

# IF69D-S11 Proc. Digital de Imagens APNP

## Aula 08

Universidade Tecnológica Federal do Paraná - UTFPR - Curitiba  
Departamento Acadêmico de Eletrônica - DAELN  
Curso de Engenharia Eletrônica  
Prof. Gustavo B. Borba

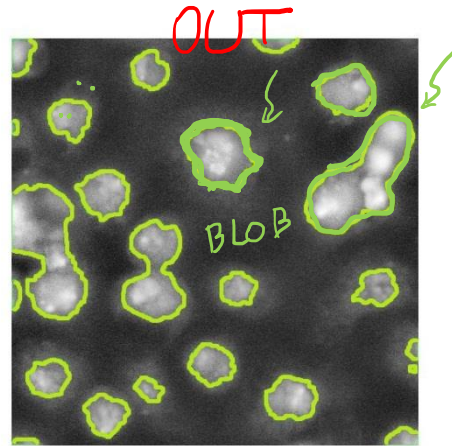
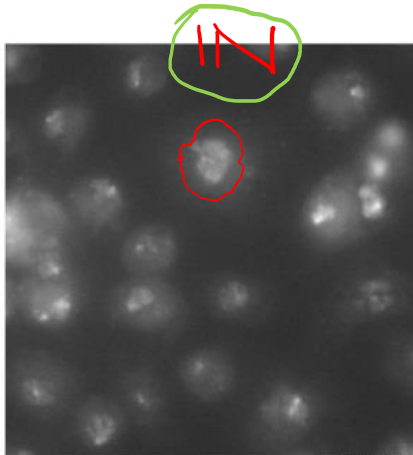
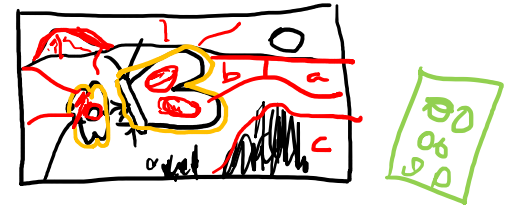
13.abr.2020

21h 50 +

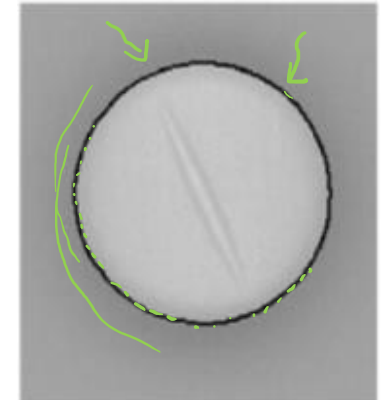


# SEGMENTAÇÃO

- Separar os objetos de interesse do fundo da imagem



<https://blogs.mathworks.com/steve/2006/06/02/cell-segmentation/>



Miha Mozina, Dejan Tomazevic, Franjo Pernus, Bostjan Likar, Real-time image segmentation for visual inspection of pharmaceutical tablets, Machine Vision and Applications, v. 22, p. 145-156, 2011, DOI 10.1007/s00138-009-0218-7.



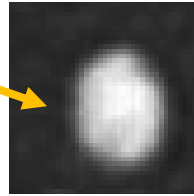
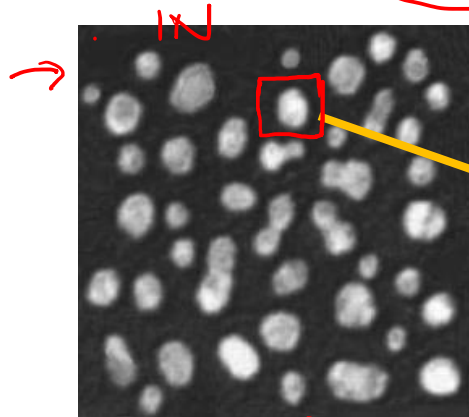
- Não vamos entrar na seguinte discussão:

*The major challenge is that the question “What is a correct segmentation” is a subtler question than “Is this digit a 5”.*

<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/papers/mftm-iccv01.pdf>

# BINARIZAÇÃO Limiarização global

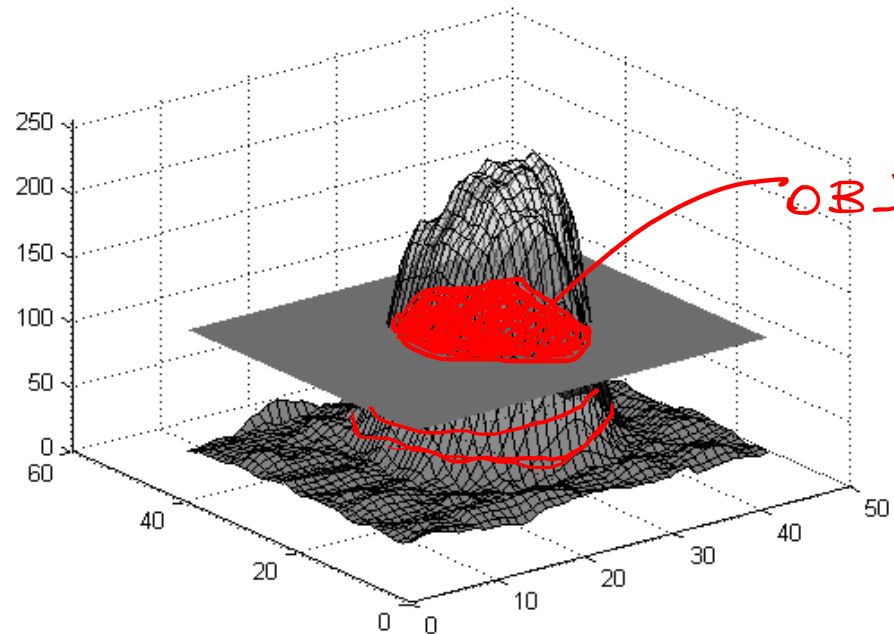
Thresholding ( $T_h$ )



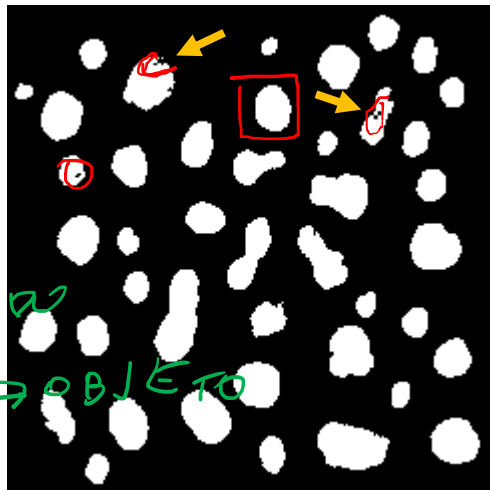
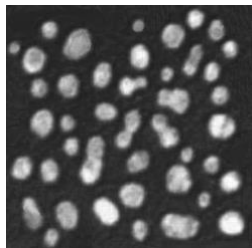
8BPP  
gray scale

255

0



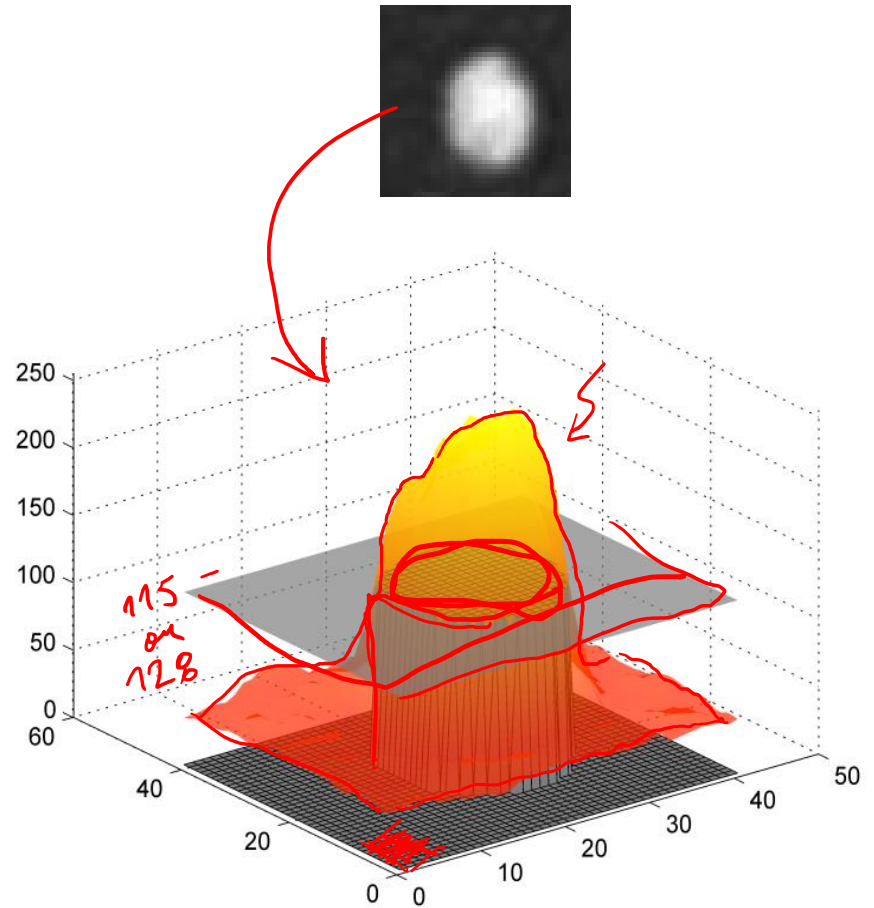
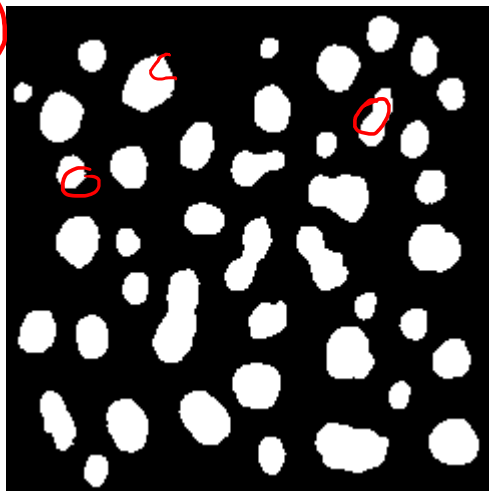
# Limiarização global



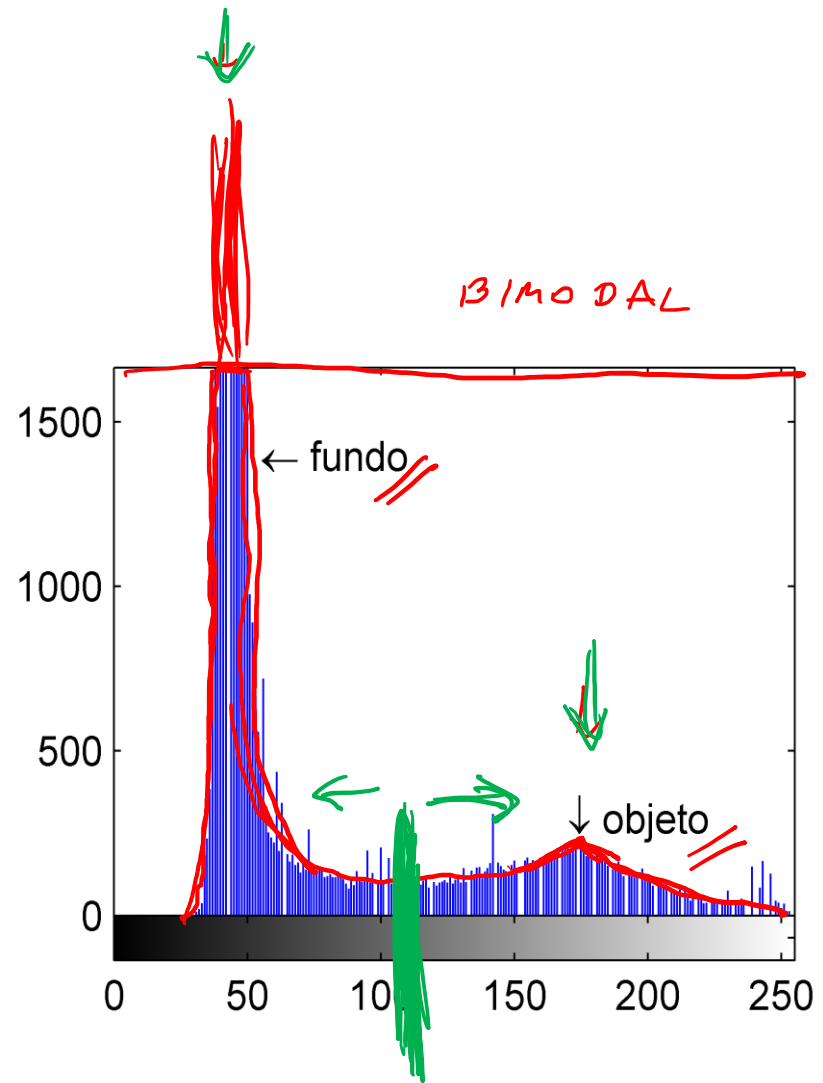
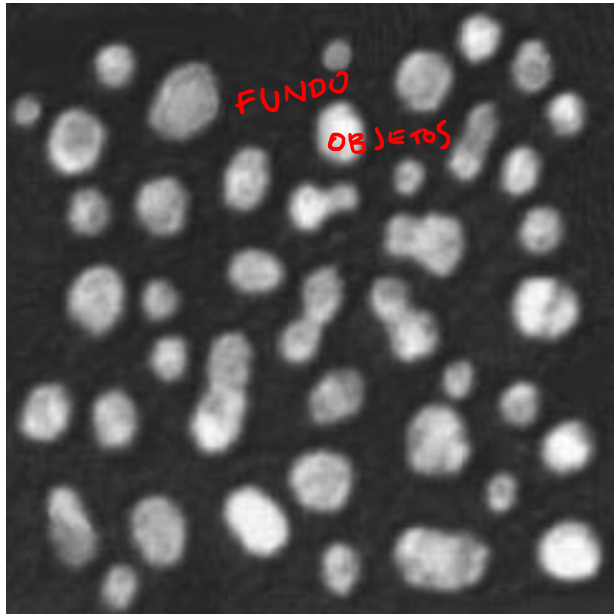
< 128  $\Rightarrow$  FUNDO  
<  $\rightarrow$  OBJETO

Limiar 128

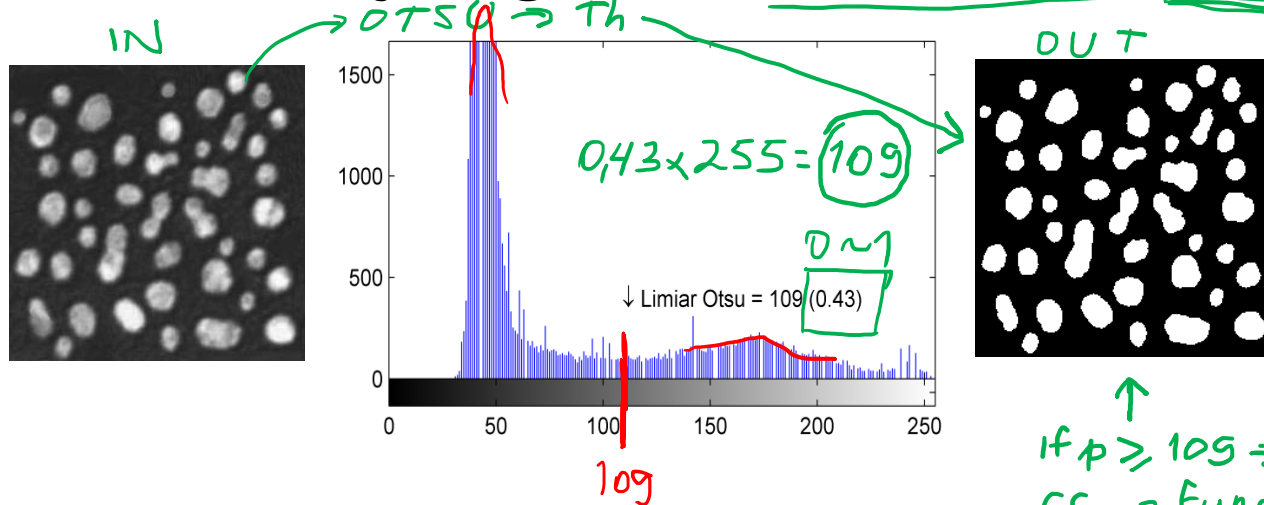
Limiar 115



# Limiarização global



# Limiarização global: método de Otsu



Otsu obtém o nível de cinza que melhor separa, segundo o critério estatístico da variância, os dois grupos de pixels.

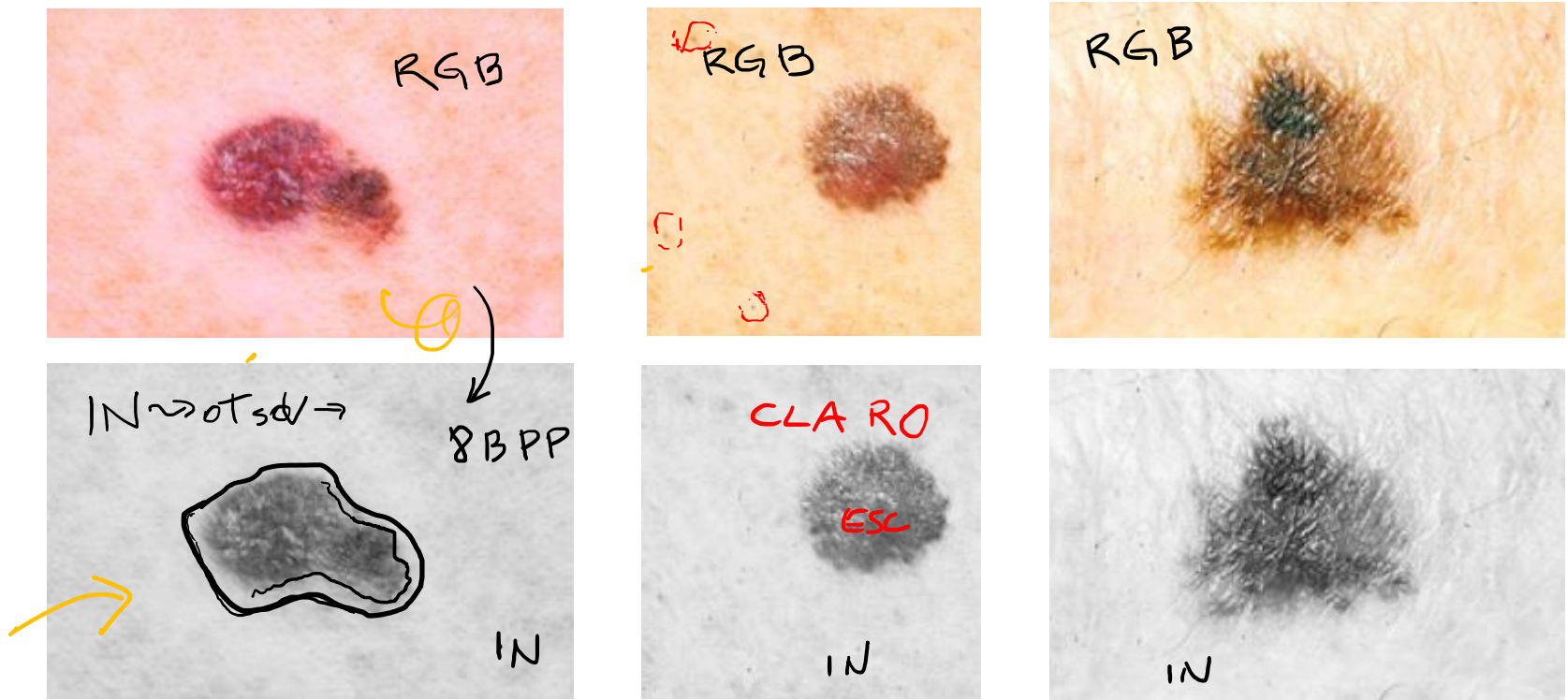
O valor ótimo de limiar é aquele que minimiza a variância interna dos grupos de pixels (*within-class variance* ou *intraclass variance*)  $\sigma_w^2$  :

$$\sigma_w^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$$

Handwritten red annotations: A red circle around  $\sigma_w^2(t)$  with an arrow pointing to it from the text '0...1,2...255'. Red arrows point from 'Prob' to  $q_1(t)$  and from 'Varian.' to  $\sigma_1^2(t)$ .

# Limiarização global

- Usar Otsu para [tentar] segmentar automaticamente as imagens de lesões de pele.



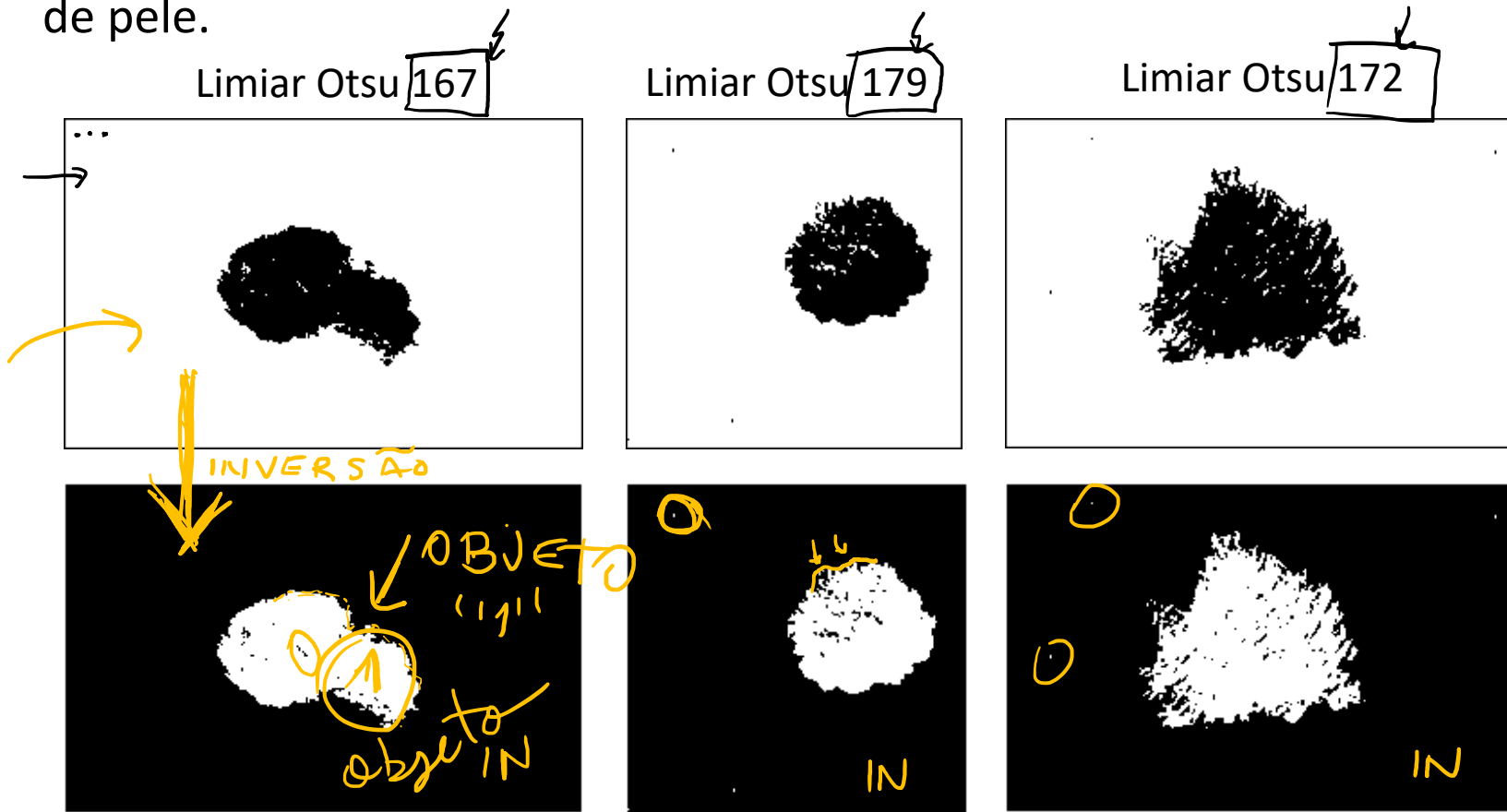
Imagens adaptadas de: Pablo G. Cavalcanti and Jacob Scharcanski, Macroscopic Pigmented Skin Lesion Segmentation and its Influence on the Lesion Classification and Diagnosis, In: Color Medical Image Analysis, Lecture Notes in Computational Vision and Biomechanics Volume 6, 2013, pp 15-39, Edited by M. Emre Celebi and Gerald Schaefer, Springer Netherlands.

[https://www.researchgate.net/publication/260713485\\_Macroscopic\\_pigmented\\_skin\\_lesion\\_segmentation\\_and\\_its\\_influence\\_on\\_lesion\\_classification\\_and\\_diagnosis](https://www.researchgate.net/publication/260713485_Macroscopic_pigmented_skin_lesion_segmentation_and_its_influence_on_lesion_classification_and_diagnosis)



# Limiarização global

- Usar Otsu para [tentar] segmentar automaticamente as imagens de lesões de pele.



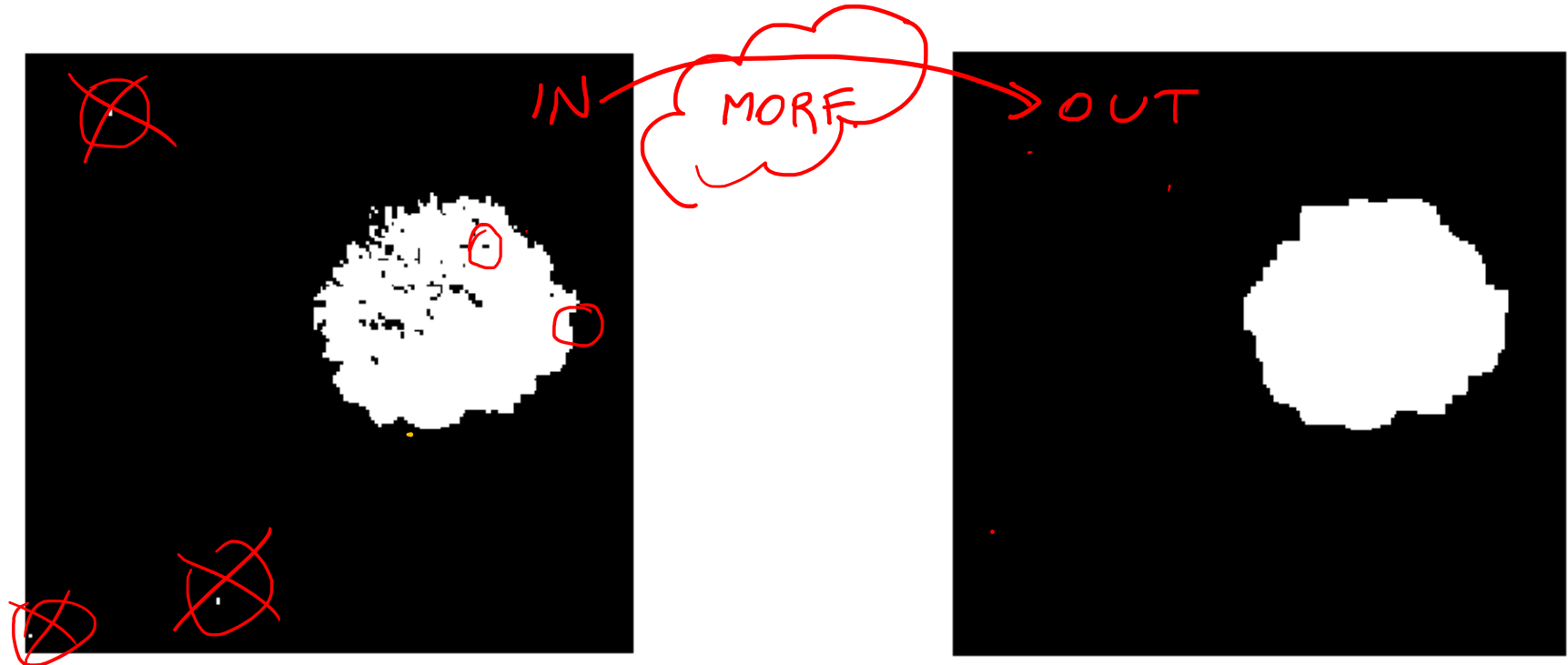
Imagens adaptadas de: Pablo G. Cavalcanti and Jacob Scharcanski, Macroscopic Pigmented Skin Lesion Segmentation and its Influence on the Lesion Classification and Diagnosis, In: Color Medical Image Analysis, Lecture Notes in Computational Vision and Biomechanics Volume 6, 2013, pp 15-39, Edited by M. Emre Celebi and Gerald Schaefer, Springer Netherlands.

[https://www.researchgate.net/publication/260713485\\_Macroscopic\\_pigmented\\_skin\\_lesion\\_segmentation\\_and\\_its\\_influence\\_on\\_lesion\\_classification\\_and\\_diagnosis](https://www.researchgate.net/publication/260713485_Macroscopic_pigmented_skin_lesion_segmentation_and_its_influence_on_lesion_classification_and_diagnosis)



# Operações morfológicas

- Morfologia significa “estudo da forma, estrutura, aparência”. Então, as operações morfológicas tem o objetivo de analisar e manipular a forma (*shape*) dos objetos na imagem.



Imagens adaptadas de: Pablo G. Cavalcanti and Jacob Scharcanski, Macroscopic Pigmented Skin Lesion Segmentation and its Influence on the Lesion Classification and Diagnosis, In: Color Medical Image Analysis, Lecture Notes in Computational Vision and Biomechanics Volume 6, 2013, pp 15-39, Edited by M. Emre Celebi and Gerald Schaefer, Springer Netherlands.

[https://www.researchgate.net/publication/260713485\\_Macroscopic\\_pigmented\\_skin\\_lesion\\_segmentation\\_and\\_its\\_influence\\_on\\_lesion\\_classification\\_and\\_diagnosis](https://www.researchgate.net/publication/260713485_Macroscopic_pigmented_skin_lesion_segmentation_and_its_influence_on_lesion_classification_and_diagnosis)

# Operações morfológicas

- A base matemática da morfologia é a teoria de conjuntos. Um conjunto de pixels binários denominado **elemento estruturante** (*structuring element*) varre a imagem (outro conjunto de pixels binários) e as operações morfológicas são realizadas entre o elemento estruturante e os pixels da imagem contidos no elemento estruturante. Este processo de 'varrer a imagem com o elemento estruturante' é similar àquele que realizamos em um box filter, por exemplo. Mas ao invés de aplicar uma convolução, **aplicamos a operação morfológica na máscara (elemento estruturante). O elemento estruturante pode ter um formato arbitrário e depende da aplicação.**
- A descrição matemática formal das operações morfológicas, na maioria das vezes utilizando notação de conjuntos, pode ser encontrada em várias referências bibliográficas.
- Aqui, para agilizar, vamos descrevê-las textualmente.

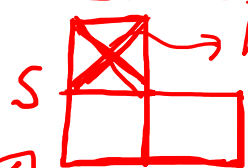
# Operações morfológicas

- O *ponto de referência* do elemento estruturante é o equivalente em morfologia ao ponto central das janelas de convolução quadradas. Este ponto de referência também é chamado de *ponto central* do elemento estruturante. Mas como este ponto central muitas vezes não é central, geometricamente falando, vamos usar o termo ***hot spot***.
- Estamos aplicando morfologia em imagens binárias (bw). Vamos convencionar que os **pixels dos objetos possuem sempre valor '1' (pixels brancos)**. **Logo, os pixels do fundo possuem sempre valor '0' (pixels pretos)**. A mesma idéia pode ser usada para os pontos do elemento estruturante: um ponto ativo do elemento estruturante tem valor '1'. Logo, um ponto não ativo (irrelevante) do elemento estruturante tem valor '0'.

# Operações morfológicas: Dilatação (dilation)

• **S**: elemento estruturante, **B**: imagem de entrada bw

Cada vez que o **hot spot** de **S** encontra um pixel '1' de **B**, **S** é replicado na imagem de saída.



DILATAÇÃO

```
clear all, close all
```

```
bw = [ 1 0 0 0 0 1
       0 0 1 1 0 0
       0 0 1 0 0 0
       0 1 1 1 1 0
       0 1 1 1 0 0
       0 0 0 0 0 0 ];
```

```
bw = logical(bw);
```

```
% O hot spot é dado por
% floor((size(NHOOD)+1)/2)
% Logo, o hotspot do el est
% abaixo é em (1,1)
```

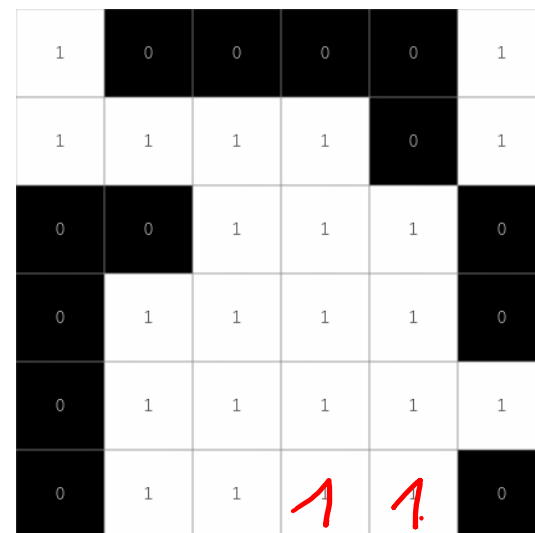
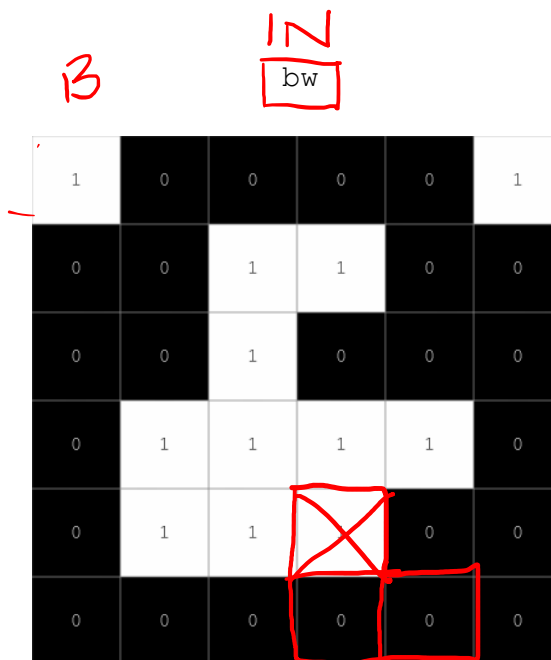
```
NHOOD = [ 1 0
           1 1 ];
```

```
SE = strel('arbitrary', NHOOD);
```

```
d = imdilate(bw, SE);
```

```
imtool(bw)
```

```
imtool(d)
```



# Operações morfológicas: Erosão (erosion)

- **S**: elemento estruturante, **B**: imagem de entrada bw
- O pixel '1' de **B** correspondente ao hot spot de **S** só é mantido na imagem de saída se *todos* os pixels '1' de **S** possuem pixel '1' correspondente em **B**.

```
clear all, close all
```

```
bw = [ 1 0 0 0 0 1
      0 0 1 1 0 0
      0 0 1 0 0 0
      0 1 1 1 1 0
      0 1 1 1 0 0
      0 0 0 0 0 0];
```

```
bw = logical(bw);
```

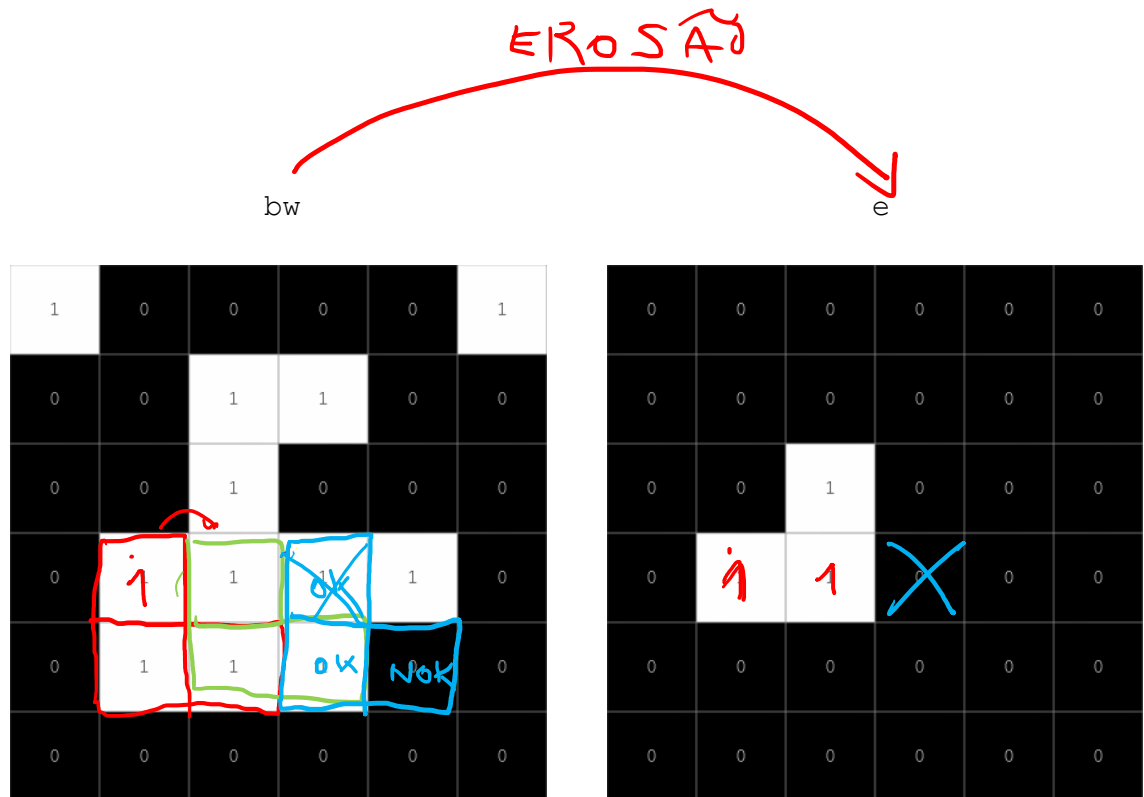
```
% O hot spot é dado por
% floor((size(NHOOD)+1)/2)
% Logo, o hotspot do el est
% abaixo é em (1,1)
```

```
NHOOD = [ 1 0
          1 1];
```

```
SE = strel('arbitrary', NHOOD);
e = imerode(bw, SE);
```

```
imtool(bw)
```

```
imtool(e)
```

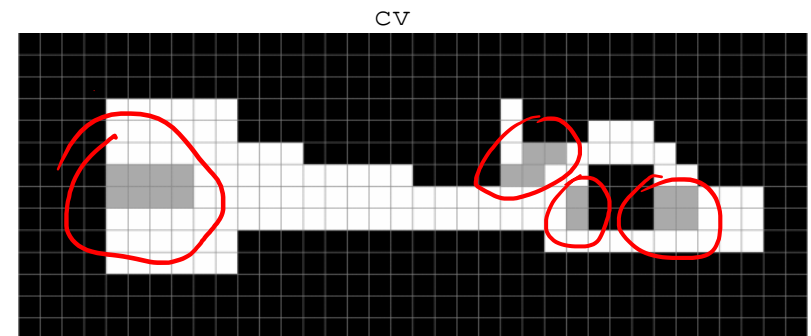
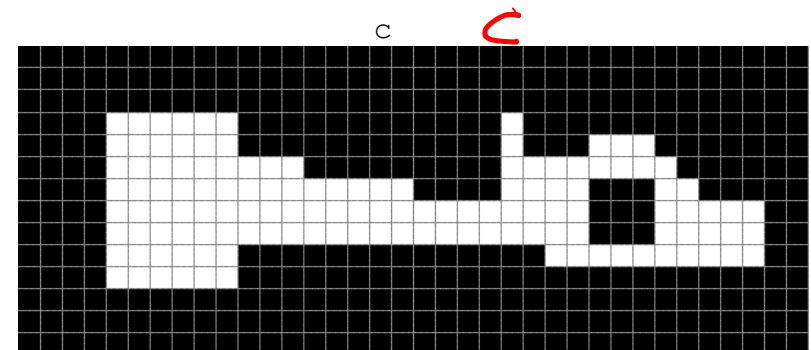
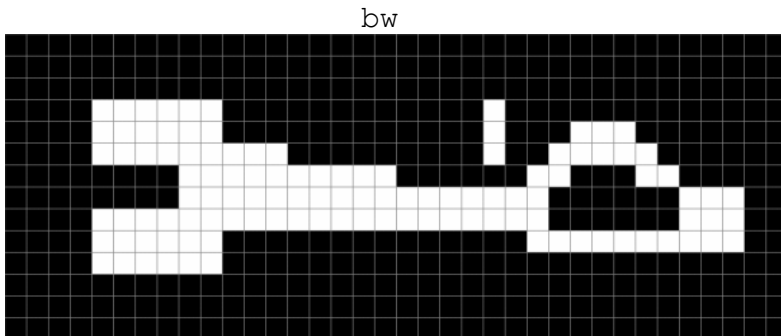


# Operações morfológicas: Fechamento (closing)

- **S**: elemento estruturante, **B**: imagem de entrada bw
- É a dilatação seguida da erosão, usando o mesmo elemento estruturante: aplicar a dilatação na imagem **B** usando **S** e depois aplicar a erosão na imagem dilatada usando **S**.

```
clear all, close all
```

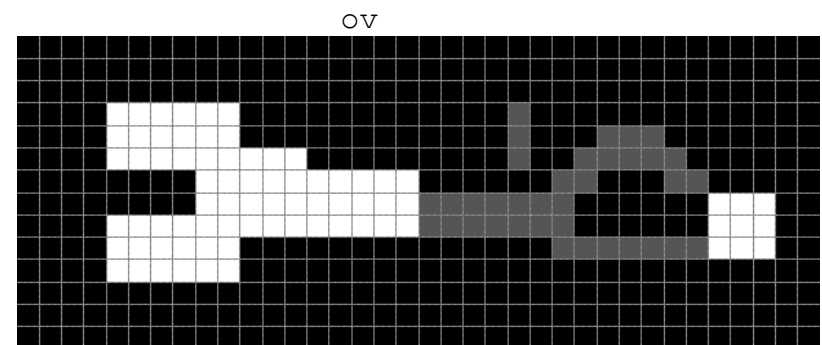
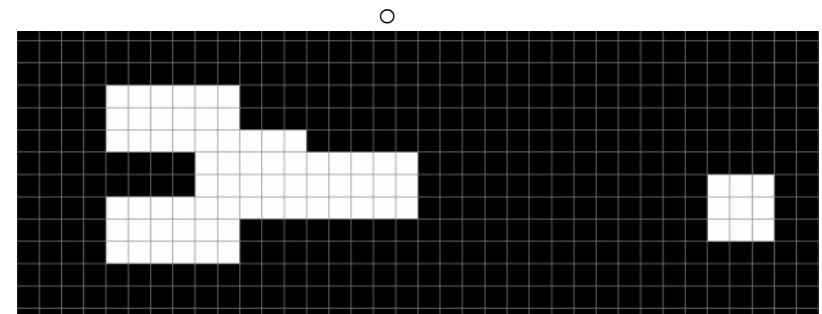
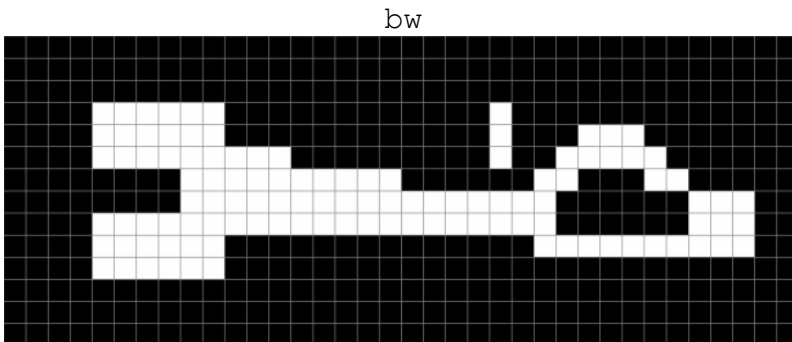
```
bw = imread('exMorph4.bmp');  
se = strel('square', 3);  
c = imclose(bw, se); ←  
cv = visualize(bw, c);  
imtool(bw);  
imtool(c);  
imtool(cv);
```



# Operações morfológicas: Abertura (opening)

- **S**: elemento estruturante, **B**: imagem de entrada bw
- É a **erosão seguida da dilatação**, usando o mesmo elemento estruturante: aplicar a erosão na imagem **B** usando **S** e depois aplicar a dilatação na imagem erodida usando **S**.

```
clear all, close all  
  
bw = imread('exMorph4.bmp');  
se = strel('square', 3);  
o = imopen(bw, se);  
ov = visualize(bw, o);  
imtool(bw);  
imtool(o);  
imtool(ov);
```

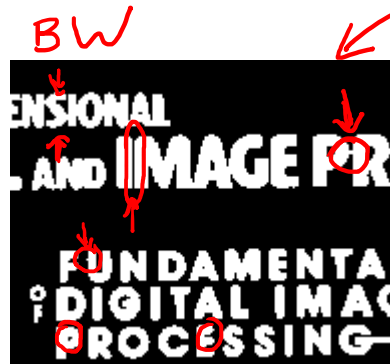




# Operações morfológicas

- Manipular a forma de regiões binárias

Original



Limiarizada (binária)

AUMENTAR  
O EL. ESTR.



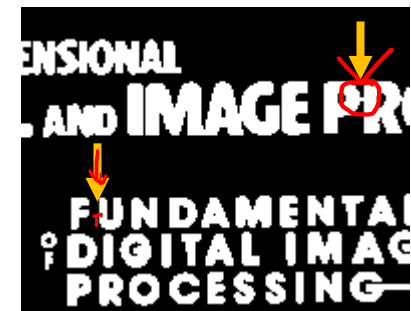
Dilatação



Erosão



Fechamento



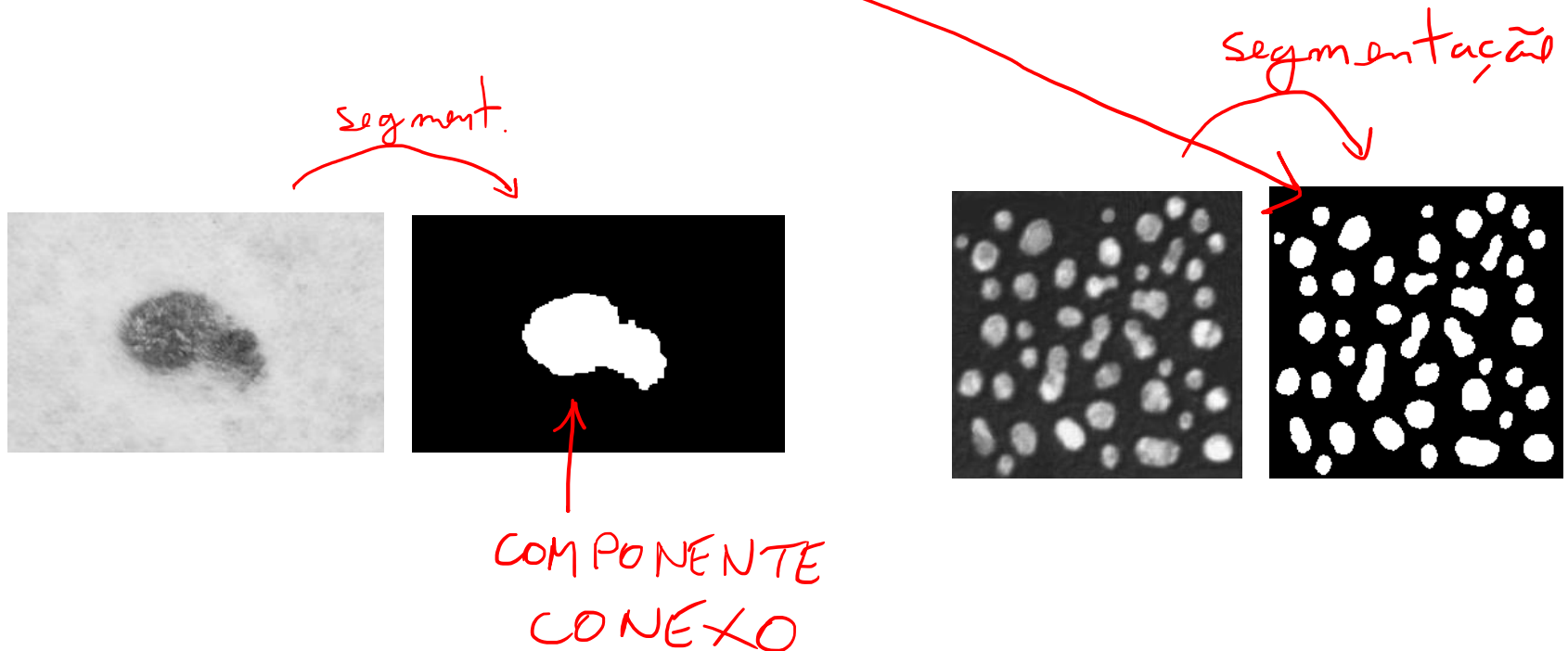
Abertura

Fechamento: Suaviza contornos, funde  
quebras estreitas, elimina pequenos buracos,  
preenche espaços de contornos.

Abertura: suaviza contornos, quebra  
ligações estreitas, elimina protruções  
finas.

# Rotulação de regiões binárias (labeling)

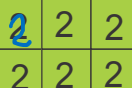
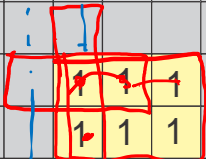
- Quantas lesões de pele tem na imagem? *área?*
- Quantas células tem na imagem?



(labeling)

- Sequential labeling  
ref: Burger & Burge

IN



●	●	●
●		●
●	●	●



# Rotulação de regiões binárias (labeling)

```
clear all, close all
clc

bw = [ 0 0 0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0 1 0
       0 0 1 1 1 0 0 1 0 0
       0 0 1 1 1 0 1 1 1 0
       0 0 0 0 0 1 1 1 0 0
       0 0 0 0 0 0 1 1 1 0
       0 0 0 1 0 0 0 1 1 0
       0 0 1 1 0 0 1 1 0 0
       0 1 1 1 0 1 1 0 0 0
       0 0 0 0 0 0 0 0 0 0];
```

```
bw = logical(bw);
```

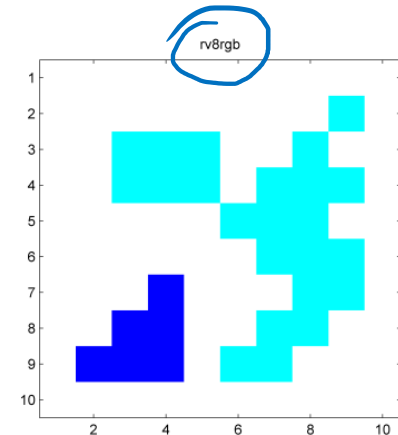
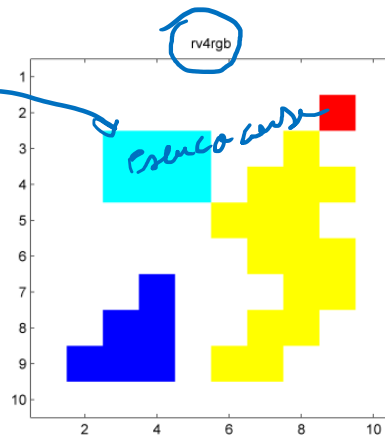
```
% A saída de bwlabel
% é da classe double
[rv4 nrv4] = bwlabel(bw, 4);
[rv8 nrv8] = bwlabel(bw, 8);
```

```
% Aplica pseudocores na
% imagem rotulada para
% visualização
rv4rgb = label2rgb(rv4);
rv8rgb = label2rgb(rv8);
disp(['n. obj em rv4 = ', ...
      num2str(nrv4)]);
disp(['n. obj em rv8 = ', ...
      num2str(nrv8)]);
```

```
% Display
figure, image(rv4rgb)
title('rv4rgb')
figure, image(rv8rgb)
title('rv8rgb')
```

```
>> rv4
rv4 =
    0    0    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    4    0
    0    0    2    2    2    0    0    3    0    0
    0    0    2    2    2    0    3    3    3    0
    0    0    0    0    0    0    3    3    3    0
    0    0    0    0    0    0    0    3    3    0
    0    0    1    1    1    0    0    3    3    0
    0    1    1    1    0    3    3    0    0    0
    0    0    0    0    0    0    0    0    0    0
```

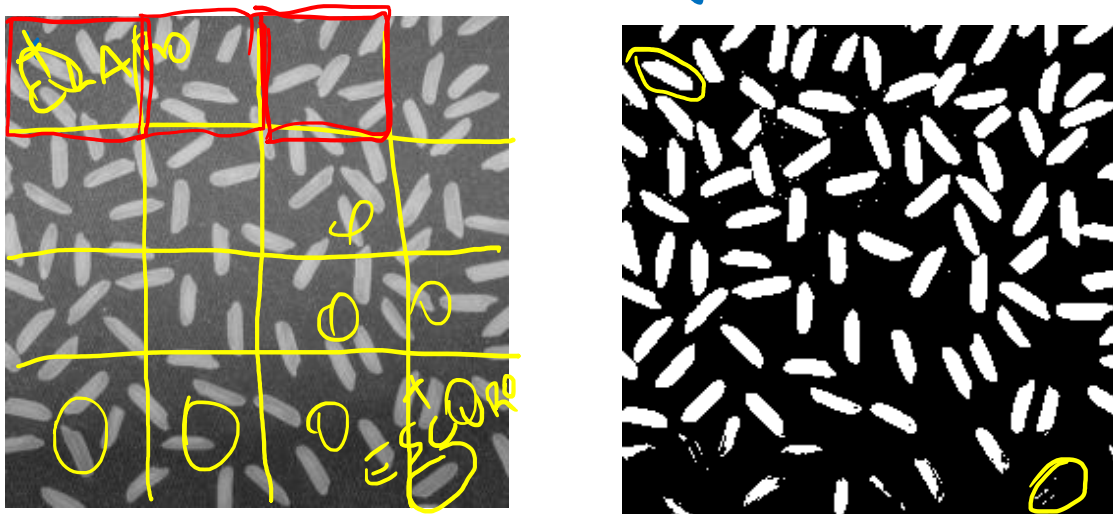
```
>> rv8
rv8 =
    0    0    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    2    0
    0    0    2    2    2    0    0    2    0    0
    0    0    2    2    2    0    2    2    2    0
    0    0    0    0    0    2    2    2    2    0
    0    0    0    0    0    0    2    2    2    0
    0    0    0    1    0    0    0    2    2    0
    0    0    1    1    0    2    2    0    0    0
    0    1    1    1    0    2    2    0    0    0
    0    0    0    0    0    0    0    0    0    0
```



## Ver Notas de Aula (Notas de aula e prática Octave #08):

- A segmentação por limiarização global pode ser bastante comprometida em situações nas quais a iluminação na imagem não é uniforme (homogênea):

Otsu, função



- Existem várias abordagens para a solução. Por exemplo:
  - Transformada top-hat ✓
  - Limiarização local (adaptativa) ✓