

# Avaliação de superpixels para segmentação de imagens

Felipe Augusto Lima Reis<sup>1</sup>

<sup>1</sup>PUC Minas - Pontifícia Universidade Católica de Minas Gerais

`falreis@sga.pucminas.br`

**Abstract.** *Superpixels are structures that group similar pixels into sets that reflect aspects of the image. This article evaluates the use of SLICO superpixels and partition hierarchy for segmentation. Using **neural networks** for segmentation, the results of superpixels images with different levels of granularity and untreated images were compared to the ground-truth. The article also evaluates the training time of neural networks for superpixels based images. For training and evaluation, the Berkeley Segmentation Data Set (BSDS500) [Arbelaez et al. 2011] image was used.*

**Resumo.** *Superpixels são estruturas que agrupam pixels semelhantes em conjuntos que refletem aspectos da imagem. Este artigo avalia a utilização superpixels SLICO e hierarquia de partições para segmentação. Utilizando **redes neurais** para segmentação, os resultados de imagens utilizando superpixels com diferentes níveis de granularidade e imagens sem tratamento foram comparadas em relação ao ground-truth. O artigo também avalia o tempo de treinamento das redes neurais para imagens com superpixels em relação às imagens originais. Para treinamento e avaliação foram utilizadas imagem do Berkeley Segmentation Data Set (BSDS500) [Arbelaez et al. 2011].*

## 1. Introdução

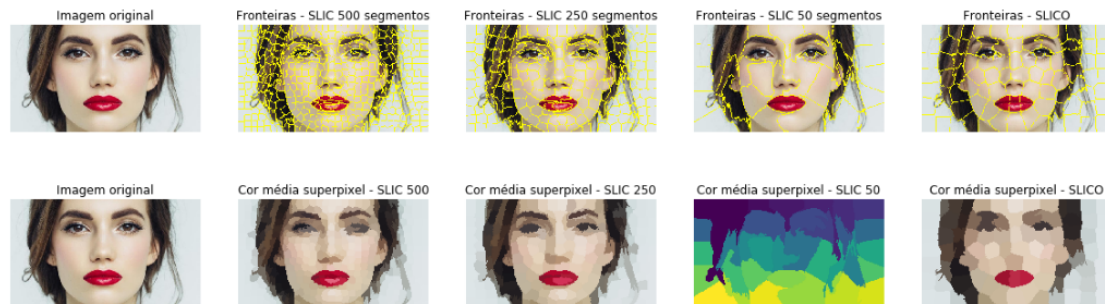
O trabalho tem como objetivo investigar a utilização de superpixels como passo de pré processamento para segmentação de imagens. O trabalho irá investigar se a utilização de métodos de oversegmentation podem facilitar ou dificultar o processo de treinamento. O trabalho também tentará identificar se o treinamento utilizando imagens pré processadas obtêm resultados semelhantes àqueles utilizando imagens originais, na etapa de validação.

Para confecção do trabalho serão usados artigos de processos de criação de superpixels como SLIC, SLICO e Felzenszwalb. Também serão utilizados artigos de redes neurais para segmentação de imagens, a fim de escolher uma arquitetura de rede adequada para o trabalho.

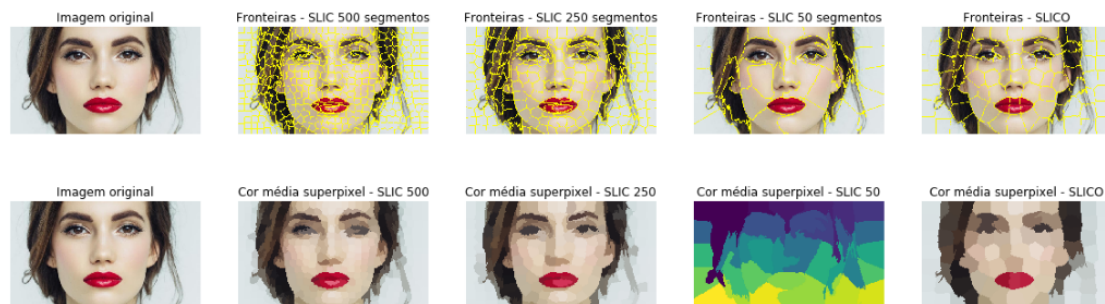
A fim de treinar a rede e avaliar os resultados, será utilizada a base de dados BSDS500 (Berkley Segmentation Data Set and Benchmarks 500), para segmentação de imagens e detecção de contornos.

Para a implementação, irei utilizar a linguagem de programação Python, junto a API SCIKIT-IMAGE, para segmentação. Esta é capaz de gerar os superpixels SLIC e SLICO. Ainda não foi definida a rede neural a ser utilizada, porém devo utilizar redes do tipo Fully Convolutional Network, como disponível no artigo <https://arxiv.org/abs/1411.4038>. Para avaliar os resultados, irei utilizar o projeto SEISM:

Supervised Evaluation of Image Segmentation Methods, para exibir a avaliação das técnicas segmentação, usando as medidas de recordação de precisão para limites e para objetos e partes.



**Figura 1. Coloração pelo valor médio dos superpixels SLIC, para diferentes quantidades de superpixels**



**Figura 2. Fronteiras das Hierarquia de partições utilizando superpixel SLICO.**

O presente trabalho apresenta a seguinte estrutura: a Seção 2 mostra o referencial teórico para construção do trabalho, a Seção 3, exibe os materiais e métodos utilizados nos testes; a Seção 4 mostra os resultados obtidos nos testes realizados e a discussões dos mesmos; a Seção 5 contém a conclusão do artigo, com as considerações finais.

## 2. Referencial Teórico

### 2.1. Superpixels

Superpixels são estruturas que agrupam pixels semelhantes em conjuntos. O agrupamento possibilita a redução de complexidade das tarefas de processamento [Achanta et al. 2012], ao reduzir a quantidade de itens a serem processados. Os superpixels são utilizados na área de visão computacional para solução de vasto número de problemas, como detecção de contorno [Wang et al. 2017], segmentação [Liu et al. 2016] e localização de objetos [Fulkerson et al. 2009].

Superpixels, segundo [Felzenszwalb and Huttenlocher 2004], devem capturar importante grupos ou regiões, refletindo aspectos da imagem. Devem também ser executados em tempo próximo ao linear em relação a quantidade de pixels. Existem diversas abordagens para a geração de superpixels [Achanta et al. 2012]. Dentre elas, podemos classificá-las, segundo o método de agrupamento em:

- *Algoritmos baseados em grafos*: utilizam abordagem baseadas em grafos para correlação entre pixels e criação dos conjuntos;
- *Algoritmos baseados em gradiente ascendente*: utilizam métodos de gradiente ascendente iterativamente até que os critérios de convergência correspondam a forma de um superpixel;
- *Algoritmos de clusterização iterativo*: utilizam métodos de clusterização, como o k-means, para produção de superpixels.

Dentre os algoritmos de clusterização iterativos podemos citar o algoritmo SLIC (*Simple Linear Iterative Clustering*) [Achanta et al. 2012]. Esse algoritmo é uma adaptação do algoritmo *k-means* para geração de superpixels, com diminuição de cálculos no processo de otimização, e combinação ponderada de cor e proximidade espacial, para permitir a definição do tamanho e capacidade dos superpixels [Achanta et al. 2012].

### 2.1.1. Superpixels SLIC

O algoritmo SLIC utiliza um único parâmetro  $k$ , correspondente a quantidade aproximada de superpixels. A fim de produzir tamanhos semelhantes, o intervalo analisado é  $S = \sqrt{N/k}$ , onde  $N$  é o número de pixels da imagem. Os centros são movidos para o local das sementes correspondentes a posição mais baixa do gradiente em uma vizinhança de  $3 \times 3$ , evitando que superpixels sejam centrados nas bordas ou em uma posição de ruído [Achanta et al. 2012]. Em seguida, cada pixel é associado com o centro do cluster mais próximo, de modo que a região de busca se sobreponham [Achanta et al. 2012]. A fim de aumentar o desempenho do algoritmo, a região de busca é limitada em 2 vezes o tamanho aproximado do superpixel  $S$ , gerando busca em uma área  $2S \times 2S$  [Achanta et al. 2012]. Em seguida, um passo de atualização dos centros dos clusters e computa o erro residual  $E$  [Achanta et al. 2012]. O algoritmo disponível na figura 3, resume as informações descritas nesse parágrafo.

Para o algoritmo descrito na figura 3, é necessário compreender o método para cálculo da medida de distância  $D$  entre os conjuntos. Devido ao algoritmo trabalhar no *colorspace* CIELAB, com o espaço-plano *labxy*, a posição do pixel pode assumir um intervalo de valores de acordo com o tamanho da imagem. Com isso, o cálculo da distância não pode ser feito utilizando uma distância euclidiana, sendo necessária uma prévia normalização da proximidade espacial e de cor. Para isso é utilizado a fórmula  $D = \sqrt{dc^2 + (ds/S)^2 * m^2}$ , onde  $D$  corresponde a distância em 5 dimensões do espaço *labxy*,  $dc$  e  $ds$  correspondem à proximidade de cores e espacial e  $m^2$  corresponde a distância máxima entre cores no cluster [Achanta et al. 2012]. Para o cálculo da distância em imagens em escala cinza, é utilizada a distância Euclidiana.

Uma etapa extra no processo de pós processamento é a união de *pixels orfãos*. Esses pixels são adicionados ao cluster mais próximo usando o algoritmo de componentes conexos [Achanta et al. 2012].

Devido a limitação do espaço de pesquisa do algoritmo SLIC, a complexidade do algoritmo é  $O(n)$ , enquanto outros algoritmos que utilizam k-means para segmentação tem custo  $O(k^N)$  [Achanta et al. 2012].

---

```

SLIC superpixel segmentation
/* Initialization */
Initialize cluster centers  $C_k = [l_k, a_k, b_k, x_k, y_k]^T$  by
sampling pixels at regular grid steps  $S$ .
Move cluster centers to the lowest gradient position in a
 $3 \times 3$  neighborhood.
Set label  $l(i) = -1$  for each pixel  $i$ .
Set distance  $d(i) = \infty$  for each pixel  $i$ .
repeat
  /* Assignment */
  for each cluster center  $C_k$  do
    for each pixel  $i$  in a  $2S \times 2S$  region around  $C_k$  do
      Compute the distance  $D$  between  $C_k$  and  $i$ .
      if  $D < d(i)$  then
        set  $d(i) = D$ 
        set  $l(i) = k$ 
      end if
    end for
  end for
  /* Update */
  Compute new cluster centers.
  Compute residual error  $E$ .
until  $E \leq \text{threshold}$ 

```

---

Figura 3. Algoritmo SLIC - Adaptado de [Achanta et al. 2012]

### 2.1.2. Superpixels SLICO

### 2.1.3. Clusters

### 2.1.4. Hierarquia de partições

## 2.2. Segmentação de Imagens

*Image segmentation is an important processing step in many image, video and computer vision applications. Extensive research has been done in creating many different approaches and algorithms for image segmentation, but it is still difficult to assess whether one algorithm produces more accurate segmentations than another, whether it be for a particular image or set of images, or more generally, for a whole class of images. To date, the most common method for evaluating the effectiveness of a segmentation method is subjective evaluation, in which a human visually compares the image segmentation results for separate segmentation algorithms, which is a tedious process and inherently limits the depth of evaluation to a relatively small number of segmentation comparisons over a predetermined set of images. Another common evaluation alternative is supervised evaluation, in which a segmented image is compared against a manually-segmented or pre-processed reference image.*

## 2.3. Redes Neurais

## 3. Materiais e Métodos

## 4. Testes, Resultados e Discussões

### 4.1. Subsections

**Tabela 1. Variables to be considered on the evaluation of interaction techniques**

	Value 1	Value 2
Case 1	$1.0 \pm 0.1$	$1.75 \times 10^{-5} \pm 5 \times 10^{-7}$
Case 2	0.003(1)	100.0

## 5. Conclusão

### Referências

- Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282.
- Arbelaez, P., Maire, M., Fowlkes, C., and Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916.
- Felzenszwalb, P. F. and Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181.
- Fulkerson, B., Vedaldi, A., and Soatto, S. (2009). Class segmentation and object localization with superpixel neighborhoods. In *2009 IEEE 12th International Conference on Computer Vision*, pages 670–677.
- Liu, T., Seyedhosseini, M., and Tasdizen, T. (2016). Image segmentation using hierarchical merge tree. *IEEE Transactions on Image Processing*, 25(10):4596–4607.
- Wang, X.-Y., Wu, C.-W., Xiang, K., and Chen, W. (2017). Efficient local and global contour detection based on superpixels. *Journal of Visual Communication and Image Representation*, 48:77 – 87.