, Kolawole O., et al. "Comparison and evaluation of segmentation techniques for subcortical structures in brain MRI." Medical Image Computing and Computer-Assisted Intervention–MICCAI 2008. Springer Berlin Heidelberg, 2008.
- [12] Manual de Instruções do CellProfiler. Disponível em:
<http://www.cellprofiler.org/manuals.shtml>. Consulta em 30 de Setembro 2015.
- [13] Altun, Z.F. and Hall, D.H., “Nervous system, neuronal support cells”. WormAtlas, 2010.

Anexo A. Funcionamento dos Módulos do CellProfiler

De acordo com [12], sendo esta a fonte de informação mais fiável para a explicação do funcionamento dos módulos de CellProfiler, segue uma lista com a descrição do funcionamento daqueles que foram utilizados na *pipeline* final.

CROP

Este módulo corta as imagens num retângulo, numa elipse, numa forma arbitrária fornecida pelo utilizador, na forma de um objeto(s) identificado por um módulo identificador, ou numa forma criada usando um módulo de cultura anterior no *pipeline*. Preciso notar que o corte muda o tamanho de imagens, que pode ter consequências inesperadas. Por exemplo, a identificação de objetos numa imagem recortada e, em seguida, tentar medir sua intensidade na imagem original não vai funcionar porque as duas imagens não são do mesmo tamanho.

Medições disponíveis:

- *AreaRetainedAfterCropping*: A área da imagem para a esquerda após o corte.
- *OriginalImageArea*: A área da imagem de entrada original.

ConFigurações:

- *Select the input image*: Escolher a imagem a ser cortada.
- *Name the output image*: Digitar o nome a ser dado à imagem cortada.
- *Select the cropping shape*: Selecionar a forma na qual se irá realizar o corte:
 - *Rectangle*: Autoexplicativo.
 - *Ellipse*: Autoexplicativo.
 - *Image*: O corte ocorrerá com base numa imagem binária que a especificar. Uma caixa de escolha com imagens disponíveis aparece, a partir da qual se deverá selecionar uma imagem.

- *Objects*: corta com base em objetos rotulados, identificados por um módulo anterior.
- *Previous cropping*: O corte gerado por um módulo de corte anterior. Ser-se-á capaz de selecionar as imagens que foram geradas por módulos de corte anteriores.
- *Select the cropping method*: Escolher se corte irá ser realizado com coordenadas de *pixel* digitadas ou utilizando o rato do computador.
 - *Coordinates*: Para *Ellipse*, deve-se inserir os parâmetros geométricos da elipse. Para *Rectangle*, deve-se especificar as coordenadas dos cantos.
 - *Mouse*: Para *Ellipse*, deve-se clicar em cinco ou mais pontos para definir uma elipse ao redor da parte da imagem que se deseja analisar. Para *Rectangle*, pode-se clicar em tantos pontos como desejado que estejam no interior da região que se deseja reter.
- *Apply which cycle's cropping pattern?*: Especificar como um dado padrão de corte deve ser aplicado a outros ciclos de imagem:
 - *First*: O padrão de corte a partir do primeiro ciclo de imagem é aplicado a todos os Ciclos subsequentes. Isto é útil se a primeira imagem destina-se a funcionar como algum tipo de modelo.
 - *Every*: Cada ciclo de imagem é cortado individualmente.
- *Left and right rectangle positions*: Especificar as posições à esquerda e direita para o retângulo delimitador, selecionando uma das seguintes opções:
 - *Absolute*: Especificar estes valores como coordenadas de *pixel* absoluto na imagem original. Por exemplo, pode-se digitar "25", "225" e "Absolute" para criar uma imagem de 200×200 *pixels* que é de 25 *pixels* a partir do canto superior esquerdo.
 - *From edge*: Especificar a posição em relação à borda da imagem. Por exemplo, pode-se digitar "25", "25", e "Edge" para cortar 25 *pixels* de ambas as margens

esquerda e direita da imagem, independentemente do tamanho original da imagem.

- **Top and bottom rectangle positions:** Especificar as posições superior e inferior para o retângulo delimitador, selecionando uma das seguintes opções:
 - *Absolute:* Especificar os valores como coordenadas de *pixel* absoluto. Por exemplo, pode-se digitar "25", "225" e "Absolute" para criar uma imagem de 200×200 *pixels* que é 25 *pixels* a partir do canto superior esquerdo.
 - *From edge:* Especificar a posição em relação à borda da imagem. Por exemplo, pode-se digitar "25", "25", e "Edge" para cortar 25 *pixels* a partir das bordas das imagens, independentemente do seu tamanho.
- *Coordinates of ellipse center:* Especificar a posição do *pixel* do centro da elipse.
- *Ellipse radius, X direction:* Especifica o raio da elipse na direção de X.
- *Ellipse radius, Y direction:* Especifica o raio da elipse na direção de Y.
- *Use Plate Fix?:* Selecionar "Yes" para tentar regularizar as bordas em torno de um objeto identificado anteriormente. Ao tentar cortar com base num objeto previamente identificado como uma placa retangular, a placa pode não ter bordas retas, pelo que pode haver um pequeno, quase imperceptível, "apêndice". Sem *Plate Fix*, o módulo de corte não cortará a imagem firmemente o suficiente, manteria o pequeno apêndice, deixando um monte de espaço em branco em torno da placa e potencialmente causando problemas com módulos posteriores (especialmente os que envolvem correção de iluminação). *Plate Fix* converte o objeto e as culturas identificadas para excluir quaisquer apêndices menores (tecnicamente, qualquer linha horizontal ou vertical, em que o objeto ocupa menos de 50% da imagem). Também define os *pixels* em torno da borda do objeto (para as regiões superiores a 50% mas menos de 100%) que de outra forma seria 0 com o valor de *pixel* do fundo da imagem, evitando assim problemas com outros módulos.
- *Remove empty rows and columns?:* Usar esta opção para remover linhas e colunas que não possuem objetos:

→ *No*: Deixa a imagem do mesmo tamanho. As áreas cortadas serão definidas como zeros, e aparece como preto.

→ *Edges*: Corta a imagem de modo que a parte superior, inferior, esquerda e direita estejam no primeiro *pixel* não-branco dessa borda.

→ *All*: Remove qualquer linha ou coluna de *pixels* em branco, mesmo a partir da parte interna da imagem.

- *Select the masking image*: Selecionar a imagem a ser usada como uma máscara de corte.
- *Select the image with a cropping mask*: Selecionar a imagem associada com a máscara de corte a usar.
- *Select the objects*: Escolher os objetos que serão usados como máscara de corte.

APPLYTHRESHOLD

Coloca as intensidades de *pixel* abaixo ou acima de um certo *threshold* a zero. *ApplyThreshold* produza uma imagem a escala de cinza ou binária com base num *threshold* que pode ser pré-selecionado ou calculado automaticamente usando um dos muitos métodos.

ConFigurações:

- *Select the input image*: Escolher a imagem a sofrer o *threshold*.
- *Name the output image*: Digitar um nome para a imagem a sofrer o *threshold*.
- *Select the output image type*: Podem ser produzidos dois tipos de imagens de saída:
 - *Grayscale*: Os *pixels* que são retidos após alguns *pixels* serem definidos como zero ou deslocados terão os seus valores de intensidade originais.
 - *Binary (black and white)*: Os *pixels* que são retidos após alguns *pixels* serem definidos como zero será branco e todos os outros *pixels* serão pretos (zeros).
- *Set pixels below or above the threshold to zero?*: Para a saída em tons de cinza, os *pixels* mais escuros abaixo do valor de *threshold* podem ser ajustados para zero ou os

pixels brilhantes acima do valor de *threshold* podem ser definidos para zero. Escolher “Below threshold” para transformar *pixels* mais escuros e “Above threshold” para transformar *pixels* brilhantes.

- *Subtract the threshold value from the remaining pixel intensities?*: Escolher “Yes” para mudar o valor dos *pixels* mais escuros pelo o valor de *threshold*.
- *Number of pixels by which to expand the thresholding around those excluded bright pixels*: Esta configuração é útil para tentar excluir objetos de artefactos brilhantes. Em primeiro lugar, define-se o *threshold* para excluir esses objetos brilhantes.
- *Threshold strategy*: determina o tipo de entrada que é usada para calcular o *threshold*. Os *thresholds* da imagem podem ser baseados em:
 - As intensidades de *pixel* da imagem de entrada (isto é a mais comum).
 - Um único valor fornecido manualmente pelo utilizador.
 - Um único valor produzido por um módulo de medição anterior.
 - Uma imagem binária (chamado de máscara), onde alguns dos valores de intensidade de *pixel* são definidos como 0, e os outros são definidos como 1.

As escolhas para *Threshold strategy* são:

- *Automatic*: Usa as configurações padrão para *thresholding*. Esta estratégia calcula o *threshold* utilizando o método MCT em toda a imagem e aplica o *threshold* à imagem, suavizada com um Gaussiano com sigma igual a 1.
- *Global*: Calcula um valor de *threshold* único com base nos *pixels* da imagem de entrada e usa esse valor para classificar os *pixels* acima do valor de *threshold* como primeiro plano e abaixo como plano de fundo.
- *Adaptive*: Distribui a imagem de entrada em telhas e calcula os valores de *threshold* para cada telha. Para cada telha, o *threshold* calculado é aplicado apenas aos *pixels* dentro dessa telha.

- *Per object*: Usa objetos de um módulo anterior como *IdentifyPrimaryObjects* para definir a região de interesse para ser aplicada o *threshold*. Calcula um *threshold* separado para cada objeto e, em seguida, aplica esse *threshold* aos *pixels* dentro do objeto. Os *pixels* fora dos objetos são classificados como fundo.
- *Manual*: Digita-se um único valor entre 0 e 1 que se aplica a todos os ciclos e é independente da imagem de entrada.
- *Binary image*: Usa uma imagem binária para classificar *pixels* como primeiro ou segundo plano. Valores de *pixels* diferentes de zero serão primeiro plano e os valores de *pixels* iguais a zero serão o fundo.
- *Measurement*: Usa uma medição de uma imagem anterior como *threshold*. A medição deve ter valores entre 0 e 1.
- *Thresholding method*: O *threshold* de intensidade afeta a decisão de se cada *pixel* será considerado com primeiro plano (região de interesse) ou fundo. Um valor de *threshold* elevado irá resultar apenas na detecção das regiões mais brilhantes, ao passo que um valor de *threshold* reduzido irá incluir regiões mais escuras. Pode-se calcular o valor de *threshold* automaticamente a partir de uma escolha de vários métodos, ou pode-se inserir manualmente um número entre 0 e 1. Há um grande número de métodos para encontrar automaticamente *thresholds*:
 - *Otsu*: Esta abordagem calcula o *threshold* que separa as duas classes de *pixels* (de primeiro plano e plano de fundo), minimizando a variância dentro da cada classe.
 - *Mixture of Gaussian (MoG)*: Esta função assume que os *pixels* na imagem pertencem a uma classe de fundo ou a uma classe de primeiro plano, usando uma estimativa inicial da fração da imagem que está coberta pelo primeiro plano.
 - *Background*: Este método encontra simplesmente o modo do histograma da imagem, que se presume ser o fundo da imagem, e escolhe um *threshold* ao

dobro desse valor. O cálculo inclui os *pixels* entre 2% e 98% do intervalo de intensidade.

→ *RobustBackground*: este método também é simples e assume que a distribuição do fundo aproxima-se de um Gaussiano aparando 5% das intensidades de *pixel* mais brilhantes e mais escuras. Em seguida, calcula a média e o desvio padrão dos *pixels* restantes e calcula o *threshold* como a média + 2 vezes o desvio padrão.

→ *RidlerCalvard*: Este método é simples e com resultados muito semelhantes ao método de Otsu. *RidlerCalvard* escolhe um *threshold* inicial e, em seguida iterativamente calcula o próximo tomando a média das intensidades médias dos *pixels* de fundo e de primeiro plano determinados pelo primeiro *threshold*. O algoritmo então repete este processo até que o *threshold* convirja para um único valor.

→ *Kapur*: Este método calcula o *threshold* de uma imagem pesquisando o *threshold* que maximiza a soma das entropias dos valores de *pixel* do primeiro e segundo plano, quando tratados como distribuições separadas.

→ *Maximum correlation thresholding (MCT)*: Este método calcula a correlação máxima entre a máscara binária criado por *thresholding* e a imagem de *threshold* e é um pouco matematicamente semelhante ao Otsu.

- *Select binary image*: Seleccionar a imagem binária de *threshold* para ser utilizada.
- *Manual threshold*: Digite o valor de 0 a 1 que funcionará como um *threshold* absoluto para as imagens.
- *Select the measurement to threshold with*: Escolha a medição da imagem que atuará como um *threshold* absoluto para as imagens.
- *Two-class or three-class thresholding?*:

→ *Two classes*: Seleccionar esta opção se os níveis de tons de cinza são facilmente distinguíveis em apenas duas classes: primeiro plano (ou seja, regiões de interesse) e o fundo.

- *Three classes*: Escolher esta opção se os níveis de cinza caírem em três classes: primeiro plano, fundo e uma intensidade média entre os dois. Em seguida, será perguntado se a classe de intensidade média deverá ser adicionada ao primeiro plano ou ao fundo, a fim de gerar a saída final de duas classes.
- *Approximate fraction of image covered by objects?*: Introduzir uma estimativa de quanto da imagem é coberta com objetos, que é usada para estimar a distribuição de intensidades de *pixel*.
- *Method to calculate adaptive window size*: O método adaptativo divide a imagem em blocos, calculando o *threshold* para cada bloco. Existem duas formas para calcular o tamanho de bloco:
 - *Image size*: O tamanho do bloco é um décimo das dimensões da imagem, ou 50×50 *pixels*, o que for maior.
 - *Custom*: O tamanho do bloco é especificado pelo usuário.
- *Size of adaptive window*: Digitar a janela para o método adaptativo. Por exemplo, pode-se usar um múltiplo do maior tamanho do objeto esperado.
- *Threshold correction factor*: Esta configuração permite ajustar o *threshold*, calculado pelo método acima. O valor introduzido aqui ajusta o *threshold* para cima ou para baixo, através da multiplicação por este valor. Um valor igual a 1 indica nenhum ajuste, 0 a 1 faz o *threshold* mais brando e >1 faz com que o *threshold* seja mais rigoroso.
- *Lower and upper bounds on threshold* : Digitar o valor de *threshold* mínimo e máximo permitido, um valor entre 0 e 1. Isto é útil como uma precaução de segurança quando o *threshold* é calculado automaticamente, substituindo o *threshold* automático.
- *Select the smoothing method for thresholding*: A imagem de entrada pode ser opcionalmente suavizada antes de sofrer *threshold*. A suavização pode melhorar a uniformidade dos objetos resultantes, removendo buracos e bordas irregulares causadas por ruído na imagem adquirida. A suavização não é provavelmente

apropriada, se a imagem entrada for binária, se já tiver sido suavizada ou se é uma saída do módulo *ClassifyPixels*. As opções são:

- *Automatic*: Alisa a imagem com um Gaussiano com um desvio padrão de um *pixel* antes do *threshold*. Isto é adequado para a maioria das aplicações de análise.
- *Manual*: Alisa a imagem com um Gaussiano com escala controlada pelo usuário.
- *No smoothing*: Não aplica qualquer suavização antes do *threshold*.
- *Threshold smoothing scale*: Esta definição controla a escala utilizada para suavizar a imagem de entrada antes do *threshold* ser aplicado. A escala deve ser aproximadamente o tamanho dos artefactos a serem eliminados por alisamento. Um Gaussiano é usado com um desvio padrão ajustado de modo a que 1/2 da distribuição Gaussiana caia dentro do diâmetro determinado pela escala (escala $\sigma = / 0,674$).

IMAGEMATH

ImageMath executa operações matemáticas simples com as intensidades da imagem.

Este módulo pode realizar operações de adição, subtração, multiplicação, divisão, ou média de duas ou mais intensidades de imagem, bem como inversão, transformada logarítmica, ou escalar por uma constante em intensidades de imagem individuais.

Após as operações solicitadas serem realizadas, a imagem final pode ter uma gama substancialmente diferente da intensidade dos *pixels* do que o original. CellProfiler assume que a imagem será dimensionada de 0 a 1 para fins de identificação e exibição de objetos, de modo que pode ser necessário redimensionamento adicional.

ConFigurações:

- *Operation*: Selecionar a operação a executar. Se forem escolhidas mais do que duas imagens, as operações serão realizadas sequencialmente da primeira para a última, por exemplo, para "Divide", (Image1 / Image2) / Image3.
 - *Add*: Adiciona a primeira imagem à segunda, e assim por diante.

- *Subtract*: Subtrai a segunda imagem da primeira.
 - *Absolute Difference*: O valor absoluto da diferença entre a primeira e segunda imagem.
 - *Multiply*: Multiplica a primeira imagem pela segunda.
 - *Divide*: Divide a primeira imagem pela segunda.
 - *Average*: Calcula a média das intensidades das imagens carregadas no módulo. Isto é equivalente à opção *Add* dividida pelo número de imagens carregadas neste módulo.
 - *Maximum*: Retorna o valor do elemento máximo em cada localização dos *pixels*.
 - *Invert*: subtrai as intensidades de imagem de 1. Isto faz com que a cor mais escura seja a mais brilhante e vice-versa.
 - *Log transform (base 2)*: Executa uma transformada logarítmica para cada intensidade dos *pixels*.
 - *None*: Esta opção é útil para apenas selecionar algumas das opções do módulo, como a adição, multiplicação, ou exponenciar uma imagem por uma constante.
- *Name the output image*: Digitar um nome para a imagem resultante.
 - *Image or measurement?*: Pode-se executar operações matemáticas utilizando duas imagens ou pode-se usar uma medida para um dos operandos. Por exemplo, para dividir a intensidade de uma imagem por outra, escolhe-se “Image” para ambos e seleciona-se as respectivas imagens. Para dividir a intensidade de uma imagem pela sua intensidade média, usa-se *MeasureImageIntensity* antes deste módulo para calcular a intensidade média, em seguida, seleciona-se “Measurement” e utiliza-se a medição como o denominador.
 - *Measurement*: Trata-se de uma medição efetuada na imagem. O valor da medição é utilizado para o operando para todos os *pixels* da outra imagem operando. Digita-se

o número que deseja-se multiplicar a imagem acima por. Esta multiplicação é aplicada antes das outras operações.

- *Raise the power of the result by*: Digitar um expoente para elevar o resultado para depois da operação escolhida.
- *Multiply the result by*: Introduzir um fator a multiplicar o resultado depois da operação escolhida.
- *Add to result*: Digitar um número para adicionar ao resultado depois da operação escolhida.
- *Set values less than 0 equal to 0?*: Os valores fora do intervalo de 0 a 1 podem não ser bem tratados por outros módulos. Selecionar “Yes” para definir valores negativos a 0.
- *Set values greater than 1 equal to 1?*: Os valores fora do intervalo de 0 a 1 podem não ser bem tratados por outros módulos. Selecionar “Yes” para definir valores superiores a 1 para um valor máximo de 1.
- *Ignore the image masks?*: Normalmente, a máscara mais pequena de todos os operandos de imagem é aplicada após o *ImageMath* ter terminado. Selecionar “Yes” para definir como iguais os zero todos os pixels anteriormente mascarados e operar nas imagens mascaradas como se nenhuma máscara tivesse sido aplicada.

ENHANCEORSUPPRESSFEATURES

EnhanceOrSuppressFeatures aprimora ou suprime certas características da imagem, o que pode melhorar uma posterior identificação dos objetos.

Este módulo aumenta ou suprime a intensidade de *pixels* determinados em relação ao resto da imagem, através da aplicação de filtros de processamento de imagem. Produz uma imagem em tons de cinza em que os objetos podem ser identificados através de um módulo identificador.

ConFigurações:

Select the input image: Seleciona a imagem com características a serem reforçadas ou suprimidas.

Name the output image: Digita um nome para a imagem suprimida ou reforçada.

Select the operation: Seleciona se quer aumentar ou suprimir as características designadas.

- *Enhance:* produz uma imagem cuja intensidade é composta em grande parte das características de interesse.
- *Suppress:* produz uma imagem com as características em grande parte removidas.

Feature size: Introduce o diâmetro do maior *speckle*, a largura da circunferência ou a largura das neurites a serem reforçadas ou suprimidas, que será utilizado para calcular o tamanho de filtro adequado. Para ver a intensidade dos *pixels* numa imagem aberta, usa-se a ferramenta de intensidade de *pixel* que está disponível em qualquer janela de exibição aberta. As intensidades dos *pixels* aparecem na barra inferior da janela do *display* quando se move o rato sobre a imagem.

Feature type: Este módulo pode melhorar três tipos de características de intensidade de imagem:

- *Speckles:* Um *speckle* é uma área de maior intensidade em relação à sua vizinhança imediata. O módulo aumenta os *speckles* utilizando um filtro *white tophat*, o qual é a imagem em tons de cinza menos a abertura morfológica da imagem. A operação de abertura suprime primeiro os *speckles* através da aplicação de uma erosão em tons de cinza para reduzir tudo dentro de um determinado raio para o valor mais baixo dentro desse raio, em seguida, usa uma dilatação em tons de cinza para restaurar objetos maiores do que o raio para uma aproximação da sua forma anterior. O filtro *white tophat* aumenta os *speckles* subtraindo os efeitos da abertura da imagem original.
- *Neurites:* Neurites são formas longas e finas de intensidade reforçada. Esta opção melhora a intensidade de neurites, utilizando os métodos *Line structures* ou *Tubeness* descritos abaixo.

- *Dark Holes*: O módulo utiliza reconstrução morfológica (o algoritmo *rolling-ball*) para identificar buracos negros dentro de áreas mais claras, ou formas de anéis mais brilhantes. A imagem é invertida de modo a que os buracos negros transformem-se em picos brilhantes. A imagem é erodida, sucessivamente, e a imagem erodida é reconstruída em cada passo, resultando numa imagem que não tem os picos. Finalmente, a imagem reconstruída é subtraída da imagem reconstruída anteriormente. Isto deixa manchas brilhantes circulares com um raio igual ao número de iterações realizadas.
- *Circles*: O módulo calcula a transformada circular de Hough da imagem pelo diâmetro introduzido em *Feature Size*. A transformada de Hough terá uma maior intensidade em pontos que são centrados dentro de um anel de *pixels* de alta intensidade em que o diâmetro do anel é o *Feature Size*.
- *Texture: EnhanceOrSuppressFeatures* produz uma imagem cuja intensidade é a variação entre os *pixels* próximos. Este método de avalia as contribuições dos *pixels* por distância usando uma função gaussiana.
- *DIC*: Este método recupera a densidade ótica de uma imagem *DIC (Differential Interference Contrast)* por integração numa direção perpendicular à direção de corte da imagem.

Range of hole sizes: A gama de tamanhos dos *Dark Holes* serem reforçados. O algoritmo irá identificar apenas orifícios cujos diâmetros se situam entre estes dois valores.

Smoothing scale: Utilizado nos métodos *Texture*, *DIC* e *Neurites*.

- *Texture*: Esta é a escala das características de textura em *pixels*. O algoritmo utiliza o valor de *Smoothing scale* inserido como o sigma da função Gaussiana utilizado para medir *pixels* próximos na determinação da variação.
- *DIC*: Especifica a quantidade de suavização da imagem na direção paralela ao eixo de corte da imagem. O método de integração linear vai deixar marcas na imagem sem alisamento à medida que encontra *pixels* ruidosos durante o curso da integração. O alisamento usa contribuições de *pixels* nas proximidades, o que diminui o ruído, mas suaviza a imagem resultante.

- *Neurites*: A opção *Tubeness* utiliza esta escala como o sigma da função Gaussiana usado para suavizar a imagem antes do gradiente de detecção.

Shear angle: O *shear angle* (ângulo de corte) é a direção do valor constante para as sombras e luzes numa imagem DIC. Os gradientes numa imagem DIC surgem na direção perpendicular ao ângulo de distorção. Por exemplo, se as sombras surgirem na diagonal do canto inferior esquerdo para o canto superior direito e os focos luminosos aparecem acima das sombras, o ângulo de corte é de 45° . Se as sombras aparecem no topo, o ângulo de corte é de $180^\circ + 45^\circ = 225^\circ$.

Decay: aplica-se um decaimento exponencial durante o processo de integração multiplicando a soma acumulada pelo decaimento de cada etapa. Isto permite à integração recuperar do erro acumulado durante o curso da integração, mas também resulta em intensidades diminuídas no meio de grandes objetos.

Enhancement method: Dois métodos podem ser utilizados para melhorar as neurites:

- *Tubeness*: Este método é uma adaptação do método utilizado pelo *plugin ImageJ Tubeness*. A imagem é suavizada com uma função Gaussiana. A matriz Hessiana é então calculada em cada ponto para medir o gradiente de intensidade e os valores próprios da matriz são computados para determinar a magnitude da intensidade. A máxima absoluta dos dois valores próprios dá uma medida da relação entre a intensidade do gradiente na direção da sua descida mais rápida vs. a direção ortogonal. A imagem de saída é a magnitude absoluta do valor próprio mais alto se esse valor próprio for negativo (neurite branca sobre um fundo preto), caso contrário, zero.
- *Line structures*: O módulo leva a diferença dos filtros *White and Black Tophat* (uma filtragem *White Tophat* é a imagem menos a abertura de tons de cinza morfológica da imagem; uma filtragem *Black Tophat* é a aproximação em tons de cinza morfológicos da imagem, menos a imagem). O efeito é o de aumentar a linhas cuja largura é o valor em *Feature size*.

IDENTIFYPRIMARYOBJECTS

IdentifyPrimaryObjects identifica componentes biológicos de interesse em imagens em tons de cinza contendo objetos brilhantes num fundo escuro.

No CellProfiler, usa-se o termo objeto como um termo genérico para referir um recurso identificado numa imagem, geralmente um subcompartimento celular de algum tipo (por exemplo, núcleos, células, colónias, vermes). Define-se um objeto como primário quando este pode ser encontrado numa imagem sem a necessidade do auxílio de uma outra característica celular como uma referência. Por exemplo: Os núcleos das células geralmente são mais facilmente identificáveis devido à sua morfologia mais uniforme, de alto contraste em relação ao fundo.

Em contraste, as células muitas vezes têm padrões irregulares de intensidade e são de menor contraste com a coloração mais difusa, tornando-as mais difíceis de identificar do que os núcleos. Além disso, as células muitas vezes encontram-se em contacto com seus vizinhos, sendo mais difícil delinear as fronteiras celulares

Para utilizar este módulo, é necessário que a imagem de entrada tenha as seguintes qualidades:

- A imagem deve ser em tons de cinza.
- O primeiro plano (isto é, as regiões de interesse) é mais claro que o fundo.

Como saída deste módulo é adquirido um conjunto de objetos primários, o qual pode ser utilizado noutros módulos para efeitos de medição ou outras operações.

Medições disponíveis:

- Medidas de imagens:
 - *Count*: O número de objetos primários identificados.
 - *OriginalThreshold*: O *threshold* global para a imagem.
 - *FinalThreshold*: Para os métodos de *threshold* globais, este valor é o mesmo que o obtido em *OriginalThreshold*. Para os métodos por objeto ou adaptativos, este valor representa a média dos *thresholds* locais.
 - *WeightedVariance*: A soma dos desvios transformação logarítmica dos *pixels* do primeiro e segundo plano, ponderada pelo número de *pixels* em cada distribuição.

→ *SumOfEntropies*: A soma das entropias computadas a partir das distribuições do primeiro e segundo plano.

- Medições de objetos:

→ *Location_X, Location_Y*: As coordenadas (X, Y) do *pixel* dos centróides dos objetos primários. O centróide é calculado como o centro de massa da representação binária do objeto.

ConFigurações:

- *Select the input image*: Selecione a imagem desejada para identificar objetos.
- *Name the primary objects to be identified*: Digite o nome desejado para os objetos identificados neste módulo.
- *Typical diameter of objects, in pixel units (Min,Max)*: Esta configuração permite ao utilizador fazer uma distinção com base no tamanho, que pode ser usado em conjunto com a configuração *Discard objects outside the diameter range?* para remover os objetos que falhem este critério.
- *Discard objects outside the diameter range?*: Selecionar “Yes” para descartar objetos fora do intervalo especificado em *Typical diameter of objects, in pixel units (Min,Max)*. Selecionar “No” para ignorar este critério.
- *Try to merge too small objects with nearby larger objects?*: Selecionar “Yes” para que objetos que são menores do que o diâmetro mínimo especificado sejam unidos, se possível, com outros objetos em redor. Isto é útil nos casos em que um objeto foi dividido em dois objetos, um dos quais é, na verdade, apenas uma pequena parte do objeto maior.
- *Discard objects touching the border of the image?*: Selecionar “Yes” para descartar objetos que tocam a borda da imagem. Selecionar “No” para ignorar este critério.
- *Threshold strategy*: determina o tipo de entrada que é usada para calcular o *threshold*. Os *thresholds* da imagem podem ser baseados em:

→ As intensidades de *pixel* da imagem de entrada (isto é a mais comum).

- Um único valor fornecido manualmente pelo utilizador.
- Um único valor produzido por um módulo de medição anterior.
- Uma imagem binária (chamado de máscara), onde alguns dos valores de intensidade de *pixel* são definidos como 0, e os outros são definidos como 1.

As escolhas para *Threshold strategy* são:

- *Automatic*: Usa as configurações padrão para *thresholding*. Esta estratégia calcula o *threshold* utilizando o método MCT em toda a imagem e aplica o *threshold* à imagem, suavizada com um Gaussiano com sigma igual a 1.
- *Global*: Calcula um valor de *threshold* único com base nos *pixels* da imagem de entrada e usa esse valor para classificar os *pixels* acima do valor de *threshold* como primeiro plano e abaixo como plano de fundo.
- *Adaptive*: Distribui a imagem de entrada em telhas e calcula os valores de *threshold* para cada telha. Para cada telha, o *threshold* calculado é aplicado apenas aos *pixels* dentro dessa telha.
- *Per object*: Usa objetos de um módulo anterior como *IdentifyPrimaryObjects* para definir a região de interesse para ser aplicada o *threshold*. Calcula um *threshold* separado para cada objeto e, em seguida, aplica esse *threshold* aos *pixels* dentro do objeto. Os *pixels* fora dos objetos são classificados como fundo.
- *Manual*: Digita-se um único valor entre 0 e 1 que se aplica a todos os ciclos e é independente da imagem de entrada.
- *Binary image*: Usa uma imagem binária para classificar *pixels* como primeiro ou segundo plano. Valores de *pixels* diferentes de zero serão primeiro plano e os valores de *pixels* iguais a zero serão o fundo.
- *Measurement*: Usa uma medição de uma imagem anterior como *threshold*. A medição deve ter valores entre 0 e 1.

- *Thresholding method*: O *threshold* de intensidade afeta a decisão de se cada *pixel* será considerado com primeiro plano (região de interesse) ou fundo. Um valor de *threshold* elevado irá resultar apenas na detecção das regiões mais brilhantes, ao passo que um valor de *threshold* reduzido irá incluir regiões mais escuras. Pode-se calcular o valor de *threshold* automaticamente a partir de uma escolha de vários métodos, ou pode-se inserir manualmente um número entre 0 e 1. Há um grande número de métodos para encontrar automaticamente *thresholds*:
 - *Otsu*: Esta abordagem calcula o *threshold* que separa as duas classes de *pixels* (de primeiro plano e plano de fundo), minimizando a variância dentro da cada classe.
 - *Mixture of Gaussian (MoG)*: Esta função assume que os *pixels* na imagem pertencem a uma classe de fundo ou a uma classe de primeiro plano, usando uma estimativa inicial da fração da imagem que está coberta pelo primeiro plano.
 - *Background*: Este método encontra simplesmente o modo do histograma da imagem, que se presume ser o fundo da imagem, e escolhe um *threshold* ao dobro desse valor. O cálculo inclui os *pixels* entre 2% e 98% do intervalo de intensidade.
 - *RobustBackground*: este método também é simples e assume que a distribuição do fundo aproxima-se de um Gaussiano aparando 5% das intensidades de *pixel* mais brilhantes e mais escuras. Em seguida, calcula a média e o desvio padrão dos *pixels* restantes e calcula o *threshold* como a média + 2 vezes o desvio padrão.
 - *RidlerCalvard*: Este método é simples e com resultados muito semelhantes ao método de Otsu. *RidlerCalvard* escolhe um *threshold* inicial e, em seguida iterativamente calcula o próximo tomando a média das intensidades médias dos *pixels* de fundo e de primeiro plano determinados pelo primeiro *threshold*. O algoritmo então repete este processo até que o *threshold* convirja para um único valor.

- Kapur: Este método calcula o *threshold* de uma imagem pesquisando o *threshold* que maximiza a soma das entropias dos valores de *pixel* do primeiro e segundo plano, quando tratados como distribuições separadas.
- *Maximum correlation thresholding (MCT)*: Este método calcula a correlação máxima entre a máscara binária criado por *thresholding* e a imagem de *threshold* e é um pouco matematicamente semelhante ao Otsu.
- *Select binary image*: Seleccionar a imagem binária de *threshold* para ser utilizada.
- *Manual threshold*: Digite o valor de 0 a 1 que funcionará como um *threshold* absoluto para as imagens.
- *Select the measurement to threshold with*: Escolha a medição da imagem que atuará como um *threshold* absoluto para as imagens.
- *Two-class or three-class thresholding?*:
 - *Two classes*: Selecionar esta opção se os níveis de tons de cinza são facilmente distinguíveis em apenas duas classes: primeiro plano (ou seja, regiões de interesse) e o fundo.
 - *Three classes*: Escolher esta opção se os níveis de cinza caírem em três classes: primeiro plano, fundo e uma intensidade média entre os dois. Em seguida, será perguntado se a classe de intensidade média deverá ser adicionada ao primeiro plano ou ao fundo, a fim de gerar a saída final de duas classes.
- *Approximate fraction of image covered by objects?*: Introduzir uma estimativa de quanto da imagem é coberta com objetos, que é usada para estimar a distribuição de intensidades de *pixel*.
- *Method to calculate adaptive window size*: O método adaptativo divide a imagem em blocos, calculando o *threshold* para cada bloco. Existem duas formas para calcular o tamanho de bloco:

→ *Image size*: O tamanho do bloco é um décimo das dimensões da imagem, ou 50×50 pixels, o que for maior.

→ *Custom*: O tamanho do bloco é especificado pelo usuário.

- *Size of adaptive window*: Digitar a janela para o método adaptativo. Por exemplo, pode-se usar um múltiplo do maior tamanho do objeto esperado.
- *Threshold correction factor*: Esta configuração permite ajustar o *threshold*, calculado pelo método acima. O valor introduzido aqui ajusta o *threshold* para cima ou para baixo, através da multiplicação por este valor. Um valor igual a 1 indica nenhum ajuste, 0 a 1 faz o *threshold* mais brando e >1 faz com que o *threshold* seja mais rigoroso.
- *Lower and upper bounds on threshold* : Digitar o valor de *threshold* mínimo e máximo permitido, um valor entre 0 e 1. Isto é útil como uma precaução de segurança quando o *threshold* é calculado automaticamente, substituindo o *threshold* automático.
- *Select the smoothing method for thresholding*: A imagem de entrada pode ser opcionalmente suavizada antes de sofrer *threshold*. A suavização pode melhorar a uniformidade dos objetos resultantes, removendo buracos e bordas irregulares causadas por ruído na imagem adquirida. A suavização não é provavelmente apropriada, se a imagem entrada for binária, se já tiver sido suavizada ou se é uma saída do módulo *ClassifyPixels*. As opções são:
 - *Automatic*: Alisa a imagem com um Gaussiano com um desvio padrão de um *pixel* antes do *threshold*. Isto é adequado para a maioria das aplicações de análise.
 - *Manual*: Alisa a imagem com um Gaussiano com escala controlada pelo usuário.
 - *No smoothing*: Não aplica qualquer suavização antes do *threshold*.
- *Threshold smoothing scale*: Esta definição controla a escala utilizada para suavizar a imagem de entrada antes do *threshold* ser aplicado. A escala deve ser

aproximadamente o tamanho dos artefactos a serem eliminados por alisamento. Um Gaussiano é usado com um desvio padrão ajustado de modo a que 1/2 da distribuição Gaussiana caia dentro do diâmetro determinado pela escala (escala $\sigma = / 0,674$)

- *Automatically calculate the size of objects for the Laplacian of Gaussian filter?:* Selecionar “Yes” para usar a faixa de diâmetro de filtragem acima ao construir o filtro de log. Selecionar “No”, a fim de especificar manualmente o tamanho. Pode-se querer especificar um tamanho personalizado, se desejar-se filtrar usando critérios livres, mas tem-se objetos que geralmente são de tamanhos semelhantes.
- *Enter LoG filter diameter:* O tamanho a usar no cálculo do filtro de log. O filtro aumenta os máximos locais de objetos cujos diâmetros são mais ou menos o número digitado ou menores.
- *Method to distinguish clumped objects:* Esta configuração permite que se escolha o método que é usado para segmentar objetos, ou seja, "desacoplar" um objeto grande que incorpora objetos individuais de interesse:
 - *Intensity:* Para objetos que tendem a ter apenas um único pico de brilho (por exemplo, objetos que são mais brilhante para os seus interiores e mais escuros para as bordas), esta opção conta cada pico de intensidade como um objeto separado. Os objetos podem ser de qualquer forma, de modo que não precisam de ser redondos e de tamanho uniforme, como seria necessário para a opção *Shape*.
 - *Shape:* Para os casos em que há recuos definidos que separam objetos. A imagem é convertida a preto e branco (binário) e determina se a forma dos objetos aglutinados será distinguida.
 - *Laplacian of Gaussian:* Para objetos que têm um gradiente de intensidade crescente em direção ao seu centro, esta opção executa uma transformada de Laplace de Gauss (ou chapéu mexicano) na imagem, o que acentua os *pixels* que são máximos locais de um tamanho desejado. Executa um *threshold* e encontra *pixels* que são máximos. Estes *pixels* são usados como as sementes para os objetos no *watershed*.

- *None*: Se os objetos estão bem separados e brilhantes em relação ao fundo, pode não ser necessário tentar separar objetos aglutinados. Usando a opção *None*, será usado um *threshold* simples para identificar objetos. Isto irá substituir qualquer método de desacoplamento escolhido nas configurações abaixo.
- *Method to draw dividing lines between clumped objects*: Esta configuração permite que escolher o método que é usado para desenhar a linha entre objetos segmentados, desde que se tenha optado por um desacoplamento dos objetos:
 - *Intensity*: Funciona melhor onde as linhas divisórias entre os objetos aglutinados são menos brilhantes do que os restantes objetos.
 - *Shape*: as linhas divisórias entre os objetos são aglutinadas com base na forma do tufo. Por exemplo, quando um grupo contém dois objetos, a linha de separação irá ser colocada onde ocorrerem entalhes entre os dois objetos. Os padrões de intensidade na imagem original são irrelevantes: as células não precisam ser mais escuras ao longo das linhas entre objetos agregados.
 - *Propagate*: Este método usa um algoritmo de propagação em vez de um *watershed*. A imagem é ignorada e os *pixels* são atribuídos aos objetos adicionando repetidamente *pixels* não atribuídos para os objetos que se encontram imediatamente adjacentes aos mesmos. Este método é adequado em casos com objetos com extensões de ramificação, por exemplo neurites, onde o objetivo é traçar para fora do corpo celular ao longo do ramo, atribuindo *pixels* no ramo ao longo do caminho.
 - *None*: Se os objetos estão bem separados e brilhantes em relação ao fundo, pode não ser necessário tentar separar objetos aglutinados. Usando a opção *None*, será usado um *threshold* simples para identificar objetos. Isto irá substituir qualquer método de desacoplamento escolhido na pergunta anterior.
- *Automatically calculate size of smoothing filter for declumping?*: Selecionar “Yes” para calcular automaticamente a quantidade de suavização aplicada à imagem para

auxiliar no desacoplamento. Selecionar “No” para inserir manualmente o tamanho do filtro de alisamento.

- *Size of smoothing filter*: Se existirem muitos objetos unidos que devem ser separados (sub-segmentação), este valor deverá ser menor. Se existirem muitos objetos divididos que devem ser unidos, o valor deve ser maior. Digitar 0 para impedir qualquer suavização de imagem em determinados casos; por exemplo, para imagens de baixa resolução com objetos pequenos. Reduzindo a textura dos objetos aumentando o alisamento, aumenta a possibilidade de que cada objeto real distinto tenha apenas um único pico de intensidade, mas também aumenta a probabilidade de dois objetos distintos serem reconhecidos como apenas um objeto.
- *Automatically calculate minimum allowed distance between local maxima?:* Selecionar “Yes” para calcular automaticamente a distância entre intensidade máxima para ajudar no desacoplamento. Selecionar “No” para inserir manualmente a distância máxima permitida.
- *Suppress local maxima that are closer than this minimum allowed distance*: Introduzir um número inteiro positivo, em unidades de *pixel*. Se existirem muitos objetos combinados que devem ser separados (sub-segmentação), o valor deve ser menor. Se existirem muitos objetos divididos que devem ser combinados, o valor deve ser maior. A distância da supressão máxima deve ser definida para ser mais ou menos equivalente ao raio mínimo de um objeto real. Quaisquer "objetos" distintos que se encontram, mas estão separados um do outro pelo dobro desta distância vão ser assumidos como sendo na verdade duas partes irregulares do mesmo objeto, e serão fundidos.
- *Speed up by using lower-resolution image to find local maxima?:* Selecione “Yes” para usar um *down-sample* da imagem para desacoplamento. Isto pode ser útil para poupar tempo de processamento em imagens grandes.
- *Retain outlines of the identified objects?:* Selecionar “Yes” para manter os contornos dos novos objetos para uso posterior.
- *Name the outline image*: Digitar um nome para os contornos dos objetos identificados.

- *Fill holes in identified objects?* Caso esta opção seja assinalada, os buracos interiores dos objetos identificados serão preenchidos.
- *Handling of objects if excessive number of objects identified:* Esta configuração lida com imagens em que são segmentados um número razoável de objetos. Isto pode acontecer se o módulo calcular um *threshold* baixo ou se a imagem tem artefactos incomuns. *IdentifyPrimaryObjects* pode lidar com esta condição de três maneiras:
 - *Continue:* Não ocorre verificação para um grande número de objetos.
 - *Truncate:* Limita o número de objetos. Apaga objetos arbitrariamente para limitar o número para o máximo permitido.
 - *Erase:* Apaga todos os objetos, se o número de objetos excede o máximo. Isto resulta numa imagem sem objetos primários. Esta opção é uma boa escolha, se um grande número de objetos indicar que a imagem não deve ser processada.

Maximum number of objects: Esta configuração limita o número de objetos na imagem.

MEASUREOBJECTSIZESHAPE

Dada uma imagem com objetos identificados (por exemplo, núcleos ou células), este módulo extraí a área e formas de cada um. Essas características são apenas fiáveis para os objetos que estão completamente dentro das fronteiras de imagem, de modo que deve-se excluir objetos que tocam a borda da imagem.

Medições disponíveis:

- *Area:* O número efetivo de *pixels* na região.
- *Perimeter:* O número total de *pixels* em torno da fronteira de cada região da imagem.
- *FormFactor:* calculado como $4 * \pi * \text{Área} / \text{Perímetro}^2$. É igual a 1 para um objeto perfeitamente circular.
- *Solidity:* A proporção dos *pixels* no casco convexo que também estão no objeto, ou seja, $\text{ObjectArea} / \text{ConvexHullArea}$. É igual a 1 para um objeto sólido (ou seja, sem

buracos ou tem um limite côncavo), ou <1 para um objeto com buracos ou que possuía um limite convexo / irregular.

- *Extent*: A proporção dos *pixels* na caixa delimitadora que também se encontra na região. Calculada como a área dividida pela área da caixa delimitadora.
- *EulerNumber*: O número de objetos na região menos o número de buracos nesses objetos, assumindo conectividade-8.
- *Center_X*, *Center_Y*: As coordenadas x e y do ponto mais distante de qualquer aresta do objeto.
- *Eccentricity*: A excentricidade da elipse que tem os mesmos *second-moments* como a região. A excentricidade é a razão entre a distância entre os focos da elipse e o seu comprimento do eixo principal (ver Figura 44). O valor está entre 0 e 1. (0 e 1 são casos degenerados. Uma elipse cujo excentricidade é 0 é, na verdade, um círculo, uma elipse, enquanto cuja excentricidade é um é um segmento de reta).

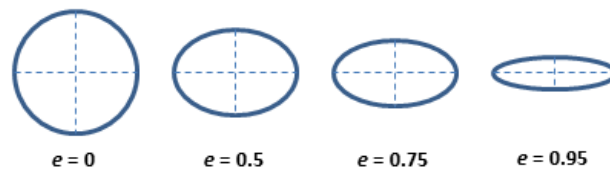


Figura 44 Excentricidade de uma elipse [12].

- *MajorAxisLength*: O comprimento (em *pixels*) do maior eixo da elipse que tem os *second-moments* centrais normalizados, tal como a região.
- *MinorAxisLength*: O comprimento (em *pixels*) do menor eixo da elipse que tem os *second-moments* centrais normalizados, tal como a região.
- *Orientation*: o ângulo (em graus que variam de -90 a 90 graus) entre o eixo X e o maior eixo da elipse que tem os mesmos *second-moments* como a região.
- *Compactness*: A variância da distância radial de *pixels* do objeto a partir do centroide dividido pela área.
- *MaximumRadius*: A distância máxima de qualquer *pixel* no objeto ao *pixel* mais próximo fora do objeto. Para objetos finos, isto é $1/2$ da largura máxima do objeto.

- *MedianRadius*: A distância média de qualquer *pixel* no objeto ao *pixel* mais próximo do lado de fora do elemento de imagem do objeto.
- *MeanRadius*: A distância média de qualquer *pixel* no objeto ao *pixel* mais próximo fora do objeto.
- *MinFerretDiameter*, *MaxFerretDiameter*: O diâmetro de Feret é a distância entre duas linhas paralelas tangentes em ambos os lados do objeto. Os diâmetros mínimos e máximos de Feret são os menores e maiores diâmetros possíveis.
- *Zernike shape features*: Mede a forma ao descrever um objeto binário (ou mais precisamente, um *patch* com fundo e um objeto no centro), numa base de polinómios de Zernike, utilizando os coeficientes como recursos. Atualmente, os polinómios de Zernike da ordem de 0 a 9 são calculados, dando um total de 30 medições.

ConFigurações:

Select objects to measure: Selecionar os objetos a medir.

Calculate the Zernike features?: Selecione “Yes” para calcular os recursos de forma Zernike. Uma vez que os primeiros 10 (polinomiais de Zernike de ordem 0 à ordem 9) são calculados, esta operação pode ser demorada quando a imagem contém uma grande quantidade de objetos.

MEASUREOBJECTINTENSITY

Mede várias características de intensidade para objetos identificados.

Dada uma imagem com objetos identificados (por exemplo: núcleos ou células), este módulo extrai as características de intensidade para cada objeto com base numa ou mais imagens em escala de cinzentos correspondentes. As medições são registadas para cada objeto.

As medições de intensidade são feitas para todas as combinações de imagens e objetos inseridos.

O comportamento padrão no CellProfiler é o de redimensionar a intensidade da imagem de 0 a 1 dividindo todos os *pixels* da imagem pelo valor máximo possível de intensidade.

Medições disponíveis:

- *IntegratedIntensity*: A soma das intensidades dos *pixels* dentro de um objeto.
- *MeanIntensity*: a intensidade média dos *pixels* dentro de um objeto.
- *StdIntensity*: O desvio padrão das intensidades dos *pixels* dentro de um objeto.
- *MaxIntensity*: A intensidade máxima dos *pixels* dentro de um objeto.
- *MinIntensity*: A intensidade mínima dos *pixels* dentro de um objeto.
- *IntegratedIntensityEdge*: A soma das intensidades dos *pixels* da borda de um objeto.
- *MeanIntensityEdge*: A intensidade média dos *pixels* da borda de um objeto.
- *StdIntensityEdge*: O desvio-padrão das intensidades dos *pixels* da borda de um objeto.
- *MaxIntensityEdge*: A intensidade máxima dos *pixels* da borda de um objeto.
- *MinIntensityEdge*: A intensidade mínima dos *pixels* da borda de um objeto.
- *MassDisplacement*: A distância entre os centros de gravidade na representação de nível de cinzento do objeto e a representação binária do objeto.
- *LowerQuartileIntensity*: O valor da intensidade dos *pixels* para a qual 25% dos *pixels* do objeto têm valores mais baixos.
- *MedianIntensity*: O valor de intensidade mediana dentro do objeto.
- *MADIntensity*: O valor mediano do desvio absoluto (Median Absolute Deviation) das intensidades dentro do objeto. O MAD é definido como a mediana*($|x_i - \text{mediana}(x)|$).
- *UpperQuartileIntensity*: O valor da intensidade do *pixel* para a qual 75% dos *pixels* do objeto têm valores mais baixos.
- *Location_CenterMassIntensity_X*, *Location_CenterMassIntensity_Y*: As coordenadas (X, Y) do *pixel* da intensidade do centróide ponderado (= centro de massa = primeiro momento) de todos os *pixels* dentro do objeto.

- *Location_MaxIntensity_X, Location_MaxIntensity_Y*: As coordenadas (X, Y) do *pixel* com a intensidade máxima dentro do objeto.

ConFigurações:

- *Select an image to measure*: Selecionar as imagens em tons de cinza, cuja intensidade deseja-se medir.
- *Select objects to measure*: Selecionar os objetos cujas intensidades deseja-se medir.

FILTEROBJECTS

Este módulo remove objetos com base em medições produzidas noutro módulo selecionado. Todos os objetos que não satisfaçam os parâmetros especificados serão descartados. Este módulo também pode remover objetos tocando a borda da imagem ou bordas de uma máscara. Os objetos que passam a etapa de filtragem compreendem um novo conjunto de objetos, e, portanto, não herdam as medidas associadas com os objetos originais

Medições disponíveis:

- *Count*: O número de objetos remanescentes após a filtragem.
- *Parent*: A identidade do objeto de entrada associado a cada objeto filtrado.
- *Location_X, Location_Y*: As coordenadas (X, Y) do centro de massa dos *pixels* dos restantes objetos.

ConFigurações:

- *Name the output objects*: Digitar um nome para a coleção de objetos que são mantidos depois de aplicar o filtro(s).
- *Select the object to filter*: Selecionar o conjunto de objetos a filtrar. Esta configuração também controla quais as opções de medição aparecerem para filtragem: apenas pode filtrar com base em medições feitas no objeto selecionado.
- *Select the filtering mode*: Escolher entre as seguintes opções:

- *Measurements*: Especificar uma medição por objeto feita por um módulo a montante na *pipeline*.
- *Rules*: Usar um arquivo contendo regras geradas pelo “CellProfiler Analyst”. Importante assegurar que as medidas especificadas pelo arquivo das regras são produzidas por módulos a montante na *pipeline*.
- *Image or mask border*: Remover objetos que tocam a borda da imagem e/ou as bordas de uma máscara de imagem.
- *Select the filtering method*: Existem cinco maneiras diferentes para filtrar objetos:
 - *Limits*: Mantem um objeto se o seu valor de medição está dentro de um intervalo especificado.
 - *Maximal*: Mantem o objeto com o valor máximo para a medição de interesse. Se vários objetos compartilharem um valor máximo, mantem um objeto selecionado arbitrariamente.
 - *Minimal*: Mantem o objeto com o valor mínimo para a medição de interesse. Se vários objetos compartilham um valor mínimo, mantem um objeto selecionado arbitrariamente.
 - *Maximal per object*: Esta opção requer a escolha de um objeto pai. O objeto pai pode conter vários objetos filho. Somente os objeto filhos cujas medidas igualarem o valor máximo entre esse conjunto de objetos filho serão mantidos.
 - *Minimal per object*: O mesmo que *Maximal per object*, exceto a filtragem é baseada no valor mínimo.
- *Rules file location*: Selecionar a localização do ficheiro de regras que vai ser usado para a filtragem. Pode-se escolher entre as seguintes opções que são comuns a todos os módulos de entrada / saída de arquivo:
 - *Default Input Folder*: Usa a pasta de entrada padrão.
 - *Default Output Folder*: Usa a pasta de saída padrão.

- *Elsewhere...*: Usa uma pasta especificada pelo utilizador.
 - *Default input directory sub-folder*: Digitar o nome de uma subpasta da pasta de entrada padrão ou um caminho que começa a partir da pasta de entrada padrão.
 - *Default output directory sub-folder*: Digitar o nome de uma subpasta da pasta de saída padrão ou um caminho que começa a partir da pasta de saída padrão.
- *Rules file name*: O nome do arquivo de regras. Este arquivo deve ser um arquivo de texto simples contendo o conjunto completo de regras. Cada linha desse arquivo deve ser uma regra de nomeação de uma medição a ser feita sobre o objeto selecionado, por exemplo: IF (Nuclei_AreaShape_Area < 351.3, [0.79, -0.79], [-0.94, 0.94]) . Esta regra vai marcar 0,79 para a categoria positiva e -0,94 para a categoria negativo para núcleos cuja área é inferior a 351,3 *pixels* e vai marcar o oposto para núcleos cuja área é maior. O filtro adiciona positivo e negativo e mantém apenas os objetos cuja pontuação positiva é maior do que a pontuação negativa.
 - *Class number*: Selecionar qual das classes para manter durante a filtragem. A interface do usuário do classificador do “CellProfiler Analyst” lista os nomes das classes, da esquerda para a direita. *FilterObjects* utiliza a primeira classe do “CellProfiler Analyst” se “1” for escolhido, etc.
 - *Retain outlines of the identified objects?*: Selecionar “Yes” para manter os contornos dos novos objetos para uso posterior no *pipeline*.
 - *Name the outline image*: Digitar um nome para os contornos dos objetos identificados. A imagem delineada pode ser selecionada em módulos a jusante, selecionando-os a partir de qualquer lista de imagem.

EDITOBJECTSMANUALLY

EditObjectsManually permite ao utilizador criar, remover e editar manualmente objetos previamente definidos.

A interface irá mostrar a imagem que selecionada como imagem orientadora, coberta com os contornos coloridos dos objetos selecionados. Este módulo permite ao utilizador remover ou editar objetos específicos apontando e clicando para selecionar objetos para a remoção ou edição. Uma vez que a edição esteja completa, o módulo exibe os objetos como originalmente identificados (à esquerda) e os objetos que permanecem após este módulo (à direita). É fornecida ajuda mais detalhada na janela de edição através do botão '?'.
O *pipeline* faz uma pausa, uma vez por imagem processada quando se chega a este módulo. Deve pressionar o botão “Done” para aceitar os objetos selecionados e continuar o *pipeline*.

O *pipeline* faz uma pausa, uma vez por imagem processada quando se chega a este módulo. Deve pressionar o botão “Done” para aceitar os objetos selecionados e continuar o *pipeline*.

Medições disponíveis:

- *Count*: O número de objetos editados na imagem.
- *Location_X, Location_Y*: As coordenadas (X, Y) do centro de massa dos *pixels* dos objetos editados.

ConFigurações:

- *Select the objects to be edited*: Escolher um conjunto de objetos previamente identificados para a edição.
- *Name the edited objects*: Digitar o nome para os objetos que permanecem após a edição. Estes objetos estarão disponíveis para uso por módulos subsequentes.
- *Retain outlines of the edited objects?*: Selecionar “Yes” para manter as imagens dos contornos dos objetos que permanecem após a edição.
- *Name the outline image*: Digitar um nome para a imagem esboço.
- *Numbering of the edited objects*: Escolher como numerar os objetos que permanecem após a edição, que controla como os objetos editados são associados com os seus antecessores:
 - *Renumber*: O módulo irá numerar os objetos que permanecem utilizando números consecutivos.

→ *Retain*: Esta opção irá reter o número original de cada objeto para que o número do objeto editado corresponda ao seu número original. Isso permite que quaisquer medições feitas a partir dos objetos a serem editados fiquem diretamente alinhadas com as medições previamente feitas dos objetos originais.

- *Display a guiding image?*: Selecionar “Yes” para exibir uma imagem e contornos dos objetos. Selecionar “No” se o utilizador não quiser uma imagem de guia durante a edição.
- *Select the guiding image*: Esta é a imagem que aparecerá na edição de objetos. Escolher uma imagem fornecida por um módulo anterior.
- *Allow overlapping objects?*: *EditObjectsManually* pode permitir ao utilizador editar um objeto para que ele sobreponha-se a outro ou pode impedir a sobreposição. Objetos como vermes ou as neurites dos neurónios podem sofrer cruzamentos entre si e podem precisar de ser editados com a sobreposição permitida, enquanto uma monocamada de células poderia ser editada melhor com a sobreposição impedida. Selecionar “Yes” para permitir sobreposições ou selecionar “No” para impedi-las.

IDENTIFYSECONDARYOBJECTS

IdentifySecondaryObjects identifica objetos (por exemplo, bordas de células) usando objetos identificados por outro módulo (por exemplo, núcleos) como ponto de partida.

No CellProfiler, usa-se o termo objeto como um termo genérico para se referir a uma característica identificada numa imagem, geralmente um sob compartimento celular de algum tipo (por exemplo, núcleos, células, colónias, vermes). Define-se um objeto como secundário quando este pode ser encontrado numa imagem usando uma outra característica celular como referência para guiar a deteção.

Para as células densamente-embaradas, determinando os limites das células usando uma mancha de um corpo celular pode ser bastante difícil, uma vez que muitas vezes têm padrões irregulares de intensidade e são de contraste baixo com uma coloração mais difusa. Além disso, as células muitas vezes tocam nos seus vizinhos, tornando mais difícil delinear as bordas das células. Muitas vezes, é mais fácil identificar um organelo que está separado

espacialmente (tal como o núcleo) como um objeto primário e, em seguida, usar esse objeto para orientar a detecção das bordas da célula.

A fim de identificar os limites dos objetos secundários, este módulo executa duas funções:

- Encontra as linhas divisórias entre objetos secundários que se tocam.
- Encontra as linhas divisórias entre os objetos secundários e o fundo da imagem.

Medições disponíveis:

- *Count*: O número de objetos secundários identificados.
- *OriginalThreshold*: O *threshold* global para a imagem.
- *FinalThreshold*: Para os métodos de *threshold* globais, este valor é o mesmo que em *OriginalThreshold*. Para a adaptação ou em métodos por objeto, este valor é a média dos *thresholds* locais.
- *WeightedVariance*: A soma dos desvios das transformadas logarítmicas dos *pixels* do primeiro plano e do fundo, ponderada pelo número de *pixels* em cada distribuição.
- *SumOfEntropies*: A soma das entropias computadas a partir das distribuições de primeiro e segundo plano.
- *Parent*: A identidade do objecto primário associado com cada objeto secundário.
- *Location_X*, *Location_Y*: As coordenadas (X, Y) do centro de massa dos *pixels* identificados como objetos secundários.

ConFigurações:

- *Select the input objects*: Selecionar os objetos para usar como "sementes" para identificar um objeto secundário em torno de cada um. Por definição, cada objeto primário deve ser associado a exatamente um objeto secundário e completamente contido dentro dele.
- *Name the objects to be identified*: Digitar o nome a chamar aos objetos identificados por este módulo.

- *Select the method to identify the secondary objects*: Existem vários métodos disponíveis para encontrar as linhas divisórias entre os objetos secundários:

→ *Propagation*: Este método irá encontrar as linhas divisórias entre objetos aglutinados em que a imagem, coberta por objetos secundários, mostra uma mudança na coloração. As linhas divisórias entre os objetos são determinadas por uma combinação da distância até o objeto e gradientes de intensidade primários mais próximos. Este algoritmo utiliza a similaridade da imagem local para orientar a localização das fronteiras entre as células. Os limites são preferencialmente colocados onde a aparência local da imagem muda perpendicularmente ao limite.

→ *Watershed - Gradient*: Este método usa um algoritmo de *watershed* para atribuir *pixels* aos objetos primários que agem como sementes no *watershed*. Nesta variante, o algoritmo opera na imagem após uma transformada de Sobel, que calcula um gradiente de intensidade. Este método funciona melhor quando a intensidade da imagem cai ou aumenta rapidamente perto da fronteira entre as células.

→ *Watershed - Image*: Este método é semelhante ao anterior, mas utiliza a intensidade da imagem invertida para o *watershed*. As áreas de menor intensidade formam os limites entre as células. Este método funciona melhor quando existe uma sela de intensidade relativamente baixa no limite célula-célula.

→ *Distance*: Neste método, as arestas dos objetos primários são expandidas para uma distância específica para criar os objetos secundários. Por exemplo, se os núcleos são marcados, mas não há nenhuma mancha para ajudar a localizar as bordas das células, os núcleos podem simplesmente ser expandidos, a fim de calcular a localização da célula. Existem dois métodos que podem ser utilizados:

- ✓ *Distance - N*: Neste método, a imagem da mancha secundária não é utilizada; os objetos expandidos são os objetos secundários finais.

- ✓ *Distance - B*: É usado um *threshold* da imagem secundária para eliminar as regiões de fundo dos objetos secundários. Isto permite que a dimensão dos objetos secundários seja limitada a uma certa distância da borda dos objetos primários sem incluir as regiões do fundo.
- *Select the input image*: A imagem selecionada é utilizada para encontrar as arestas dos objetos secundários.
- *Threshold strategy*: determina o tipo de entrada que é usada para calcular o *threshold*. Os *thresholds* da imagem podem ser baseados em:
 - As intensidades de *pixel* da imagem de entrada (isto é a mais comum).
 - Um único valor fornecido manualmente pelo utilizador.
 - Um único valor produzido por um módulo de medição anterior.
 - Uma imagem binária (chamado de máscara), onde alguns dos valores de intensidade de *pixel* são definidos como 0, e os outros são definidos como 1.

As escolhas para *Threshold strategy* são:

- *Automatic*: Usa as configurações padrão para *thresholding*. Esta estratégia calcula o *threshold* utilizando o método MCT em toda a imagem e aplica o *threshold* à imagem, suavizada com um Gaussiano com sigma igual a 1.
- *Global*: Calcula um valor de *threshold* único com base nos *pixels* da imagem de entrada e usa esse valor para classificar os *pixels* acima do valor de *threshold* como primeiro plano e abaixo como plano de fundo.
- *Adaptive*: Distribui a imagem de entrada em telhas e calcula os valores de *threshold* para cada telha. Para cada telha, o *threshold* calculado é aplicado apenas aos *pixels* dentro dessa telha.
- *Per object*: Usa objetos de um módulo anterior como *IdentifyPrimaryObjects* para definir a região de interesse para ser aplicada o *threshold*. Calcula um *threshold* separado para cada objeto e, em seguida, aplica esse *threshold* aos

pixels dentro do objeto. Os *pixels* fora dos objetos são classificados como fundo.

→ *Manual*: Digita-se um único valor entre 0 e 1 que se aplica a todos os ciclos e é independente da imagem de entrada.

→ *Binary image*: Usa uma imagem binária para classificar *pixels* como primeiro ou segundo plano. Valores de *pixels* diferentes de zero serão primeiro plano e os valores de *pixels* iguais a zero serão o fundo.

→ *Measurement*: Usa uma medição de uma imagem anterior como *threshold*. A medição deve ter valores entre 0 e 1.

- *Regularization factor*: O fator de regularização λ pode assumir qualquer valor no intervalo de 0 até infinito. Esse método tem em conta dois fatores na decisão de onde traçar a linha divisória entre dois objetos secundários tocantes: a distância ao objeto primário mais próximo e a intensidade da imagem do objeto secundário. O fator de regularização controla o equilíbrio entre estas duas considerações. Um valor de λ igual a 0 significa que a distância ao objeto primário mais próximo é ignorada e a decisão é feita inteiramente no gradiente de intensidade entre os dois objetos primários concorrentes. Valores elevados de λ colocam mais e mais peso sobre a distância entre os dois objetos. Esta relação é tal que as pequenas mudanças no λ terão resultados bastante diferentes. No infinito, o resultado será semelhante a *Distance – B*.
- *Fill holes in identified objects?*: Selecionar “Yes” para preencher todos os “buracos” dentro de objetos.
- *Discard secondary objects touching the border of the image?*: Selecionar “Yes” para descartar objetos secundários que tocam a borda da imagem. Selecionar “No” para manter os objetos.
- *Discard the associated primary objects?*: Pode ser adequado descartar o objeto primário para qualquer objeto secundário que toca a borda da imagem. Selecionar “Yes” para criar um novo conjunto de objetos que são idênticos aos objetos primários

originais estabelecidos, menos os objetos para os quais o objeto secundário associado toca a borda da imagem.

- *Name the new primary objects*: Nomeia os objetos primários que permanecem após a etapa de descarte.
- *Retain outlines of the new primary objects?*: Selecionar “Yes” para manter os contornos dos novos objetos para uso posterior no *pipeline*.
- *Name the new primary object outlines*: Digitar um nome para os contornos dos objetos identificados.
- *Retain outlines of the identified secondary objects?*: Selecionar “Yes” para manter os contornos dos novos objetos para uso posterior no *pipeline*.
- *Name the outline image*: Digitar um nome para os contornos dos objetos identificados.

Anexo B. Tabela das Características dos Objetos Detetados pelo CellProfiler

Imagem	Objeto	Área (pixels)	Excentricidade	Intensidade Integrada (pixels)
1	1	2669	0,95	2093,29
1	2	2195	0,82	1737,15
1	3	328	0,66	197,64
1	4	478	0,61	267,42
2	1	2794	0,88	2412,32
2	2	1758	0,92	1194,16
2	3	2108	0,82	1775,11
2	4	1859	0,84	576,04
3	1	2091	0,91	1305,97
3	2	717	0,85	357,30
3	3	1463	0,81	989,42
3	4	1050	0,76	600,49
4	1	1371	0,91	767,18
4	2	339	0,68	158,97
4	3	2009	0,91	1754,08
4	4	2138	0,92	1890,06
4	5	2361	0,88	939,39
5	1	1715	0,87	1321,35
5	2	1051	0,83	450,05
5	3	957	0,75	600,49
5	4	1585	0,91	1326,77
6	1	974	0,79	508,91
6	2	1088	0,75	738,26
6	3	2001	0,82	1594,28
6	4	1489	0,79	1163,46
7	1	2013	0,88	1558,40
7	2	1474	0,85	1005,72
7	3	2191	0,88	1896,99
7	4	1413	0,89	668,94
7	5	2221	0,86	847,32
8	1	1629	0,85	864,04
8	2	2387	0,92	1967,22
8	3	622	0,76	294,17
8	4	1479	0,79	895,69
8	5	1852	0,82	1259,23
9	1	1040	0,61	570,70
9	2	835	0,83	368,12
9	3	2268	0,83	1876,81
9	4	1408	0,78	710,37
9	5	1661	0,74	1335,71
10	1	598	0,69	214,00
10	2	1583	0,61	743,44
10	3	576	0,82	188,41
10	4	1626	0,77	659,53
10	5	941	0,83	234,75
11	1	824	0,78	455,10
11	2	393	0,80	205,15
11	3	1351	0,84	529,60
11	4	2217	0,79	1747,50
12	1	1219	0,73	830,79
12	2	333	0,52	152,17
12	3	1761	0,83	1328,59
12	4	1933	0,69	1820,52
12	5	1351	0,37	965,66
13	1	2169	0,83	1481,97
13	2	1715	0,89	1117,98
13	3	2057	0,78	1843,06
14	1	977	0,79	683,80
14	2	1515	0,79	1216,35
14	3	560	0,54	509,97
14	4	1612	0,70	1292,37
15	1	1643	0,73	975,63

15	2	1252	0,76	703,10
15	3	1238	0,91	602,70
16	1	1600	0,83	1221,58
16	2	1654	0,81	1537,55
16	3	1385	0,79	1136,95
16	4	964	0,84	720,23
16	5	1179	0,61	996,29
17	1	2014	0,91	1468,78
17	2	1785	0,86	1234,79
17	3	2128	0,84	1921,99
17	4	1240	0,78	741,34
18	1	379	0,71	219,76
18	2	1993	0,82	1483,61
18	3	793	0,81	555,76
18	4	1786	0,71	1631,19
18	5	2079	0,76	1548,96
19	1	2121	0,98	1706,71
19	2	1087	0,74	434,82
19	3	3409	0,93	1609,96
20	1	1668	0,83	778,62
20	2	1320	0,80	614,81
20	3	1624	0,81	1040,54
20	4	838	0,83	369,80
20	5	1240	0,67	609,35
21	1	435	0,69	251,73
21	2	1520	0,85	1283,01
21	3	1562	0,87	1317,91
22	1	1428	0,72	754,67
22	2	1234	0,73	568,01
22	3	1819	0,42	1499,93
23	1	1513	0,72	865,61
23	2	1814	0,84	1298,72
23	3	2055	0,85	1715,08
24	1	1458	0,92	57,81
24	2	1348	0,84	44,39
24	3	486	0,55	11,92
24	4	1178	0,64	53,06
24	5	666	0,85	20,88
25	1	1130	0,92	424,66
25	2	1315	0,86	354,70
25	3	856	0,88	217,93
25	4	1000	0,71	394,48
26	1	797	0,88	55,69
26	2	1495	0,83	200,15
26	3	1033	0,88	84,64
27	1	1431	0,82	108,88
27	2	1274	0,85	93,22
27	3	1073	0,92	70,64
28	1	849	0,80	85,26
28	2	1054	0,77	125,10
28	3	1049	0,91	89,06
28	4	1249	0,88	108,66
29	1	780	0,90	45,53
29	2	670	0,84	37,54
29	3	1538	0,59	179,19
29	4	2062	0,90	110,12
30	1	1300	0,79	92,34
30	2	893	0,71	48,01
30	3	1158	0,60	88,05
31	1	775	0,90	57,32
31	2	2009	0,92	183,53
31	3	1188	0,91	81,03
31	4	1554	0,92	108,37
32	1	1150	0,85	67,49
32	2	658	0,81	31,93
32	3	861	0,89	46,97
32	4	1514	0,48	121,59
33	1	1074	0,82	167,92
33	2	1106	0,92	133,58
33	3	1398	0,60	286,42
34	1	1959	0,75	1563,83
34	2	1135	0,40	626,01

35	1	1594	0,83	216,49
35	2	1141	0,92	116,31
35	3	1537	0,82	190,62
36	1	784	0,56	57,90
36	2	826	0,88	53,48
36	3	1223	0,78	114,39
36	4	1110	0,89	81,99
36	5	1284	0,80	113,83
37	1	1105	0,82	109,55
37	2	972	0,93	72,95
37	3	1092	0,82	143,51
37	4	808	0,88	88,98
37	5	1237	0,74	57,65
38	1	565	0,91	73,93
38	2	1359	0,88	171,18
38	3	1242	0,88	146,29
38	4	841	0,96	102,74
39	1	787	0,77	64,02
39	2	1368	0,86	173,05
39	3	1338	0,85	164,21
39	4	1325	0,81	184,98
40	1	675	0,87	89,54
40	2	972	0,83	138,76
40	3	445	0,91	48,32
40	4	1411	0,89	219,86
40	5	1061	0,87	119,96
41	1	995	0,89	124,80
41	2	564	0,90	54,77
41	3	1075	0,82	177,44
41	4	977	0,79	143,27
41	5	854	0,90	107,36
41	6	1262	0,85	168,09
41	7	599	0,87	66,49
41	8	1087	0,79	245,19
41	9	847	0,91	158,77
41	10	1362	0,90	193,09
42	1	1824	0,92	257,71
42	2	1385	0,92	166,62
42	3	1729	0,85	79,26
42	4	1277	0,97	144,28
42	5	936	0,96	71,68
43	1	386	0,75	45,45
43	2	1079	0,88	171,57
43	3	1428	0,69	305,28
43	4	1114	0,90	165,98
43	5	1795	0,78	69,92
44	1	697	0,89	49,85
44	2	973	0,85	74,32
44	3	1369	0,89	115,17
44	4	447	0,92	26,17
44	5	1211	0,87	82,36
44	6	1318	0,87	39,98
45	1	1607	0,43	1244,97
46	1	1291	0,84	182,08
46	2	1090	0,87	144,27
46	3	686	0,83	72,31
46	4	1230	0,85	201,04
46	5	949	0,87	153,57
46	6	1661	0,73	90,77
47	1	1254	0,62	215,10
47	2	951	0,85	151,81
47	3	684	0,81	101,38
47	4	1113	0,86	168,03
48	1	1292	0,81	192,59
48	2	1347	0,81	206,91
48	3	1008	0,85	133,43
49	1	910	0,91	118,48
49	2	686	0,90	83,68
49	3	663	0,74	76,01
49	4	509	0,73	57,49
49	5	1687	0,95	243,24
50	1	466	0,75	39,75

50	2	1143	0,89	129,95
50	3	1564	0,77	237,33
50	4	1206	0,90	144,74
51	1	1324	0,69	104,81
51	2	1167	0,84	74,82
51	3	1283	0,69	107,09
52	1	883	0,81	139,69
52	2	1247	0,81	190,63
52	3	1379	0,84	69,62
52	4	1057	0,77	78,02
52	5	1630	0,81	103,50
53	1	1021	0,85	72,77
53	2	1384	0,84	126,02
53	3	1319	0,72	108,11
53	4	1050	0,60	80,90
54	1	1700	0,87	245,69
54	2	1429	0,58	180,38
54	3	1226	0,76	165,95
54	4	868	0,73	56,58
55	1	1723	0,70	142,27
55	2	1393	0,81	133,46
55	3	611	0,65	39,90
55	4	1327	0,72	107,61
56	1	1345	0,76	839,64
56	2	500	0,90	321,25
56	3	2030	0,87	1783,88
56	4	1419	0,94	1188,67
57	1	1630	0,80	220,07
57	2	1540	0,81	163,46
57	3	1032	0,65	80,36
57	4	1473	0,82	137,88
57	5	738	0,73	63,51
57	6	1219	0,55	122,60
57	7	605	0,81	49,02
57	8	869	0,86	65,42
57	9	1611	0,83	78,94
58	1	1946	0,88	247,57
58	2	1270	0,83	103,45
58	3	1306	0,89	104,60
58	4	370	0,80	21,11
58	5	1508	0,89	114,78
59	1	957	0,85	88,57
59	2	913	0,86	90,98
59	3	1769	0,89	210,61
59	4	1030	0,89	84,19
60	1	353	0,83	33,15
60	2	1279	0,88	182,06
60	3	1143	0,84	154,82
60	4	1453	0,61	71,39
60	5	1459	0,62	88,39
61	1	1279	0,75	102,37
61	2	1385	0,81	123,73
61	3	1694	0,83	167,89
61	4	1367	0,79	56,93
62	1	823	0,80	57,43
62	2	1133	0,68	113,36
62	3	951	0,51	88,49
62	4	947	0,76	84,88
62	5	1560	0,86	165,57
63	1	760	0,65	54,68
63	2	1229	0,68	154,50
63	3	590	0,74	51,23
63	4	960	0,55	90,91
63	5	807	0,81	65,32
64	1	1461	0,80	97,09
64	2	438	0,76	17,36
64	3	1331	0,83	87,82
64	4	829	0,86	47,32
64	5	2257	0,74	342,52
65	1	1264	0,77	105,68
65	2	1858	0,75	306,34
65	3	1417	0,74	131,21

65	4	1107	0,59	98,09
66	1	1047	0,73	112,51
66	2	581	0,80	53,35
66	3	1236	0,65	190,53
66	4	1766	0,92	221,85
67	1	1721	0,56	1612,64
67	2	847	0,72	753,85
67	3	785	0,90	237,72
68	1	1313	0,85	188,15
68	2	1175	0,78	205,70
68	3	519	0,72	52,76
68	4	1866	0,72	136,55
68	5	1251	0,18	85,13
69	1	330	0,72	29,82
69	2	1457	0,88	230,07
69	3	331	0,74	32,24
69	4	1210	0,82	186,39
69	5	1233	0,84	90,22
70	1	645	0,85	60,02
70	2	1049	0,84	149,84
70	3	659	0,90	77,88
70	4	1513	0,88	283,04
71	1	1565	0,83	1312,48
71	2	1879	0,89	1583,88
72	1	1604	0,85	1114,28
72	2	1227	0,76	957,85
72	3	444	0,80	301,82
72	4	2196	0,89	2034,21
72	5	1462	0,80	1210,64

