

# Trabalho 3 de Introdução ao Processamento de Imagens

Victor André Gris Costa  
Universidade de Brasília - UnB  
Departamento de Ciências da Computação  
Brasília, Brasil  
victorandr98@gmail.com

**Resumo**—Este trabalho tem como objetivo fixar conceitos de processamento de imagens através da resolução de 2 problemas propostos: a segmentação de células utilizando o algoritmo de *watershed* e a segmentação por método a escolha do aluno para um frame do vídeo do *foreman*. Para o primeiro problema foi obtido um resultado bom em parte. Para o segundo problema utilizou-se o algoritmo de K-means e o resultado foi satisfatório.

## I. INTRODUÇÃO

Processamento de imagens é uma subárea de processamento digital de sinais na qual são aplicados processos em imagens para obter novas imagens. Hoje em dia com o maior poder de processamento esta área se tornou essencial para muitas áreas. Por exemplo na medicina ela é utilizada para melhorar imagens médicas. Este trabalho foca em uma parte importante de processamento de imagens, a segmentação de elementos.

A segmentação é uma subárea muito importante em processamento de imagens para realizar processos de forma automatizada. A segmentação consiste em separar uma imagem em regiões homogêneas relativas a um critério. Existem vários métodos para segmentar uma imagem e dependem da aplicação. Neste trabalho utilizaram-se as técnicas de *watershed* e de *K-means*.

O método de segmentação de *watershed* consiste em considerar a imagem como um mapa de alturas baseado no brilho dos pixels e "preencher" este mapa com "água" a partir do fundo. A borda é onde ocorre a junção da "água" por cima dos picos de brilho.

O método de segmentação por *K-means* consiste em analisar os pixels da imagem como vetores e tentar agrupar em K grupos. Esse agrupamento procura achar uma média com a menor variância dentro do grupo. Inicialmente o algoritmo escolhe K médias aleatórias e depois prossegue passo a passo minimizando as diferenças dentro do grupo até obter a segmentação final. Geralmente esse algoritmo é aplicado em todos os canais de uma imagem para classificar cada pixel por cor e brilho.

## II. METODOLOGIA

### A. Questão 1

Como primeiro problema foi proposto segmentar as células do fundo da imagem 1. Para a segmentação foram utilizadas técnicas para obter uma imagem preto e branco com o fundo

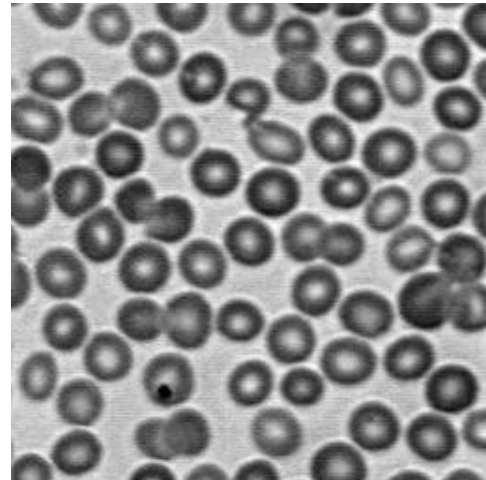


Figura 1. Imagem para a questão 1.

branco e as células em preto. A partir dessa imagem de células utilizou-se a transformada da distância e um limiar para encontrar o centro das células e gerar os marcadores para o *watershed*. Ao fim executou-se o algoritmo de *watershed* na imagem das células com os marcadores obtidos e assim obtendo a segmentação final.

Primeiramente realiza-se uma operação de limiar de brilho na imagem original de forma a obter uma separação entre o fundo e as células inicial. Essa imagem possuía muitos buracos após essa limiarização, então aplicou-se a operação de erosão de forma a expandir as células em preto e reduzir a descontinuidade. Após essa operação, ainda havia células com buracos e então esse problema foi resolvido com a operação de *floodfill* no fundo. Esta operação fez com que todos os pixels conectados do fundo fossem brancos e os pixels brancos e pretos dentro das células fossem pretos. Enfim obteve-se uma imagem sem buracos.

A partir da imagem sem buracos realizou-se a transformada de distância, que basicamente faz com que o brilho de cada pixel seja igual a distância euclidiana do pixel branco mais próximo. Após essa operação, aplicou-se uma normalização de mínimo e máximo de forma a mapear a menor distância para 0 e a maior distância para 1. Os pixels brancos nessa imagem se tornaram pretos e os pretos ficaram em escala de cinza

proporcional a distância. Aplicou-se um limiar nessa imagem de forma a filtrar os centros das células como branco e o resto como preto. Para cada conjunto de pixels brancos foi fornecida uma numeração inteira de 1 em diante para atuar como os marcadores das células.

Com essa matriz de marcadores e a imagem binária contendo as células em preto e o fundo em branco, realizou-se o algoritmo de *watershed* e foi obtida uma matriz do tamanho da imagem com as bordas marcadas com -1. Aplicou-se uma cor para esses pixels específicos da matriz de marcadores na imagem original de forma a visualizar as bordas.

#### B. Questão 2



Figura 2. Imagem para a questão 2.

Como segundo problema foi proposto a extração de *frames* de um vídeo no formato YUV 4:2:0 e a segmentação do primeiro *frame* do vídeo para obter somente a pele. Para tanto fez-se uso dos conhecimentos do formato YUV 4:2:0 para extração e o algoritmo de *K-means* para extração da pele.

Para a leitura de cada *frame* realizou-se a leitura de 3 matrizes de canais: Y, U e V. O canal Y possuía o tamanho original do vídeo de 352 colunas e 288 linhas. Os canais U e V, de forma a compactar mais a imagem, foram armazenados com um quarto do tamanho, ou seja, metade das linhas e metade das colunas. Caso o usuário da função pedisse por mais de um *frame*, era necessário somente repetir o procedimento de leitura das 3 matrizes dos 3 canais.

Após a extração dos 3 canais do primeiro *frame* utilizando a função de extração criada acima realizou-se o aumento do tamanho das canais menores para igualar com o canal Y. Com os canais no tamanho correto, realizou-se o algoritmo de *K-means* somente nos canais U e V que representam a cor. O motivo dessa decisão é por conta da sombra que há no rosto da pessoa na imagem. Quando o algoritmo é feito com a componente de brilho Y, a imagem é mal segmentada. A segmentação realizada utilizou  $K=3$ , pois há 3 grandes conjuntos de cores na imagem: O prédio, a pele e a gola da camisa. A gola da camisa possui uma cor muito próxima da pele e com  $K=2$  ela era geralmente classificada como parte dela. Utilizando  $K=3$  esse problema era reduzido a somente a borda da gola. O resultado dessa segmentação é uma matriz

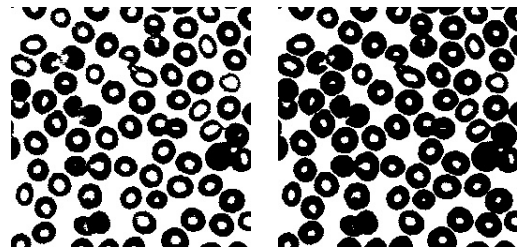
de marcadores com números inteiros de 0 a 2. Como não é possível saber qual marcador será o da pele, utiliza-se o marcador do centro da imagem como referência, tendo em vista que o pixel central da imagem com certeza é pele.

Para refinar a segmentação, primeiro se binariza a imagem entre pele com o valor de 1 e resto com o valor de 0. Por fim realiza-se operações morfológicas para reduzir as imperfeições da segmentação. Para fins de visualização, multiplica-se a matriz binária com a imagem colorida, escurecendo o fundo.

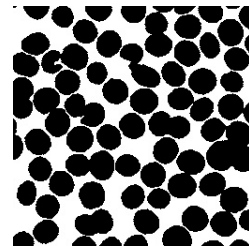
### III. RESULTADOS

#### A. Questão 1

Realizando-se a binarização da imagem original 1 obteve-se a imagem 3(a). Para melhorar a extração, realizou-se uma erosão, obtendo a imagem 3(b). A partir disso realizou-se o *floodfill* para remover os buracos completamente, obtendo a imagem 3(c).



(a) Imagem após aplicação de limiar obtido de forma empírica. (b) Imagem após aplicação de erosão para reduzir desconituidades.



(c) Imagem após aplicação de *floodfill* para retirar buracos completamente.

Figura 3. Processo para extração de células em preto e fundo em branco.

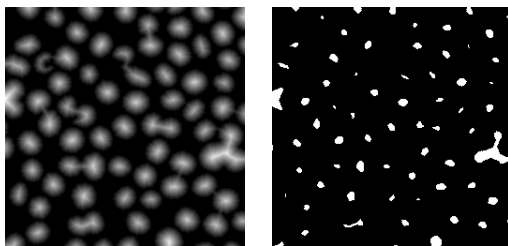
A partir das células anteriores se aplica a transformada de distância e normalização, obtendo a figura 4(a). A partir dessa imagem transformada, realiza-se uma binarização de forma a sobrar somente os centros das células, assim obtendo a figura 4(b).

Por fim realizou-se o algoritmo do *watershed* e as bordas foram desenhadas na imagem original da figura 1. Assim obteve-se a imagem da figura 5.

Infelizmente a segmentação não funcionou bem com células muito coladas, especialmente as 3 células da direita.

#### B. Questão 2

Realizando-se a extração da imagem, foi possível obter 3 componentes da imagem, a Y, U e V. Elas podem ser visualizadas na figura 6.



(a) Imagem após aplicação da (b) Imagem após aplicação li-transformada da distância e miar obtido de forma arbitrária para obter os centros.

Figura 4. Processo para extração do centro das células para obtenção de marcadores.

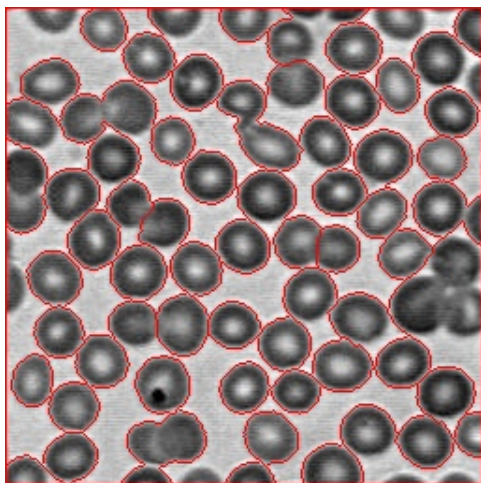
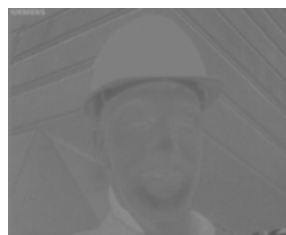


Figura 5. Imagem com células segmentadas.



(a) Componente Y da imagem, que representa o brilho.



(b) Componente U da imagem, que representa parte da cor.



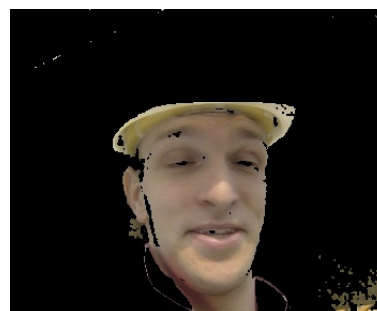
(c) Componente V da imagem, que representa parte da cor.

Figura 6. Componentes separadas da imagem original.

Após a extração dos canais da imagem, foi possível realizar a segmentação. Logo após a segmentação, obteve-se a imagem da figura 7(a). Para visualizar melhor o resultado, multiplica-se na imagem original para visualizar o que foi marcado como pele e o que não foi marcado como não pele. Obteve-se assim a imagem da figura 7(b).



(a) Imagem com marcadores de pele em 1 (branco) e não pele em 0 (preto).



(b) Imagem com a pele a mostra e não pele em preto.

Figura 7. Primeiro resultado para a segmentação.

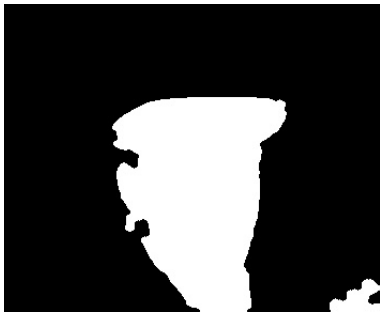
Por fim, com o objetivo de melhorar o resultado, aplica-se um filtro morfológico e se obtém as imagens da figura 8

#### IV. CONCLUSÃO

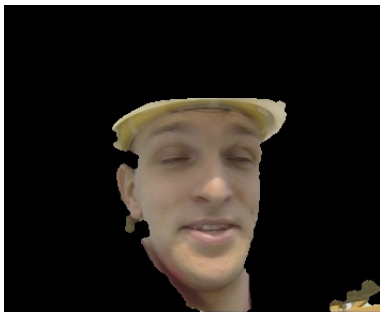
A partir do problemas solucionados anteriormente é possível ver a aplicação da segmentação do processamento de imagens em situações reais para a automatização de processos. Também é possível ver a dificuldade que é segmentar corretamente sem erros utilizando-se apenas de processamento de imagens. Por fim se vê que *watershed* pode ser uma poderosa ferramenta para segmentação, aliada de outros pré-processamentos e que o algoritmo de *K-means* é simples, porém útil para a classificação.

#### REFERÊNCIAS

- [1] Slides de Aula do professor Bruno Macchiavello.
- [2] "OpenCV: OpenCV modules", Docs.opencv.org, 2019. [Online]. Disponível: <https://docs.opencv.org/4.1.0/>. [Acessado: 21- Abril- 2019]
- [3] "NumPy Reference — NumPy v1.16 Manual", Docs.scipy.org, 2019. [Online]. Disponível: <https://docs.scipy.org/doc/numpy/reference/>. [Acessado: 21- Abril- 2019]



(a) Imagem com marcadores de pele em 1 (branco) e não pele em 0 (preto).



(b) Imagem com a pele a mostra e não pele em preto.

Figura 8. Resultado melhorado para a segmentação.