



ANÁLISE DE IMAGENS PARA A QUANTIFICAÇÃO DE COLÔNIAS DE BACTÉRIAS

Silva, E. R.¹, Santos, J. M.¹, Galvão Filho, A. R.¹², Gomes, R.P.³,
Carneiro, C.R.³, Coelho, C.J.¹²

¹ Laboratório de Computação Científica - Escola de Ciências Exatas e da Computação
Pontifícia Universidade Católica de Goiás
Goiânia-Goiás-Brasil

² Mestrado em Engenharia de Produção e Sistemas
Pontifícia Universidade Católica de Goiás
Goiânia-Goiás-Brasil

³ Laboratório de Biotecnologia de Microrganismos - Instituto de Patologia Tropical e Saúde Pública
Universidade Federal de Goiás
Goiânia-Goiás-Brasil

RESUMO: A contagem manual de colônias de bactérias em placas de Petri é uma tarefa típica na rotina laboratorial. No entanto, a contagem manual está propensa a erros de interpretação e amostragem. Este trabalho apresenta um sistema de visão computacional, para a segmentação e contagem automática de colônias de bactérias a partir de imagens de placas de Petri. O método proposto baseia-se na ideia de remover o ágar da imagem da placa de Petri e considerar como colônias as regiões conectadas que não foram removidas. Os resultados apresentaram um acerto de 92,02%, entre a contagem manual e a contagem automática, em um intervalo de tempo menor comparado à contagem manual.

Palavras-chave: Visão Computacional, Colônias de Bactérias, Imagens de placas de Petri.

ABSTRACT: Manually counting bacterial colonies on Petri dishes is a typical task in laboratory routine. However, manual counting is prone to misinterpretation and sampling errors. The proposed method is based on the idea of removing agar from Petri disk image and considering as colonies connected regions that were not removed. This work presents a computer vision system for automatic segmentation and counting of bacterial colonies from Petri dish images. Results showed an accuracy of 92,02%, between manual counting and automatic counting, in a shorter time interval compared to manual counting.

Keywords: Computer Vision, Bacteria Colony, Petri Dish Images.

1. Introdução

A contagem de colônias de bactérias é uma etapa fundamental em pesquisas e avaliações microbiológicas [8]. O processo de contagem permite estimar as taxas de sobrevivência e de proliferação de espécies de bactérias e é comumente empregado em diversas aplicações, como

na seleção de antibióticos [6], na avaliação da qualidade de água potável, em testes toxicológicos [1] e na avaliação da segurança de alimentos [2].

Normalmente, espécies de bactérias são cultivadas para análise em placas transparentes e isolados do ambiente externo, chamados de placas de Petri. O processo de cultura envolve preparar um ambiente favorável ao desenvolvimento da bactéria, contendo nutrientes necessários para o crescimento bacteriano, chamado de meio de cultivo, juntamente com um solidificante chamado ágar. Posteriormente, as bactérias são espalhadas sobre o ágar e o recipiente é então mantido em temperatura controlada por cerca de horas ou dias até que as colônias se tornem visíveis [5].

De forma geral, o processo de contagem de colônias de amostras é realizado manualmente e por esse motivo os resultados podem ser muito imprecisos. Além disso, a quantificação manual é muito trabalhosa, demorada e em grande quantidade de amostras o trabalho pode se tornar inviável [8].

Tendo em vista esse cenário, surge a necessidade de uma estratégia automatizada para a contagem de colônias de bactérias em placas de Petri. Vários métodos já foram propostos utilizando sistemas de visão computacional e machine learning, como por exemplo [9 - 14]. O principal desafio a ser resolvido por tais modelos é a heterogeneidade das imagens das placas. Existem divergências de cor e textura apresentadas por diferentes tipos de meio de cultura, componentes e nutrientes, coloração, indicadores, tamanho, espessura, formato das colônias de diferentes espécies de bactérias, possíveis artefatos como umidade, defeitos no vidro e diversas outras heterogeneidades a serem tratadas [5].

Diante do exposto, o objetivo deste artigo é desenvolver um algoritmo capaz de executar a detecção e contagem automática de colônias de bactérias a partir de imagens obtidas de placas de Petri, utilizando visão computacional.

2. Materiais e Métodos

O processo de contagem automático das colônias de bactérias foi dividido em duas etapas: seleção da região interna da placa e detecção automática das colônias de bactérias em imagens de placas de Petri. Durante o processo a ser tratado neste estudo, a Figura 1 será utilizada como exemplo. Para os testes foram utilizadas outras 30 imagens.



Figura 1: Placa de Petri com crescimento bacteriano formando diversas colônias de bactérias.

2.1. Seleção da Região Interna da placa

Foi considerado durante a contagem somente o interior da região circular da placa de Petri, pelo fato de que é nesta região que as colônias estão localizadas. Para achar esta região circular foram inseridos três pontos da borda da placa, para que fosse encontrado o círculo circunscrito do triângulo formado por estes pontos [7], como mostra a Figura 2.

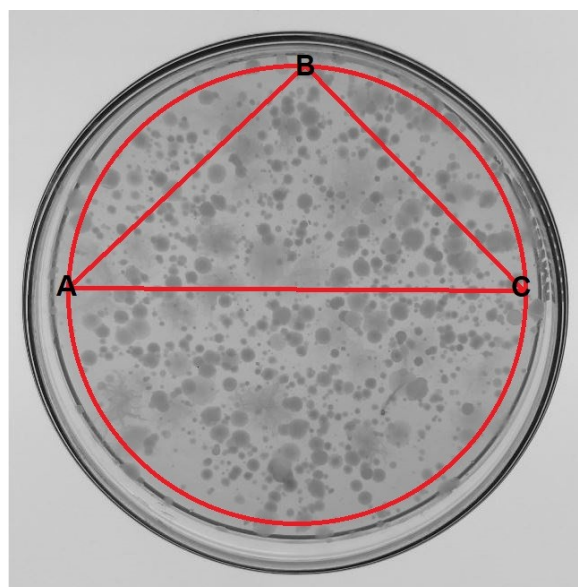


Figura 2: Círculo circunscrito de um triângulo formado pelos pontos A, B e C.

2.1.1. Algoritmo Utilizado para Encontrar o Raio e o Centro do Círculo

Foram dados três pontos A, B e C, pertencentes à circunferência da placa de Petri, o raio e o centro do círculo foram calculados através das seguintes etapas:

1. Calcular as distâncias d_{AB} , d_{BC} e d_{CA} do triângulo (A,B,C) usando a equação (1):

$$d_{PQ} = \sqrt{(x_P - x_Q)^2 + (y_P - y_Q)^2}, \quad (1) \text{ onde } P(x_P, y_P) \text{ e } Q(x_Q, y_Q) \text{ são dois pontos.}$$

2. Calcular a área do triângulo usando a equação (2):

$$area = \sqrt{s(s - d_{AB})(s - d_{BC})(s - d_{CA})}, \quad (2) \quad \text{onde } s = \frac{(d_{AB} + d_{BC} + d_{CA})}{2} \text{ é o semiperímetro do triângulo (A,B,C).}$$

3. Calcular o raio R do círculo usando a equação (3):

$$R = \frac{d_{AB} d_{BC} d_{CA}}{4 area} \quad (3)$$

4. Calcular as coordenadas baricêntricas usando as equações (4), (5) e (6):

$$x = \frac{d_{BC}^2 (-d_{AB}^2 + d_{BC}^2 + d_{CA}^2)}{d_{AB}^2 (d_{AB}^2 - d_{BC}^2 + d_{CA}^2)} \quad (4)$$

$$y = \frac{d_{BC}^2 (d_{AB}^2 - d_{BC}^2 + d_{CA}^2)}{d_{AB}^2 (d_{AB}^2 - d_{BC}^2 + d_{CA}^2)} \quad (5)$$

$$z = \frac{d_{CA}^2 (d_{AB}^2 + d_{BC}^2 - d_{CA}^2)}{d_{AB}^2 (d_{AB}^2 - d_{BC}^2 + d_{CA}^2)} \quad (6)$$

5. Calcular as coordenadas do centro do círculo P_C usando a equação (7):

$$P_C = xA + yB + zC. \quad (7)$$

2.2. Detecção Automática de Colônias de Bactérias em Imagens de placa de Petri

O método de contagem proposto neste artigo baseia-se na tentativa de identificar e remover da imagem, a região correspondente ao ágar, isto é, o fundo da região de interesse selecionada pelo usuário. Após a remoção, cada região isolada restante na imagem é considerada uma colônia de bactérias.

Para realizar a detecção do ágar, foi empregada uma técnica de processamento digital de imagens que seleciona, a partir de um pixel inicial, todos os pixels conectados a ele que possuam intensidades similares na escala de cinza. Duas intensidades são ditas similares se estas possuem diferença absoluta inferior a um certo valor de tolerância. Por conveniência, ao longo do artigo, essa técnica foi chamada de *grayconnected*, o mesmo nome da rotina do MATLAB que implementa esse método.

Inicialmente, um ponto qualquer do ágar foi inserido como o ponto de partida do *grayconnected*. A partir desse ponto o *grayconnected* foi executado repetidamente com a tolerância de similaridade entre os pixels cada vez maior. Considere *DiffMax*, a tolerância de similaridade inicial do *grayconnected*. Considere também *DiffInc*, o valor que foi adicionado ao *DiffMax* a cada iteração do algoritmo.

Na Figura 3 os pixels selecionados como parte do ágar são representados na cor preta, e as regiões isoladas (prováveis colônias) são representadas em branco.

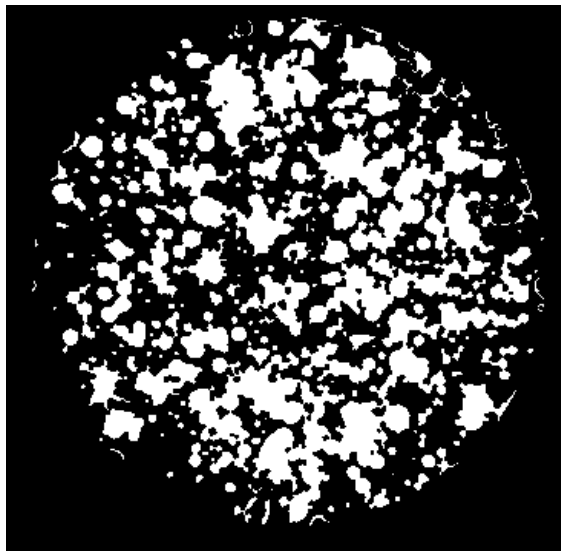


Figura 3: Regiões após o uso do *grayconnected* na primeira iteração.

Cada região isolada da imagem obtida é contabilizada como uma única colônia. Contudo, é possível notar na Figura 3 que várias regiões isoladas são formadas por mais de uma colônia. A fim de solucionar esse problema é que a tolerância de similaridade *DiffMax* é incrementada a cada iteração do algoritmo utilizando a equação (8):

$$DiffMax = DiffMax + DiffInc \quad (8)$$

Quando o *grayconnected* é executado com o valor de *DiffMax* atualizado pode ocorrer a separação de regiões que antes eram consideradas apenas uma colônia e agora são contabilizadas como mais de uma. Isso ocorre pelo fato de que geralmente as regiões que conectam duas ou mais colônias têm a diferença de intensidade com o fundo menor que a diferença de intensidade entre a região central da colônia que está sendo conectada e o fundo.

Quando a intensidade do *grayconnected* é aumentada, algumas regiões podem deixar de ser contabilizadas, como mostra a Figura 4. Para continuar tendo essas regiões contabilizadas

uma outra imagem auxiliar *ImAux* foi atualizada a cada iteração sem que se perdesse as regiões que não podiam ser localizadas pelo *grayconnected*.

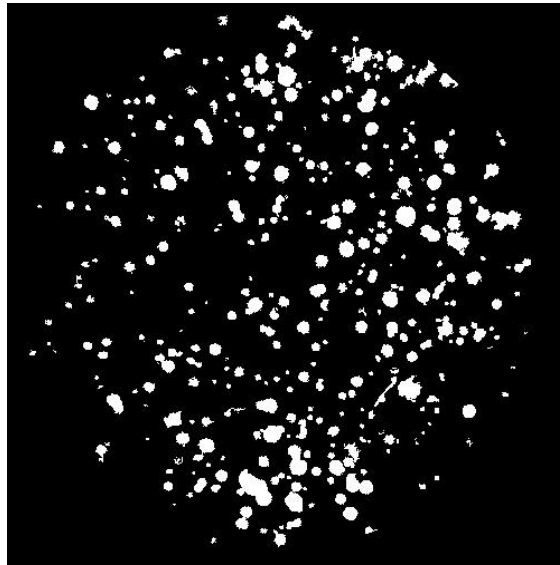


Figura 4: Regiões após o uso do *grayconnected* na terceira iteração.

O algoritmo foi repetido, executando o *grayconnected* com tolerâncias cada vez maiores, até que nenhuma colônia fosse mais identificada, isto é, até que toda a região de interesse da imagem fosse considerada ágar.

Como a área de ruídos e artefatos, geralmente, é inferior à área de uma colônia, como mostra a Figura 5, foi definido que a menor área que uma região isolada pode ter, para que seja contabilizada como colônia, é $AreaInt \times MinCol$, em que *AreaInt* é a área da região interna da placa de Petri (região de interesse inserida) e *MinCol* é uma constante que foi obtida através de testes experimentais.



Figura 5: Artefatos não contabilizados como colônias de bactérias.

2.2.1. Processo de Detecção de Regiões que se Dividiram

Foi proposto um algoritmo robusto capaz de otimizar o processo de detecção de regiões que se dividiram durante as iterações da contagem automática de colônias de bactérias, utilizando visão computacional.

Considere as Figuras 6, 7, 8 e 9 como sendo partes de um exemplo de execução do algoritmo citado acima em uma imagem obtida a partir de uma placa de Petri.

1. Atribuir valores representativos aos pixels de cada região de uma imagem A processada pelo *grayconnected*, de tal forma que $V(A_{ij}) = V(A_{kl})$, se A_{ij} e A_{kl} forem quaisquer pixels da mesma região e $V(A_{ij}) \neq V(A_{kl})$, se A_{ij} e A_{kl} forem quaisquer pixels de regiões diferentes e $V(A_{ij}) = 0$ se A_{ij} pertencer ao fundo.

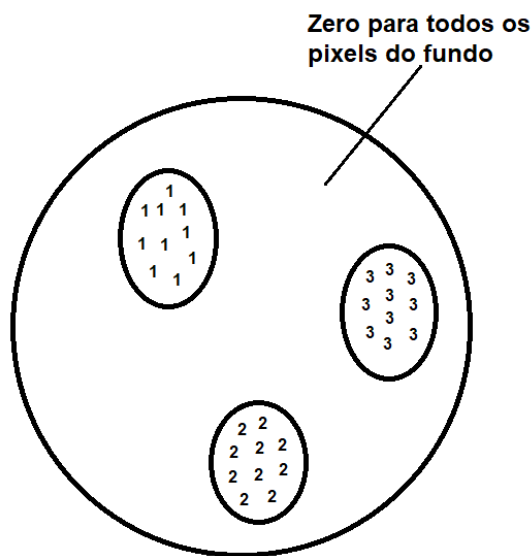


Figura 6: Primeira etapa da execução do algoritmo.

2. Entrar na repetição e continuar até que não haja mais regiões a serem contabilizadas.
 - a. Criar imagem B através do processo do *grayconnected*, com o *DiffMax* incrementado.
 - b. Repetir a etapa 1 na imagem B .

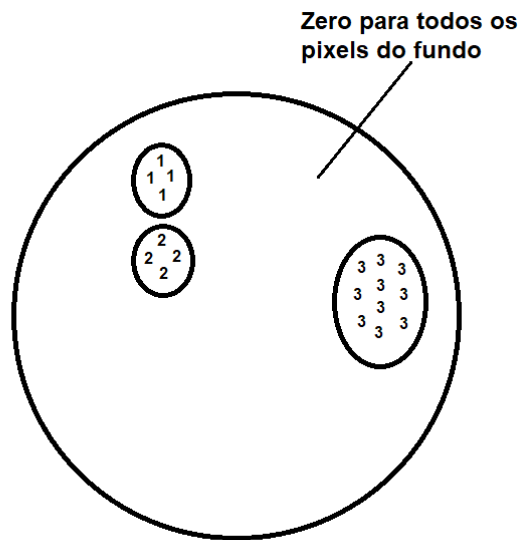


Figura 7: Quarta etapa da execução do algoritmo.

- c. Criar uma variável Qt , tal que Qt é igual ao número de regiões da figura $B + 1$.
- d. Criar uma imagem C tal que $C_{ij} = A_{ij} * Qt + B_{ij}$.

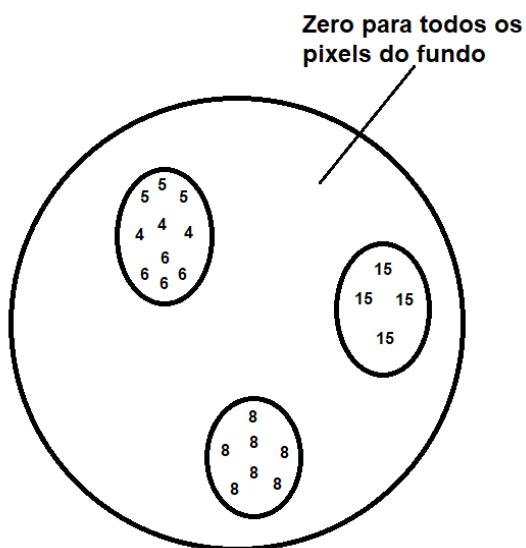


Figura 8: Sexta etapa da execução do algoritmo.

- e. Criar uma estrutura de dados Vet tal que Vet recebe todos os valores referenciados aos pixels da imagem C menos o valor referenciado ao fundo que é zero.

$$Vet = [4 \ 6 \ 5 \ 8 \ 15]$$

- f. Entrar na repetição que percorre todo o Vet .

- i. Se $Vet_i \bmod Qt \equiv 0$ e o valor truncado de $\frac{Vet_i}{Qt}$ for igual ao valor truncado de $\frac{Vet_{i+1}}{Qt}$ então continue a repetição.
- ii. Se $Vet_i \bmod Qt \equiv 0$ e o valor truncado de $\frac{Vet_i}{Qt}$ for diferente do valor truncado de $\frac{Vet_{i+1}}{Qt}$ então adicione a região conectada pelo valor do Vet_i na imagem C a imagem $ImAux$.
- iii. Se $Vet_i \bmod Qt \neq 0$ então adicione a região conectada pelo valor do Vet_i na imagem C a imagem $ImAux$.

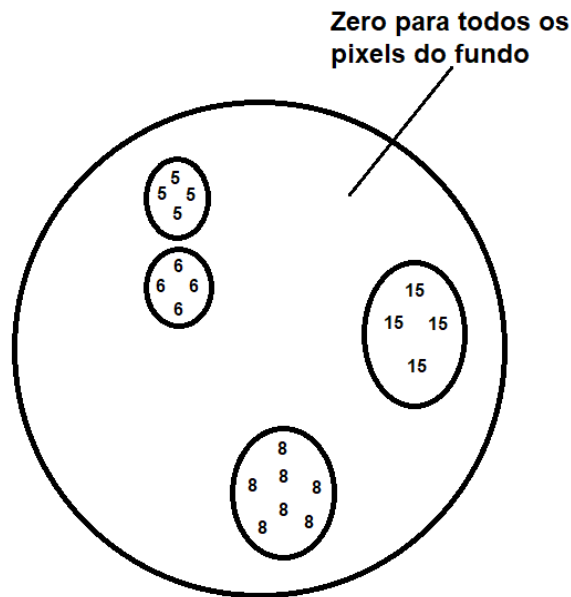


Figura 9: Oitava etapa da execução do algoritmo.

2.4. Definição dos Parâmetros

Para melhorar a qualidade dos resultados obtidos com o método proposto, foi utilizado um algoritmo de *iterated racing* para definir os valores dos parâmetros *DiffMax*, *MinCol* e *DiffInc*. O algoritmo *iterated racing* utiliza métodos estatísticos para verificar quais os melhores valores a serem atribuídos aos parâmetros utilizados em um determinado algoritmo [3].

O refinamento dos parâmetros com o *iterated racing* foi realizado através do pacote *irace* versão 3.1. O pacote *irace* necessita que seja definida a quantidade de execuções do algoritmo a ser refinado que podem ser realizadas. Para o refinamento empregado, a quantidade de execuções permitidas foi definida em 2000.

Os melhores valores a serem atribuídos às variáveis *DiffMax*, *MinCol* e *DiffInc*, segundo o *irace*, foram 29, 0.00002952 e 6 respectivamente.

Para calcular a taxa de acerto foi utilizada a equação (9) :

$$taxa\ de\ acerto = 100 - \frac{(|ContagemManual - ContagemAutomática|)}{ContagemManual} 100 \quad (9)$$

3. Resultados

Os resultados apresentados são referentes à contagem das colônias de bactérias em 30 imagens obtidas a partir de placas de Petri. Para ilustrar os resultados, a Figura 10 foi utilizada como exemplo.

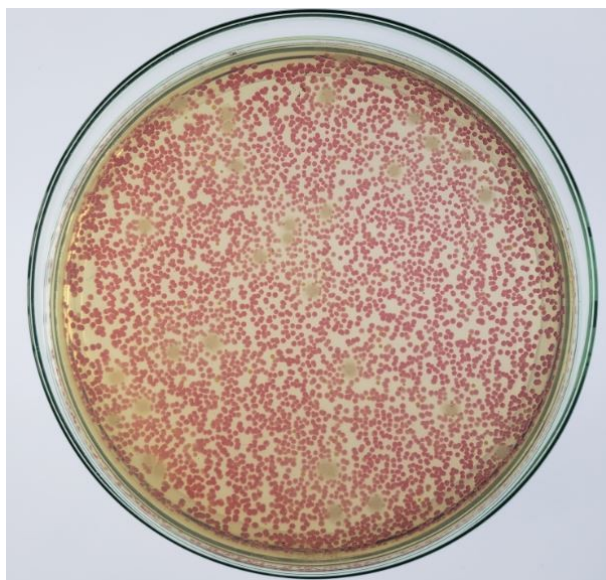


Figura 10: Placa de Petri populada com colônias de bactérias.

3.1. Contagem das Colônias de Bactérias

Várias colônias foram selecionadas, no entanto algumas colônias que estão sobrepostas não foram contabilizadas, como mostra a Figura 11. A Figura 11 exibe diferentes regiões, a), b), c) e d), da mesma imagem ampliada. As colônias que foram contabilizadas possuem um ponto azul em suas regiões centrais.

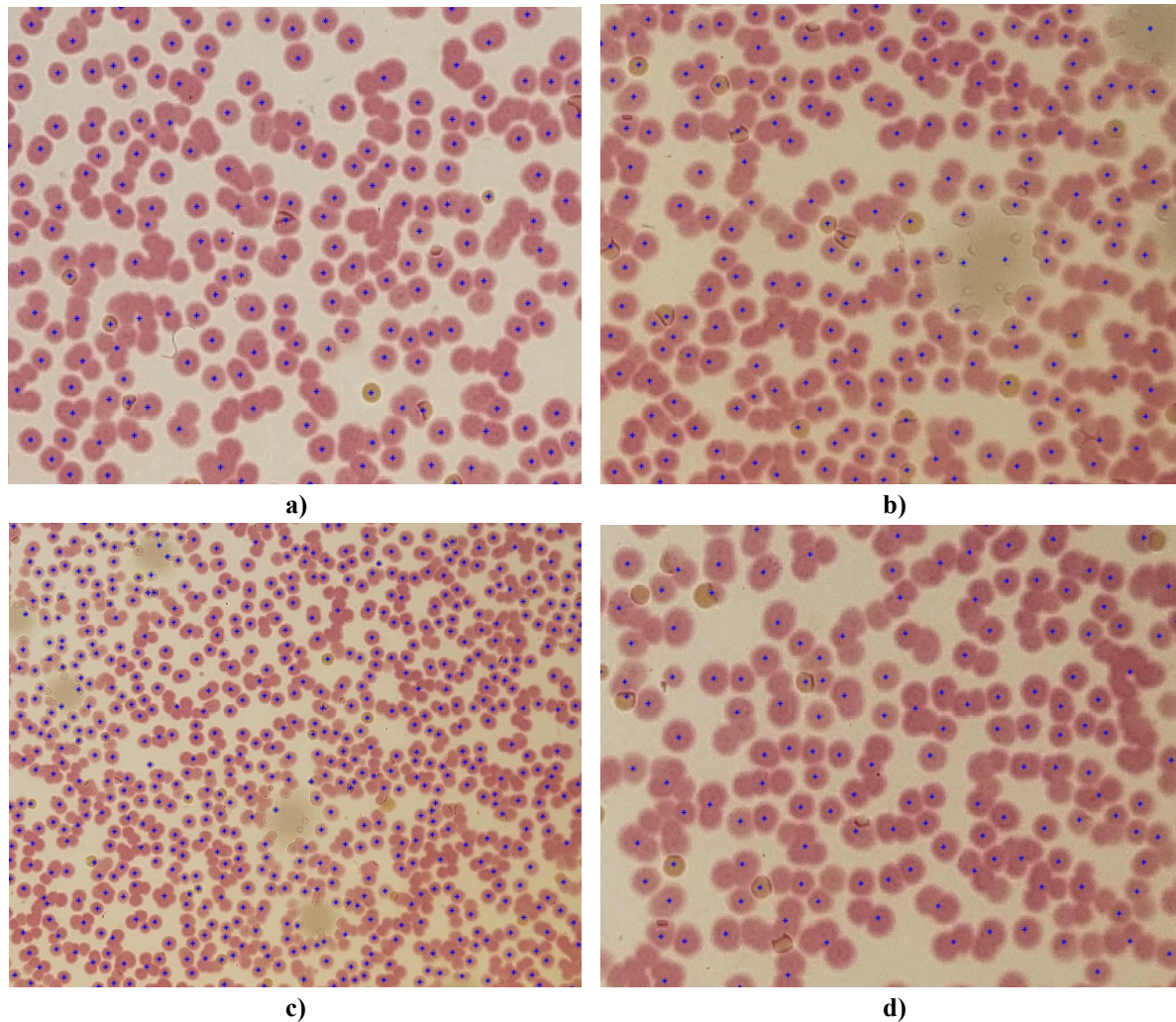


Figura 11: Resultados da detecção de colônias de bactérias.

O mesmo processo foi executado para outras 30 imagens, resultantes de campos visuais diferentes. O método proposto obteve um acerto de 92,02%, entre a contagem manual e a contagem automática, com resultados aproximados aos obtidos por especialistas e processados em menor tempo comparado a contagem manual

4. Conclusão

Neste artigo foi proposto um método de contagem automática de colônias de bactérias em imagens obtidas a partir de placas de Petri usando técnicas de processamento digital de imagem. Considerando que os objetos de interesse, colônias de bactérias, são regiões conectadas, formadas por tons de pixels semelhantes, que se diferenciam em relação a sua área e localização na imagem é possível o emprego de técnicas de processamento digital de imagens para a segmentação e contagem dos objetos de interesse. Os resultados obtidos aproximam-se dos produzidos por especialistas com a vantagem de serem automatizados e desempenhados em menor período de tempo quando comparado ao processo manual de contagem.

5. Referências Bibliográficas

1. Chen, B.-Y., Chen, H.-L., Liu, Y.-W., Cheng, Y.-C.: Dose–response assessment of metal toxicity upon indigenous *Thiobacillus thiooxidans* BC1. *Process Biochem.* 39, 737–748; (2004).
2. Itoh, Y., et al.: Enterohemorrhagic *Escherichia coli* O157:H7 present in radish sprouts. *Appl. Environ. Microbiol.* 64, 1532; (1998).
3. López-Ibáñez, Manuel; Cáceres, Leslie Pérez; DUBOIS-LACOSTE, Jérémie. The irace Package: User Guide. Belgium: Iridia (2019).
4. Szeliski, Richard. *Computer Vision: Algorithms and Applications*. Washington: Springer (2011).
5. Tortora, Gerard J.; FUNKE, Berdell R.; CASE, Christine L.: *Microbiologia*. [S.L.]; Artmed (2012).
6. Van Doorn, L.J., et al.: Importance of *Helicobacter pylori* cagA and vacA status for the efficacy of antibiotic treatment. *Gut* 46, 321–326; (2000).
7. ATEŞ, Hüseyin; GEREK, Ömer Nezih. An Image-Processing Based Automated Bacteria Colony Counter. 2009. 6 f. Tese (Doutorado) - Curso de Electrical And Electronics Engineering, Anadolu University, Eskisehir, 2009.
8. CHIANG, Pei-ju; TSENG, Min-jen. Automated counting of bacterial colonies by image analysis. *Journal Of Microbiological Methods*. Taiwan, p. 1-9. jul. 2014. CHIANG, Pei-ju; TSENG, Min-jen. Automated counting of bacterial colonies by image analysis. *Journal Of Microbiological Methods*. Taiwan, p. 1-9. jul. 2014.
9. MAROTZ, J.. Effective object recognition for automated counting of colonies in Petri dishes (automated colony counting). 2000. 16 f. Tese (Doutorado) - Curso de Medicine, Lubeck, Germany, 2000.
10. M.R. Garcia-Armesto, M.L. Prieto, Garcia-Lopez, A. Otero, B. Moreno, Modern micro-biological methods for foods: colony counting and direct count methods. A review, *Microbiologia* 9 (1) (1993) 1–13.
11. R.L. Parry, T.W. Chin, P.K. Donahoe, Computer-aided cell colony counting, *BioTechniques* 10 (6) (1991) 772–774.

12. N. Belyaev, S. Paavilainen, T. Korpela, Characterization of bacterial growth on solid medium with image analysis, *J. Biochem. Biophys. Methods* 25 (1992) 125–132.
13. P. Jasiobedzki, W.J. Martin, Processing of bacterial colony images for automatic isolation and transfer, *J. Phys. E Sci. Instrum.* 22 (1989) 364–367.
14. I.G. Wilson, Use of the IUL counterplate automatic colony counter for spiral plated total viable counts, *Appl. Environ. Microbiol.* 61 (1995) 3158–3160.