

Segmentação, crescimento e seguimento de células *Escherichia coli* em imagens microscópicas

Diana Isabel Campos Santos

Relatório Final do Trabalho de Projeto apresentado à
Escola Superior de Tecnologia e Gestão
Instituto Politécnico de Bragança

para obtenção do grau de Mestre em
Tecnologia Biomédica

Orientador

Dr. Fernando Monteiro

Julho de 2017

Agradecimentos

A realização deste relatório só foi possível devido ao apoio incondicional de várias pessoas que o fizeram de diversas formas e durante todas as etapas do trabalho e em especial gostaria de agradecer:

ao Professor Doutor Fernando Monteiro por a sugestão do tema da tese, pela sua disponibilidade, pela sua paciência em transmitir-me todos os seus conhecimentos ao longo deste trabalho e por a incentivar-me a fazer sempre melhor.

a todos os meus familiares, em especial aos meus pais e à minha irmã pelo apoio incondicional, por todo o carinho, ajuda, força e compreensão durante a realização deste trabalho e por estarem sempre ao meu lado em todos os momentos da minha vida.

a todos os meus amigos pelas palavras de incentivo, por todos os momentos de apoio e de convívio.

Muito obrigada a todos por estarem ao meu lado.

Resumo

Este trabalho tem como finalidade a obtenção do seguimento celular, sendo importante para os profissionais da área da microbiologia, visto que auxilia na contagem de células e na evolução e crescimento das mesmas de modo a diminuir os erros de contagem e seguimento manual das células.

Neste projeto aborda-se o crescimento celular de células *Escherichia coli* (*E.coli*), a segmentação de células e o seguimento celular. Primeiramente analisou-se o software *schnitzcells* que é um programa de seguimento de células e deste modo procedeu-se a um estudo na tentativa de obter melhores resultados. Desta forma, através de modificações no processo de segmentação obtiveram-se melhores resultados do que no programa *schnitzcells*. De seguida, aplicou-se o método de sobreposição de imagens aos pares de modo que os centróides das células na imagem seguinte estivessem sobrepostos com os centróides das células na imagem anterior e assim proceder-se ao seguimento celular. No seguimento celular verificou-se os centróides dos objetos na imagem seguinte encontravam-se no interior de uma célula na imagem anterior e também calculou-se a distância mínima entre objetos, uma vez que existiam células não sobrepostas e deste modo obteve-se o seguimento celular.

Assim, constata-se melhores resultados na metodologia implementada, visto que o processo realiza-se de forma automática enquanto no software *schnitzcells* necessita de correções manuais a nível da segmentação e do seguimento celular para obterem-se resultados corretos.

Palavras-chave: Células E.Coli, Segmentação de Células, Seguimento Celular, Software *Schnitzcells*.

Abstract

This work aims to obtain the cell tracking, being important to the professionals in the area of microbiology, as it aids in the counting of cells and in the evolution and growth of them in order to decrease the errors of counting and manual tracking of cells.

On this thesis it is described *Escherichia Coli* (*E.coli*) cellular growth, cellular segmentation and its tracking. At first it was conducted an analysis of the software *schnitzcells*, which is a program that tracks cells so that a study in an attempt to achieved better results could be executed. In this way, through some modifications in the process of segmentation, better results were indeed achieved than the ones presented in the program *schnitzcells*. Afterwards, the method of overlapping images was applied in pairs so that the centroids of the cells in the next image were overlapping with the centroids in the previous image in order to ensue to the tracking of the cells. In the process of tracking the cells, it was verified whether the centroids of the objects in the next image were inside a cell in the previous image.

Thus, it was calculated the minimum distance between objects and it was obtained the segmentation of each cell. In short, better results are obtained in the implemented methodology, since the process is performed automatically while with the software *schnitzcells* manual corrections are required in the segmentation and in the cellular tracking in order to obtain correct results.

Keywords: E.coli Cells, Cells Segmentation, Cellular Tracking, *Schnitzcells* Software.

Índice

Índice de Figuras	xi
Índice de Tabelas	xv
Capítulo 1. Introdução	1
Estrutura do relatório	2
Capítulo 2. Fundamentação Teórica.....	3
2.1 Definição de Célula	3
2.1.1 Estrutura celular	3
2.1.2 Escherichia Coli	5
2.1.3 Reprodução e crescimento celular.....	5
2.2 Segmentação de imagens	7
2.3 Seguimento celular.....	8
Capítulo 3. Revisão Bibliográfica.....	9
Trabalhos que estudam o seguimento de células	9
3.1 Softwares de segmentação e de seguimento	9
3.1.1 CellTracer	9
3.1.2 CellTracker.....	10
3.1.3 CellTrack	10
3.1.4 MicrobeTracker	11
3.1.5 TLM-tracker	11
3.1.6 CellAging	11
3.1.7 BactImAS	12
3.1.8 Schnitzcells	12

3.1.9 ‘miSimBa’ - Simulador de imagens microscópicas de células bacterianas	13
3.10 Oufiti	13
3.11 Molyso	13
Capítulo 4. Metodologia implementada	15
4.1 Pré-processamento	15
4.2 Segmentação de imagens	16
4.3 Pré-processamento para o seguimento celular	17
4.4 Seguimento celular	21
4.5 Gráfico em árvore dos resultados obtidos no seguimento celular	33
Capítulo 5. Análise de resultados	37
5.1 Comparação entre o método implementado e o software TLM-Tracker	42
Capítulo 6. Conclusão	45
Referências Bibliográficas	47
Anexos	

Índice de Figuras

Figura 1. Ilustração e fotografia de uma bactéria Escherichia Coli a partir de visualização em microscópio.....	3
Figura 2. Ilustração e fotografia de um protozoário a partir da visualização em microscópio	4
Figura 3. Células Escherichia coli	5
Figura 4. Crescimento celular através do método fissão binária.....	6
Figura 5. a) Regiões resultantes da segmentação de imagens. b) Fronteiras ou contornos obtidos pelo processo de segmentação.	7
Figura 6. c) Gráfico de seguimento celular obtido através do software TLM-tracker. d) Seguimento celular resultante do software CellTracer.[14][15]	8
Figura 7. Imagem original com a aplicação da função imadjust.....	15
Figura 8. a) Imagem original. b) Imagem resultante da segmentação.....	17
Figura 9. c) Imagem resultante da segmentação. d) Imagem obtida através das operações morfológicas.	18
Figura 10. Primeira imagem com as células identificadas (controlo do seguimento celular).	18
Figura 11. e) Exemplo da 1ª iteração que representa a primeira imagem, a segunda imagem e a sobreposição dos centros desalinhados através da função imoverlay, respetivamente. f) Exemplo da 8ª iteração que representa a oitava imagem, a nona imagem e a sobreposição dos centros desalinhados através da função imoverlay, respetivamente.	19
Figura 12. g) Primeira imagem, segunda imagem e sobreposição dos centros alinhados, respetivamente, deslocação do centro da segunda imagem para o centro da primeira imagem através do vetor de translação. h) Oitava imagem, nona imagem e sobreposição dos centros alinhados através do vetor de translação, respetivamente, deslocação do centro da nona imagem para o centro da oitava imagem.	21
Figura 13. Sobreposição entre as imagens 6 e 5.....	22
Figura 14. Sobreposição das imagens 8 e 7.....	22
Figura 15. Visualização de uma sequência de imagens (imagem 1 até à imagem 8) em que ocorre um erro no seguimento.	23

Figura 16. Visualização de uma sequência de imagens (imagem 1 até à imagem 8) com o erro de seguimento corrigido.	24
Figura 17. Visualização de uma sequência de imagens (imagem 9 até á imagem 14) em que deteta-se falhas no seguimento.	26
Figura 18. Visualização da sequência de imagens (imagem 9 até á imagem 14) com o seguimento corrigido.	27
Figura 19. a) Sobreposição da imagem 14 com a imagem 13 através da função imoverlay sem alterações ao vetor translação. b) Sobreposição da imagem 14 com a imagem 13 através da função imoverlay com modificações no vetor translação.	28
Figura 20. c) Imagem 15 com o seguimento celular correto. d) Imagem 16 com o seguimento celular incorreto.	28
Figura 21. e) Imagem 15. f) Imagem 16 com o seguimento correto após modificação ao vetor translação.	29
Figura 22. Representação de uma sequência de imagens desde da imagem 17 até à imagem 19, onde observa-se um erro no seguimento.	30
Figura 23. Representação de uma sequência de imagens desde da imagem 17 até à imagem 19 com o seguimento corrigido.	31
Figura 24. g) Imagem 19 com seguimento correto. h) Imagem 20, onde detetou-se um erro no seguimento.	31
Figura 25. i) Imagem 19 com o seguimento correto. j) Imagem 20 com o seguimento corrigido.	32
Figura 26. Visualização de um exemplo de gráfico em árvore no MATLAB.	33
Figura 27. Gráfico em árvore (seguimento célula 1).	34
Figura 28. Gráfico em árvore.	35
Figura 29. Gráfico em árvore com os nós numerados.	36
Figura 30. a) Imagem resultante da segmentação do schnitzcells original. b) Imagem resultante da segmentação da metodologia implementada.	37
Figura 31. Representação em sequência desde da imagem 1 até à imagem 7 dos resultados obtidos do seguimento no software schnitzcells, onde se detetou um erro... ..	38
Figura 32. Sequência de imagens desde da imagem 1 até á imagem 7 dos resultados obtidos através da metodologia implementada.	38

Figura 33. Representação da imagem 1 até á imagem 6 dos resultados obtidos do seguimento com o software schnitzcells.	39
Figura 34. Sequência de imagens desde da imagem 1 até á imagem 6 dos resultados obtidos através da metodologia implementada.....	39
Figura 35. Representação de imagens consecutivas (desde da imagem 1 até à imagem 7), onde verifica-se a divisão correta da célula número dois.....	39
Figura 36. Sequência da imagem 4 até à imagem 7, problema no seguimento executado pelo software schnitzcells.....	40
Figura 37. Representação em sequência desde a imagem 4 até à imagem 7 dos resultados obtidos na metodologia.	40
Figura 38. Sequência de imagens desde da imagem 8 até á imagem 11, deteta-se erro no seguimento na execução do software schnitzcells.....	41
Figura 39. Representação da sequência de imagens desde a imagem 8 até á imagem 11 com o seguimento correto.....	41
Figura 40. Visualização do gráfico obtido do software TLM-Tracker	43

Índice de Tabelas

Tabela1. Comparação dos resultados do software Schnitzcells com o método implementado.....	42
Tabela2. Comparação dos resultados do software TLM-Tracker com o método implementado.....	44

Capítulo 1. Introdução

Ao longo de vários anos, realizaram-se diversos trabalhos no campo da microscopia em que utilizaram-se um conjunto de procedimentos de análise em imagens microscópicas. A quantificação das células e a evolução celular das imagens em estudo normalmente é um procedimento lento, podendo apresentar uma percentagem de erro significativa devido principalmente às inúmeras observações a serem efetuadas.

O crescimento celular é definido como o aumento coordenado de todos os constituintes celulares. O crescimento microbiano é normalmente associado ao crescimento de uma população de células de um dado microrganismo, ou seja, com o aumento do número de células. A maior parte das células multiplicam-se pelo processo de fissão binária, em resultado em que uma célula ao fim de um certo tempo originará duas células.^[1]

O processo de segmentação de imagens tem como objetivo simplificar e/ou mudar a representação de uma imagem para facilitar a sua análise e é tipicamente utilizada para localizar objetos e formas (linhas, curvas) em imagens. A segmentação consiste na realidade em dividir a imagem em diferentes regiões que posteriormente serão analisadas por algoritmos especializados em retirar informação das imagens.^[2]

O seguimento celular consiste em conhecer a origem das células desde das primeiras células que constituem uma colónia de células, detetando as divisões celulares e identificando as células mãe e as células filha, sendo que o objetivo do seguimento de células é em criar uma trajetória da célula ao longo do tempo.

Deste modo, este trabalho tem como finalidade a obtenção de melhores resultados do que *softwares* de seguimento existentes através da melhoria do *software schnitzcells* tanto a nível da segmentação como a nível do seguimento celular. Desta forma, produziu-se um programa em formato automático que faz a contagem do número de células e demonstra o seguimento celular numa colónia de células.

Assim, o desenvolvimento de programas de processamento de imagem que permitam a criação de processos de contagem de células e o histórico da evolução celular de forma automática em imagens microscópicas em estudo torna-se uma mais-valia para os colaboradores de laboratório, uma vez que o processo para obtenção de resultados reduz-se e diminui a percentagens de erros que ocorriam manualmente.

Estrutura do relatório

Este relatório encontra-se dividido em seis capítulos, sendo que no capítulo 1 descreve-se uma introdução ao trabalho desenvolvido.

O capítulo 2 corresponde à fundamentação teórica, ou seja, é uma parte introdutória do tema da tese.

O capítulo 3 diz respeito à revisão bibliográfica, onde encontram-se descritos diversos softwares e trabalhos de seguimento celular.

No capítulo 4 descreve-se os métodos, as operações aplicadas e explica-se a metodologia implementada no trabalho.

O capítulo 5 corresponde à análise de resultados, comparando-se o método desenvolvido com o software *Schnitzcells* e com o software *TLM-Tracker*.

O capítulo 6 diz respeito à conclusão do método desenvolvido.

Capítulo 2. Fundamentação Teórica

2.1 Definição de Célula

A menor unidade estrutural básica e funcional dos seres vivos denomina-se de célula, sendo que os organismos são formados a partir destas unidades, pois o corpo humano na sua constituição possui cerca de 100 trilhões de células.[3][4]

2.1.1 Estrutura celular

Existem dois tipos de células: as células procarióticas e as células eucarióticas. Na sua constituição as células procarióticas possuem simplesmente uma única célula (seres unicelulares), uma vez que foram as primeiras células a surgirem na Terra. As células procarióticas apresentam uma estrutura mais simplificada do que as células eucarióticas, pois não possuem organelos nem núcleo apenas apresentam uma área nuclear denominada de nucleóide. O material genético (DNA) encontra-se difuso no citoplasma devido à ausência de delimitação de um invólucro (membrana nuclear), ou seja, não existe um núcleo organizado como verifica-se na Figura 1.[4][5][6]

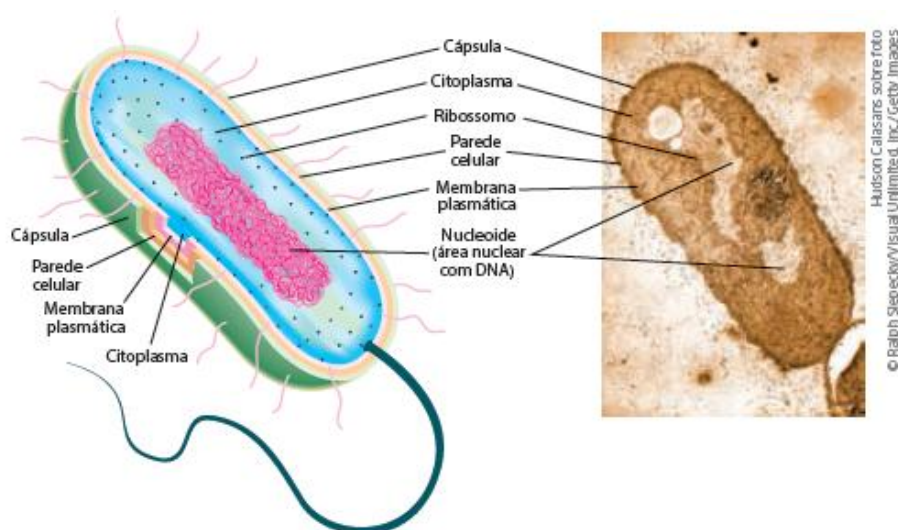


Figura 1. Ilustração e fotografia de uma bactéria Escherichia Coli a partir de visualização em microscópio.[6]

As células procarióticas existem em todos os tipos de ambiente (água, ar, solo) e apresentam diversas formas de vida como:

- Bactérias - algumas são decompositoras de cadáveres de animais e de plantas, tendo grande importância nas cadeias alimentares.[3]
- Simbiontes – vivem em harmonia com indivíduos de outras espécies, como por exemplo as bactérias do nosso intestino.[3]
- Bactérias parasitas – que causam doenças nas plantas, nos animais e no Homem. A tuberculose, a pneumonia, o tétano, a lepra, a meningite e sífilis são alguns exemplos de doenças causadas por bactérias na espécie humana.[3]

Em relação às células eucariontes possuem uma estrutura mais complexa do que as células procariontes, uma vez que possuem membrana nuclear individualizada, núcleo organizado e diversos tipos de organelos, como visualiza-se na Figura 2. A maioria dos seres vivos tem na sua constituição células eucarióticas como os protozoários, algas, fungos, vegetais e animais.[4][5][6]

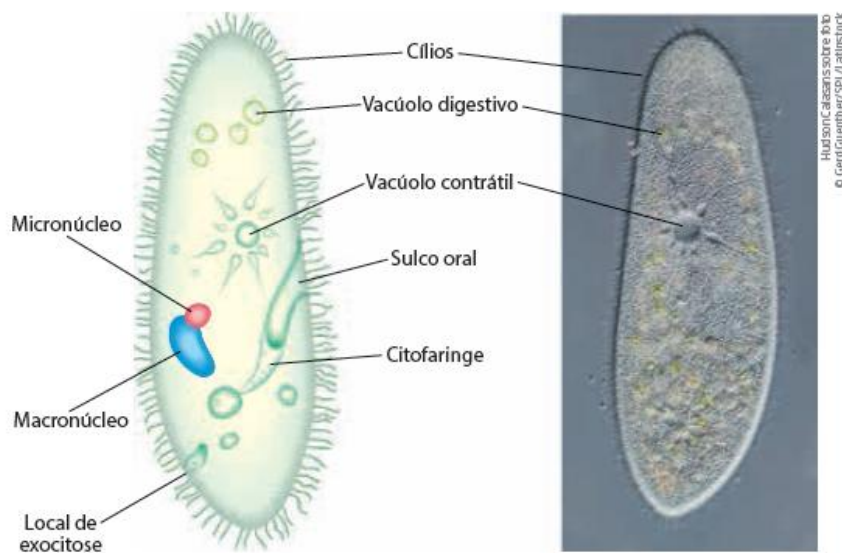


Figura 2. Ilustração e fotografia de um protozoário a partir da visualização em microscópio.[6]

2.1.2 Escherichia Coli

A célula *Escherichia coli* é uma célula procariótica bastante utilizada para estudos de evolução celular, uma vez que apresenta uma estrutura simplificada e multiplica-se com elevada rapidez. Possui uma membrana plasmática semelhante à das células eucarióticas e apresenta-se em forma de bastonete. A célula *E.coli* (Figura 3) constitui um grupo de bactérias Gram-negativas, anaeróbicas facultativas que pertencem à família das *Enterobacteriaceae*. O habitat natural é o organismo de animais endotérmicos (seres humanos e animais) localizando-se no lúmen intestinal.[7][8]

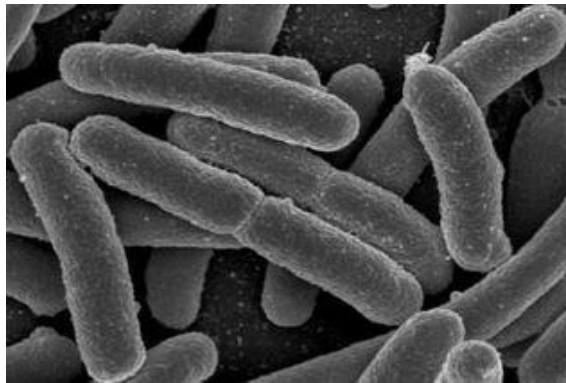


Figura 3. Células *Escherichia coli*. [9]

2.1.3 Reprodução e crescimento celular

A maioria das células são frequentemente reproduzidas e substituídas durante a vida de um indivíduo. Em microbiologia o crescimento celular refere-se ao aumento do número de células, ou seja, é um conjunto de processos metabólicos que conduz à divisão de uma célula originando duas células filha.

Deste modo, a reprodução celular consiste no processo de divisão celular em que as células dividem-se para formar novas células obtendo-se duas células semelhantes.

As células procarióticas, nomeadamente as células E.coli dividem-se por o método de fissão binária, aumentando o tamanho original para o dobro. Em que o DNA das células está contido num único cromossoma e deste modo o processo começa com a replicação do mesmo. De seguida, ocorre a ligação do novo cromossoma à membrana do plasma e os dois cromossomas migram para as extremidades opostas da célula. A membrana de plasma no meio cresce para o interior da célula até ao ponto em que fecha-se para a separação em dois compartimentos, em que cada um contém um conjunto completo de material genético.

Assim, a célula sofre fissão no centro, formando duas novas célula filhas como verifica-se na Figura 4.[1]

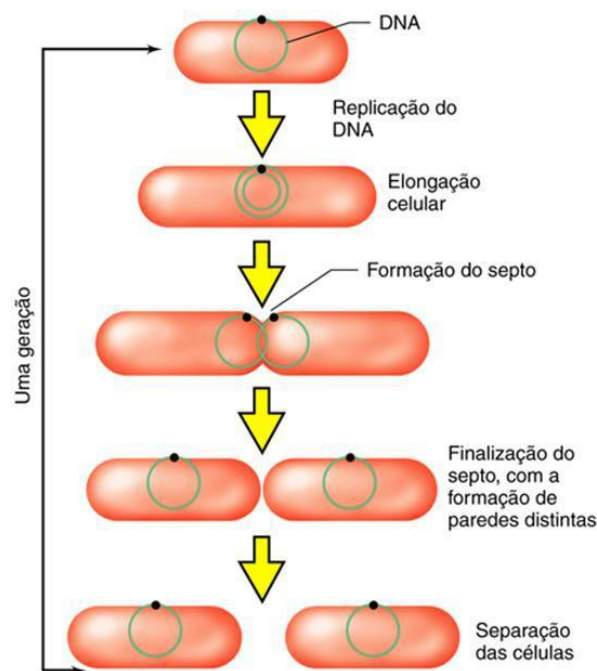


Figura 4. Crescimento celular através do método fissão binária [1]

2.2 Segmentação de imagens

A segmentação de imagens consiste num dos processos mais complexos num sistema de processamento de imagens, uma vez que esta área consiste na extração ou identificação de objetos e regiões contidos na imagem, como visualiza-se na Figura 5. Tem como finalidade a divisão de uma imagem em objetos ou regiões que a constituem de maneira a que os pixéis de cada região possuam um conjunto idêntico de propriedades ou atributos, ou seja, o processo de segmentação deve ser finalizado quando o objeto de interesse se encontrar isolado. Estes atributos podem incluir propriedades de textura, níveis de contraste, valores espectrais, cor, área, forma que podem ser retirados dos objetos e utilizados no processo de análise de imagem. Deste modo, a imagem segmentada resultará num conjunto de regiões que encontram-se ligadas mas não sobrepostas de maneira a que cada pixel na imagem pertença apenas a uma região.

Os algoritmos de segmentação normalmente baseiam-se em duas propriedades básicas de níveis de intensidade:

- Descontinuidade – A divisão da imagem pode ser feita com base nas alterações bruscas da intensidade da imagem nos níveis de cinzento, tais como os contornos dos objetos.
- Similaridade – A segmentação de imagem é efetuada com base no parâmetro de similaridade entre pixels, seguindo um determinado critério como por exemplo: a binarização, crescimento de regiões, divisão e junção de regiões. [2][10][11]

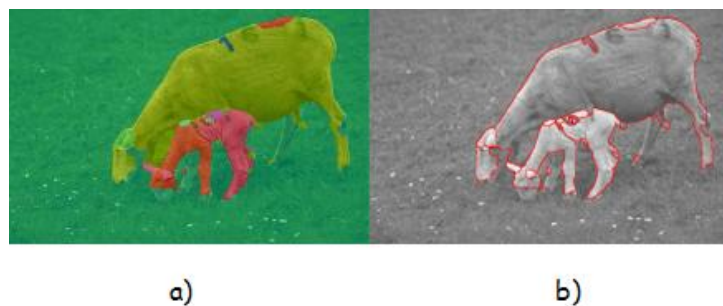


Figura 5. a) Regiões resultantes da segmentação de imagens. b) Fronteiras ou contornos obtidos pelo processo de segmentação. [10]

2.3 Seguimento celular

O estudo do comportamento celular encontra-se diversas vezes interligado com a avaliação de várias características biológicas. A área de processamento de imagem está relacionada com o seguimento temporal e automático de determinadas amostras celulares e deste modo desenvolvem-se algoritmos de seguimento automático em imagens microscópicas de maneira a facilitar o trabalho de laboratório, como se observa na Figura 6.

O seguimento celular tem como objetivo a criação de uma trajetória das células que constituem uma imagem ao longo do tempo. A forma mais fácil de obter o seguimento é associar cada célula previamente detetada à célula mais próxima na imagem seguinte, considerando por exemplo a posição dos centróides de cada célula.[12][13]

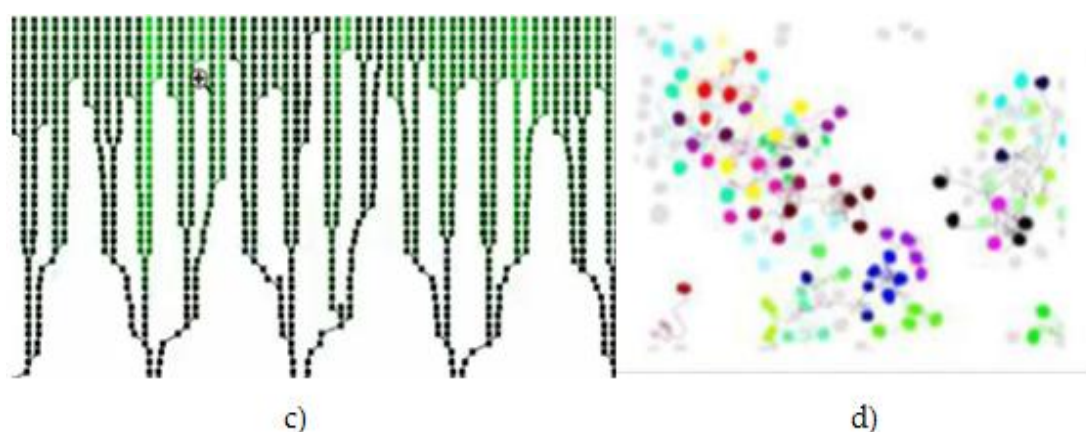


Figura 6. c) Gráfico de seguimento celular obtido através do software TLM-tracker. d) Seguimento celular resultante do software CellTracer.[14][15]

Capítulo 3. Revisão Bibliográfica

Trabalhos que estudam o seguimento de células

Existem diversos trabalhos que estudam e criam métodos para o seguimento celular de colónia de células nomeadamente:

- “High-throughput gene expression analysis at the level of single proteins using a microfluidic turbidostat and automated cell tracking.”^[16]
- “Tracking single-cell gene regulation in dynamically controlled environments using an integrated microfluidic and computational setup.”^[17]
- “Tracking Single-Cells in Overcrowded Bacterial Colonies.”^[18]
- “Bacterial Acrobatics on a Surface: Swirling Packs, Collisions, and Reversals during Swarming.”^[19]
- “Image Segmentation and Dynamic Lineage Analysis in Single-Cell Fluorescence Microscopy.”^[20]
- “Automated tracking of migrating cells in phase-contrast video microscopy sequences using image registration.”^[21]

3.1 Softwares de segmentação e de seguimento

De seguida, são apresentadas algumas aplicações mais usadas para o seguimento celular.

3.1.1 CellTracer

CellTracer consiste num software que se baseia nas ferramentas de processamento de imagem facultadas pelo MATLAB e possibilita a utilização de alguns procedimentos semiautomáticos ou manuais de correção de alguns problemas resultantes do processo automático. CellTracer é uma aplicação que permite a utilização de algumas ferramentas automáticas de segmentação e

reconstrução temporal. A aplicação pode ser testada em vários tipos de células nomeadamente E.coli, humanas e leveduras. O processo de seguimento automático e segmentação é dividido em diferentes etapas, passando por uma extração da região de fundo da imagem e de seguida por um processamento das regiões dos contornos existentes. O software encontra-se disponível gratuitamente e também disponibiliza manual de utilização.[15]

3.1.2 CellTracker

O software CellTracker, apresenta-se como um sistema para o seguimento nuclear e citoplasmático através dos picos de intensidade fluorescente e foi desenvolvido para a avaliação de processos em células eucariontes. A aplicação utiliza os canais de deteção de objetos, em que as próprias células podem ser detetadas com base em intensidades de imagem, exemplo via thresholding. O CellTracker encontra-se disponível gratuitamente e também disponibiliza um manual de utilização do software.[22]

3.1.3 CellTrack

O programa *CellTrack* foi divulgado no ano de 2008 e o mesmo consiste num sistema desenvolvido em *Open-Source* (C++) que utiliza uma combinação de métodos e permite a análise automática de seguimento celular. Possui um sistema de seguimento dos vários segmentos a partir dos contornos correspondentes. Este software fornece uma imagem geral de processamento e funções de melhoria, tais como alisamento e subtração de fundo. Para cada um destes métodos, o utilizador pode alterar os parâmetros padrão de diferentes aplicações de seguimento celular. O CellTrack também mede a velocidade e a área das células tendo em conta a deformação celular.[23]

3.1.4 MicrobeTracker

MicrobeTracker é um programa desenvolvido de forma automática para a análise de imagens em organismos pequenos. Esta aplicação é de fácil acesso e encontra-se disponível para qualquer utilizador. *MicrobeTracker* utiliza uma interface gráfica elaborada no programa MATLAB permitindo uma análise detalhada das células. O software permite a identificação de células a partir do algoritmo *SpotFinder*, determina quando as células foram divididas e pode ser utilizado para rastrear a linhagem (família) de células através das divisões celulares.[24]

3.1.5 TLM-tracker

O software *TLM-tracker* permite a identificação de células individuais e o seguimento celular através de sequências temporais de imagens de células *Bacillus megaterium*. O algoritmo é constituído por quatro etapas: pré processamento, segmentação, seguimento e exportação de dados, ou seja, visualização da árvore de linhagem das células. O software também disponibiliza diversos algoritmos para o processo de segmentação de imagens.[14]

3.1.6 CellAging

O programa *CellAging* foi desenvolvido para a segmentação e rastreamento celular em estirpes (linhagem) de células *E.coli* com o intuito de estudar a segregação e separação que ocorre num processo de divisão celular. O programa é executado através de uma interface gráfica do utilizador. Esta ferramenta permite o estudo e o cálculo de características relacionadas com o envelhecimento celular. Também apresenta a possibilidade do estudo *in vivo* da dinâmica de distribuição espacial na divisão de qualquer elemento intracelular fluorescente marcado.[25]

3.1.7 BactlmAS

A aplicação *BactlmAS* é descrita como uma multiplataforma desenvolvida em open-source e complementada com o software baseado em *Java*. O software tem como objetivo a extração e visualização da quantidade de dados resultantes dos filmes microscópicos de lapso de tempo das bactérias. O *BactlmAS* contém alguns procedimentos e ferramentas semiautomáticos, uma vez que o utilizador tem a possibilidade de definir as células iniciais e corrigir manualmente o processo de segmentação através do programa *Image J*. O software disponível para a análise de sequências de imagens de microscopia de lapso de tempo somente funciona para determinadas bactérias e sob condições limitadas. Esta aplicação é projetada para a segmentação e seguimento de células, assim como para armazenamento, análise e visualização dos dados adquiridos a partir das imagens microscópicas.[26]

3.1.8 Schnitzcells

O software *schnitzcells* encontra-se desenvolvido em linguagem Matlab, fornecendo uma interface de fácil acesso, uma vez que fornece duas possibilidades de utilização do programa. O algoritmo foi criado para executar a segmentação e seguimento de imagens celulares, nomeadamente de células *E.Coli* e *Bacillus subtilis*. Para além dos processos automáticos de segmentação e seguimento celular contém um sistema que permite editar as informações das imagens resultantes, de modo a corrigir determinados erros ocorridos, como por exemplo divisão celular não simétrica. A informação obtida pela execução do programa é guardada numa única estrutura, que pode ser posteriormente utilizada para uma simples visualização, como para procedimentos de avaliação adicionais.[27]

3.1.9 ‘*miSimBa*’ - Simulador de imagens microscópicas de células bacterianas

O ‘*miSimBa*’ consiste num simulador de imagens artificiais de células *Escherichia. Coli*. Esta aplicação reproduz a organização espacial e temporal das células bacterianas através de modelos realistas de morfologia celular (forma, tamanho e arranjo espacial), o crescimento e divisão celular, a motilidade celular e algumas funções internas e estruturas intracelulares, ou seja, o nucleóide celular. O simulador incorpora parâmetros de aquisição de imagens que simulam a iluminação e as principais fontes de ruído. O ‘*miSimBa*’ é uma ferramenta que baseia-se nas capacidades de programação orientada a objetos no programa *MATLAB*. A aplicação permite a simulação de um número quase ilimitado de imagens a serem processadas, enquanto num laboratório real não seria possível devido aos custos financeiros e ao tempo limitado.[28]

3.10 Oufiti

O programa *oufiti* permite a deteção de divisões celulares e obtenção do seguimento celular em microrganismos e possibilita a análise de grandes conjuntos de dados. Possui módulos interativos para a segmentação e permite análises pós-processamento. O algoritmo permite a deteção e caracterização de objetos de difração e a caracterização quantitativa de nucleóide. Fornece ferramentas para a análise de dados, para análise do ciclo celular e construção do filme. A interface do programa é acessível para qualquer tipo de utilizador, independentemente das suas habilidades computacionais.[29]

3.11 Molyso

O software *molyso* resulta da combinação da tecnologia microfluídica de laboratório em chip com imagens de células vivas que possibilita a observação de células isoladas em seu contexto espaço/temporal. *Molyso* define-se como um software “máquina mãe” que permite a análise do seguimento temporal e automático de diversas imagens celulares. Este software contém diversas etapas nomeadamente: registo de imagem, correção da orientação, deteção de células, seguimento celular e visualização de resultados.[30]

Capítulo 4. Metodologia implementada

Neste capítulo descrevem-se os procedimentos desenvolvidos para a melhoria do software *schnitzcells*, uma vez que utilizou-se este software para a criação da metodologia que será apresentada de seguida.

Deste modo, a metodologia implementada encontra-se dividida em cinco etapas:

- Pré-processamento;
- Segmentação de imagens;
- Pré-processamento para o seguimento celular;
- Seguimento celular;
- Visualização dos resultados do seguimento celular (gráfico em árvore).

4.1 Pré-processamento

As imagens originais fornecidas pelo software *schnitzcells* mostravam qualidade reduzida, visto que apresentavam ruído, sendo a nitidez das imagens também reduzida, pois as mesmas encontravam-se escuras, como visualiza-se na Figura 7. Deste modo, tornou-se necessário a aplicação do filtro mediana para diminuir o ruído presente nas imagens de forma a melhorar a qualidade das mesmas.



Figura 7. Imagem original com a aplicação da função *imadjust*.

4.2 Segmentação de imagens

Após a melhoria da qualidade das imagens procedeu-se ao método da segmentação que consiste na divisão da imagem em partes que sejam semelhantes entre si, ou seja, os pixels devem-se agrupar de acordo com características similares como a intensidade, a forma, a textura entre outras. Neste caso, as imagens separaram-se em duas partes diferentes, ficando o fundo com a cor preta e os objetos (células) com a cor branca. Deste modo, aplicou-se o filtro denominado de laplaciano do gaussiano (LOG) que para além de mostrar os contornos das imagens ainda possibilita a eliminação de ruído que permaneceu após a utilização do filtro mediana. De seguida, utilizaram-se operações morfológicas com o intuito de melhorar os resultados obtidos no processo de segmentação. Primeiramente aplicou-se a operação morfológica denominada de dilatação que tem como função garantir que as extremidades das células continuam incluídas nas células, visto que a forma dos objetos é recuperada por esta operação. Seguidamente utilizou-se a abertura que consiste numa operação morfológica que tende a abrir pequenos espaços entre objetos próximos e suaviza o contorno na imagem. A operação de dilatação foi aplicada novamente, mas somente para células que possuam número mínimo de pixels inferiores a trezentos, pois este parâmetro implementou-se através de estudos feitos nas células *E.coli* e *Bacillus subtilis*. Seguidamente, determinou-se a distância ao centro das imagens de modo a manter o conjunto de células o mais próximo do centro de cada imagem. De modo, a melhorar ainda mais a segmentação utilizou-se novamente o filtro *log*, mas aplicou-se um limite para mais facilmente encontrar as células nas imagens. Implementou-se a função *imfill* para preencher os espaços que permaneceram nas células após a aplicação do filtro *log*. De modo, a identificar cada célula presente nas imagens procedeu-se à aplicação do comando *regionprops* e obtiveram-se diversos parâmetros nomeadamente a área, o centróide, o comprimento do eixo maior e a área convexa. Analisaram-se estas características com o intuito de considerar os objetos que não são considerados células sejam eliminados das imagens. Fizeram-se diversos testes para quebrar

grandes células, eliminar ângulos afiados ao longo das células e eliminar células pequenas.

Assim, a etapa da segmentação encontra-se finalizada obtendo-se as imagens com as regiões isoladas (células), uniformes e homogêneas, como verifica-se na Figura 8.

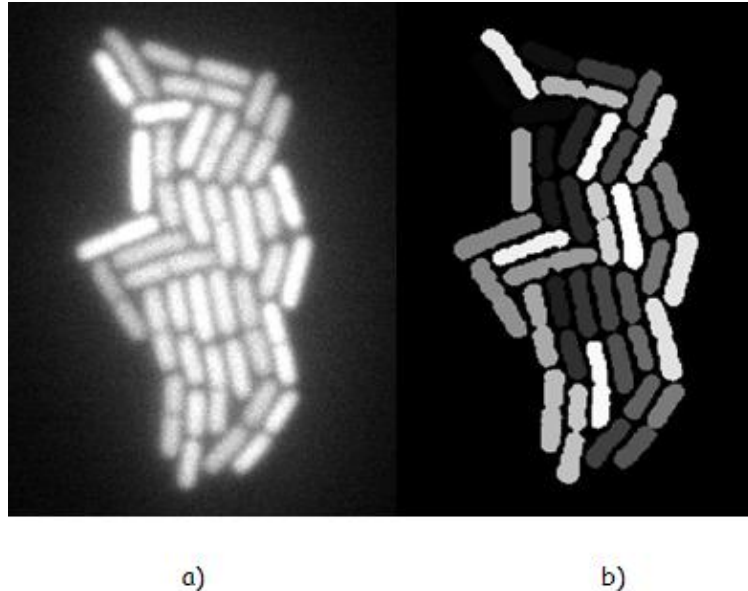


Figura 8. a) Imagem original. b) Imagem resultante da segmentação.

4.3 Pré-processamento para o seguimento celular

Nesta etapa implementaram-se procedimentos que preparam as imagens para a próxima fase denominada de seguimento celular e nesta etapa compararam-se as imagens aos pares, movendo-se os centróides dos objetos que constituem a imagem seguinte para os centróides dos objetos que constituem a imagem anterior de modo a obter-se um seguimento correto das células.

Inicialmente aplicaram-se operações morfológicas designadas de erosão e dilatação, de modo a acelerar o processo de divisão celular, uma vez que através da erosão reduziu-se o tamanho das células e a através da dilatação recuperou-se a forma das mesmas. Como existiam células que encontravam-se em processo de divisão, ou seja, unidas simplesmente por um pixel com este conjunto de operações dividiram-se mais rapidamente, como observa-se na Figura 9.

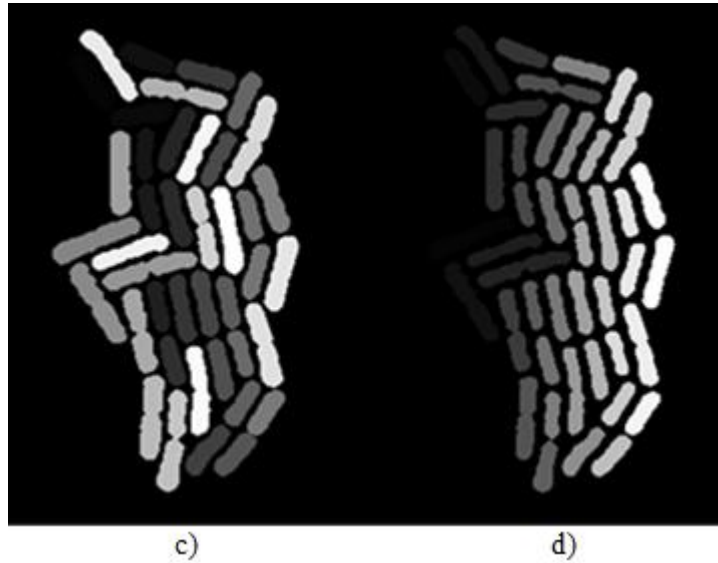


Figura 9. c) Imagem resultante da segmentação. d) Imagem obtida através das operações morfológicas.

Como detetou-se que o comando *regionprops* e a função *bwconncomp* apresentavam resultados diferentes na identificação dos objetos, recorreu-se a um ciclo *for para* que os números de objetos obtidos nos dois comandos fossem iguais. De seguida, retirou-se as coordenadas dos centróides dos objetos na primeira imagem e numerou-se as células de acordo com a ordem de contagem de objetos do comando *bwconncomp*. A primeira imagem tem como função a iniciação e controlo do seguimento celular como visualiza-se na Figura 10, visto que as restantes imagens são orientadas por esta, uma vez que o centro da primeira imagem iguala-se ao centro das imagens seguintes.

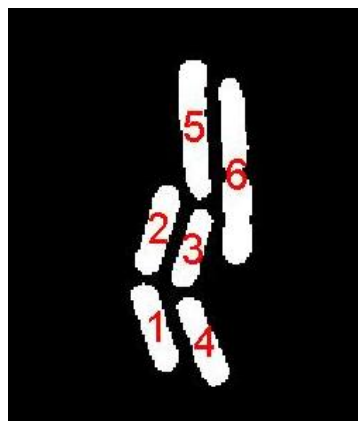


Figura 10. Primeira imagem com as células identificadas (controlo do seguimento celular).

Seguidamente representa-se a relação inicial entre as imagens consecutivas com o objetivo de perceber se os centros das imagens são coincidentes de modo a proceder-se ao seguimento das células. No entanto, os centros das imagens diferem significativamente, uma vez que comparou-se a imagem anterior com a imagem seguinte através da função *imoverlay* que possibilita a visualização dos contornos das duas imagens em simultâneo. Como exemplos dos centros entre duas imagens em sequência apresentarem-se diferentes, na primeira iteração que consiste na comparação do centro da primeira imagem com o centro da segunda imagem (Figura 11 e)) e na oitava iteração comparou-se os centros entre a oitava imagem e a nona imagem (Figura 11 f)) e verificou-se que os mesmos não são coincidentes.

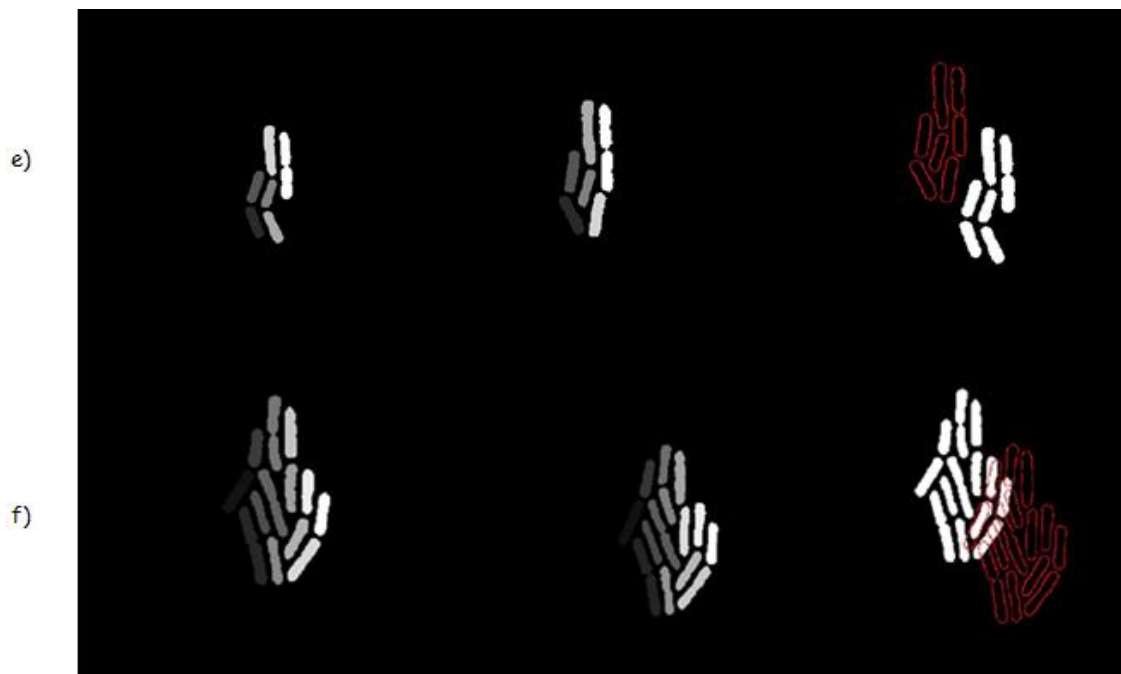


Figura 11. e) Exemplo da 1ª iteração que representa a primeira imagem, a segunda imagem e a sobreposição dos centros desalinhados através da função *imoverlay*, respetivamente. f) Exemplo da 8ª iteração que representa a oitava imagem, a nona imagem e a sobreposição dos centros desalinhados através da função *imoverlay*, respetivamente.

De modo que os centros das imagens coincidissem criou-se um vetor de translação que tem como função a obtenção de coordenadas dos centros de duas imagens consecutivas de modo a deslocar-se os centros das células da imagem seguinte para os centros das células da imagem anterior.

Primeiramente aplicou-se a função *bwlabel* que determina o número de objetos presentes numa imagem binária. Para obter-se as coordenadas dos centros criou-se um vetor que armazena os valores encontrados na pesquisa das imagens de valores inferiores ou iguais ao número de células presentes nas imagens e valores superiores ou iguais ao valor um que representa as células a branco, uma vez que no início do processo os objetos com a cor branca foram colocados com o índice de valor igual a um de forma a facilitar a obtenção das coordenadas dos centros das imagens. Testou-se também nas imagens se os valores eram superiores a um, visto que na primeira imagem os índices das células não foram alterados e por isso as células estão identificadas com índices de acordo com a numeração da função *bwconncomp*, ou seja, o conjunto de células da primeira imagem apresenta as células numeradas desde do valor um até seis. Desta forma, calculou-se a soma entre o máximo linha e o mínimo linha e a soma entre o máximo coluna e o mínimo coluna e dividiu-se as somas por o número dois, obtendo-se a coordenada linha e a coordenada coluna das duas imagens em sequência, respetivamente. Para deslocar os centros das células da imagem seguinte para os centros das células na imagem anterior criou-se um vetor de translação que consiste na subtração das coordenadas linha e coluna da imagem anterior com a imagem seguinte. De seguida, corrigiu-se as coordenadas dos centróides na imagem seguinte em que as coordenadas dos centróides obtidas pela função *regionprops* armazenaram-se num vetor com duas colunas. Primeiramente alterou-se a primeira coluna através da soma da mesma com o vetor translação coluna e corrigiu-se a segunda coluna através da soma da segunda coluna do vetor com o vetor translação linha, ou seja, na primeira iteração deslocou-se os centróides dos objetos da segunda imagem para os centros dos objetos da primeira imagem resultando em centros com as células sobrepostas. Com o intuito de perceber-se a deslocação de centros recorreu-se à função *imoverlay* para visualizar os contornos das imagens em simultâneo e verificar-se a coincidência dos centros. Deste modo, criou-se uma imagem com o tamanho da imagem seguinte e constituída por duas colunas, percorreu-se as células da imagem e obteve-se os pixéis de todas as células e realizou-se a operação de soma entre a primeira coluna das coordenadas de pixéis com o vetor translação coluna e a soma da segunda coluna das coordenadas de pixéis e o vetor translação linha.

E de seguida coloca-se as novas coordenadas na imagem criada no início e desta forma aplica-se a função *imoverlay* e desta forma verifica-se a concordância entre centros como observa-se na Figura 12.

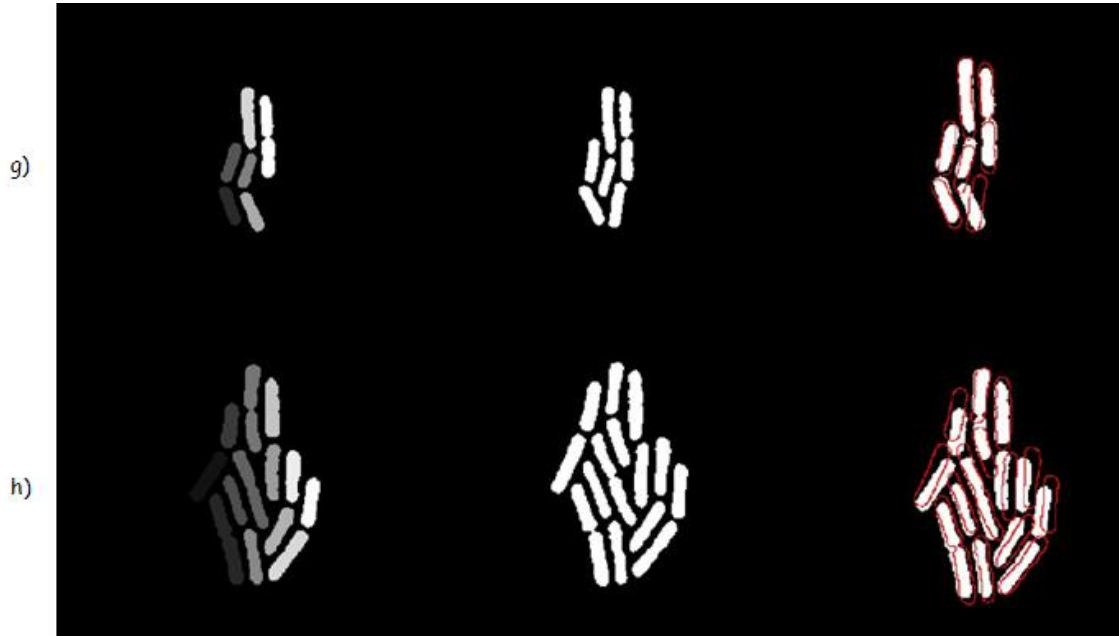


Figura 12.g) Primeira imagem, segunda imagem e sobreposição dos centros alinhados, respectivamente, deslocação do centro da segunda imagem para o centro da primeira imagem através do vetor de translação. h) Oitava imagem, nona imagem e sobreposição dos centros alinhados através do vetor de translação, respectivamente, deslocação do centro da nona imagem para o centro da oitava imagem.

4.4 Seguimento celular

Após a deslocação dos centróides das células procedeu-se ao seguimento celular e esta etapa consiste na compreensão da origem de cada célula desde a primeira imagem até à última, identificando as divisões que as mesmas sofrem ao longo do seu crescimento e por fim obteve-se um histórico da evolução celular.

Primeiramente, verificou-se se o centróide dos objectos da imagem seguinte encontrava-se no interior de uma célula da imagem anterior. Deste modo, numa sequência de duas imagens percorreram-se as células da imagem seguinte e da imagem anterior e armazenaram-se as coordenadas dos pixels das células da imagem anterior. Procedeu-se a um teste em que verificou-se se a primeira coluna das coordenadas dos centróides de todos os objetos na imagem seguinte continha valores iguais à primeira coluna das coordenadas dos pixels de todas as células na

imagem anterior. Averiguou-se também se a segunda coluna das coordenadas dos centróides de todas as células na imagem seguinte incluía valores iguais à segunda coluna das coordenadas dos pixels de todas as células na imagem anterior. No caso em que as células na imagem anterior e na imagem seguinte possuíam as mesmas coordenadas atribuiu-se o número da célula correspondente na imagem anterior. Como verificou-se que existiam objetos na imagem seguinte em que o centróide não se encontrava no interior das células da imagem anterior tornou-se necessário calcular a distância entre os objetos, uma vez que em algumas imagens existem células que não estão sobrepostas, como mostram as Figuras 13 e 14.



Figura 13.Sobreposição entre as imagens 6 e 5.



Figura 14.Sobreposição das imagens 8 e 7.

Desta forma testou-se se existiam células que não estavam a ser seguidas, ou seja, verificou-se se a terceira coluna da imagem seguinte encontrava-se a zero e guardou-se a informação, sendo que a terceira coluna corresponde à identificação das células no seguimento. Percorreu-se os objetos da imagem seguinte em que o centróide das células não se encontravam no interior das células da imagem anterior, ou seja, a terceira coluna que continha zeros. Também obteve-se as coordenadas dos pixels das células da imagem anterior para utilizar a função *euclidean distance* para calcular a distância entre objetos, ou seja, a distância entre as células que encontravam-se na terceira coluna com o valor zero e as células da imagem anterior. Deste modo, calculou-se a distância mínima entre as células para atribuir-se o número da célula mais próxima ao seguimento.

No entanto, ao executar o programa constatou-se que o seguimento não estava correto para todas as imagens, como verifica-se na Figura 15.

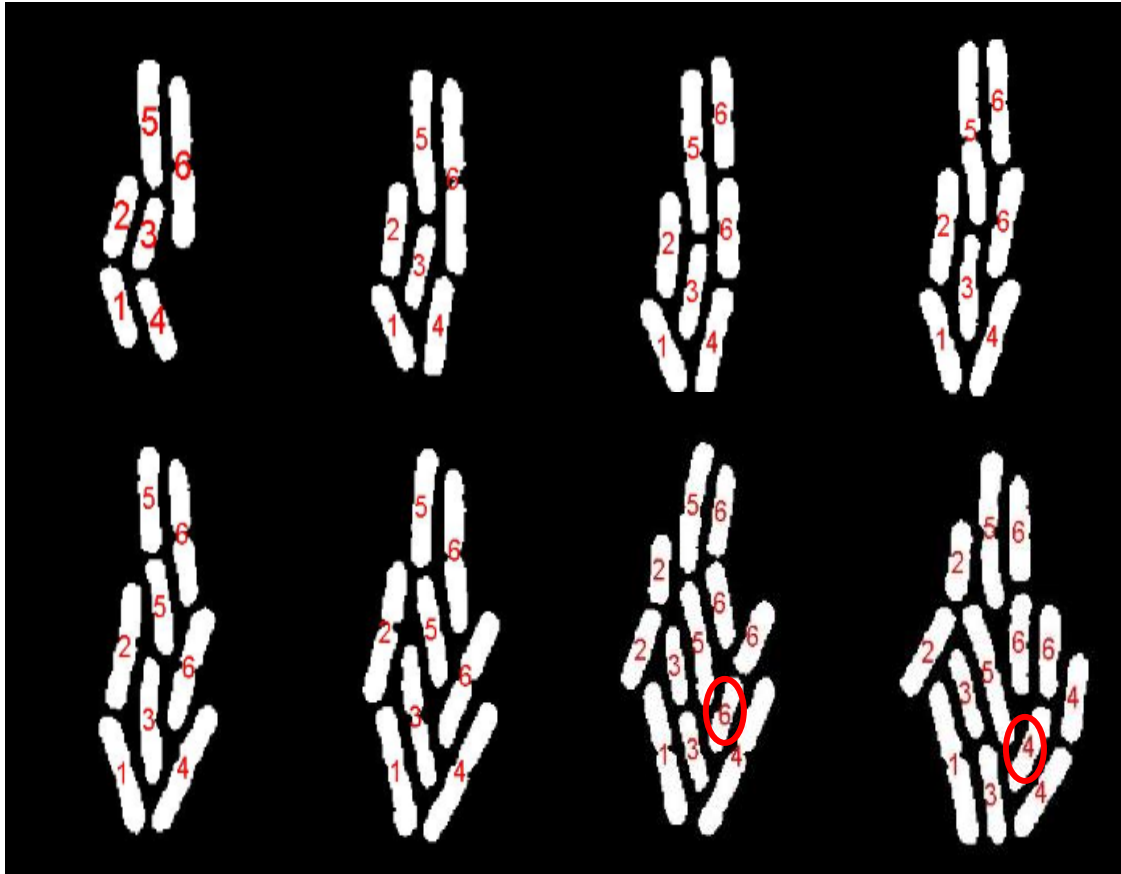


Figura 15. Visualização de uma sequência de imagens (imagem 1 até à imagem 8) em que ocorre um erro no seguimento.

Deste modo, tornou-se necessário alterar o vetor translação para todas as imagens, visto que como as células encontravam-se em constante crescimento modificam as posições das células de uma imagem para outra. Por isso, adaptou-se o vetor de translação para cada imagem para seguidamente corrigir-se os centróides de todas as células nas imagens de maneira a que o seguimento celular apresenta-se resultados corretos. Como observa-se na Figura 15 em que ocorreu um erro de seguimento, uma vez que na imagem 7 a célula seis aparece com uma numeração diferente na imagem 8, pois a mesma surge com o número quatro o que aconteceu devido à deslocação ligeira das células representadas com o número quatro na imagem oito.

Por isso, detetou-se o primeiro erro no seguimento na sequência da imagem sete para a imagem oito, onde as imagens sofrem modificações nas posições das células. Desta forma, desenvolveu-se um teste para imagens que fossem constituídas por mais de dez objetos em que os resultados do vetor coluna ou do vetor linha (resultante da subtração das coordenadas dos centros da imagem anterior com a imagem seguinte) tivessem valor superior ao número dez, o vetor translação sofre uma alteração de redução de 95% o que significa que as células na imagem diminuiram em 95% (imagens 9, 10 e 11). Caso contrário o vetor de deslocação multiplica-se pelo valor de 1.3 (imagens 8 e 12) e desta forma o erro no seguimento ficou corrigido, como demonstra-se na Figura 16.

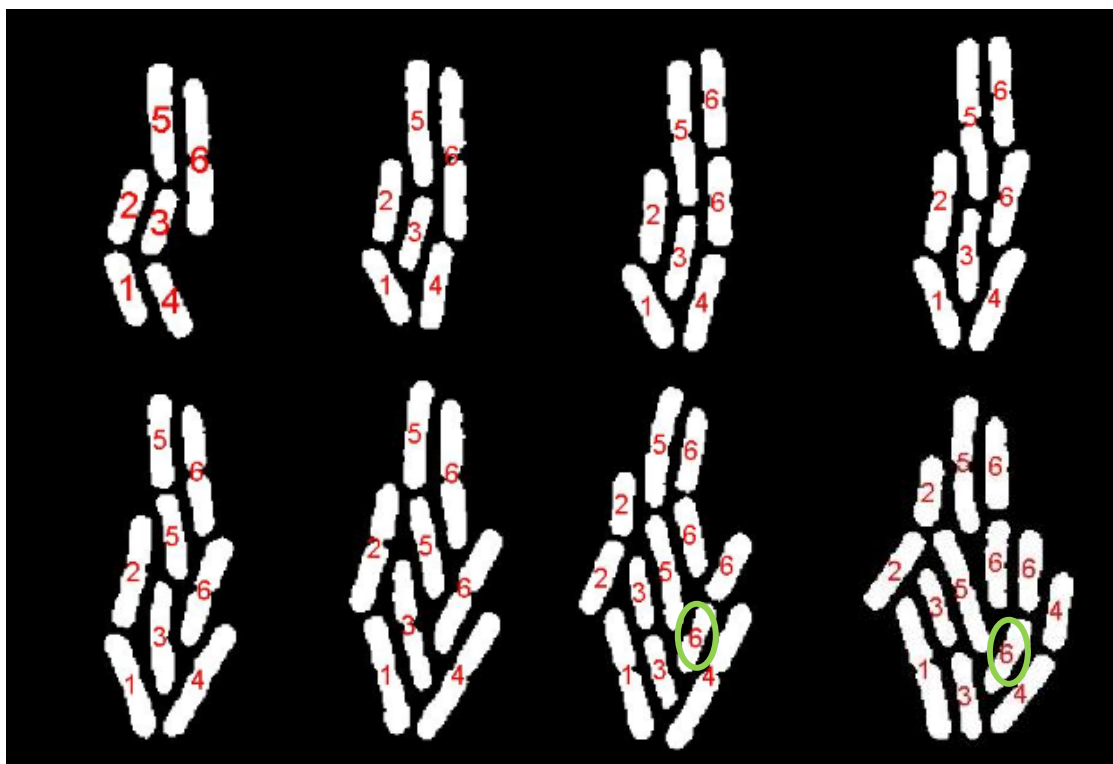


Figura 16. Visualização de uma sequência de imagens (imagem 1 até à imagem 8) com o erro de seguimento corrigido.

De seguida, alterou-se novamente o vetor translação para a sequência de imagens seguinte, ou seja, a partir da imagem 12, visto que as imagens posteriores apresentavam características diferentes e deste modo desenvolveu-se duas análises nomeadamente a expansão da imagem em largura e a expansão da imagem em altura, pois com o desenvolvimento das células as imagens sofreram alterações.

Para a análise em largura recorreu-se a uma primeira operação de subtração entre o máximo linha e o mínimo linha da imagem seguinte, a segunda subtração entre o máximo linha e o mínimo linha da imagem anterior e por fim subtraiu-se a primeira operação com a segunda. Este teste da largura tornou-se possível através dos resultados do teste feito para obter as coordenadas dos centros, ou seja, os valores armazenados no vetor quando nas imagens encontravam-se valores superiores ou iguais ao valor um que são relativos aos indexes dos objetos brancos e quando na imagem encontram-se valores inferiores ou iguais ao número de objetos obtidos através da função *bwlabel*. Deste modo, quando a subtração entre as duas operações de subtração apresentava resultado superior ao número dez a imagem aumentou em largura, mas caso o resultado da operação exibe-se valor inferior ao valor dez, então a imagem não sofreu alterações na largura. Em relação à análise em altura assemelha-se com a análise em largura, pois a primeira operação de subtração ocorreu entre o máximo coluna e o mínimo coluna da imagem seguinte, a segunda subtração entre o máximo coluna e o mínimo coluna da imagem anterior e por fim subtraiu-se a primeira operação de subtração à segunda. Realizou-se um teste em que no caso da operação de subtração final apresenta-se resultados superiores a zero significava que a imagem expandia em altura em caso contrário não existem alterações na imagem a nível da altura. As modificações no vetor de translação foram necessárias em cada imagem, uma vez que continuavam a surgir erros no seguimento como verifica-se na Figura 17, pois não alterou-se o vetor referente à imagem 14 e deste modo o seguimento apresentou-se incorreto.

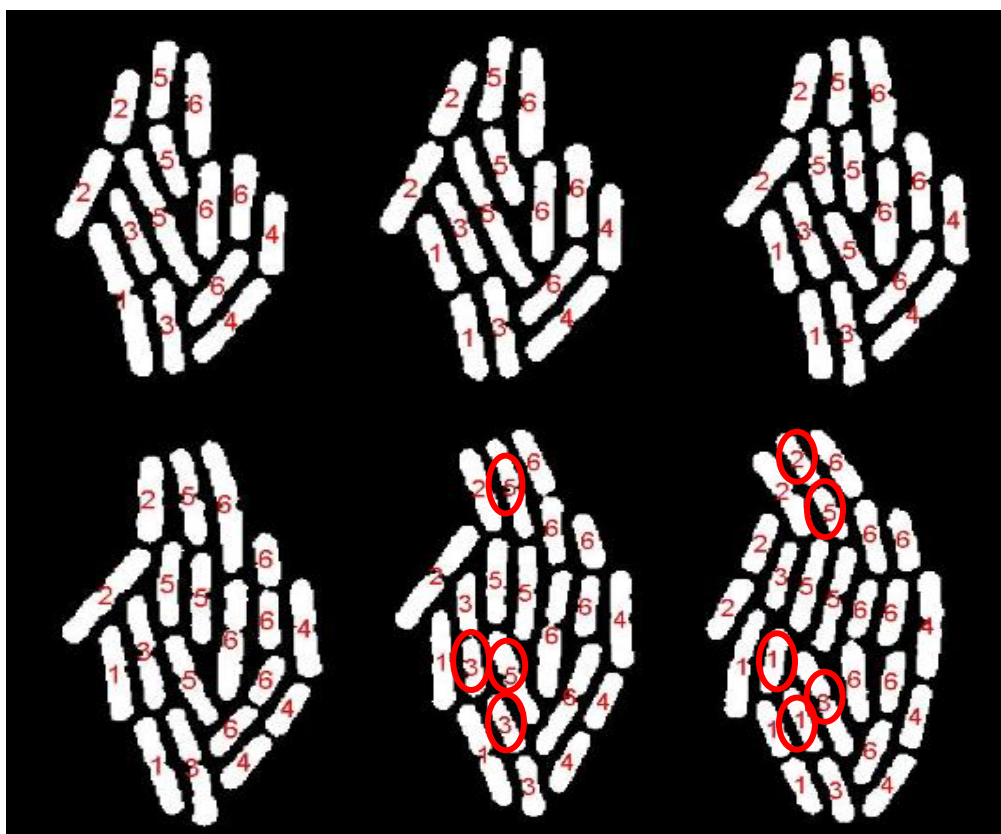


Figura 17. Visualização de uma sequência de imagens (imagem 9 até á imagem 14) em que deteta-se falhas no seguimento.

Para imagens constituídas por um número igual ou superior a dezoito células a análise de altura e largura aplicou-se, uma vez que as imagens expandem em largura ou em altura. Deste modo, o vetor translação alterou-se dependendo das hipóteses mencionadas acima, quando testou-se no caso de os objetos na imagem expandirem em largura ocorreu uma operação de divisão do vetor deslocação por o valor de 1.8 em que o mesmo reduziu-se quase para metade do valor original. Sendo que os centróides de todas as células na imagem 13 diminuíram quase para metade do valor inicial. No caso em que os objetos na imagem aumentarem em altura a variável que contém o vetor linha de deslocação alterou-se para negativo, reduziu-se o mesmo em um terço do seu valor original e ao resultado final adicionou-se sete unidades. A variável que contém o vetor coluna de deslocação alterou-se para negativo e dividiu-se o mesmo pelo valor três e aos dois vetores subtraiu-se cinco unidades e deste modo a falha apresentada na imagem 14 (Figura 17) ficou resolvida.

Sendo que dependendo do valor do vetor translação inicial os centróides na imagem deslocaram-se em diferentes direções a nível das linhas que podem ter-se deslocado para a esquerda ou para a direita e a nível das colunas para cima ou para baixo.



Figura 18. Visualização da sequência de imagens (imagem 9 até á imagem 14) com o seguimento corrigido.

Desta forma, com as modificações no vetor translação na imagem 14 constatou-se que o seguimento celular encontrava-se correto como se verifica na Figura 18. Também é notório que existia a necessidade de ajustar o vetor translação para cada imagem, visto que as células nas imagens 13 e 14 quando sobrepostas apresentavam-se ligeiramente deslocadas, como se observa na Figura 19 a) e após a correção do vetor e o deslocamento de todos os centróides das células da imagem 14 o grau de sobreposição

entre as células tornou-se mais elevado (Figura 19 b)), proporcionando um seguimento celular correto.

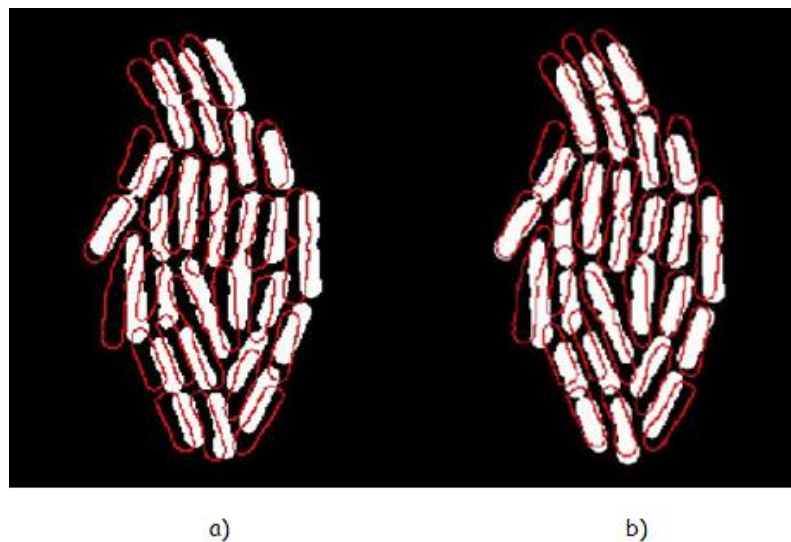


Figura 19. a) Sobreposição da imagem 14 com a imagem 13 através da função *imoverlay* sem alterações ao vetor translação. b) Sobreposição da imagem 14 com a imagem 13 através da função *imoverlay* com modificações no vetor translação.

Deste modo, segue-se mais um exemplo de um seguimento incorreto quando o vetor translação não sofreu modificações de uma imagem para outra, como visualiza-se na Figura 20. Verifica-se a necessidade de adaptar o vetor a cada imagem de modo a obter-se um seguimento assertivo.

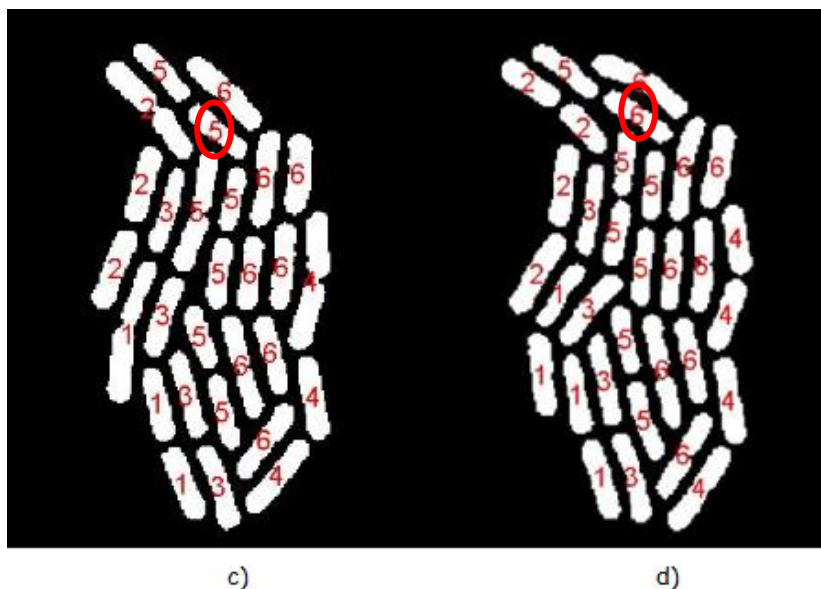


Figura 20. c) Imagem 15 com o seguimento celular correto. d) Imagem 16 com o seguimento celular incorreto.

Seguidamente desenvolveu-se mais um teste para imagens com um valor igual ou superior a vinte e cinco objetos e analisou-se duas hipóteses, sendo a primeira expansão dos objetos na imagem em altura (imagem 16) e a segunda os objetos nas imagens amplificaram em largura e em altura (imagem 15 e imagens 17 e 18). Para a expansão em altura altera-se o sinal do vetor translação linha para negativo e reduziu-se um terço do mesmo e por fim subtraiu-se três unidades e ao vetor translação coluna modifica-se o sinal para negativo e diminui-se o vetor em um terço do seu valor original que aplicou-se na imagem da Figura 21 f), ou seja, em todas as células alterou-se as coordenadas dos centróides de acordo com os resultados da correção do vetor translação. Quanto à segunda hipótese que consiste em os objetos nas imagens expandirem-se em altura e em largura o vetor translação linha diminui-se para metade e subtraiu-se duas unidades e meia enquanto no vetor translação coluna também reduziu-se para metade e subtraiu-se uma unidade que aplicou-se na Figura 21 e), alterando todas as coordenadas dos centróides nas células da imagem.

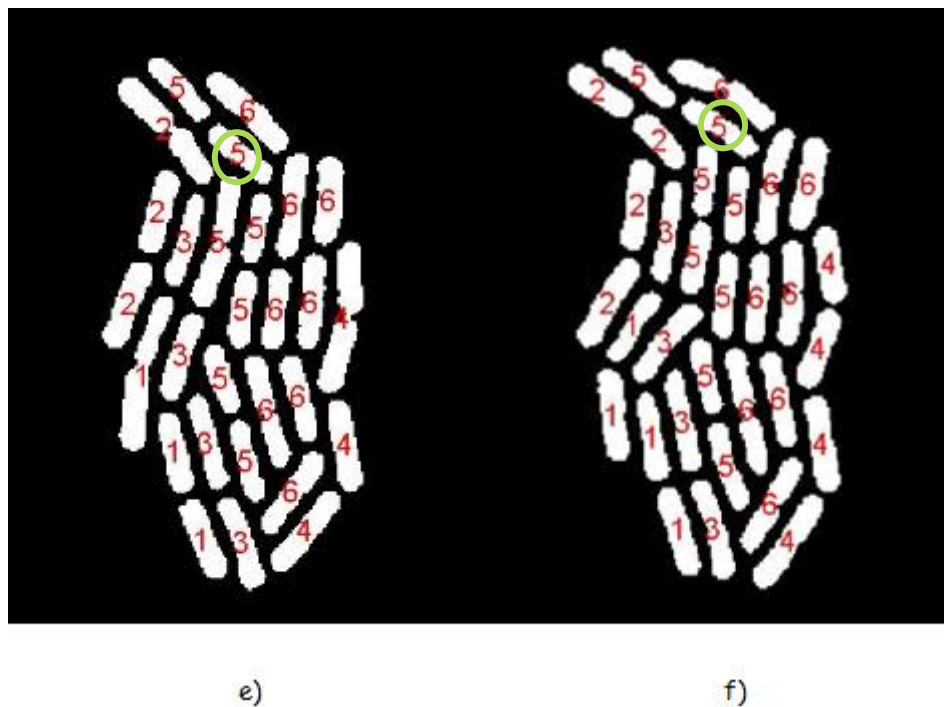


Figura 21. e) Imagem 15. f) Imagem 16 com o seguimento correto após modificação ao vetor translação.

Segue-se mais um exemplo de seguimento incorreto, uma vez que o vetor translação não foi alterado e deste modo obteve-se a imagem 19 com seguimento incorreto como verifica-se na Figura 22.

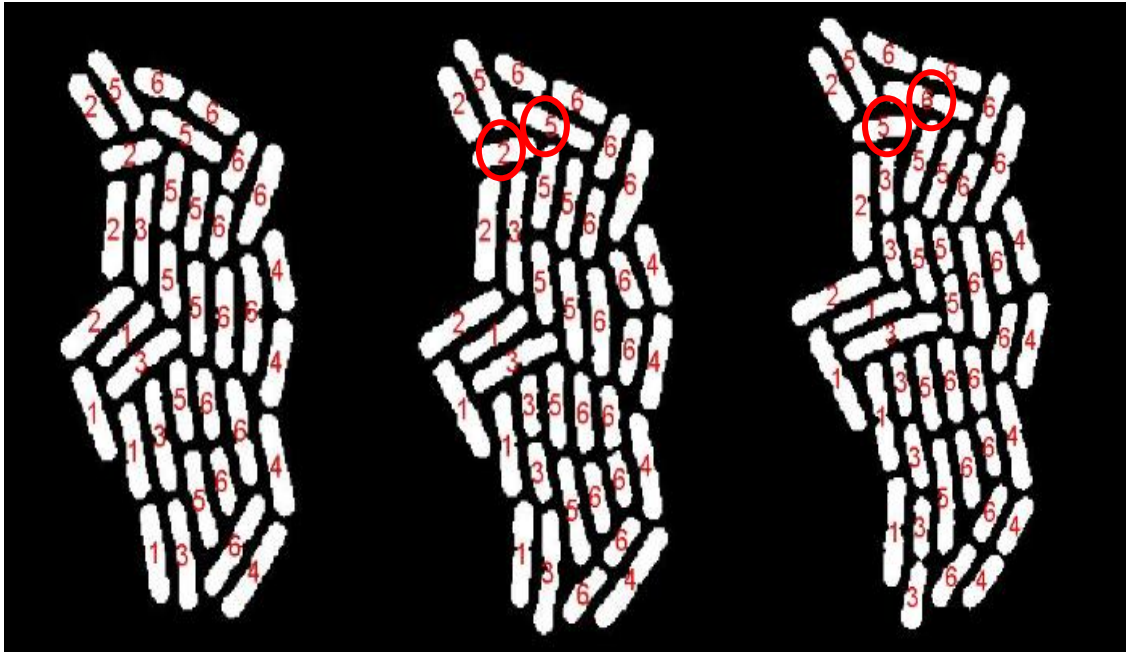


Figura 22. Representação de uma sequência de imagens desde da imagem 17 até à imagem 19, onde observa-se um erro no seguimento.

Desta forma, realizou-se mais um teste para imagens com um número de objetos igual ou superior a trinta e nove objetos (imagem 19) e em que os mesmos amplificaram em largura e em altura de modo que o vetor translação linha diminui-se para metade do seu valor original e adicionou-se três unidades enquanto ao vetor translação coluna também reduziu-se para metade e adicionou-se ao resultado final do vetor translação linha e coluna 1.4.

Assim, o seguimento na imagem 19 aparece corretamente, uma vez que acertou-se os valores das coordenadas dos centróides de todas as células de acordo com as modificações no vetor translação, como se observa na Figura 23.



Figura 23. Representação de uma sequência de imagens desde da imagem 17 até à imagem 19 com o seguimento corrigido.

Por fim apresenta-se mais um erro no seguimento, onde não adaptou-se o vetor translação na imagem 20 e deste modo a célula seis na imagem 19 encontra-se identificada com o número seis enquanto na imagem 20 a mesma célula encontra-se numerada com o número quatro, como analisa-se na Figura 24.

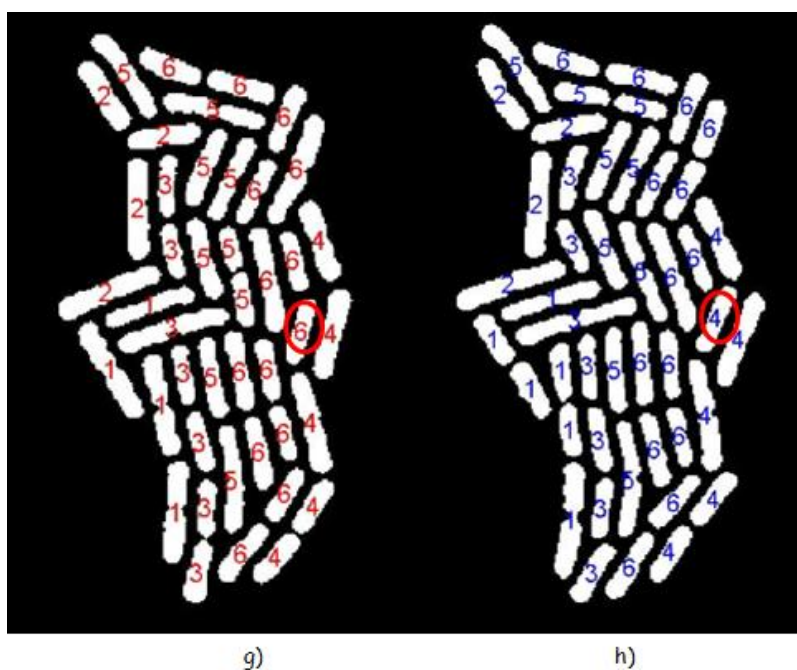


Figura 24. g) Imagem 19 com seguimento correto. h) Imagem 20, onde detetou-se um erro no seguimento.

Mais uma vez desenvolveu-se um teste para imagens que possuíam na sua constituição o número de objetos igual ou superior a quarenta e dois objetos e os mesmos expandiam em altura e em largura na imagem 20. Deste modo, o vetor translação linha diminuiu-se para metade e adiciona-se duas unidades enquanto o vetor translação coluna reduziu-se para metade e subtraiu-se 0.5 ao vetor translação linha e coluna e assim modificou-se todos os centróides das células de acordo com a correção do vetor translação. Com as alterações obteve-se um seguimento correto, visto que a célula número seis na imagem 19 mantém a identificação seis na imagem 20, como verifica-se na Figura 25.

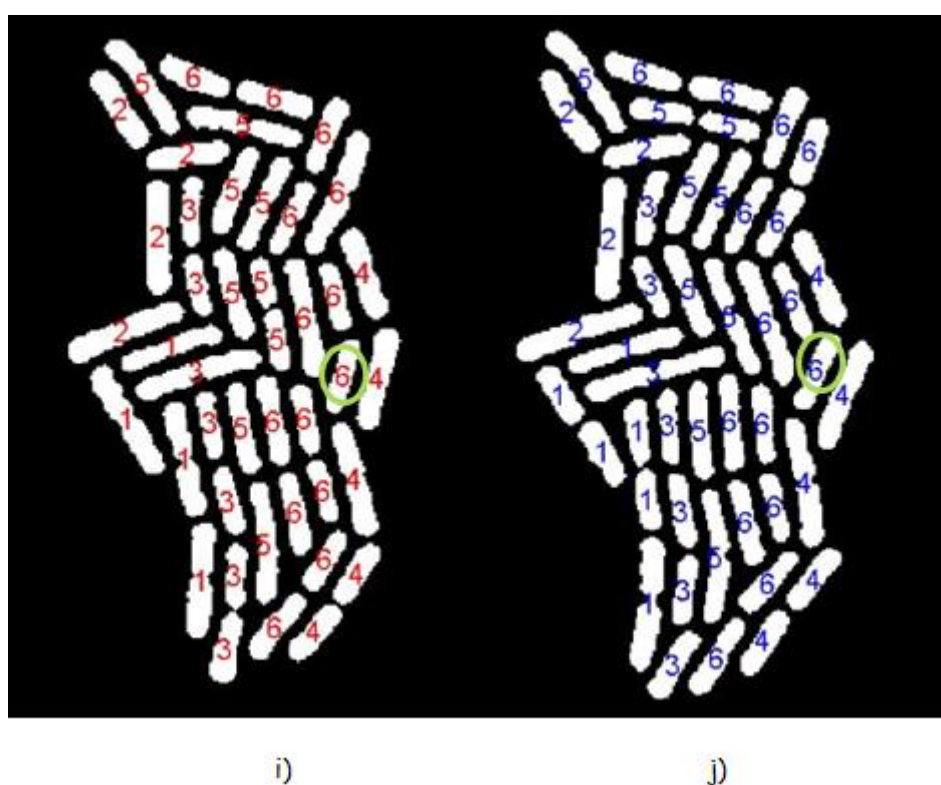


Figura 25. i) Imagem 19 com o seguimento correto. j) Imagem 20 com o seguimento corrigido.

Assim, com as modificações ao vetor translação adaptando o mesmo a cada imagem obteve-se um seguimento correto desde da primeira até á ultima imagem (pode-se observar todas as imagens disponíveis nos anexos). No fim do programa obtêm-se as imagens em que ocorreram as divisões, o número de células presente em cada imagem e a descendência das células.

4.5 Gráfico em árvore dos resultados obtidos no seguimento celular

O gráfico em árvore tem como função a demonstração do seguimento das células, pois deste modo visualiza-se as divisões celulares e a origem de cada célula, identificando-se as células mães e as células filhas. A elaboração deste gráfico teve de ser feita manualmente, uma vez que produzir o mesmo enquanto o programa executava não resultava, pois ocorriam muitos erros e desta forma não se obtinha um gráfico com os resultados do seguimento correto. Para fazer o gráfico pesquisou-se uma função no MATLAB denominada de *treeplot* que produz o gráfico em árvore (Figura 26), no entanto tornou-se necessário criar um vetor manual, visto que a maneira como numera os nós não resultaria para o seguimento.

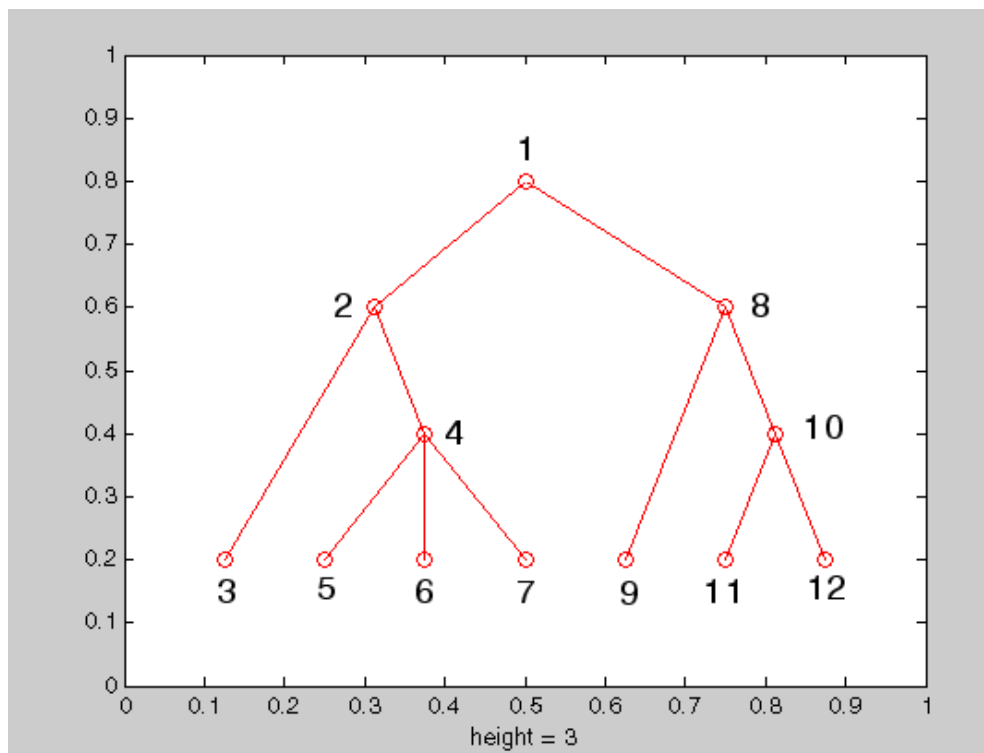


Figura 26. Visualização de um exemplo de gráfico em árvore no MATLAB.

Como verifica-se na Figura 26 a numeração atribuída aos nós é um pouco aleatória e para os resultados deste seguimento necessita-se de seis nós iniciais (que correspondem às seis células da primeira imagem) e deste modo não só aplicando esta função não resultaria, pois precisava-se de seis nós em linha.

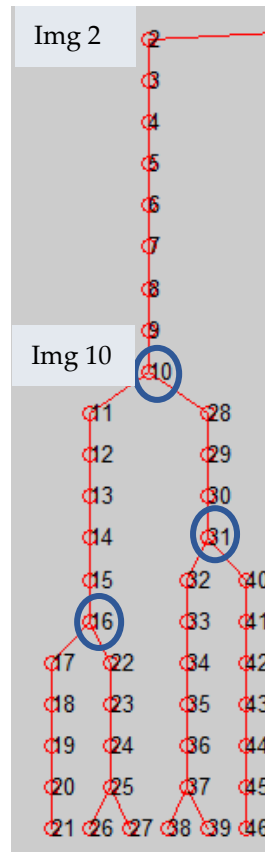


Figura 27. Gráfico em árvore (seguimento célula 1).

Assim, criou-se um vetor manual colocando os números nas posições do vetor de modo a que os resultados no gráfico de árvore fossem concordantes com os resultados do seguimento obtidos através do trabalho desenvolvido e assim numerou-se por ordem crescente os nós na vertical, por exemplo a célula 1 (primeira imagem) encontra-se numerada desde do nó 2 até ao nó 46, como se observa na Figura 27.

nodes = [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 16 22 23 24 **25 25** ...
10 28 29 30 31 32 33 34 35 36 **37 37** 31 40 41 42 43 44 45...

Deste modo, primeiramente numerou-se o nó 2 correspondente à primeira célula, sendo que o primeiro nó é considerado zero (sem mãe), pois é a partir deste que se obtém os outros nós, ou seja é o nó de raiz. Percebe-se que o nó 3 encontra-se ligado ao nó dois, que o nó 5 encontra-se ligado ao nó 4 o que significa que desde a imagem 1 até à imagem 9 a célula 1 não sofreu divisão. No nó 10 ocorreu divisão (na imagem 10) e

primeiramente no vetor é demonstrado o seguimento da célula esquerda dividida e depois volta a posicionar-se o nó 10 e mostra-se o seguimento da célula do lado direito.

É de notar que no vetor quando ocorrem divisões o número da célula que sofre divisão repete-se duas vezes e caso tenha seguimento descreve-se primeiro o da célula originada na esquerda e depois o seguimento da célula originada na direita.

Após a criação do vetor aplicou-se a função *treelayout* em que tem como função guardar as coordenadas (x e y) para colocar os nós da árvore para criar uma figura organizada do gráfico. Seguidamente utilizou-se a função *treeplot* que traça uma imagem de árvore como verifica-se na Figura 28.

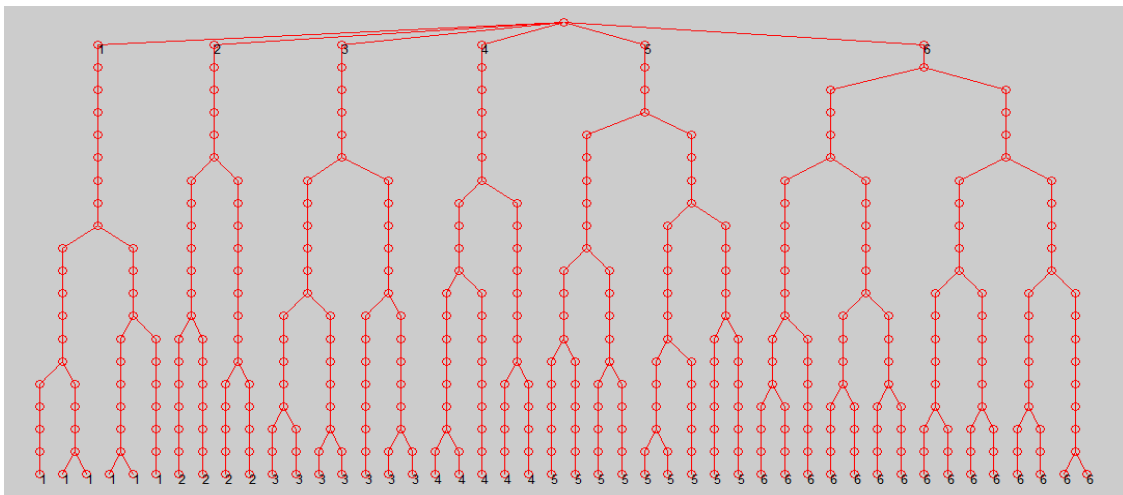


Figura 28. Gráfico em árvore.

Desta forma através do gráfico identificam-se perfeitamente as células mãe e as células filha, sendo que o nó 1 que corresponde à célula um da primeira imagem em que esta é a célula mãe e a mesma originou seis células. Em relação ao nó 2 que representa a célula dois (primeira imagem), a célula mãe dividiu-se originando quatro células. A célula três da primeira imagem (célula mãe) que encontra-se no nó 3 sofreu processos de divisão originando sete células. O nó quatro que corresponde à célula quatro (primeira imagem), célula mãe dividiu-se e originou cinco células. Em relação ao nó 5 que representa a célula cinco da primeira imagem (célula mãe) sofreu processos de

divisão e obteve-se nove células. Por fim, a célula seis da primeira imagem representada pelo nó 6 que é a célula mãe, multiplicou-se e originou quinze células.

De modo a perceber-se mais facilmente o vetor elaborado manualmente colocaram-se os números correspondentes em cada nó através das coordenadas guardadas no vetor obtido através da função *treelayout* e para este efeito percorreu-se o comprimento do vetor que contém a informação dos nós e desse modo identificou-se todos os nós, como se visualiza na Figura 29.

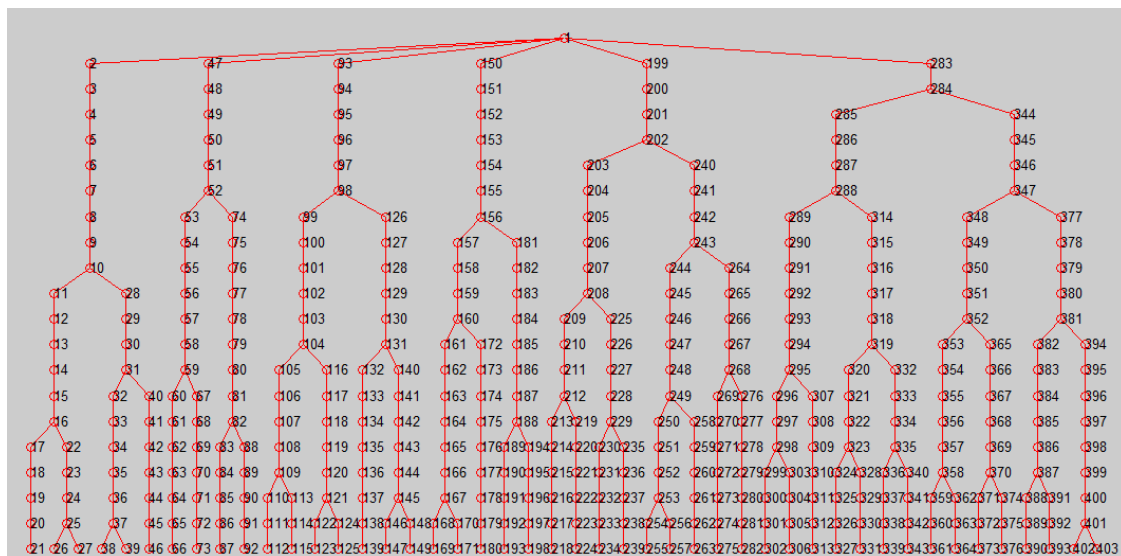


Figura 29. Gráfico em árvore com os nós numerados.

Assim, deste modo obteve-se o gráfico em árvore para visualizar o seguimento das células de forma mais organizada e perceber-se mais facilmente a evolução e o crescimento das células.

Capítulo 5. Análise de resultados

Na comparação do software schnitzcells original com a metodologia de estudo desenvolvida neste trabalho constata-se que as modificações implementadas contribuíram para resultados satisfatórios, começando pela acessibilidade à plataforma do trabalho que encontra-se facilitada. Enquanto no schnitzcells original copiava-se caminhos de diretoria através de um ficheiro fornecido pelo software para prosseguir para as fases seguintes do trabalho, nesta metodologia o programa fornece um ficheiro em que altera-se as diretorias de todas as etapas do mesmo e deste modo o programa executa até ao fim. Para além da modificação na plataforma de trabalho a metodologia implementada proporcionou melhores resultados a nível da segmentação como verifica-se na Figura 30.

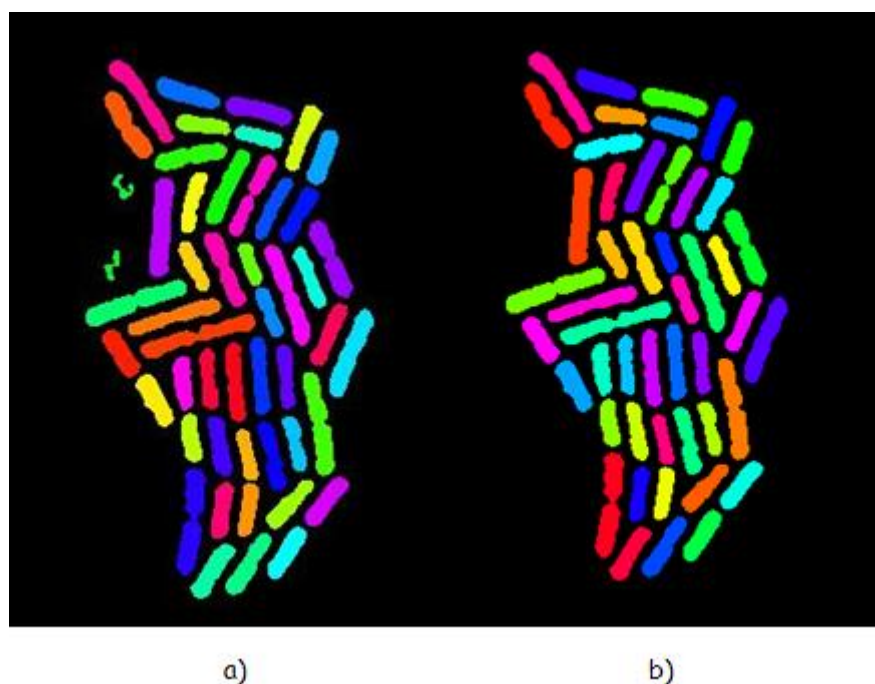


Figura 30. a) Imagem resultante da segmentação do schnitzcells original. b) Imagem resultante da segmentação da metodologia implementada.

Como verifica-se na Figura 30 b) os resultados obtidos na segmentação através do trabalho desenvolvido apresentaram melhores resultados do que os obtidos no software original. Enquanto na Figura 30 a) aparecem restos de células que não foram eliminados na totalidade, na Figura 30 b) esses pequenos fragmentos desapareceram alterando

apenas um valor numa parte da segmentação em que eliminavam-se células em que a sua área fosse menor do que cinquenta e esse mesmo valor alterou-se para setenta eliminando deste modo os fragmentos.

Para além da melhoria a nível da segmentação também obtiveram-se melhores resultados a nível do seguimento celular. Na Figura 31 demonstra-se um erro decorrido durante o seguimento, sendo que a célula com a cor branca representa a primeira célula da linhagem e as células com a cor amarela representam o seguimento da mesma célula. Deste modo, percebe-se que na imagem 6 a célula com a cor amarela aparece na célula errada, a falha contínua na imagem 7 e caso o erro permanecesse a célula seria seguida incorretamente.

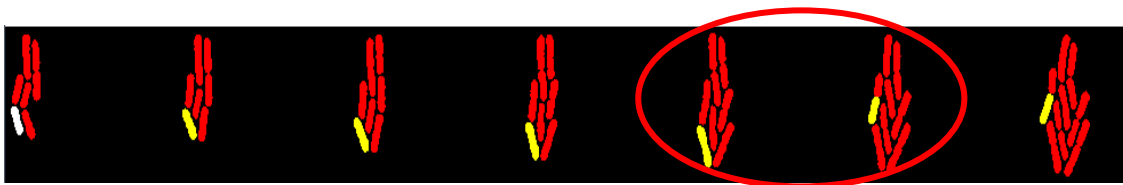


Figura 31. Representação em sequência desde da imagem 1 até à imagem 7 dos resultados obtidos do seguimento no software schnitzcells, onde se detetou um erro.

O programa schnitzcells resolve o problema manualmente no entanto no trabalho desenvolvido este tipo de erro não acontece, pois a célula identificada como a célula um (célula amarela na Figura 31) apresenta um seguimento correto desde a imagem 1 até à imagem 7, como se observa na Figura 32.

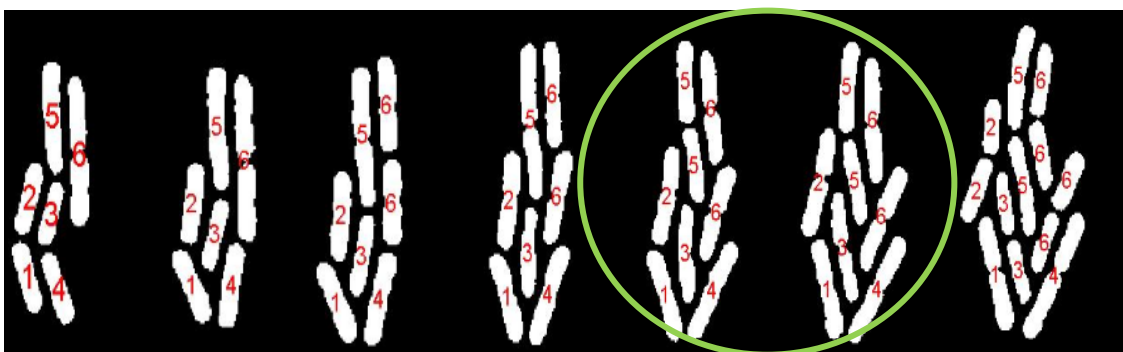


Figura 32. Sequência de imagens desde da imagem 1 até á imagem 7 dos resultados obtidos através da metodologia implementada.

Seguidamente detetou-se mais uma falha nos resultados do seguimento executado através do software schnitzcells. Neste caso, ocorreu num processo de divisão, ou seja, na imagem 5 a célula com a cor amarela representa a mãe das células que aparecem divididas na imagem 6 (Figura 33) que deveriam estar representadas com cores diferentes nomeadamente com as cores branca e lilás, pois desta forma seriam distinguidas das restantes células individuais.

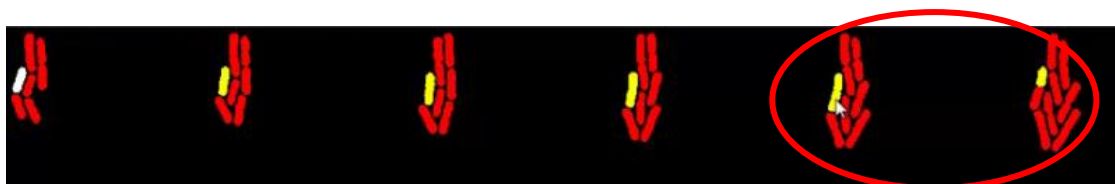


Figura 33. Representação da imagem 1 até á imagem 6 dos resultados obtidos do seguimento com o software schnitzcells.

O software schnitzcells mais uma vez corrige o erro no seguimento manualmente enquanto no trabalho desenvolvido tal não acontece, uma vez que a célula 2 (célula amarela na Figura 33) apresenta-se individualizada nas imagens 5 e 6, pois o processo de divisão que decorre corretamente só ocorrerá na frame seguinte (Figura 35) devido às operações morfológicas aplicadas após a segmentação, como visualiza-se na Figura 34.

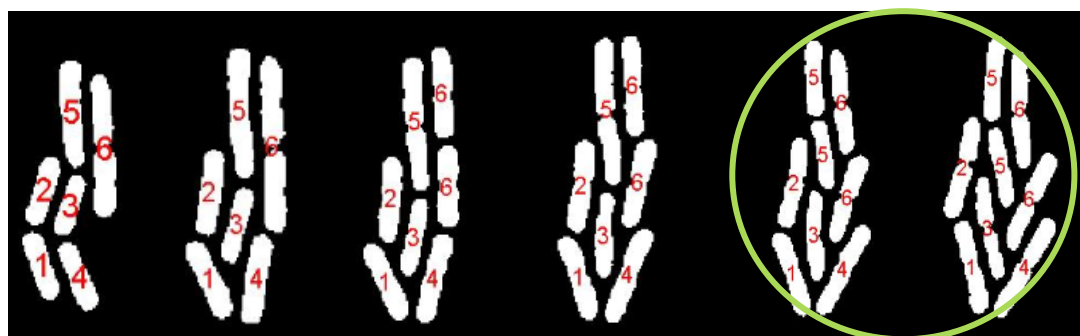


Figura 34. Sequência de imagens desde da imagem 1 até á imagem 6 dos resultados obtidos através da metodologia implementada.



Figura 35. Representação de imagens consecutivas (desde da imagem 1 até à imagem 7), onde verifica-se a divisão correta da célula número dois.

Detetou-se mais um erro ocorrido durante o processo de seguimento celular na execução do software schnitzcells em que na Figura 36 a célula representada com a cor branca localizada na frame 4 apresenta-se individualizada, na imagem 5 encontra-se dividida no entanto na imagem 6 a célula encontra-se novamente isolada o que não é possível, uma vez que o processo de divisão celular é irreversível.



Figura 36. Sequência da imagem 4 até à imagem 7, problema no seguimento executado pelo software schnitzcells.

Como verifica-se na Figura 37 a falha que surgiu no programa schnitzcells não acontece na metodologia desenvolvida, uma vez que na imagem 4 a célula 6 (célula amarela na Figura 36) encontra-se individualizada, nas imagens 5 e 6 a célula número seis continua na forma isolada e a célula seis só se divide na frame 7, mas não surge o problema de a célula seis estar na forma individual, depois na imagem precedente apresentar-se dividida e na imagem seguinte encontrar-se isolada novamente.



Figura 37. Representação em sequência desde a imagem 4 até à imagem 7 dos resultados obtidos na metodologia.

De seguida, apresenta-se mais uma falha a nível do processo de divisão celular no software schnitzcells, como observa-se na Figura 38.



Figura 38. Sequência de imagens desde da imagem 8 até á imagem 11, deteta-se erro no seguimento na execução do software schnitzcells.

O software schnitzcells corrige a falha manualmente enquanto na metodologia desenvolvida esse erro não surge, visto que as células apresentam-se corretamente seguidas. O seguimento de células obtido nos resultados no trabalho desenvolvido o erro não surge, pois a célula três (célula amarela na Figura 38) encontra-se seguida corretamente (Figura 39) e divide-se nas imagens seguintes, visto que aplicaram-se operações morfológicas após os resultados da segmentação.



Figura 39. Representação da sequência de imagens desde a imagem 8 até á imagem 11 com o seguimento correto.

Tabela 1. Comparação dos resultados do software schnitzcells com o método implementado.

	Schnitzcells	Método implementado
Tipo de programa	Manual e automático	Automático
Processo de segmentação	Mais lento e com correções manuais	Rápido e automático
Processo de seguimento	Mais lento e com correções manuais	Rápido e automático
Falhas no programa	Algumas falhas nos processos de segmentação e de seguimento	Não ocorrem falhas

Assim, constata-se que os resultados obtidos no trabalho desenvolvido tornaram o programa mais fácil para os utilizadores a nível de compreensão e de execução. Deste modo a metodologia implementada consiste num programa automatizado enquanto o software schnitzcells apesar de ser automático contém partes manuais, nomeadamente as correções da segmentação e do seguimento, como verifica-se na Tabela 1.

5.1 Comparação entre o método implementado e o software TLM-Tracker

O programa TLM-Tracker é um programa elaborado através da interface gráfica do *GUI* e em termos de processamento é semelhante à metodologia implementada que encontra-se dividido em quatro partes nomeadamente:

- Pré-processamento;
- Segmentação;
- Tracking;
- Exportação em forma de gráfico.

No pré-processamento o programa remove imagens que estão desfocadas e desta forma a sequência de imagens encontra-se preparada para o processo seguinte designado de segmentação, enquanto na metodologia implementada o pré-processamento consiste em retirar ruído nas imagens através da aplicação de filtros.

Relativamente ao processo de segmentação o software TLM-Tracker possui diversos processos para segmentar as imagens nomeadamente a função *watershed*, o comando *threshold* e a função *active contours* e também possibilita a aplicação de

operações morfológicas entre as quais a erosão. Enquanto no método implementado o processo de segmentação utiliza filtros, operações morfológicas e também existem processos para eliminação de objetos que não são considerados células. Deste modo, o TLM-Tracker possibilita mais opções para o utilizador, no entanto se o utilizador não compreender a linguagem do programa MATLAB torna-se mais difícil do que o método desenvolvido, uma vez que o mesmo apenas necessita de mudar as diretorias na função *main* que o programa executa até ao fim.

Em relação ao processo de seguimento celular o programa identifica as células em todas as imagens, fazendo a trajetória de cada célula e não necessita de pré-processamento como no caso da metodologia implementada em que primeiramente necessitou-se de criar um processo de modo a que todas as células nas imagens em sequência estivessem sobrepostas para prosseguir-se ao seguimento celular.

Por fim, o software TLM-Tracker exporta a informação do seguimento celular através de um gráfico em árvore como na metodologia desenvolvida. No entanto, no programa TLM-Tracker ocorre uma falha (Figura 40), uma vez que no gráfico aparecem duas células sobrepostas e a célula que encontra-se errada no “ramo” da árvore tem de ser eliminada. No método desenvolvido isso não acontece, uma vez que o gráfico em árvore também elaborou-se manualmente, enquanto no programa TLM-Tracker o programa produz o gráfico automaticamente.

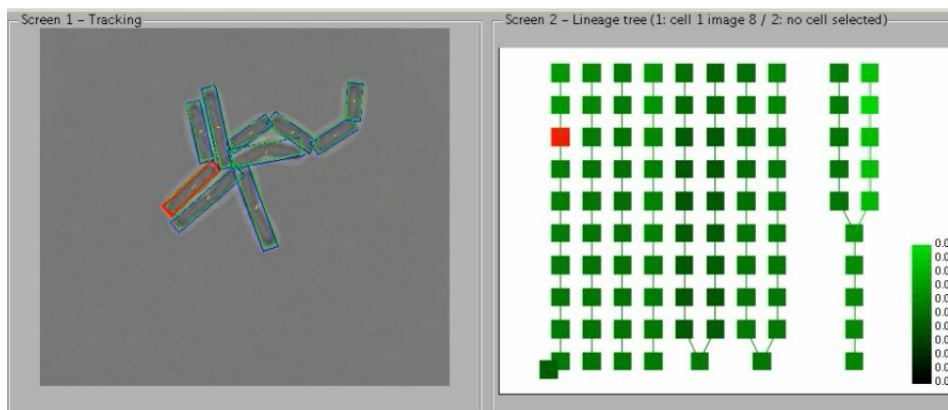


Figura 40. Visualização do gráfico obtido do software TLM-Tracker. [14].

Tabela 2. Comparação dos resultados do software TLM-Tracker com o método implementado.

	TLM-Tracker	Método implementado
Tipo de programa	Automático (interface <i>GUI</i>)	Automático
Processo de segmentação	Maior número de opções de escolha	Rápido e automático
Processo de seguimento	Não necessita de pré-processamento antes do seguimento.	Rápido e automático
Falhas no programa	Falha na execução do gráfico em árvore	Não ocorrem falhas

Assim, constata-se que o trabalho desenvolvido para o utilizador é mais fácil para compreender e executar, mesmo para pessoas que não entendam a linguagem do programa, enquanto o software TLM-Tracker apresenta maior número de opções de escolhas nas diversas etapas do programa, mas para utilizadores que não compreendam e não conheçam a linguagem MATLAB torna-se mais complicado. Deste modo, os dois programas funcionam automaticamente, sendo que no TLM-Tracker o utilizador tem a possibilidade de escolher como executar cada etapa do programa, como analisa-se na Tabela 2.

Capítulo 6. Conclusão

Concluindo, o objetivo principal deste trabalho consistia na melhoria do software *schnitzcells*, uma vez que o programa apesar de ser em formato automático contém correções manuais a nível da segmentação e do seguimento celular. Deste modo, para alterar as correções manuais para automáticas recorreu-se a diversas operações com o intuito de se obterem melhores resultados. Primeiramente modificou-se o processo de segmentação, visto que algumas imagens continham fragmentos de células que deveriam ser eliminadas. Desta forma, aumentou-se o valor da área de modo a que os restos fossem excluídos das imagens resultando em imagens limpas, simplesmente com o fundo e com os objetos considerados células pelo programa. Seguidamente as alterações ocorreram no processo de seguimento celular, visto que no programa original surgiram diversos erros que tiveram de ser corrigidos manualmente pelo utilizador. Deste modo, colocou-se a primeira imagem da sequência de vinte imagens como imagem de controlo do seguimento, de seguida calculou-se a distâncias entre os centros das imagens (aos pares) e criou-se um vetor translação que deslocou os centróides das células da imagem seguinte para os centróides das células da imagem anterior resultando em imagens com células sobrepostas. Com o intuito de se obterem resultados corretos no seguimento adaptou-se o vetor translação para cada imagem e seguidamente verificou-se os centróides dos objetos da imagem seguinte encontravam-se no interior de uma célula da imagem anterior. Também calculou-se a distância mínima entre os objetos entre duas imagens consecutivas, uma vez que existiam células que não se encontravam sobrepostas nas imagens e deste modo atribui-se o número da célula mais próxima ao seguimento. Constata-se que através da metodologia de estudo implementada obtiveram-se melhores resultados do que no software *schnitzcells*, visto que o programa desenvolvido apresenta-se em forma automática não necessitando de o utilizador de copiar linhas de código para avançar com o programa e de corrigir os processos decorridos durante a execução do mesmo. Para além do acesso facilitado à execução do trabalho também conseguiu-se melhores resultados a nível da segmentação, pois as imagens apresentam-se somente com os objetos considerados células e não foi necessário realizarem-se correções manuais para se prosseguir para o seguimento celular. As células seguem-se corretamente devido às operações

implementadas para o processo de seguimento celular, sendo que decorre de forma automática e assertiva o que não acontecia no software original, pois necessitava de o utilizador corrigir as falhas manualmente.

Para além da comparação com o software *schnitzcells* também comparou-se o trabalho desenvolvido com o software *TLM-Tracker* e constatou-se que os dois programas são semelhantes. Enquanto o *TLM-Tracker* possibilita diversas escolhas para o utilizador em todas as etapas do processo, o método desenvolvido possui uma interface mais acessível e a nível da execução e compreensão do programa é mais vantajoso para o utilizador, visto que mesmo que não se compreenda a linguagem do programa é possível executar o mesmo rapidamente.

Em trabalhos futuros poder-se-ia resolver o fato de acontecer de uma célula encontrar-se dividida numa imagem e na imagem seguinte a mesma célula aparecer na forma isolada e depois novamente na imagem posterior encontrar-se dividida. Na verdade isso não é possível, uma vez que o processo de divisão é irreversível, no entanto seria uma mais-valia para o programa, pois assim visualmente não apareceria esse erro.

Assim, pode-se concluir que os objetivos do trabalho encontram-se concretizados, uma vez que os resultados obtidos pelo trabalho desenvolvido desde o acesso facilitado à plataforma de trabalho até ao seguimento celular correto apresentam melhorias satisfatórias comparadas com o software original. Pode-se também acrescentar que em relação ao software *TLM-Tracker* que é um programa de seguimento bastante desenvolvido e com falhas reduzidas e que os resultados obtidos no método desenvolvido não se apresentam melhores do que o *TLM-Tracker*, contudo apresenta resultados semelhantes.

Referências Bibliográficas

- [1] I. Camargo, “Nutrição e cultivo de microrganismos - Laboratório de Biologia- IFSC,” 2015. [Online]. [Acedido em 8 Junho 2017].
- [2] J. L. Sobral, “Segmentação de imagem-uminho,” 2002. [Online]. [Acedido em 8 Junho 2017].
- [3] Maringá, “Programa de nivelamento biologia- cesumar,” 2011. [Online]. [Acedido em 8 Junho 2017].
- [4] C. Augusto, “Célula - A unidade da vida - CNEC,” 2009. [Online]. [Acedido em 8 Junho 2017].
- [5] T. C. Monteiro, “Biologia celular células procariontes- UFMS-CPCS,” 2012. [Online]. [Acedido em 8 Junho 2017].
- [6] T. S. M. Arias, *Biologia caderno do estudante- Ensino médio volume 3*, São Paulo: EJA, 2015.
- [7] A. L. S. P. Cardoso, E. N. C. Tessari, A. G. M. Castro, and G. F. Zanatta, “Avaliação da susceptibilidade a antimicrobianos de cepas de *Escherichia coli* de origem aviária,” *Arq. Inst. Biol. (Sao. Paulo)*, vol. 69, no. 2, pp. 1–5, 2002.
- [8] S. Mittelstaedt, “*Escherichia coli* enterohemorrágica (EHEC) O157:H7 – revisão,” *Rev Inst Ciência Saúde*, vol. 24, no. 3, pp. 175–182, 2006.
- [9] K. Snyder, “Genomes of Two Popular Research Strains of *E.Coli* Sequenced,” *Media & Communications Office*, pp. 1-3, 2009.
- [10] F. Monteiro, “Regras para o processo de segmentação,” Bragança, 2012.
- [11] M. P. d. A. e. M. P. d. Albuquerque, “Processamento de Imagens: Métodos e Análises,” Centro Brasileiro de Pesquisas Físicas – CBPF/MCT, Brasil.
- [12] J. C. M. É. Rodrigues, “Segmentação, seguimento e avaliação automática de bactérias em imagens de microscópio,” Faculdade de ciências e tecnologia- universidade nova de lisboa, Lisboa, 2013.

- [13] B. R. P. Taboada, “Seguimento de Células e Cálculo da sua Deformação em Microcanaís:,” Insituto Politécnico de Bragança- ESTIG, Bragança, 2013.
- [14] J. Klein, S. Leupold, I. Biegler, R. Biedendieck, R. Munch, and D. Jahn, “TLM-tracker: Software for cell segmentation, tracking and lineage analysis in time-lapse microscopy movies,” *Bioinformatics*, vol. 28, no. 17, pp. 2276–2277, 2012.
- [15] Q. Wang, L. You, and M. West, “Celltracer: Software for automated image segmentation and lineage mapping for single-cell studies,” ... *Duke. Edu/Research/Software/West/Celltracer*, pp. 1–2, 2008.
- [16] G. Ullman, M. Wallden, E. G. Marklund, A. Mahmutovic, I. Razinkov, and J. Elf, “High-throughput gene expression analysis at the level of single proteins using a microfluidic turbidostat and automated cell tracking,” 2013.
- [17] M. Kaiser, F. Jug, O. Silander, S. Deshpande, T. Julou, G. Myers, and E. Van Nimwegen, “Tracking single-cell gene regulation in dynamically controlled environments using an integrated microfluidic and computational setup .,” pp. 1–30, 2016.
- [18] A. D. Balomenos, S. Member, P. Tsakanikas, E. S. Manolakos, and S. Member, “Tracking Single-Cells in Overcrowded Bacterial Colonies *,” pp. 6473–6476, 2015.
- [19] L. L. McCarter, “Bacterial acrobatics on a surface: Swirling packs, collisions, and reversals during swarming,” *J. Bacteriol.*, vol. 192, no. 13, pp. 3246–3248, 2010.
- [20] Q. Wang, J. Niemi, C. Tan, L. You, and M. West, “Image Segmentation and Dynamic Lineage Analysis in Single-Cell Fluorescence Microscopy,” 2010.
- [21] S. Macneil, “Automated tracking of migrating cells in phase-contrast video,” vol. 234, no. November 2008, pp. 62–79, 2009.

- [22] H. Shen, G. Nelson, D. E. Nelson, S. Kennedy, D. G. Spiller, T. Griffiths, N. Paton, S. G. Oliver, M. R. H. White, and D. B. Kell, "Automated tracking of gene expression in individual cells and cell compartments.," *J. R. Soc. Interface*, vol. 3, no. 11, pp. 787–794, 2006.
- [23] A. Sacan, H. Ferhatosmanoglu, and H. Coskun, "CellTrack: An open-source software for cell tracking and motility analysis," *Bioinformatics*, vol. 24, no. 14, pp. 1647–1649, 2008.
- [24] E. C. Garner, "Micro Commentary MicrobeTracker□: quantitative image analysis designed for the smallest organisms," vol. 80, no. March, pp. 577–579, 2011.
- [25] A. Muthukrishnan, A. Mora, J. M. Fonseca, and A. S. Ribeiro, "Systems biology CellAging□: a tool to study segregation and partitioning in division in cell lineages of Escherichia coli," pp. 1–2, 2013.
- [26] I. Mekterovi and D. Mekterovi, "BactImAS□: a platform for processing and analysis of bacterial time-lapse microscopy movies," pp. 1–10, 2014.
- [27] P. S. Swain, E. Mjolsness, and M. B. Elowitz, "using fluorescence time-lapse microscopy," vol. 7, no. 1, pp. 80–88, 2014.
- [28] L. Martins, J. Fonseca, and A. Ribeiro, "'miSimBa' - A simulator of synthetic time-lapsed microscopy images of bacterial cells," *Proc. - 2015 IEEE 4th Port. Meet. Bioeng. ENBENG 2015*, no. February, pp. 26–28, 2015.
- [29] A. Paintdakhi, B. Parry, M. Campos, I. Irnov, J. Elf, I. Surovtsev, and C. Jacobs-wagner, "Oufiti□: an integrated software package for high-accuracy , high-throughput quantitative microscopy analysis," no. 2015, pp. 767–777, 2016.
- [30] S. Helfrich, C. Probst, C. C. Sachs, A. Gru, W. Wiechert, D. Kohlheyer, and K. No, "Image-Based Single Cell Profiling□: High- Throughput Processing of Mother Machine Experiments," pp. 1–15, 2016.

Anexos

- Imagens resultantes do seguimento celular (desde da imagem 1 até à imagem 20)

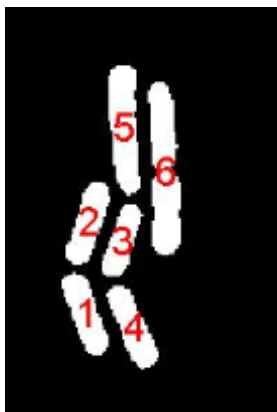


Imagem 1

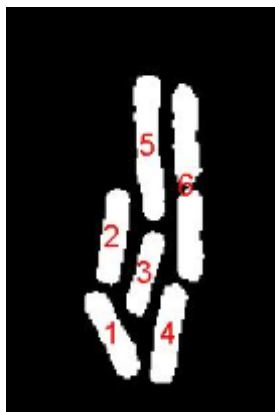


Imagem 2

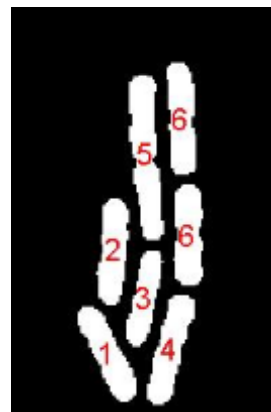


Imagem 3

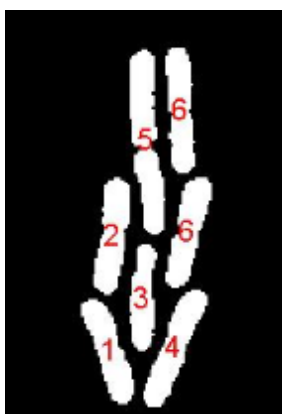


Imagem 4

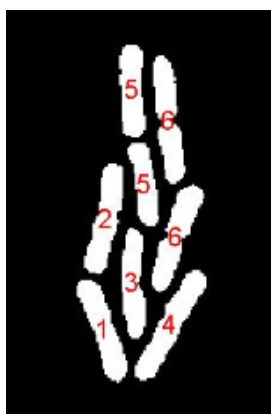


Imagem 5

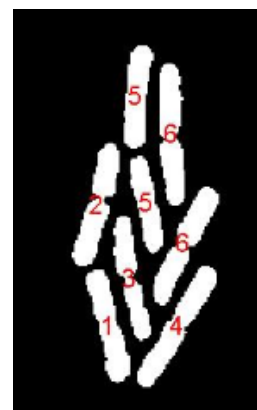


Imagem 6



Imagem 7



Imagem 8



Imagem 9



Imagem 10



Imagem 11



Imagem 12



Imagem 13



Imagem 14

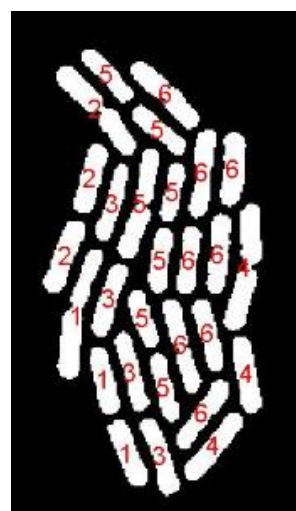


Imagem 15



Imagem 16



Imagem 17



Imagem 18



Imagem 19

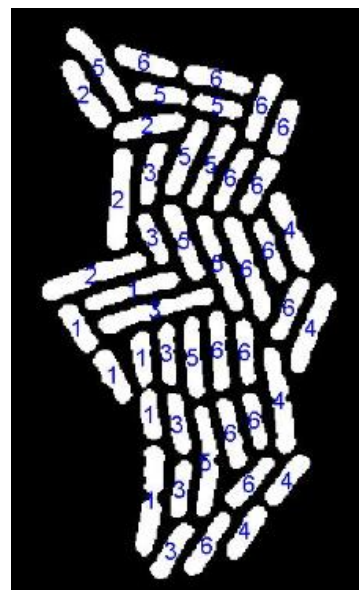


Imagem 20

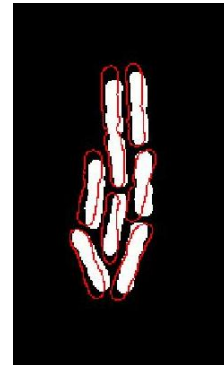
- Imagens resultantes da sobreposição de imagens



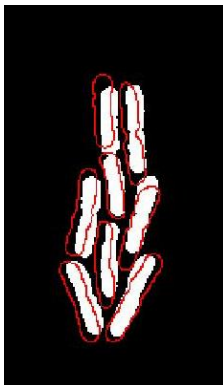
Imoverlay_1_2



Imoverlay_2_3



Imoverlay_3_4



Imoverlay_4_5



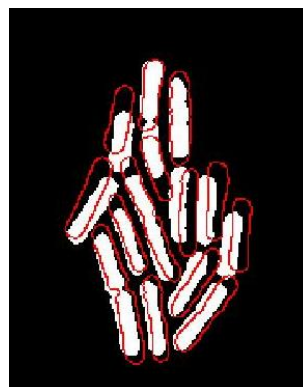
Imoverlay_5_6



Imoverlay_6_7



Imoverlay_7_8



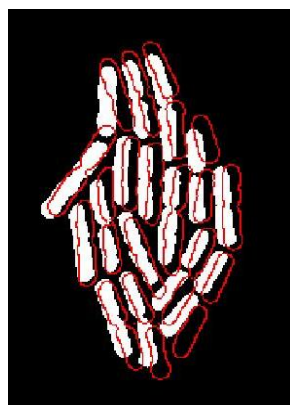
Imoverlay_8_9



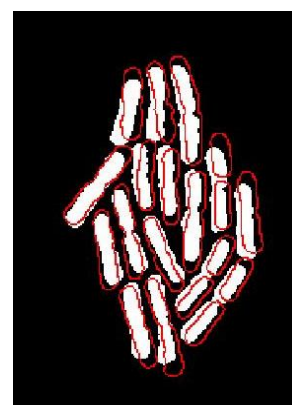
Imoverlay_9_10



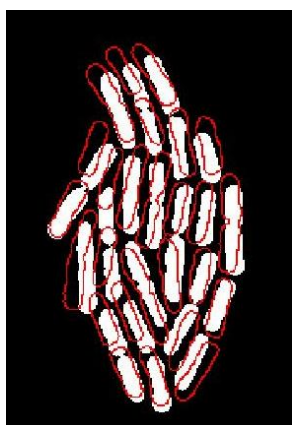
Imoverlay_10_11



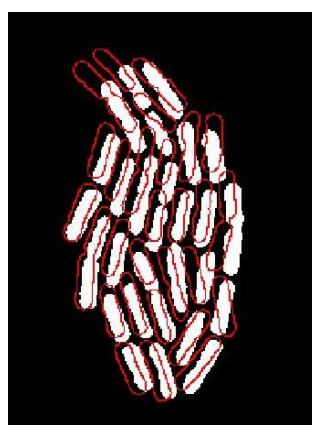
Imoverlay_11_12



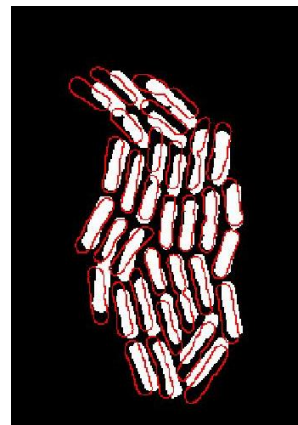
Imoverlay_12_13



Imoverlay_13_14



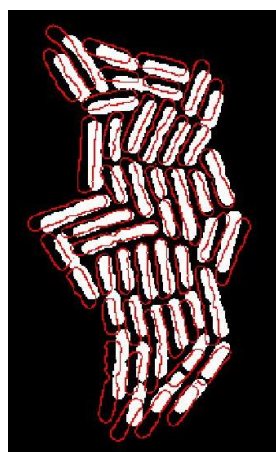
Imoverlay_14_15



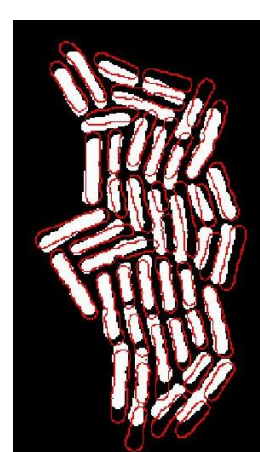
Imoverlay_15_16



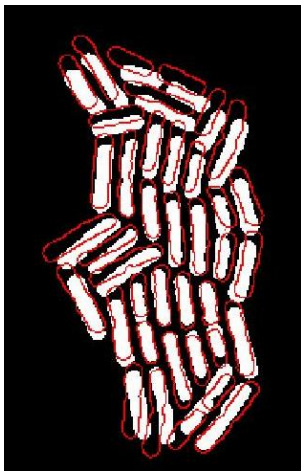
Imoverlay_16_17



Imoverlay_17_18



Imoverlay_18_19



Imoverlay_19_20