

Da Avaliação e Quantização de Superpixels em Segmentação de Lesões de Pele

Davi Vilarinho, João Pedro Ramires, Mateus Carmo

I. INTRODUÇÃO

A pele é um órgão fundamental para a vida humana e, dentre suas funções, destaca-se a protetiva. Na medida em que proporciona ao organismo mecanismos de defesa contra o ambiente externo, com o tempo ocorrem surgimento de lesões que podem ser benignas ou malignas. Uma dessas é o melanoma. É fundamental que pessoas, notando mudanças na pele, novas pintas ou sinais, verifiquem com um profissional o estado de risco que aquilo acarreta para a pessoa. Entretanto, o diagnóstico pode não ser necessariamente trivial e, com os avanços tecnológicos, profissionais da área da saúde cada vez mais utilizam de mecanismos computacionais para realizá-lo, o que configura os *Computer-Aided Diagnostics* (CAD, Diagnóstico Assistido por Computador). A principal técnica médica é a chamada *ABCD*, onde avalia a assimetria, borda, cor e diâmetro de uma lesão. Para alcançá-los via computador, utiliza-se, principalmente, técnicas de melhoramento de imagem, segmentação e inteligência artificial, assistidas ou não por profissionais na área.

O melhoramento, segmentação, descrição e reconhecimento de um objeto são áreas tênuas e próximas que atuam em conjunto no estudo de uma imagem, bem como dependem do atual conhecimento da área em questão. Um dos artigos na área propõe o uso de um método semi-automático [1], ou seja, com apontamento de um profissional das regiões importantes na imagem, de forma a encontrar a forma da lesão em específico e fornecer bordas fiéis aos testes de segmentação pré-existentes em *datasets* públicos. O artigo em questão utilizou principalmente 1200 e 1400 superpixels.

Este artigo tem por objetivo *melhorar a etapa de pré-processamento* do artigo base [1], ao alinhar e testar o menor particionamento possível de superpixel e analisar quão próximo com métodos de segmentação simples e aceitos pela comunidade científica eles se aproximam da *ground truth* dos *datasets* utilizados. A premissa é que, se tais análises são simples e geram bons resultados, um médico também seria capaz de julgar facilmente os resultados obtidos com o refino de métodos de segmentação. Assim, *diminui-se o esforço do profissional, o custo computacional, e maximiza-se a performance geral do sistema*.

II. RELACIONADOS E ESTADO DA ARTE

No ramo de estudo de lesões na pele, as soluções podem ser classificadas de acordo com o método empregado para encontrá-los. Alguns métodos são por inteligência artificial, outros por limiarização, ou clusterização ou modelos deformáveis. Em todos eles cabe uso automático ou não dos



Figura 1. Lesão de Pele

mesmos, ou seja, pode haver um especialista escolhendo amostras ou ser integralmente independente da ação humana.

Os métodos tradicionais de segmentação de imagem como limiarização (bem representado pelo algoritmo de Otsu [2]), crescimento de regiões, “active contour” [3] ou detecção de descontinuidades usam de atributos da própria imagem como cor, descritores de textura e formato para separar a lesão da pele. São métodos (em geral) de fácil implementação, mas são sensíveis à aparência das lesões, limitada com ruído, luminosidade, cabelos, pintas e outros fatores comuns.

Com auxílio do *arxiv*, foi pesquisado modelos posteriores à 2019 pelos seguintes termos: *melanoma segmentation*, *skin lesions segmentation*, *melanoma auxiliary image segmentation*, todos no ramo de ciência da computação. O objetivo é avaliar métodos já existentes e entender as barreiras existentes e quais resultados são possíveis, uma vez que este artigo visa implementar um método que também será analisado na sequência.

Tabela I
FREQUÊNCIA E ATRIBUTOS DE ARTIGOS COM AS PALAVRAS CHAVES COMUNS

Palavra chave	Frequência	Semi-automático	Performance
Inteligência Artificial	13	0	4
Limiarização	2	1	1
Contornos Deformáveis	2	0	1
Clusterização	4	2	2
TOTAL	21	3	8

A tabela I expressa os métodos comuns no ramo de segmentação de lesões e, principalmente, do melanoma. É importante verificar que o método de inteligência artificial apresentou o menor número de orientação à performance, bem como é importante levar em consideração o viés de maturação, isto é, o fato de métodos semi-automáticos ou não baseados em inteligência artificial terem menor quantidade de

artigos escritos recentemente pode implicar no fato de que estes métodos já são mais maduros, independentemente da sua performance, ou seja, são métodos que alcançaram bons resultados e não se tem uma visão clara de pontos possíveis para melhorias, enquanto métodos de inteligência artificial ou automáticos, por mais que tenham desempenhos inclusive superiores, estão em evolução clara e ainda será possível maior desempenho que o atual.

O artigo [1] que será usado como base implementou um método semi-automático de segmentação de lesões de pele e utilizou métodos de pré-processamento baseado em particionamento em superpixel, clusterização em *Fuzzy C-means* e extração de descritores de textura. Baseia-se na escolha semi-automática de superpixels por um profissional e utilizava apenas datasets públicos. Foi utilizado o *Métrica do índice de Jaccard (TJI)*, e teve acurárias (mínimas) que variavam de 65% à 94.28%.

Observado os métodos recentes e do artigo que oferece as bases para implementação deste, propor-se-à, em seguida, um método baseado em técnicas tradicionais e estáveis afim de medir, descrever e melhorar a etapa de pré-processamento já descrita do artigo base.

III. MÉTODO PROPOSTO

O método proposto tem interesse em realizar de forma eficiente um pré-processamento suficiente para que a etapa de escolha de superpixel por um profissional capacitado [1] possa ser simples e não acarrete num alto número de superpixels, de forma a facilitar a análise da imagem por humano e máquina, bem como encontrar níveis ótimos do sistema.

Como o objetivo do artigo não é criar um método para segmentação, e sim testar o atual existente, então será descrito à seguir as etapas necessárias para geração de material para o teste. Na seção de métricas será mais explícito como será a avaliação com os resultados obtidos pela aplicação dos passos propostos.

A. Remoção de Cabelos

Assim como o artigo base [1], utilizou-se uma implementação do *Dull Razor* [4], que busca frequência de traços longos, finos e escuros que caracterizam pelos para serem removidos, revertendo às cores medianas ao contorno das áreas removidas.

Para realizar esta tarefa, o algoritmo realiza uma série de etapas. Primeiramente, a imagem é convertida em escala de cinza. Ela então é passada por um *black hat filter*, um filtro que enfatiza regiões de tom escuro na imagem, tornando os fios de cabelo (que normalmente se encontram mais escuros nas fotos) mais pronunciados. Após isso, utiliza-se um desenfoque Gaussiano (*Gaussian Blur*) para aumentar e englobar melhor as regiões ao redor dos fios de cabelo, e então utiliza-se um *threshold binário* para transformar essa imagem com os cabelos pronunciados em uma máscara. Por fim, utiliza-se a função *inpaint* da biblioteca OpenCV2 para retocar a imagem original nas regiões demarcadas pela máscara (que, no caso, são as regiões com pelos).

A imagem final terá seus pelos removidos até um certo grau. O algoritmo não é perfeito, e não consegue garantir uma remoção ou reconstrução exata das regiões abaixo dos pelos, mas facilita a aplicação de outros métodos de segmentação ou reconhecimento posteriormente.

O algoritmo original [4] não teve uma boa performance nas imagens dos datasets utilizados nos nossos experimentos, então foram realizadas algumas modificações. Em particular, foram utilizados *kernels* de (20,20) e (9,9) para o *black hat filter* e desenfoque Gaussiano respectivamente, e utilizou-se um valor de 10 para o *threshold binário*.

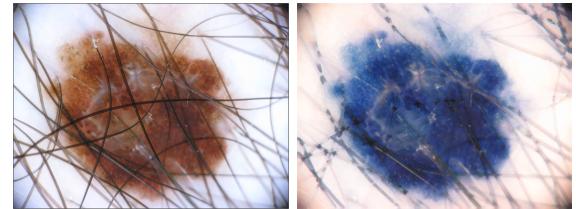


Figura 2. Removendo cabelos difíceis.

B. Particionamento em superpixel

O particionamento em superpixel foi feito com a utilização do SLIC ("Simple Linear Iterative Clustering") implementado pela biblioteca Scikit-image [5] em Python. Este método usa de *k-means* para encontrar superpixels e assim particionar a imagem em porções relevantes [6]. É a parte inicial do artigo base [1] e o objetivo será particionar de 200 a 1550 superpixels, recolorir a imagem original com base nos superpixels e comparar com a *ground truth* disponível.



Figura 3. Segmentando em 450 superpixels ISIC_0000227

O particionamento em superpixel depende de parâmetros como a compactividade (que resulta em superpixels mais quadrados), mantido em 30, e número de segmentos, que é o objeto de estudo deste artigo.

O número de segmentos em uma partição por superpixel é o aspecto chave porque indica a quantidade de pixels que foram agrupados com as k-médias, e diminui o *overhead* das operações com a imagem, bem como cria regiões menos heterogêneas, destacando mudanças.

C. Limiarização de Otsu

O Método de Otsu [2] é um método muito famoso e simples na segmentação de imagens. Seu uso se resume à encontrar automaticamente um limiar e dividir os pixels das imagens em primeiro e segundo plano (ou duas regiões).

No caso deste artigo, imagem já processada nas demais etapas, em escala de cinza do componente luminosidade do HSI superpixel serão submetidos ao algoritmo que é rápido e encontrará as regiões da lesão em questão.

Como encontra regiões bimodais, o resultado obtido é bem similar ao que o método ABCD busca e é binário como a *ground truth* dos *datasets*, sendo assim a escolha do método tem por objetivo simular a técnica mais rápida e simples, de forma que, ainda na fase de seleção por um especialista, já tenha a máxima chance de encontrar a região em que a lesão se expande.

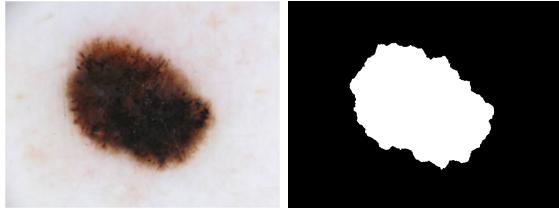


Figura 4. Aplicação do Otsu em imagens já particionada em superpixel

D. Chan-Vese Active Contour

Outro método de segmentação, mas baseado em "Active Contour", um dos principais métodos em modelos deformáveis, tem por objetivo ser alternativo ao Otsu neste artigo. Encontra diferença de pontos com base na área em que já está inserido e possui implementação em Python[5], portanto foi um candidato natural para realizar testes do que considera-se um método simples e abrangente.

O Chan-Vese considera os pesos de diferentes intensidades na imagem em múltiplas iterações, bem como as iterações são distintas para o fundo e o primeiro plano.

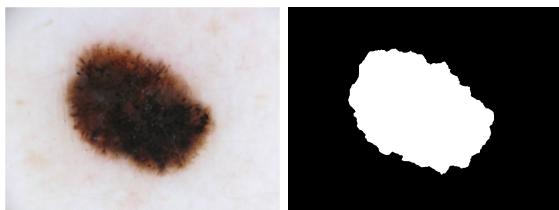


Figura 5. Como o Chan-Vese se comporta em imagens já particionada em superpixel

IV. EXPERIMENTOS

A. Datasets

Um fator muito importante na avaliação de imagens é a fonte de dados envolvida para testes e desenvolvimento. Como não é necessário treinar e testar em fases distintas, este método exige apenas imagens de lesões e a respectiva *ground truth* para verificação.

Foram utilizadas as imagens do dataset público ISIC 2016 - Task 1 [7] para testar a eficiência de cada particionamento. Os resultados são então comparados à "*ground truth*" correspondente de cada imagem, também disponibilizadas no dataset.

São um total de 900 fotos de lesões, totalizando então 1800 imagens e um total de 607.8MB.

As imagens do dataset estão no formato JPEG (*Joint Photographic Experts Group*), com 3 canais de cor. As fotos possuem imensa variedade visual: foram tiradas com câmeras diferentes (resultando em qualidades de imagem diferentes) e em situações de iluminação variadas, possuem tamanhos diferentes e também foram fotos de pessoas com tons de pele diferentes. Ainda, algumas das fotos foram tiradas com câmeras de microscopia, o que gerou um círculo escuro ao redor do foco da imagem. Essas características são capazes de afetar a qualidade da pré-análise e pré-processamento, mas alguns dos algoritmos descritos em no trabalho original [1], como a seleção do envoltório convexo e os modelos morfológicos, poderiam ser capazes de eliminar a maioria destes problemas.

As *ground truths* de cada imagem são PNGs (*Portable Network Graphics*) com apenas um canal de cor *greyscale*. Elas refletem suas fotos correspondentes, coloridas em branco (valor de byte 255) na região onde há uma lesão, e em preto (valor de byte 0) onde não há a lesão. Elas são formatadas dessa forma para que possam ser facilmente comparadas por meio de métodos como *Score de Jaccard*.

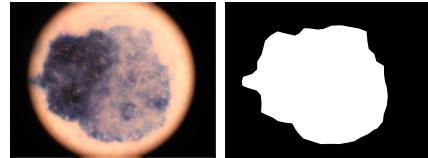


Figura 6. Imagens em que a câmera pode impactar os resultados.

B. Métricas de Avaliação

As métricas de avaliação, para fins de compatibilidade de resultados, serão todos iguais ao [1], ou seja: 'positivo' será usado para designar áreas que são lesão e 'negativo' as que não são, mas será considerada o *Score de Jaccard*.

Ao contrário do artigo base, esta escolha se deve ao fato de que o *Índice de Jaccard* (TJI) é uma aplicação que cria limiares mínimos e é mais adequada para testes do eficácia do método de segmentação [1]. No caso, a escolha do TJI implicaria no fato de que não se compararia a eficiência entre múltiplos tamanhos de superpixel, mas sim a eficácia do método referência, que não serão exploradas neste artigo, mas sim nos de sua procedência [2][3][4][6].

Os resultados não têm como objetivo superar as métricas já existentes no artigo base, mas sim indicar quais apresentaram melhor pontuação inicial, uma vez que o objetivo é melhorar a imagem disponibilizada ao especialista, e não necessariamente a segmentação final.

Para comparação entre diferentes superpixels e imagens foi usada métricas estatísticas elementares como desvio padrão e média, com objetivo de avaliar o *Score de Jaccard* em cada um dos métodos escolhidos, incluindo o produto de ambos, que foi uma métrica criada aqui para avaliação para considerar o efeito de ambos métodos usados.

C. Método de Experimentação

Para realizar as análises, utilizamos um programa na linguagem de programação *Python*. Este programa coleta as imagens do dataset e as *ground truths* correspondentes, e aplica sobre elas o método proposto para cada uma das quantidades de superpixel descritas (após a aplicação do algoritmo *DullRazor*). Calcula-se os valores do *Score de Jaccard* para tanto Otsu e Chan-Vese, e então os dados pertinentes são escritos em um arquivo CSV, tabelando o nome da imagem com seu valor de superpixel e *Scores*.

É importante notar que o algoritmo de *thresholding* Otsu tenta marcar as partes escuras como fundo, principalmente. Em muitos casos, isso faz com que as regiões de lesão segmentadas fiquem marcadas com a cor preta, enquanto o resto fica marcado com branco. Isso é um problema, uma vez que as *ground truths* marcam de forma inversa. Porém, simplesmente inverter os resultados do algoritmo resolveram o problema. Por outro lado, o algoritmo de Chan-Vese sofre de um problema similar, mas a maneira como é escolhida a cor de fundo e frente pelo algoritmo é muito incerta, qualquer resultado gera uma quantidade significativa de problemas, então optou-se por manter a original.

D. Avaliação dos Resultados

Para que possamos observar o desempenho do método para diferentes valores de superpixel, executa-se o algoritmo de superpixelização com o parâmetro de quantidade de segmentos. Usa-se valores no intervalo entre 200 e 1600, em incrementos de 50. Isso totaliza 28 superpixelizações de cada imagem, e os classificadores Otsu e Chan-Vese são então computados para cada uma.

A análise de parte do dataset então gera uma tabela com 448 entradas, 28 para cada imagem das 16 primeiras imagens do dataset original, onde serão disponíveis o *Score de Jaccard* para cada um dos métodos em separado e seu produto.

E. Resultados

O experimento [8] foi realizado em cerca de duas horas, com aproximadamente 30s por amostra, e foi exportado também as imagens resultantes e da etapa de remoção dos cabelos para fins de estudo.

A tabela II mostra um sumário por quartil dos resultados obtidos, existe a presença de ruído (JAC zero) que será explicado em seguida.

Tabela II
SUMÁRIO DOS RESULTADOS

	Superpixels	Otsu JAC	Chan-vese JAC	Produto JAC
cont	448	448	448	448
média	875.000000	0.619266	0.511514	0.347943
d.p.	404.338884	0.188829	0.339803	0.280630
min	200.000000	0.198860	0.000000	0.000000
25%	537.500000	0.512144	0.196063	0.072384
50%	875.000000	0.639253	0.674996	0.420001
75%	1212.500000	0.762524	0.795482	0.595427
máx.	1550.000000	0.916789	0.942162	0.818969

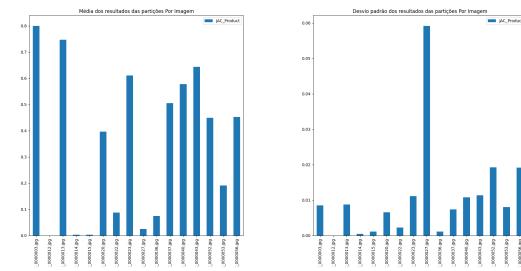


Figura 7. Média e Desvio padrão por imagem

Na tabela III é visualizado a média dos resultados por cada imagem, e é possível visualizar de quem era o ruído (a figura 13).

Tabela III
MÉDIA POR IMAGEM DO DATASET

Imagen	Otsu JAC	Chan-vese JAC	Produto JAC
ISIC_0000003.jpg	0.856081	0.934985	0.800412
ISIC_0000012.jpg	0.436453	0.000022	0.000009
ISIC_0000013.jpg	0.836921	0.893421	0.747705
ISIC_0000014.jpg	0.721524	0.004348	0.003137
ISIC_0000015.jpg	0.597391	0.004689	0.002812
ISIC_0000020.jpg	0.570484	0.694306	0.396094
ISIC_0000022.jpg	0.435892	0.201833	0.087980
ISIC_0000023.jpg	0.727448	0.839452	0.610631
ISIC_0000027.jpg	0.555008	0.044106	0.024784
ISIC_0000036.jpg	0.206507	0.362307	0.074810
ISIC_0000037.jpg	0.701943	0.719436	0.504997
ISIC_0000040.jpg	0.893158	0.647265	0.577993
ISIC_0000043.jpg	0.797516	0.806847	0.643512
ISIC_0000052.jpg	0.654426	0.685645	0.448866
ISIC_0000053.jpg	0.336257	0.566430	0.190482
ISIC_0000056.jpg	0.581240	0.779141	0.452868

Também foi exportado o desvio padrão das imagens na tabela IV e é possível visualizar como cada imagem em si é impactada pela aplicação da metodologia proposta. Uma discussão será feita de como as propriedades da imagem impactam os resultados. Foi plotado também o gráfico das tabelas IV e III na figura 7, o que possibilita observar visualmente o comportamento das imagens.

Tabela IV
DESVIO PADRÃO POR IMAGEM

Imagen	Otsu JAC	Chan-vese JAC	Produto JAC
ISIC_0000003.jpg	0.009704	0.003647	0.008530
ISIC_0000012.jpg	0.014264	0.000041	0.000018
ISIC_0000013.jpg	0.009296	0.007157	0.008780
ISIC_0000014.jpg	0.010630	0.000616	0.000443
ISIC_0000015.jpg	0.015435	0.001686	0.001087
ISIC_0000020.jpg	0.008294	0.004608	0.006547
ISIC_0000022.jpg	0.008575	0.002977	0.002275
ISIC_0000023.jpg	0.009594	0.013140	0.011136
ISIC_0000027.jpg	0.017590	0.103576	0.059204
ISIC_0000036.jpg	0.003861	0.004173	0.001144
ISIC_0000037.jpg	0.008628	0.006711	0.007386
ISIC_0000040.jpg	0.019789	0.010799	0.010798
ISIC_0000043.jpg	0.008630	0.006583	0.011339
ISIC_0000052.jpg	0.017948	0.011645	0.019216
ISIC_0000053.jpg	0.013094	0.003629	0.008031
ISIC_0000056.jpg	0.024196	0.006653	0.019200

Tabela V
MÉDIA POR PARTIÇÃO EM SUPERPIXEL

Superpixel	Otsu JAC	Chan-vese JAC	JAC Product
200	0.631006	0.551500	0.379458
250	0.623813	0.512227	0.351384
300	0.621029	0.511415	0.350449
350	0.618554	0.511941	0.349821
400	0.619202	0.516029	0.349185
450	0.620385	0.510233	0.347928
500	0.621747	0.512196	0.351031
550	0.614522	0.509897	0.345602
600	0.618627	0.512640	0.349728
650	0.613594	0.512197	0.344808
700	0.617505	0.509721	0.346471
750	0.620283	0.508814	0.345673
800	0.618953	0.509332	0.345981
850	0.618796	0.510542	0.347735
900	0.618374	0.509174	0.346079
950	0.620110	0.509899	0.345777
1000	0.615087	0.507831	0.343219
1050	0.615621	0.507653	0.343795
1100	0.621221	0.508833	0.347078
1150	0.616629	0.509193	0.344408
1200	0.618494	0.509201	0.345236
1250	0.619115	0.507488	0.343586
1300	0.620048	0.509659	0.346157
1350	0.617567	0.508911	0.345556
1400	0.618094	0.508894	0.345756
1450	0.621208	0.509258	0.347777
1500	0.621488	0.509716	0.348482
1550	0.618364	0.508011	0.344252

As tabelas V e VI, bem como seus respectivos plots na figura 8 explicitam o análise da dispersão e dos resultados por superpixel. É importante destacar a diferença entre as 3 análises abordadas. A primeira é geral, têm-se apenas resultados do que houve no experimento. No escopo de imagem destaca-se as propriedades internas da imagem e como se relaciona aos resultados obtidos. Enquanto por superpixel trata-se do objetivo deste artigo que é descobrir o ponto ótimo de particionamento em superpixel para segmentar lesões de pele.

Estas tabelas e imagens foram obtidas a partir da análise com a biblioteca Pandas [9] do csv disponibilizado [8] e as análises estão todas presentes no repositório relacionado ao artigo [10].

V. DISCUSSÃO

A. Análise por imagem

Antes de analisar o particionamento, é importante destacar que a natureza de cada lesão muda, então é importante entender que os métodos de Otsu e Chan-Vese naturalmente terão resultados distintos em imagens diferentes.

O método de Chan-Vese se provou muito mais inconsistente, com resultados péssimos, mas também outros quase perfeitos. Como já foi abordado anteriormente, a técnica utilizada para analisá-los em conjunto foi o produto do score em ambas segmentações clássicas e podem ser visualizados na figura 7.

Conclui-se que houve outliers que serão explicados na subseção de ruído, mas removendo-os, verifica-se que abrange um intervalo de aproximadamente 0,19 à 0,8. Este é um resultado abrangente e fica visível que pode ser explicado pelo

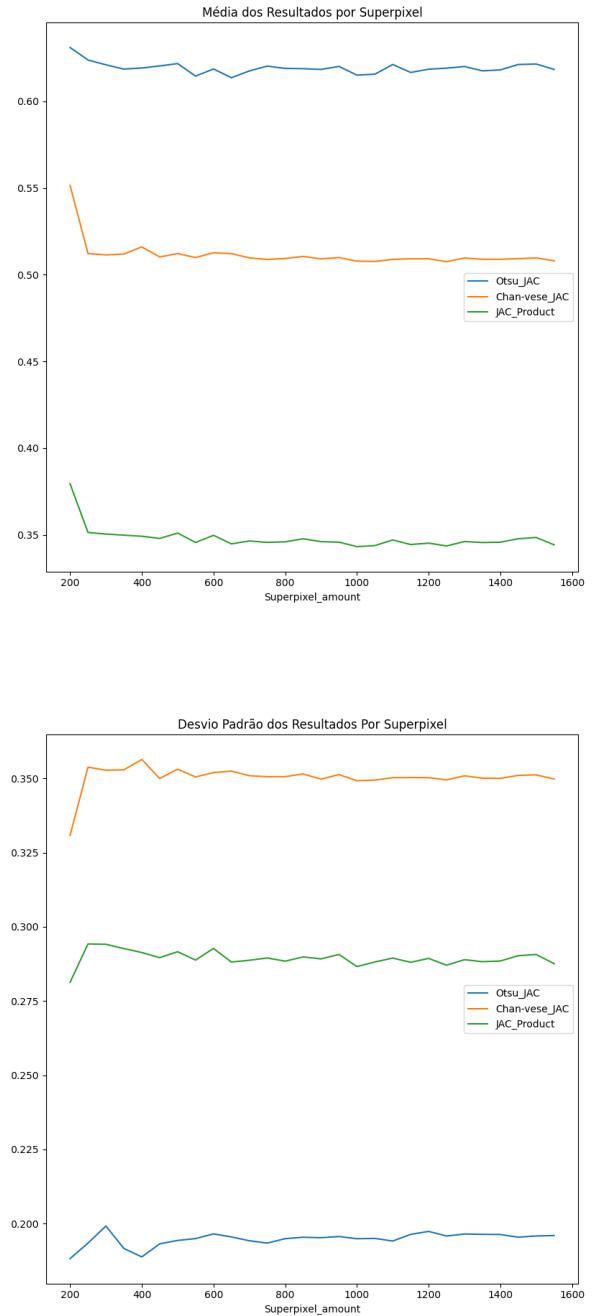


Figura 8. Média e desvio padrão dos Scores de Jaccard por superpixel

Tabela VI
DESVIO PADRÃO POR PARTIÇÃO EM SUPERPIXELS

Superpixels	Otsu JAC	Chan-vese JAC	Produto JAC
200	0.188152	0.330837	0.281303
250	0.193392	0.353822	0.294227
300	0.199139	0.352804	0.294114
350	0.191575	0.352944	0.292677
400	0.188720	0.356407	0.291347
450	0.193143	0.350005	0.289604
500	0.194286	0.353144	0.291611
550	0.194901	0.350536	0.288791
600	0.196499	0.352006	0.292719
650	0.195461	0.352497	0.288128
700	0.194177	0.350944	0.288733
750	0.193376	0.350606	0.289501
800	0.194882	0.350622	0.288428
850	0.195362	0.351558	0.289865
900	0.195203	0.349792	0.289189
950	0.195599	0.351342	0.290695
1000	0.194864	0.349249	0.286608
1050	0.194960	0.349471	0.288137
1100	0.194080	0.350292	0.289476
1150	0.196327	0.350316	0.288035
1200	0.197324	0.350291	0.289365
1250	0.195773	0.349534	0.287065
1300	0.196451	0.350916	0.288933
1350	0.196343	0.350128	0.288239
1400	0.196281	0.350054	0.288482
1450	0.195370	0.351059	0.290268
1500	0.195789	0.351220	0.290676
1550	0.195936	0.349823	0.287589

alto desvio padrão já visto na tabela sumário II quanto ao método de Chan-Vese.

No entanto, as imagens com maior desvio padrão (exetuando-se outliers) são também as com melhor desempenho do método, o que indica que sua eficácia alterna entre as imagens, mas sua precisão muda internamente de acordo com a quantidade de superpixels submetida, o que valida o objeto de estudo deste artigo. O método de Otsu mantém os resultados em níveis satisfatórios e consistentes.

B. Análise por Superpixel

O resultado obtido, ao contrário do esperado, mostrou que quantidades menores de superpixel na verdade tiveram resultados melhores que maiores particionamentos, como vistos na tabela V e no plot de suas médias na figura 8. Isto é um fenômeno interessante e provavelmente se deve à simplicidade dos métodos[2][3] escolhidos. No entanto, é importante destacar que o desvio padrão, tem maior oscilação em níveis inferiores, enquanto alcança uma estabilidade a partir de 450 superpixels.

É importante destacar que o intervalo escolhido foi aritmético, não geométrico. Isso implica que particionamentos iniciais tendem a ter muitas mudanças entre si e diminui na medida em que aumenta (note que de uma quantidade de superpixel 200 para 400 dobra-se, enquanto de 800 para 1000 é apenas 25%).

Um aspecto extremamente relevante é o fato de que a partir de 550 superpixels o resultado na média (de ambos métodos tradicionais) manteve-se próximo (diferenças na ordem de 10^{-3}), ou seja, a partir deste valor os resultados são, em média, iguais.



Figura 9. Da esquerda para direita, aplicação do Otsu em imagens particionadas em 550, 850 e 1550 superpixels. À direita a *ground truth*

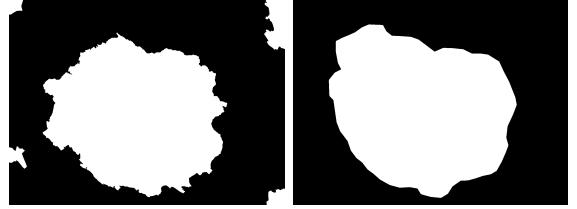


Figura 10. ISIC_0000043 sob Otsu em particionamento de 200 superpixel respectivamente. À direita a *ground truth*.

Bem como o desvio padrão para valores a partir de 800 superpixels também estagnou, mas em ordem maior 10^{-2} . A partir deste ponto as imagens, por outro lado, apresentam menos variação nos resultados da aplicação dos métodos.

São pontos importantes porque indicam onde possivelmente implementações do particionamento em superpixel em lesões de pele podem manter-se e que o aumento para o dataset implica em menores ganhos e maior custo computacional.

Note como grandes diferenças na quantidade de superpixels representados na figura 9 não incorreram em grandes avanços conquistados e teria sido uma alternativa melhor mudar ou aperfeiçoar os métodos de processamento em si, e não a quantidade de superpixel, que é parte do pré-processamento.

Outra discussão pertinente é em relação a menor quantidade de superpixel ter performado melhor. É visível pela tabela V que o melhor resultado foi de 200 superpixels, bem como teve o menor desvio padrão dentre todos os outros valores, implicando que não é algo localizado. Isso pode ser observado com as figuras 10, 11 e 12.

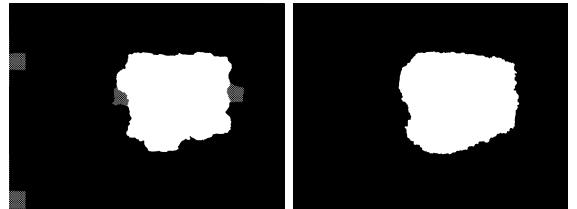


Figura 11. ISIC_0000023 sob Chan-Vese em particionamento de 200 superpixel. À direita a *ground truth*.



Figura 12. ISIC_0000003 sob Chan-Vese em particionamento de 200 superpixel. À direita a *ground truth*.

C. Ruído

É importante salientar que existem imagens cujas metodologia de avaliação empregadas no artigo foram insuficientes.

1) *Dull Razor*: os resultados obtidos e os testes empíricos do método mostraram que existem imagens que o processamento de remoção de cabelos foi insuficiente este é um caso visível na figura 2.

2) *Inconsistência do Chan-Vese*: embora o método de Otsu mantenha uma consistência no que é o resultado positivo de uma lesão ou negativo, o mesmo não ocorreu para o método de Chan-Vese. Durante a execução, é visível que ocorre alternância do que considera fundo ou primeiro plano. Isso se deve ao fato de que o Otsu é um método de limiarização, então como as lesões de pele naturalmente são mais escuras que o tom de pele, a imagem a encontra como a região 0 e, no código, inverte-a. No entanto, o método alternativo por ser *Active Contour*, foca nas bordas e formatos das lesões, então é suscetível à mudanças na própria pele também.

A figura 13 demonstra bem como causa um grande impacto, justamente por se mostrar um outlier na tabela III, bem como as imagens ISIC_0000027, ISIC_0000036, ISIC_0000015 e ISIC_0000014, onde obtiveram resultado próximos à zero com o Chan-Vese e, portanto, o produto resultou em zero (e não se alterou significativamente entre diferentes particionamentos em superpixel).



Figura 13. Lesão de pele ISIC_0000012 em que o resultado do Chan-Vese (à direita) discrepou bastante do Otsu (ao centro) à 1050 superpixels e da ground truth.

3) *Câmera*: a presença de efeito na câmera de microscopia utilizada pode diminuir o limiar nas etapas de remoção de cabelos (com a limiarização binária), Otsu e Chan-Vese, uma vez que interpreta as bordas como parte da imagem, como visto na figura 14.

Ainda assim, o produto do Score de Jaccard da figura 14 foi, em média, alto III (0.747705). Isso implica no fato de que este ruído não foi significativo quanto os demais.

VI. CONCLUSÃO E TRABALHOS FUTUROS

Com a utilização de técnicas de segmentação de imagens tradicionais, foi possível avaliar que, para o dataset utilizado

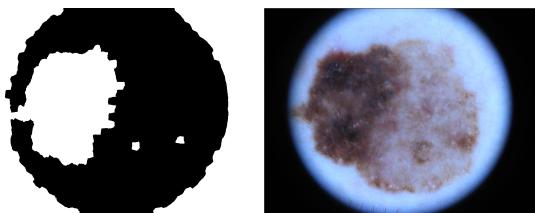


Figura 14. Na imagem ISIC_0000013 é visível como as bordas da lente de microscopia impactam o resultado, com particionamento de 1000 superpixels e o método de Otsu. À direita a imagem original. Note que a borda preta foi marcada em branco, como se fosse parte do melanoma.

no escopo deste artigo, a partir de 800 superpixels as imagens apresentam uma estagnação dos resultados em dispersão, ou seja, apresentam resultados similares entre distintas imagens inclusive. Em média, este valor é alcançado a partir de 550 pixels, mas levemente com mais variação.

Além disso, quantidades menores de superpixels performam melhor, em média e dispersão, que quantidades maiores, com bons resultados para 400, 450 superpixels e, o mais surpreendente, 200 superpixels.

Portanto, mostra-se desnecessário o enfoque em 1200 superpixels que foi mostrado à profissionais no artigo base [1]. Embora exista uma imagem que demonstre utilização de 400, não é reiterada depois. Logo, pelos resultados deste estudo, poderia ser utilizado uma menor quantidade.

O autor também comenta que em certos *datasets*, as imagens possuem muita distinção entre si. Neste caso, uma quantidade ótima que neutraliza o efeito das imagens é 800 superpixels.

É importante destacar que a abrangência deste artigo não pôde ser ampliada devido às limitações de *hardware* dos autores e destaca-se para trabalhos futuros a avaliação de outras técnicas de segmentação para avaliar a quantidade de superpixels, substituindo ou aperfeiçoando principalmente o Chan-Vese, ou também a avaliação de um intervalo próximo ao ótimo de 200 superpixels. Também é importante a adequada parametrização e descrição do método empregado para remoção de cabelos *Dull Razor* e a remoção do ruído causado pela câmera de microscopia.

REFERÊNCIAS

- [1] E. Santos, R. Veras, H. Miguel, K. Aires, M. L. Claro, and G. B. Junior, "A skin lesion semi-supervised segmentation method," in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*. IEEE, 2020, pp. 33–38.
- [2] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [3] P. Getreuer, "Chan-Vese Segmentation," *Image Processing On Line*, vol. 2, pp. 214–224, 2012, <https://doi.org/10.5201/ipol.2012.g-cv>.
- [4] J. Velasquez, "Dullrazor-algorithm," 2021. [Online]. Available: <https://github.com/BlueDokk/Dullrazor-algorithm>
- [5] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "scikit-image: image processing in python," *PeerJ*, vol. 2, p. e453, 2014.
- [6] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.
- [7] D. Gutman, N. C. F. Codella, E. Celebi, B. Helba, M. Marchetti, N. Mishra, and A. Halpern, "Skin lesion analysis toward melanoma detection: A challenge at the international symposium on biomedical imaging (isbi) 2016, hosted by the international skin imaging collaboration (isic)," 2016. [Online]. Available: <https://arxiv.org/abs/1605.01397>
- [8] J. P. R. Davi Vilarinho, Matheus Carmo, "2023-01-22_output.zip," 2023. [Online]. Available: https://s3.amazonaws.com/vilarinho-mounting-point/2023-01-22_output.zip
- [9] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020. [Online]. Available: <https://doi.org/10.5281/zenodo.3509134>
- [10] J. P. R. Davi Vilarinho, Matheus Carmo, "multimidia-artigo-final," 2023. [Online]. Available: <https://github.com/DaviVilarinho/multimidia-artigo-final>