

Métodos de análise de imagem para contagem celular automatizada

Ana Maria Sousa^{1,2}, Bernardo Pereira^{1,3}, Jacinta Ferreira^{1,4}

¹ Faculdade de Engenharia da Universidade do Porto

²up201707312, ³up201909521, ⁴up201705451

U.C.: Análise de Imagem Biomédica (EBE0056)

Docentes: Ana Maria Mendonça, Guilherme Moreira Aresta e Teresa Manuel Sá Finisterra Araújo

Resumo — A contagem de células é indispensável em experiências de cultura celular, porque permite obter dados relativos à densidade celular, sendo esta uma informação de grande relevância no estudo de condições de cultura. Em algumas universidades e outros institutos de ensino e investigação, esta contagem ainda é feita recorrendo a métodos como a contagem manual, cuja precisão é baixa, o que pode resultar em conclusões erradas ou indevidamente inferidas. Por isso, diversas ferramentas para a contagem celular têm vindo a ser desenvolvidas e/ou investigadas.

Métodos de análise de imagem para contagem celular automatizada propõem ser uma alternativa eficiente na contagem, tanto pela questão de precisão de resultados como na organização/gestão do tempo. Existem atualmente no mercado alguns sistemas de contagem celular automatizado. No entanto, o seu elevado preço não permite uma maior utilização desta ferramenta. Neste trabalho foram utilizados os conhecimentos adquiridos na unidade curricular de Análise de Imagem Biomédica (AIBI) para detetar células cancerígenas- HL60 - em diferentes imagens representativas de variadas culturas.

I. INTRODUÇÃO

A contagem de células é uma parte fulcral de qualquer estudo que envolva experiências de cultura celular, pelo que a sua contagem deve ser o mais exata possível. Tendo isto em conta, foi proposto pelos docentes recorrer aos conhecimentos adquiridos em AIBI para criar um algoritmo, usando o software MATLAB, que permita detetar as células cancerígenas- HL60 -sendo este uma possível alternativa económica em relação aos sistemas existentes no mercado para a contagem celular e uma alternativa mais eficaz e exata em relação à contagem manual. [1]. As células HL60 (*Human promyelocytic leukemia cell line*) fazem parte de uma linhagem celular da leucemia promielocítica humana.[2] Numa fase inicial do projeto é feito um delineamento da região de interesse da imagem- região retangular da imagem onde são encontradas células- e numa segunda fase efetua-se a segmentação e contagem das células dentro da região de interesse (ROI).

II. METODOLOGIA

A. Delineamento da Região de Interesse da Imagem

O algoritmo para delineamento da região de interesse (ROI) desenvolvido tem como *input* a imagem original e o *output* uma imagem binária, com a área da ROI representada como um retângulo branco num fundo preto. Posteriormente, foi avaliado o desempenho do algoritmo comparando com o *ground truth* (GT), usando (a) o índice de Jaccard e (b) os valores médio e

máximo da distância Euclideana entre os vértices detetados e GT ROIs.

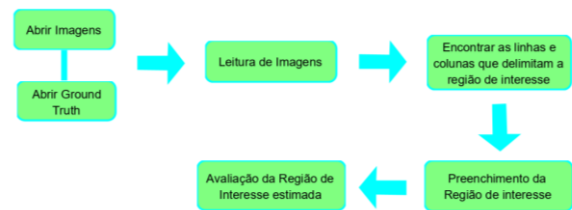


Figura.1. Diagrama representativo dos passos essenciais no delineamento da região de interesse da imagem.

Para determinar a região de interesse de cada uma das imagens presentes na DATASET, começamos por converter a imagem RGB para tons cinzentos, de forma a podermos posteriormente converter essa imagem para imagem binária.

Esta conversão deve ser gradual (RGB-Tons cinzentos-imagem binária) (fig.1) uma vez que não é possível converter diretamente uma imagem RGB para uma imagem binária. A conversão para uma imagem binária é possível recorrendo ao método de *Otsu*, que permite obter um valor de *threshold* automático da imagem. Uma binarização corresponde à conversão dos valores de intensidade abaixo do *threshold* para 0 (preto) e os dos demais para 1 (branco). O nível do *threshold* varia para cada imagem, uma vez que cada imagem tem diferentes intensidades e contrastes.

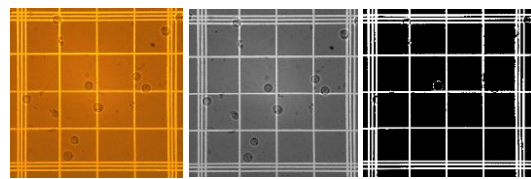


Figura.1. Imagens original, em tons de cinzento e imagem binária.

As imagens binarizadas permitem identificar as transições de zonas brancas para zonas pretas, ou seja, identificar as linhas brancas que limitam a região de interesse. O algoritmo permite encontrar as linhas de baixo e de cima (limites superior e inferior) e do lado esquerdo e direito (colunas).

Todas as imagens são de 1600x1200 pixels. Assim para encontrar as linhas horizontais que restringem, inferiormente, a ROI, foram analisados apenas os primeiros 600 pixéis, desde a sua origem até meio da imagem, no sentido ascendente. De seguida, foi realizado o mesmo procedimento, mas para o limite superior, pelo que foram analisados os restantes 600 pixéis, no sentido descendente. Para esta análise, foi necessário determinar a espessura das linhas brancas, assim como a

espessura da região a preto entre cada uma das linhas.

O mesmo procedimento foi aplicado para encontrar os limites laterais. No entanto, como estamos perante os extremos laterais, a análise deve ser feita relativamente aos 1600 pixéis, no sentido convergente, de fora para o interior da imagem.

A espessura das linhas brancas e zonas pretas que ficam entre cada uma das linhas brancas foram determinadas através de um *plot* das imagens e utilizando as ferramentas disponíveis obtendo-se aproximadamente 10 pixéis de espessura das mesmas. Com esta informação foi criado um algoritmo que deteta a 1ª linha branca e armazena a sua posição caso se trate de um limite da ROI. Para averiguar se estamos perante uma linha pertencente aos limites ROI, analisa-se se 15 pixéis após esta linha branca voltamos a estar perante outra. Caso contrário, esta linha corresponde a uma linha isolada e não a um limite da ROI. Caso a condição anterior se verifique, significa que encontramos linhas que delimitam a ROI, visto que esta condição corresponde a encontrar 2 linhas brancas próximas, o que apenas ocorre nos limites da região de interesse.

Apesar de as linhas brancas e pretas terem cerca de 10 pixéis de espessura, existem zonas da linha preta onde são encontrados pixéis brancos e zonas em que a linha branca possui pixéis pretos. Isto acontece quer devido a pequenas variações (ruído), quer devido à impossibilidade de encontrar um nível de *threshold* que permita isolar completamente as linhas da imagem, das células, sem que as linhas sejam um pouco deterioradas, havendo em certos casos sobreposição de células e linhas o que culmina em falhas (regiões escuras) nas linhas brancas. Estas questões tornam-se um problema na deteção das linhas. Para colmatar esta situação, é feita a mediana dos pontos coordenados que estão a variar (x no caso das linhas horizontais e y no caso das verticais) para 1ª linha detetada no código, pelo que conseguimos obter os limites da zona de interesse (fig.2).

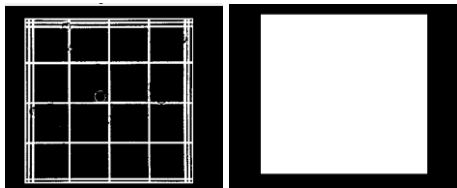


Figura.2. Imagens representativa dos limites e da ROI estimados

Tendo os limites bem definidos, é possível criar uma *mask*, que será posteriormente analisada através da comparação com o *ground truth*. Para esta comparação, começa-se por calcular o índice de *Jaccard* entre as duas *masks*. Posteriormente, calculam-se as distâncias entre os vértices. Os vértices do *ground truth* são encontrados procurando os pontos onde o gradiente tem máximos e mínimos. De seguida, estes têm de ser ordenados, para que se saiba qual das coordenadas corresponde a cada um dos vértices. Pode-se então fazer a comparação entre as coordenadas, determinando as distâncias. Por fim, são determinados valores médios do índice de *Jaccard* e da distância máxima e média entre os vértices.[3]

B. Segmentação das células e respetiva contagem

Tendo a região de interesse definida, é necessário agora implementar algoritmo capaz de detetar as células dentro da região. Para isso, é necessário fazer uma segmentação das células dentro da ROI e prosseguir com a respetiva contagem.

A ROI que foi utilizada foi a ROI *mask* disponibilizada na

DATASET. Para criar o algoritmo responsável pela deteção das células, é necessário definir que células devem ser contabilizadas. Apenas devem ser consideradas as células que se encontrem totalmente dentro da ROI, parcialmente ou totalmente sobrepostas nos limites esquerdo e superior, e células que ao se sobrepor nos limites direito e inferior não ultrapassem a linha do meio. Para uma melhor compreensão o algoritmo pode ser então dividido em 3 partes, que apesar de distintas tem alguns passos comuns. O objetivo é identificar as células sobrepostas às linhas limite esquerdas e superiores; as células dentro da ROI; as células que apenas tocam na linha mais interna dos limites inferior e direito.

Inicialmente as imagens foram convertidas para tons de cinzento e *double*. Foi ainda efetuada uma dilatação das *masks*, para facilitar a deteção de células que intersejam as linhas do limite do lado esquerdo ou superior. No entanto, esta dilatação não deve ser exagerada uma vez que pode levar à inclusão de células que se encontrem totalmente fora da ROI, não devendo estas ser consideradas. As dimensões da máscara para a dilatação da ROI foram obtidas recorrendo a diversas tentativas e respetivas avaliações qualitativas. Posteriormente, é feita uma equalização adaptativa do histograma. Esta é relevante, na medida em que permite aumentar o contraste. Este passo permite que o algoritmo seja menos suscetível a falhas decorrentes de variações de intensidade, tornando-o mais consistente. Este passo permite, então, preservar as características da imagem, aumentar o contraste, e minimizar problemas decorrentes das diferenças de cor e contraste das imagens.

Como as imagens são afetadas por ruído, em análise de imagem é usual utilizar algum tipo de processamento prévio, para minimizar problemas resultantes destas interferências. Neste caso, as imagens foram suavizadas recorrendo a um filtro gaussiano, que em comparação a outros filtros não deterioram tanto a imagem, preservando melhor as suas características.

O processamento da imagem na região correspondente ao interior da ROI baseia-se na utilização do gradiente da imagem suavizada (pelo filtro gaussiano) seguida de Binarização da magnitude do gradiente (*Gmag*), usando um valor de *threshold* de 0,35 determinado por tentativas, e respetivas avaliações qualitativas. O gradiente permite determinar as regiões de variações de intensidade, o que em termos práticos corresponde aos contornos.

Relativamente ao processamento dos limites superior e esquerdo da ROI, inicialmente foi feita uma binarização da imagem, recorrendo a um *threshold* ligeiramente superior ao usado na primeira tarefa. O *threshold* possui 6 tons, sendo o 6º correspondente ao “esqueleto” da imagem. Usando este valor de *threshold*, é possível destacar as células nas zonas sobrepostas aos limites da ROI, obtendo-se uma imagem binária, na qual as células assumem o valor 0 (preto) e as linhas 1 (branco). Para melhorar a segmentação das células, recorreu-se a uma operação de fecho seguido de abertura, com uma *mask* de dimensão 3 e de inclinação de 90º (para as colunas) e de 0º (para as linhas), reduzindo assim o ruído - pontos brancos em zonas pretas e vice-versa. As células que estão sobrepostas nestas linhas são convertidas para branco, facilitando a deteção pela função *imfindcircles*, que *default* deteta círculos brancos.

As etapas referidas anteriormente correspondem ao processamento da imagem para a região interior à ROI e limites

esquerdo e superior, nos quais é necessário detetar e contabilizar células. Resta apenas os limites inferior e direito, onde esta deteção não deve ocorrer. A estratégia seguida baseia-se em binarização da imagem inicial em tons de cinzento sendo que o valor de *threshold* corresponde ao último dos 3 tons determinados para esta imagem. Após a binarização a abordagem recorre a operadores morfológicos e utilização de Hit-or-Miss para obtenção de barras correspondentes aos limites ROI, que serão sobrepostas nos limites direito e inferior, impedido a deteção das células por parte da função *imfindcircles*.

Para tal, efetuou-se uma abertura e fecho da imagem binarizada (correspondentes às linhas referida anteriormente) uma mask de dimensão 51 e inclinação 0° e 90°, o que permite obter as linhas verticais e horizontais, bem como as barras dos limites perpendiculares. Posteriormente, efetuou-se uma dilatação, que permite juntar as linhas dos limites da ROI e eliminar possíveis descontinuidades. A combinação das duas imagens após dilatação resulta na obtenção dos limites da ROI e interseções das linhas.



Figura.3. Imagens representativa dos passos para definição das linhas horizontais e verticais e a sua interseção dentro da ROI

A individualização de limites da região foi conseguida usando Hit-or-Miss com *arrays* de dimensões 25 e 100. Esta individualização permite limitar regiões processadas de formas distintas (Fig.4) e combiná-las numa só imagem (Fig.5).



Figura.4- Imagens resultantes dos diferentes processamentos.

A deteção das células é feita na imagem combinada representada na figura 4 e utilizando na função *imfindcircles*, com uma gama de raios [23 51] e sensibilidade 0,885. Estes parâmetros foram afinados iterativamente. Verificou-se, também, que o algoritmo possui um maior desempenho quando se procede ao preenchimento das linhas no interior da ROI, facilitando a deteção de células que se sobrepõem às linhas.[3]

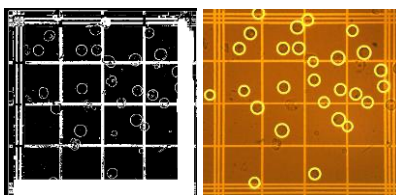


Figura.5. Figura representativa da imagem final e da deteção das células.

III. DISCUSSÃO

A avaliação e discussão dos resultados é feita para as

imagens disponibilizadas na DATASET e deve também ser realizada, numa fase final, perante imagens que não são conhecidas aquando da criação do algoritmo para testar a sua eficácia e se a sua alteração é ou não significativa perante as novas imagens.

A. Avaliação da ROI definida em relação ao ground truth

A definição da ROI foi possível recorrendo a diversas tentativas, de forma a aprimorar os resultados obtidos. Com o intuito de avaliar a eficiência do nosso algoritmo, foi implementado código que permita comparar a ROI definida com os *ground truths* cedidos pelos docentes.

Tabela 1. Valores obtidos para análise da ROI

Imagens Analisadas	Valor médio Jaccard	Valor máximo da distância Euclid.	Valor médio da distância Euclid.
Imagens de Treino	0,9815	12,0751	7,8616
Imagens Teste	0,9797	13,1084	8,5471

Pela análise dos resultados obtidos verifica-se que a ROI foi bem determinada, uma vez que, se obteve um índice Jaccard (0,9815) bastante próximo de 1 e uma distância Euclidiana máxima e média (12,0751 e 7,8616, respetivamente) reduzidas, tendo em conta as dimensões das imagens, 1200x1600.

Após testar o algoritmo nas imagens teste, foram determinados os valores que pretendem avaliar a eficiência do algoritmo. Concluímos que a diferenças nos valores médio do Jaccard, valor máximo da distância Euclidiana e o valor médio da distância Euclidiana, obtidos para as imagens de treino e imagens teste, não são significativas, o que significa que não ocorreu um decréscimo significativo na eficiência do algoritmo. Assim conclui-se que este é um algoritmo relativamente generalizado, e eficaz. No entanto verificou-se que este algoritmo tem uma menor eficiência perante linhas inclinadas, uma vez que a inclinação não considerada no algoritmo. Este problema não é significativo uma vez que a inclinação das linhas é geralmente muito reduzida. Uma opção para colmatar esta pequena falha seria recorrer a 2 ou mais pontos de cada aresta e definir uma reta que os inclui-se formando assim os limites da ROI estimada.

B. Avaliação da segmentação das células e respetiva contagem

Para averiguar se o nosso raciocínio permitiu a criação do algoritmo de deteção das células eficiente, efetuamos a avaliação de número de células contadas, o número de *true positives* (TP), o número de *false positives* (FP), e o número de *false negatives* (FN). Foram ainda aplicadas medidas de avaliação como *Recall* (R), *Precisão* (P) e *F1-measure* (F1)

Quando à área segmentada corresponder um índice de *Jaccard* superior a 0,5, considera-se *true positive* -células detetadas corretamente – quando o índice é inferior a 0,5, considera-se FP. Células do GT que não correspondem a nenhuma região segmentada são *false negatives* (FN).

Tabela 2. Valores obtidos dos parâmetros a partir do algoritmo para avaliação da eficácia do mesmo.

Imagens Analisadas	Nº células contadas	TP	FP	FN
Imagens de treino	13,6400	12,9200	0,700	1,8200
Imagens de teste	12,7400	12,3400	0,4000	2,5200

Com base nos valores obtidos referentes aos parâmetros, podemos inferir, em relação às imagens de treino, que o algoritmo foi bem idealizado, uma vez que o número de FN e principalmente FP é bastante reduzido sendo TP relativamente próximo do número de células contadas. Quando o algoritmo foi sujeito a novas imagens assumiu um cenário semelhante embora se tenha verificado um ligeiro decréscimo na de células (maior número de FN) e um maior desempenho na distinção das células relativamente a artefactos (menos FP) (tabela.2).

Podemos assim concluir que o algoritmo é relativamente eficaz e generalizado, o que permite a sua utilização perante outras imagens, visto que os valores de P, R obtidos se mantiveram relativamente elevados e foram obtidos valores de F1 consideráveis e muito próximos nas imagens de treino e teste

Tabela 3. Valores obtidos dos parâmetros a partir do algoritmo para avaliação da eficácia do mesmo.

Imagens Analisadas	R	P	F1
Imagens de Treino	0,8612	0,9508	0,8966
Imagens de teste	0,8145	0,9813	0,8825

Por fim, o algoritmo foi analisado tendo em vista a identificação de falhas do mesmo. Verificou-se que a forma de processamento dos limites da ROI não é totalmente eficiente, podendo não ser detetadas todas as células, principalmente se estiverem muito próximas, ou em casos raros pode ocorrer a deteção de células nos limites direito e inferior. Foram também identificadas algumas situações muito específicas em que o algoritmo é incapaz de detetar células, como o caso dos cantos da ROI. Qualquer célula nesta região é ignorada pelo algoritmo, uma vez que estas células se encontram “camufladas” pelas linhas pertencentes aos limites da ROI (fig.6).

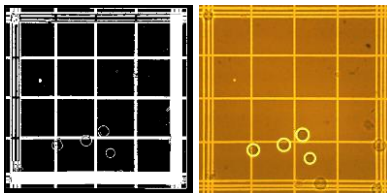


Figura.6. Representação do problema de deteção de células no canto superior esquerdo.

Foi ainda testada uma alternativa a este tipo de processamento dos limites esquerdo e superior, que se baseava num processamento semelhante ao do interior da ROI (gradiente, binarização), seguido se um conjunto de operações que pretendiam retirar as linhas e preservar os contornos das células. No entanto, esse método deteriorava demasiado os contornos das células, sendo menos eficiente que a abordagem seguida. Para além disso observou-se que processamento nos limites da ROI pode falhar em situações muito específicas, em que existem células cujo tamanho é superior à largura de toda barra do limite, resultando numa falha nesse local nas barras correspondentes ao processamento dos limites da ROI.

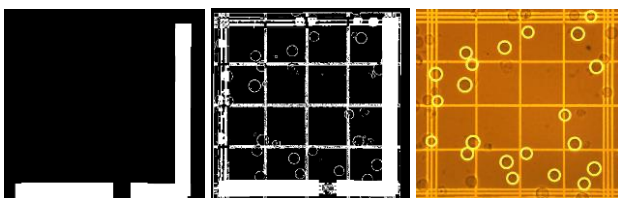


Figura.7. Representação da falha na deteção das células principalmente nos limites e da possível falha na formação das barras dos limites.

Isto acontece porque o valor de *threshold*, embora seja determinado usando o método de *otsu* para cada imagem ainda permite preservar células do limite, pelo que, após efetuar as operações de fecho e abertura como a máscara utilizada não possui uma dimensão suficientemente grande para eliminação da falha, esta prevalece sendo acentuada pelo Hit-or-Miss, usando *arrays* de dimensões 25, 100. Esta falha irá resultar numa zona no qual não irá ocorrer qualquer tipo de processamento. Este problema decorre também das diferenças de cor e intensidades das imagens, podendo ser mais ou menos afetadas por este efeito. O nível de *threshold* e dimensões das máscaras foram afinadas iterativamente, de forma a minimizar ao máximo a ocorrência deste problema.

Pela análise de resultados verifica-se que o principal problema do algoritmo diz respeito aos falsos negativos, ou seja, à determinação deficiente das verdadeiras células. Este aspeto poderia ser melhorado incrementando a sensibilidade da função *imfindcircles*. No entanto, isto conduziria a um aumento dos falsos positivos, pelo que este parâmetro deve ser equilibrado, de forma a obter a melhor eficiência geral do algoritmo. Uma opção interessante seria usar *Marr & Hildreth Edge Detector* com filtros *laplacianos* de diferentes dimensões, de forma a detetar a máxima resposta para diversos tamanhos das células, e, por fim, combinar as imagens, preservando as respostas de maior intensidade em cada local e definindo um valor de *threshold*. Esta abordagem poderia facilitar a deteção de células no interior da ROI, sem necessidade de aumentar a sensibilidade do *imfindcircles*.

IV. CONCLUSÃO

Este trabalho permitiu utilizar os conhecimentos adquiridos ao longo do leccionamento da unidade curricular assim como adequar as nossas escolhas perante o objetivo e possíveis adversidades em que por vezes tivemos de permitir a deterioração de certas propriedades para conseguirmos melhorar outras. Este caso prático deu-nos uma nova perspetiva relativamente ao papel de um engenheiro biomédico no mundo da investigação e trabalho. Resolvemos um problema concreto e relevante de forma muito económica. Apesar de ter algum erro na sua eficácia não deixa de ser vantajoso a elaboração deste projeto e visto que os resultados foram relativamente bons pode ser utilizado para auxiliar investigadores, professores e todos aqueles que trabalhem com cultura celular.

V. REFERÊNCIAS

- [1] Özkan, A., İsgör, S. B., Şengül, G., & İsgör, Y. G. (2018). Computer vision based automated cell counting pipeline: A case study for HL60 cancer cell on hemocytometer. *Biomedical Research (India)*, 29(14), 2956–2962. <https://doi.org/10.4066/biomedicalresearch.29-18-575>
- [2] Birnie, G. D. (1988). The HL60 cell line: A model system for studying human myeloid cell differentiation. *British Journal of Cancer*, 58(SUPPL. 9), 41–45.
- [3] Gonzalez, Rafael C; *Digital image processing using Matlab*. ISBN: 0-13-008519-7