# Introdução

Uma das tarefas mais comuns e cruciais na área de processamento de imagens digital é a segmentação, essa tarefa é visa dividir ou separar objetos de uma imagem em diferentes camadas. [1, 2] O nível de detalhe dessas camadas depende do problema em questão, ou seja, a segmentação termina quando os objetos/regiões de interesse são detectados. A precisão nessa faze é vital para o sucesso da aplicação final. Existem técnicas para segmentação que analisam as bruscas mudanças de intensidade (como bordas) ou então a divisão da imagem em regiões semelhantes de acordo com um critério prévio, o que é feito na limiarização, crescimento de região e fusão de regiões, por exemplo. Não existe um método padrão que funciona bem para todos os problemas a escolha do mesmo deve ser baseada no resultado final que esse apresenta.

Entre os métodos citados anteriormente existem os de agrupamento (*clustering),* neste trabalho discutiremos sobre o K-Means que pertence a esse grupo. O método é utilizado em diversas áreas principalmente na mineração de dados, onde esse divide as informações e m k diferentes grupos, o foco nesse trabalho é usar a mesma ideia e separar a imagem em k grupos diferentes.

# Segmentação por Agrupamento

A segmentação por agrupamento é um processo onde um conjunto de dados é separado em grupos (clusters), onde os elementos possuem similaridades entre si, mas são dissimilares entre elementos de outros grupos[2]. Como já dito usaremos o método k-means que usa o mesmo conceito para segmentar a imagem. Com isso iremos separar os pixels da imagem em grupos que por sua vez serão compostos por pixels que se assemelham as características desse grupo, como textura, nível de cinza e brilho, por exemplo.

A métrica mais utilizada é a Euclidiana, mas também pode ser aplicada Distância Quarteirão e Distancia Xadrez [2]. No caso desse trabalho será utilizado a Euclidiana, na forma:

Onde p1 ={r1, g1, b1} e p2={r2, g2, b2} são pixels.

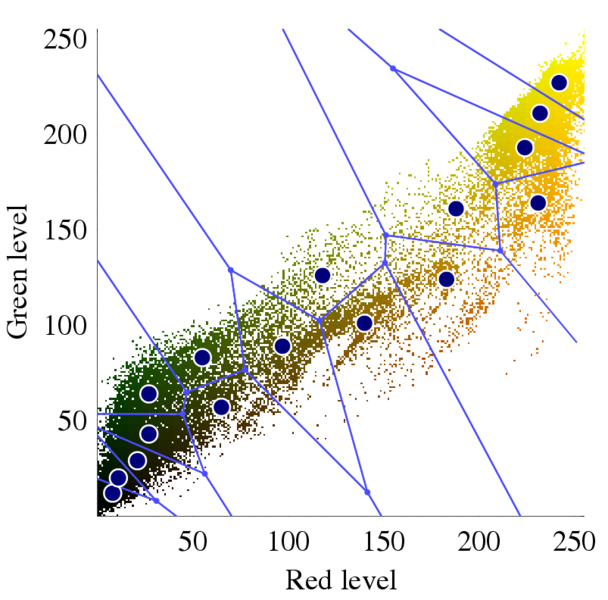
Para encontrar os pontos mais próximos do cluster preciso calcular a distância entre todos e definir qual deles é mais próximo dos k clusters que estou usando. Após encontrar esses pontos calculo a média entre todos e atribuo esse valor ao meu cluster. Farei isso por uma quantidade de vezes especificas ou até que a posição do cluster pare de variar.

Neste exemplo o valor dos clusters foi atribuído aleatoriamente, após isso calcularei a distância entre todos os pontos(pretos) em relação a cada cluster. Com esses valores comparo qual deles tem a menor distancia em relação ao cluster, por exemplo:

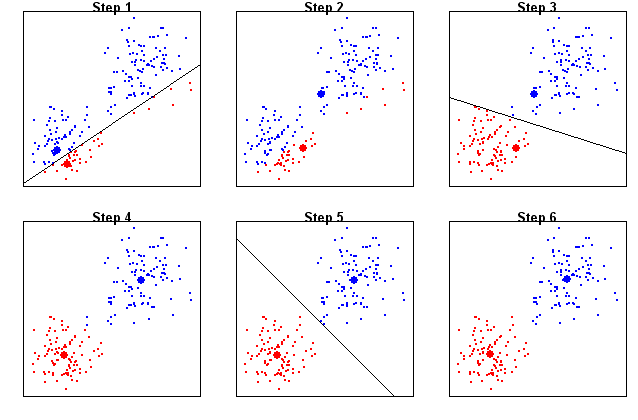
|  |  |  |  |
| --- | --- | --- | --- |
| Pontos | Cluster A | Cluster B |  |
| P1 | 20 | 10 | Distância |
| P2 | 30 | 50 |
| P3 | 30 | 25 |
| P4 | 7 | 25 |
| P5 | 35 | 80 |

Analisando a tabela acima vemos que os pontos P1 e P3 estão mais “próximos” ao ponto B e o restante ao ponto B.

Sabendo disso calculo a média entre esses dois pontos, como são cores RGB ou tom de cinza o valor a ser usado é essa cor, então se P1 é um pixel cinza de 50 e P2 outro pixel de valor 70, o novo valor do cluster A é: . O mesmo processo é feito com o cluster A e é repetido até que os mesmos parem de se mover ou a partir de um limite de iterações.



As imagens acima demonstram o que o processo de agrupamento faz. À esquerda temos a imagem original e a direita um gráfico demonstrando a forma que os pixels se espalharam no plano. Cada um dos pontos azuis maiores representa um centroide e as linhas delimitam os grupos. Ambas foram retiradas de [5].

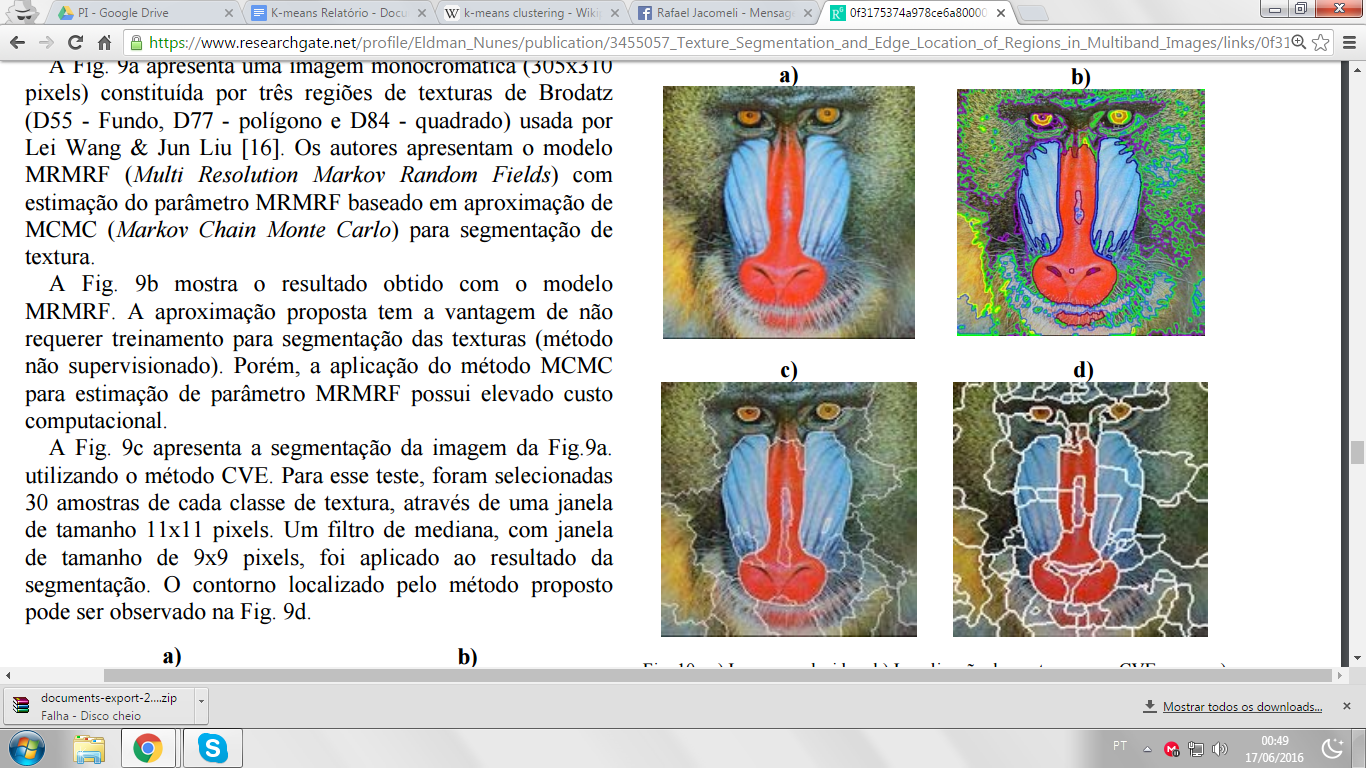


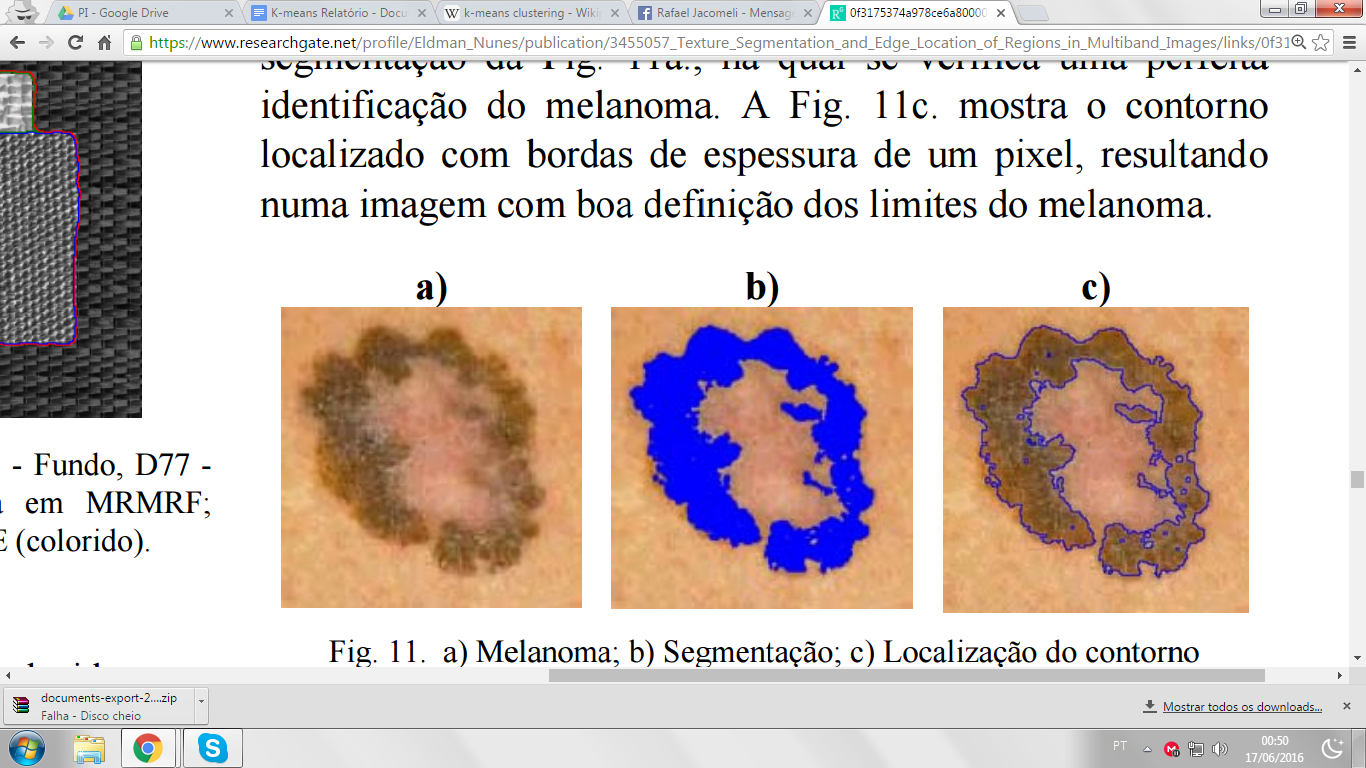
Na imagem acima podemos ver passo a passo como o método se comporta. No Step/Passo 1 os dois clusters foram jogados em pontos aleatórios, podemos ver pelas cores quais pontos eles atingem, então calculamos a média entre suas distâncias e encontramos uma nova posição para cada cluster. Isso é repetido até que os mesmos se estabilizem.

Aplicações

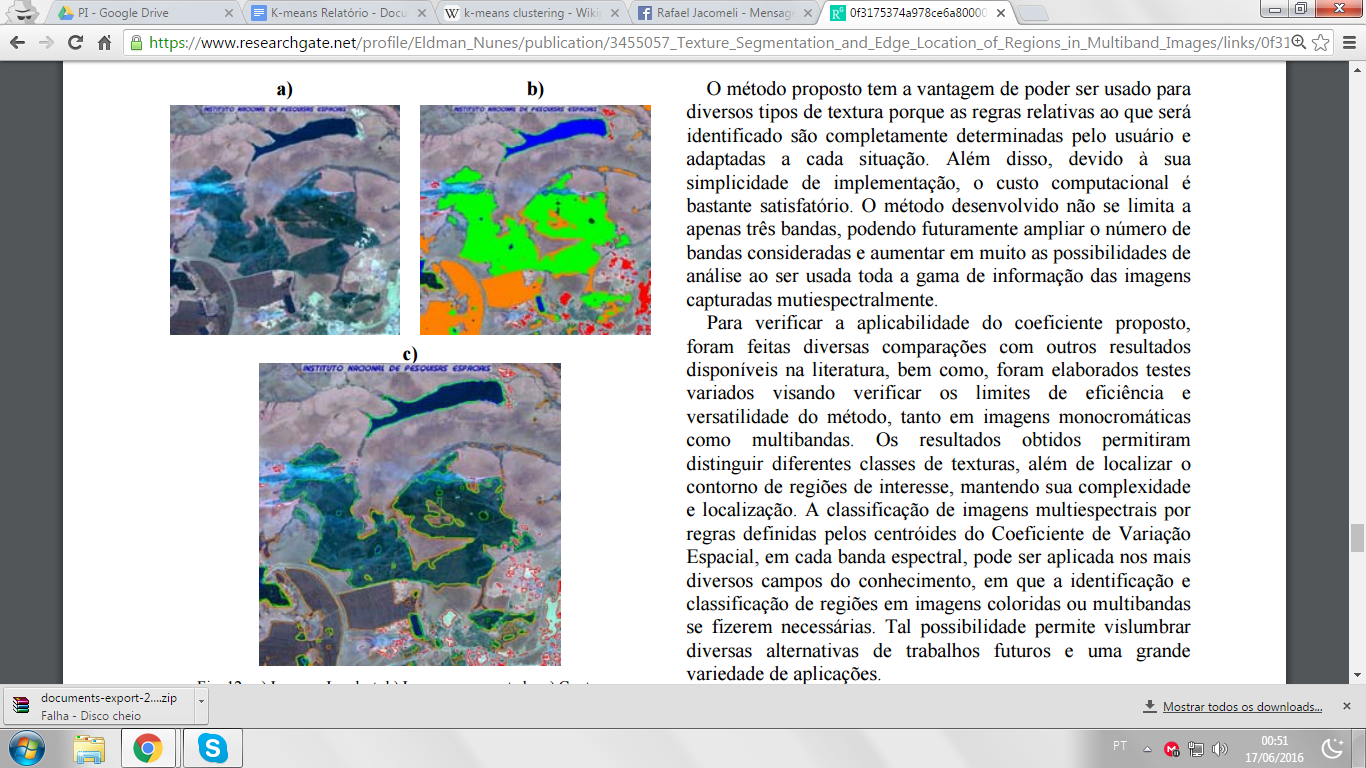
Como k-means é um método de aprendizado não supervisionado ele acaba sendo usado em diversas áreas como astronomia, agricultura, geologia, data mining e visão computacional [5]. Como o objetivo do método é agrupar grupos semelhantes para criar a segmentação logo vemos uma gama extensa de formas de uso, já que a segmentação é utilizada em varias aplicações. Para exemplificar melhor apresentaremos alguns exemplos:

## SEGMENTAÇÃO DE TEXTURA + CONTORNO DE REGIÕES



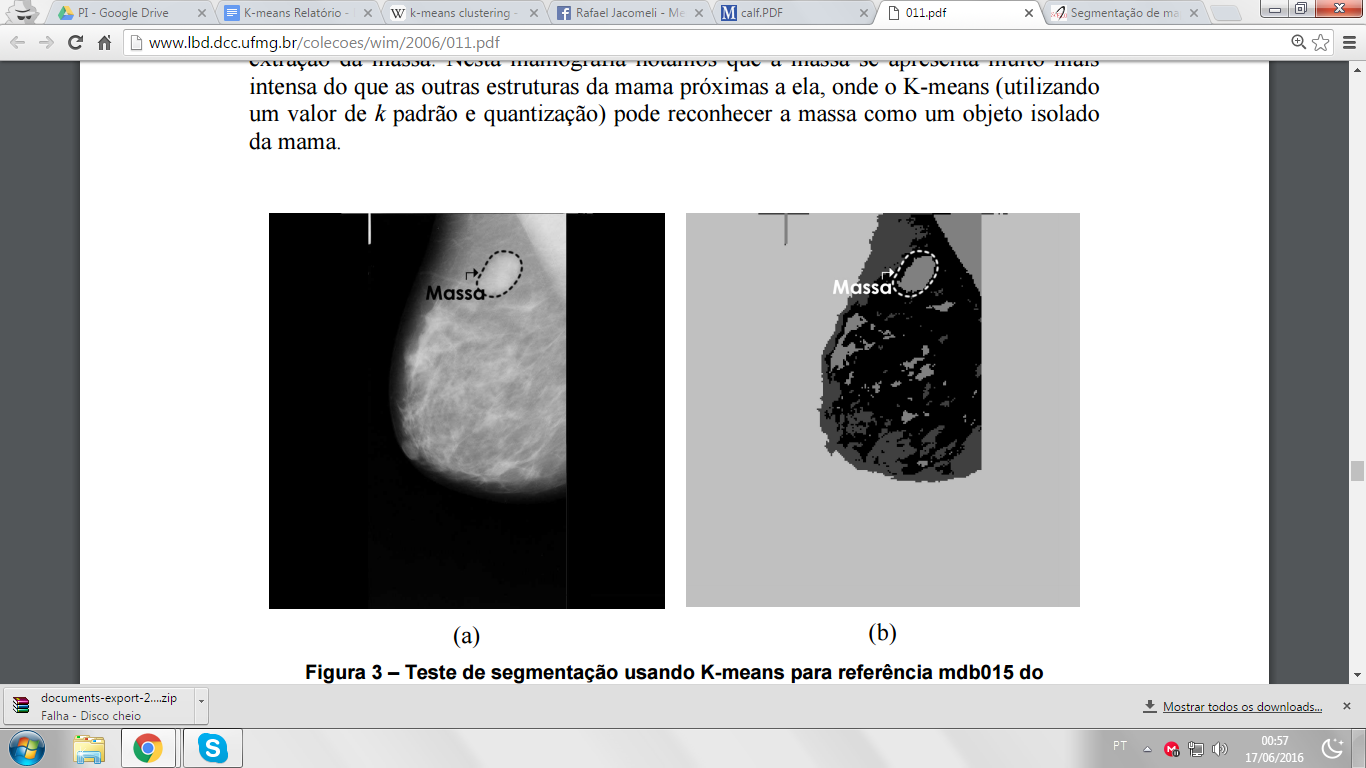


1. Melanoma; b) Segmentação; c)Localização do Contorno.



As três imagens acima foram retiradas de [7], nele o autor enfatiza o fato de que o método é muito vantajoso computacionalmente, é de fácil implementação e produziu resultados melhores do que as imagens que foram utilizadas como base.

## MEDICINA



Nessa aplicação o autor utilizou o k-means e knn para a classificação das imagens para identificar câncer de mama sob a analise da mamografia. Ele destaca que apesar de uma taxa de acerto de 85,13% os resultados poderiam ser melhores se possuíssem mais fotos e o Fuzzy C-Means como algoritmo de agrupamento e classificação baseado na matriz de pertinência.

# Algoritmo/Código

Para a aplicação criamos uma classe chamada cluster que armazena os pontos que cada cluster agrupa e suas cores em RGB. Dentro dessa classe encontramos o método que agrupa os pontos, calcula a distância euclidiana e gera valores aleatórios para os clusters. Foi testado uma forma alternativa em que se atribuía um valor do intervalo de 0 à 255 de acordo com a quantidade de clusters, mas como não ouve melhora no resultado optei por utilizar o valor randômico.

Dentro da classe kmeans temos o código que atribui o valor do cluster para cada pixel. Como nossa entidade Cluseter possui um atributo que é uma lista de pon

# Referências

[1] R.C. Gonzalez e R.E. Woods, “Processamento de Imagens Digitais”, Edgard Blucher Ltda, 2000. pp. 454 – 460.

[2] Uma Versão Intervalar do Método de Segmentação de Imagens Utilizando o K-means

[3] k-Means Image Segmentation Disponível em: http://www.cs.otago.ac.nz/cosc450/assignments/assignment2\_2015.pdf Acessado: 17/06/2016

[4] Entendendo como funciona o algoritmo de Cluster K-Means - Disponível em: http://www.diegonogare.net/2015/08/entendendo-como-funciona-o-algoritmo-de-cluster-k-means/ Acessado: 17/06/16

[5] K-Means Clustering

Disponível em: https://en.wikipedia.org/wiki/K-means\_clustering Acessado: 17/06/16

[6] B, A. R. - **Segmentação de Texturas por Análise de Complexidade**

Disponível em: <http://www.dcc.ufla.br/infocomp/index.php/INFOCOMP/article/viewFile/126/111> Acessado: 17/06/2016.

[7] Texture Segmentation and Edge Location of Regions in Multiband Images Disponível em: ttps://www.researchgate.net/profile/Eldman\_Nunes/publication/3455057\_Texture\_Segmentation\_and\_Edge\_Location\_of\_Regions\_in\_Multiband\_Images/links/0f3175374a978ce6a8000000.pdf Acessado: 17/06/2016.

[8] Identificação de Massas em Mamografias usando Textura, Geometria e Algoritmos de Agrupamento e Classificação. Disponível em: <http://www.lbd.dcc.ufmg.br/colecoes/wim/2006/011.pdf> Acessado: 17/06/2016.

[9] Unsupervised Image Classification: K Means

Disponível em: https://www.youtube.com/watch?v=9LqdSDxQNgw Acessado 17062016.

[10] K-means clustering Disponível em: <http://sherrytowers.com/2013/10/24/k-means-clustering/> Acessado: 17/06/2016

Obs.: Algumas das referencias não foram citadas, mas foram de vital importância para o entendimento do método, principalmente [9].