

# 1. Imagens digitais

Malha digital. Conectividade e vizinhança. Resolução.

# Malha digital

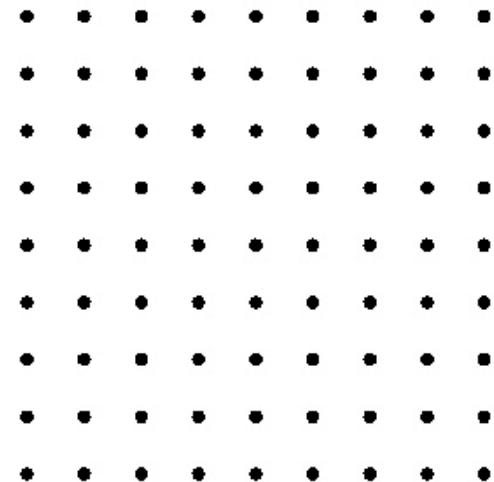
**Malha digital:** É o suporte no qual as imagens digitais são representadas e através do qual se estabelecem as conexões entre pontos vizinhos.

**Estrutura da malha digital:** antes de se fazer qualquer operação sobre uma imagem digital, deve-se conhecer primeiro a estrutura de relações espaciais entre os pixels, ou seja, definir quais as relações de vizinhança entre os pixels da malha digital, para se poder saber de que forma se distinguirão objectos constituídos por conjuntos de pixels, assim como definir as suas respectivas fronteiras.

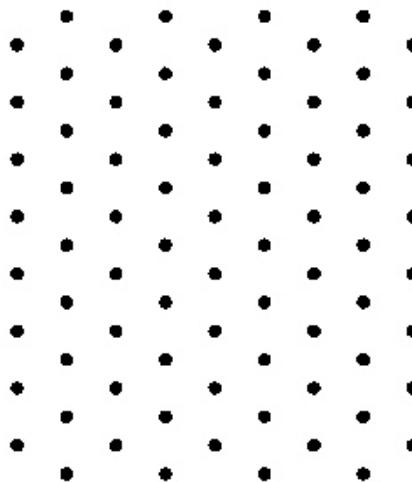
- Para tal, há que estabelecer a geometria da malha digital de representação e caracterizá-la topologicamente.

# Malha digital

Num largo número de sistemas digitais de aquisição de imagem, a sua estrutura espacial é definida segundo uma malha ortogonal e equidistante, sendo cada ponto dessa malha chamado de “pixel”. Regra geral, os pixels têm forma quadrada. A geometria da malha é que pode variar.



Malha rectangular



Malha hexagonal

# Conectividade e vizinhança

Fernando Soares

As regras de conectividade estabelecem a relação entre dois ou mais pixels.

De uma forma geral, para que dois pixels sejam considerados como conectados, têm que verificar condições de adjacência radiométrica e adjacência espacial.

Em primeiro lugar, para que dois pixels se considerem conectados radiometricamente, têm ambos que ter valores pertencentes a um mesmo intervalo numérico (classe numérica), ou conjunto de valores numéricos, podendo dizer-se que assim pertencerão a um mesmo tipo de objeto.

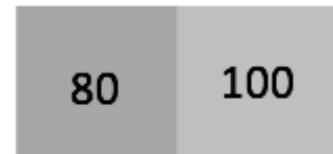
# Conectividade e vizinhança

Fernando Soares

Por exemplo, numa imagem de cinzentos de 8-bits, estabelecendo o critério de similitude de serem semelhantes todos os pixels com valores dentro do intervalo  $I = [60, \dots, 120]$ , dois pixels com valores respectivamente de 80 e 100 são semelhantes, estando espacialmente próximos, ou não. Outros dois com valores de 80 e 170, não são semelhantes.



$I$



Semelhantes



Não semelhantes

# Conectividade e vizinhança

Fernando Soares

Para formular o critério de conectividade espacial, introduz-se primeiro o conceito de **vizinhança**.

A vizinhança de conectividade-4 do pixel de coordenadas  $(x, y)$  é o conjunto de pixels  $V_4$ .

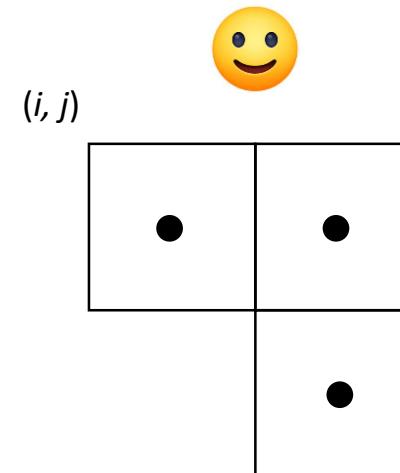
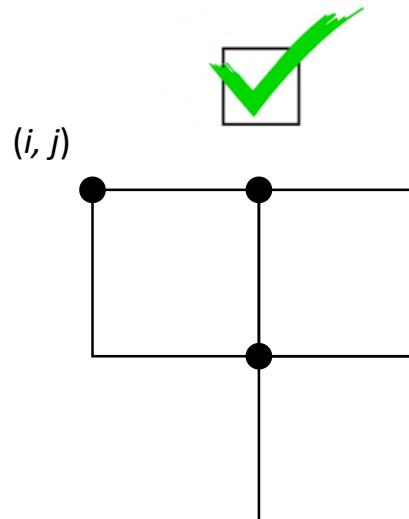
$$V_4(x, y) = \{(x, y), (x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)\}$$

A vizinhança de conectividade-8 do pixel de coordenadas  $(x, y)$  é o conjunto de pixels  $V_8$ .

$$V_8(x, y) = V_4(x, y) \cup \{(x + 1, y + 1), (x + 1, y - 1), (x - 1, y + 1), (x - 1, y - 1)\}$$

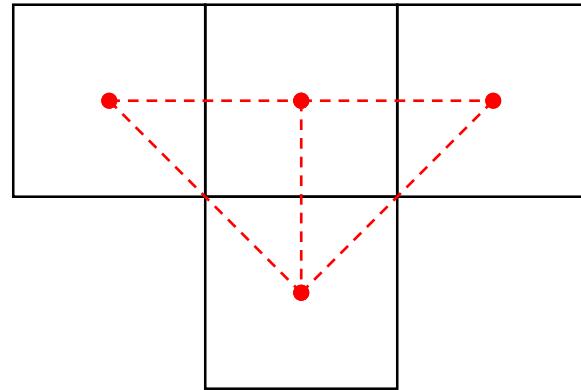
# Conectividade e vizinhança

Por definição, as coordenadas imagem de um pixel devem referir-se ao seu canto superior esquerdo. Contudo, para facilitar a compreensão de vários conceitos, será por vezes ilustrada a sua posição no seu centro geométrico.

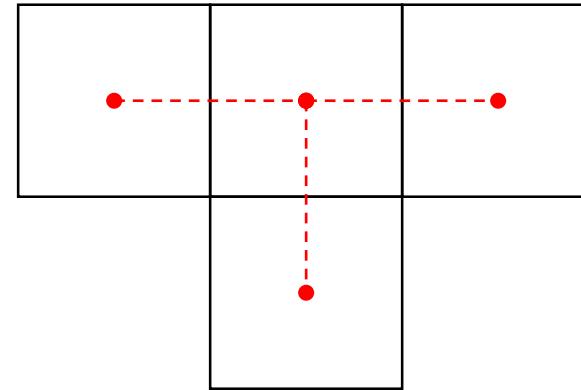


# Conectividade e vizinhança

O **grafo de adjacência** de um certo conjunto de pixels é representado pelo conjunto de ligações elementares que conectam esses pixels.



Grafo de conectividade 8

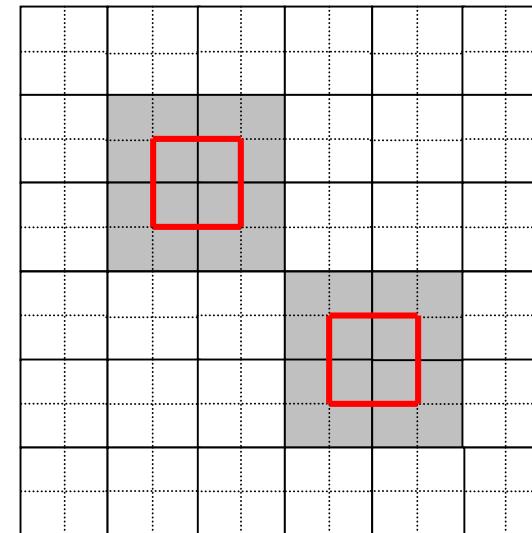


Grafo de conectividade 4

# Conectividade e vizinhança

Na malha de conectividade-4 há um máximo de quatro pixels vizinhos a que cada pixel se pode conectar.

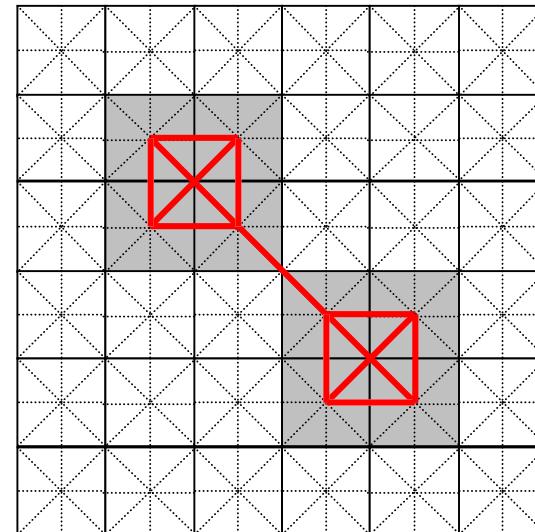
Na figura, os dois conjuntos de pixels sombreados tocam-se aparentemente num vértice. Contudo não se consideram conectados entre si, pela razão de que a conectividade da malha só permite a conexão de pixels segundo as direcções horizontal e vertical. Há portanto dois objectos distintos.



# Conectividade e vizinhança

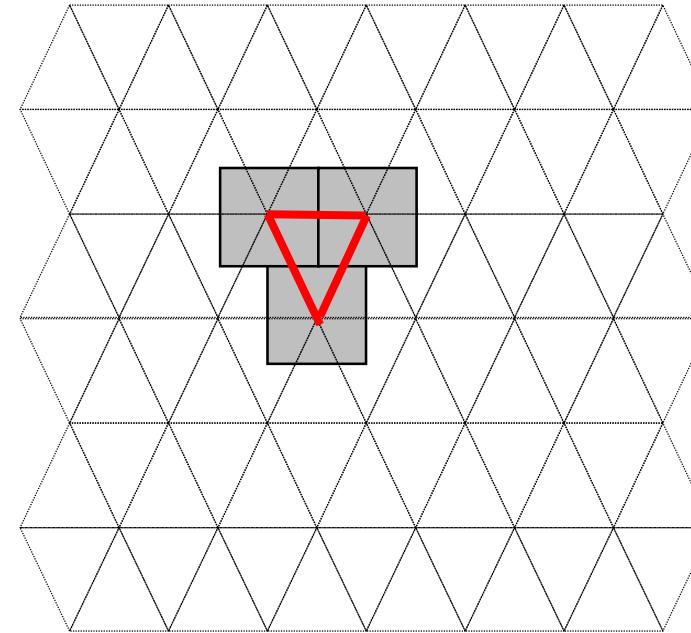
Na malha de conectividade-8 há um máximo de oito pixels vizinhos a que cada pixel se pode conectar.

Assim, havendo conexão na diagonal, os pixels sombreados constituem um só conjunto conexo.



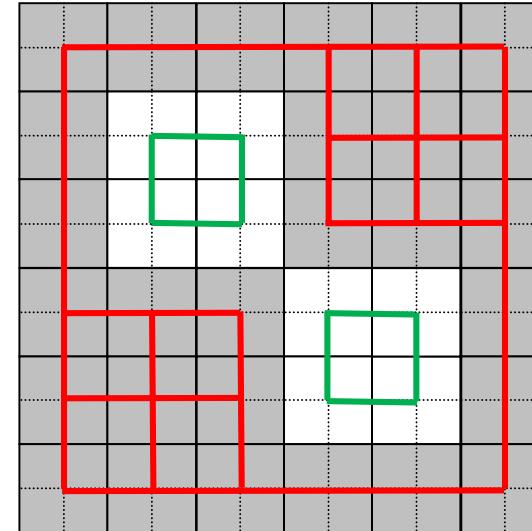
# Conectividade e vizinhança

A malha hexagonal é de conectividade-6 e, neste caso, a vizinhança de cada pixel é constituída sempre por sete pixels.



# Conectividade e vizinhança

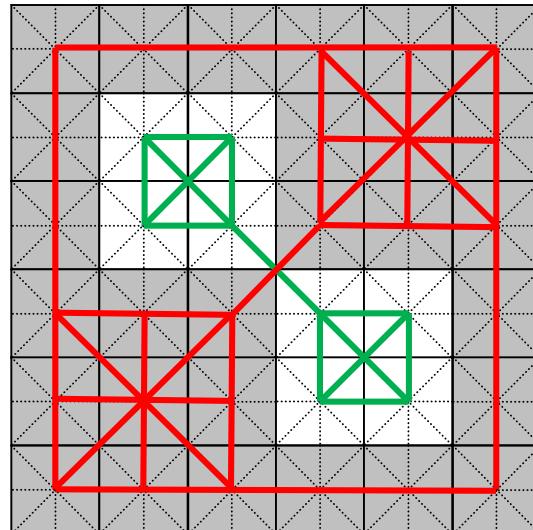
Para a malha de conectividade-4, os grafos correspondentes ao conjunto e ao seu complementar, não se intersectam, o que não cria ambiguidades lógicas.



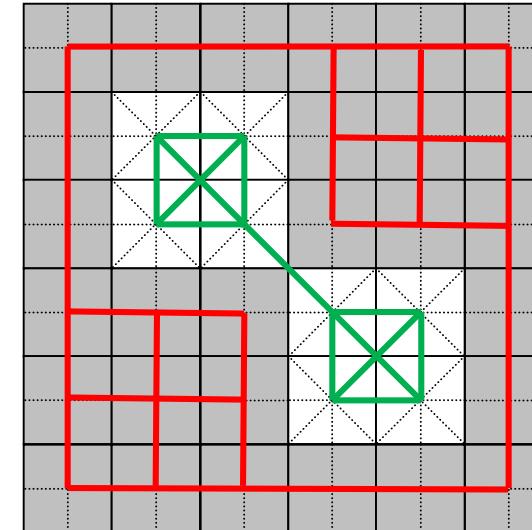
# Conectividade e vizinhança

No caso da conectividade-8, há uma ambiguidade lógica.

A intersecção entre os grafos do conjunto e do seu complementar é diferente de conjunto vazio. Como tal, quando se usa a malha digital quadrada, constitui-se a regra de considerar-se a conectividade-8 para um conjunto e a conectividade-4 para o seu complementar, ou vice-versa.



Ambiguidade

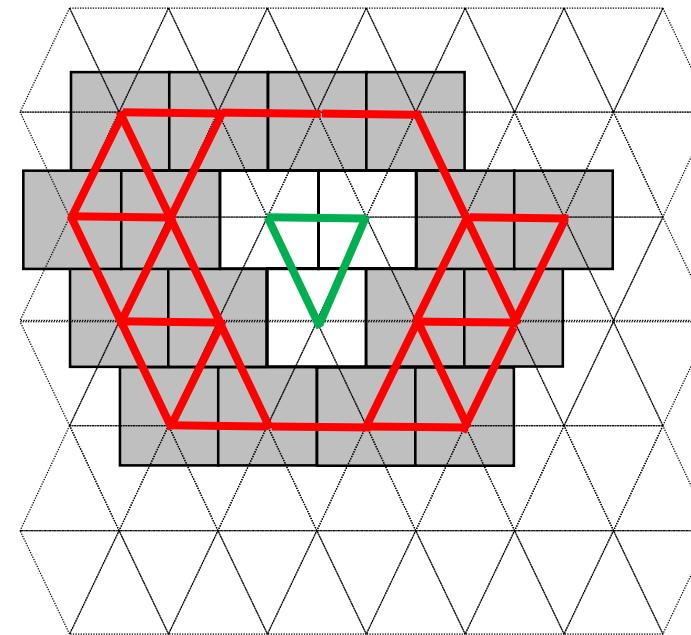


Solução

# Conectividade e vizinhança

Fernando Soares

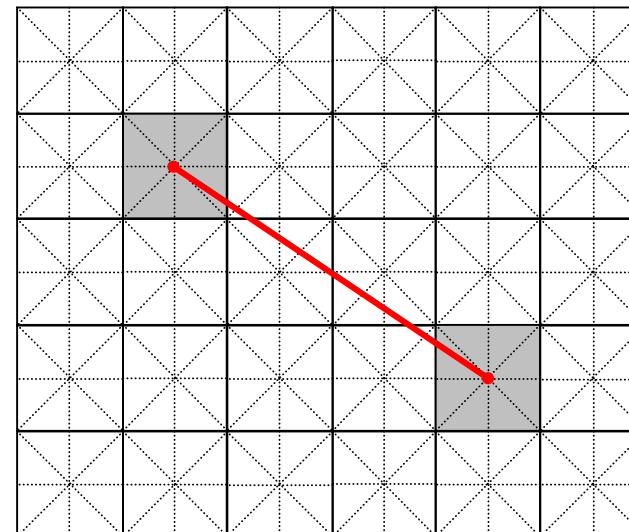
No caso da malha hexagonal não existe ambiguidade lógica entre um conjunto e o seu complementar.



# Conectividade e vizinhança

Numa malha digital, as distâncias entre pixels são geralmente duas: **distância euclidiana** e **distância grafológica**.

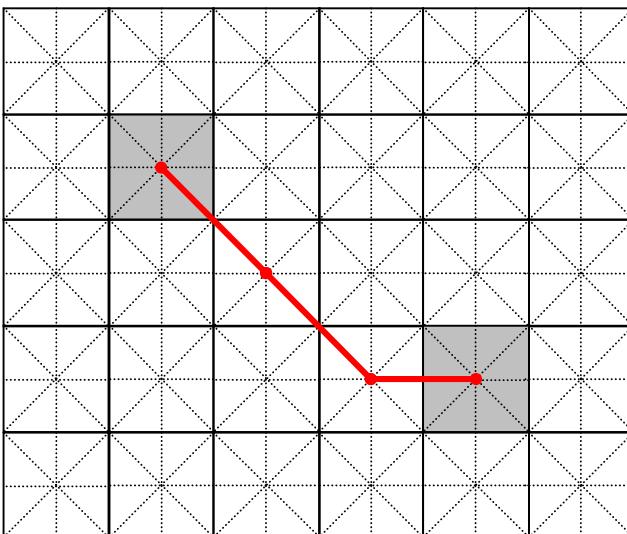
**Distância euclidiana:** calculada através das diferenças de coordenadas-pixel entre os dois pixels.



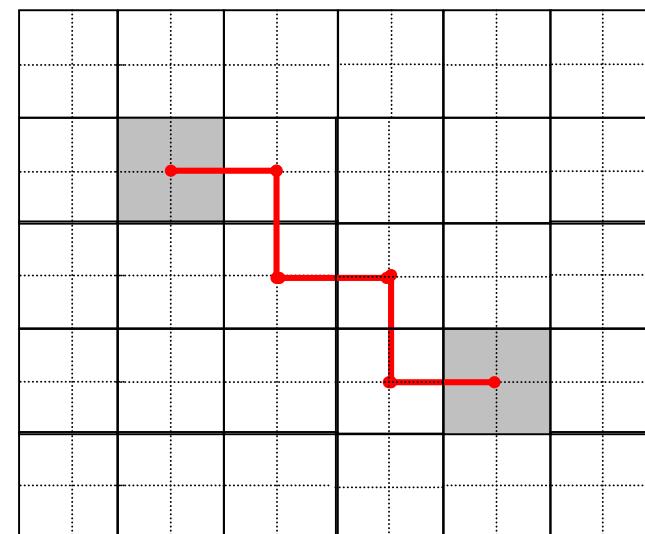
$$|\Delta M| = 3; |\Delta P| = 2; Dist = \sqrt{13} \approx 3.6;$$

**Distância grafológica:** corresponde ao valor da menor soma de conexões que ligam os dois pixels dentro do grafo de adjacência da malha digital.

(a)



(b)



(a): distância grafológica numa malha de conexidade-8 (Chessboard Distance):  $Dist = 3$ ;

(b): distância grafológica numa malha de conexidade-4 (City Block Distance):  $Dist = 5$ .

# Resolução

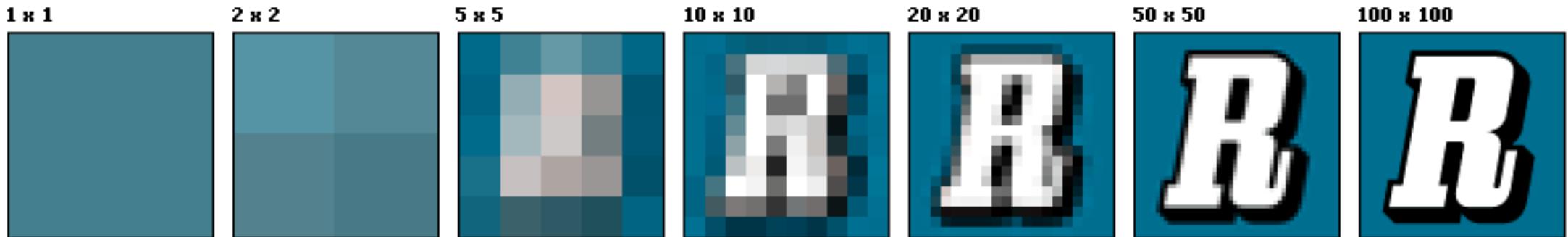
Descreve o nível de detalhe de uma imagem. O termo aplica-se a imagens digitais, imagens em sequências de vídeo, ou outros tipos de imagem. Resoluções mais altas significam maior detalhe.

Tipos de resolução associados às imagens digitais:

1. Número de pixels (*pixel count*)
2. Pixels por polegada (*PPI*)
3. Pontos por polegada (*DPI*)
4. Espacial (*spatial resolution*)
5. Espectral (*spectral resolution*)
6. Radiométrica (*radiometric resolution*)
7. Temporal (*temporal resolution*)

# Resolução

**Número de pixels (*pixel count*):** quando a contagem dos pixels de uma imagem é referida como “resolução”, a convenção descreve-a como o conjunto de dois números inteiros positivos em que o primeiro corresponde ao número de colunas (*width*) e o segundo corresponde ao número de linhas (*height*).



# Resolução

Fernando Soares

Outra convenção comum para quantificar o número de pixels consiste na contagem em mega-pixels, que pode ser calculada multiplicando o número de colunas pelo número de linhas, e dividindo por um milhão ( $1 \text{ mega-pixel} = 10^6 \text{ pixels}$ ).

Por exemplo: uma imagem de 2048 colunas por 1536 linhas tem um total de  $2048 \times 1536 = 3145728 \text{ pixels}$ , ou seja, 3.1 mega-pixels.

O tamanho de uma imagem digital está, pois, relacionado com a quantidade de pixels que tem e não com o tamanho da sua representação num ecrã.

**“Pixels por polegada”, ou “pixels per inch” (PPI):** é uma medida de densidade ou resolução de uma tela de computador, tela de televisão ou outro dispositivo de exibição. O valor de *PPI* indica quantos pixels estão em uma linha de uma polegada, num ecrã, independentemente se ela está na vertical, ou na horizontal (tratando-se de pixels quadrados).

O tamanho do ecrã não quer dizer absolutamente nada quando se fala em qualidade. Saber o número em polegadas de um ecrã, ou *display* de um *smartphone*, não diz nada acerca da sua resolução. Tratando-se de qualidade, o número que realmente importa é o que se expressa em pixels. Basicamente, quanto mais pixels houver em um mesmo espaço, melhor será a definição das imagens e, consequentemente, melhor será a qualidade visual.

# Resolução

Cálculo do valor da resolução PPI de um ecrã:

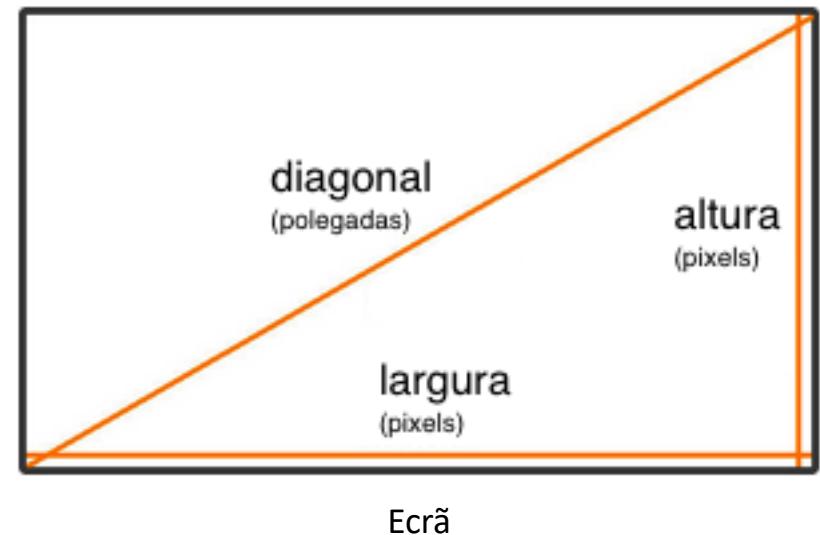
$$\text{diagonal (pixels)} = \sqrt{\text{largura}^2 + \text{altura}^2}$$

$$PPI = \frac{\text{diagonal (pixels)}}{\text{diagonal (polegadas)}}$$

Exemplo:

Resolução de um ecrã de  $1920 \times 1080$  pixels (Full HD), com 15"

$$PPI = \frac{\sqrt{1920^2 + 1080^2}}{15} = 96$$

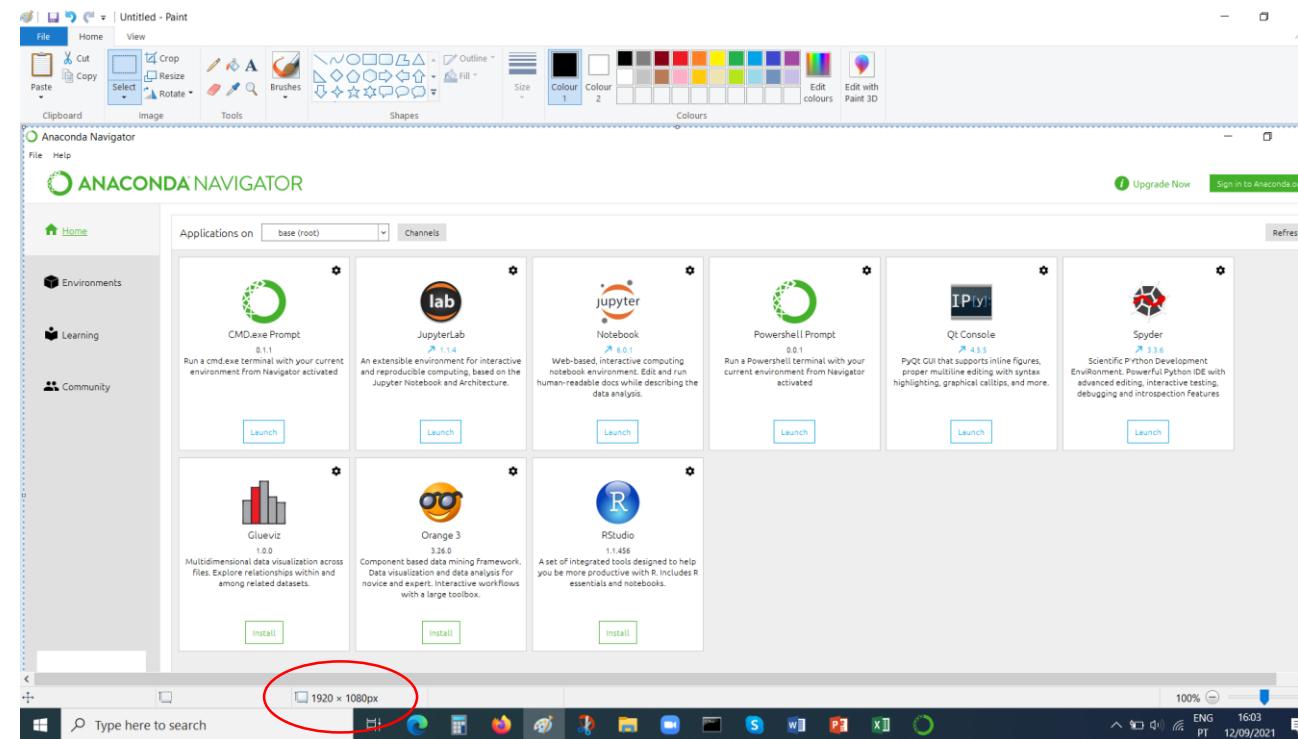


Ecrã

# Resolução

Como saber rapidamente qual a dimensão de um ecrã em pixels:

1. Fazer um *printscreen*.
2. Abrir a aplicação *Paint* e fazer *Ctrl+V*.
3. Ler o tamanho da imagem ( $1920 \times 1080$  pixels) no rodapé.



# Resolução

**“Pontos por polegada”, ou “*dots per inch*” (DPI):** no momento da impressão de uma imagem em papel, o pixel deixa de existir e “o que se vê” no papel são apenas minúsculos pontos que a compõem. O valor de DPI corresponde à resolução com que a imagem será imprimida e depende dos valores em pixels, já que é necessário fazer uma conversão antes.

Nas fases, quer de impressão, quer de digitalização de uma imagem, deve-se ter em atenção a escolha do valor de DPI, pois tal influencia diretamente o tamanho e a qualidade da foto. Normalmente não é preciso um valor muito alto, podendo-se usar, para a maior parte dos casos, os valores de referência de 72, 100, ou 300.

# Resolução

Partindo da imagem digitalizada, coloca-se a questão ao contrário: conhece-se o valor de DPI e a sua dimensão (em pixels). Pode-se então determinar uma das suas dimensões de impressão em papel. Se este comprimento for ultrapassado, na impressão, ou na sua apresentação num ecrã, a qualidade pode ficar comprometida.



Imagen  
 $256 \times 256$   
96 PPI

$$\text{diagonal (polegadas)} = \frac{\text{diagonal (pixels)}}{\text{DPI}}$$

Exemplo:

Imagen de  $256 \times 256$  pixels

$$\begin{aligned} \text{diagonal (cm)} &= \text{diagonal (polegadas)} \times 2.5400051 = \\ &= \frac{\sqrt{256^2 + 256^2}}{96} \times 2.5400051 = 3.77 \times 2.5400051 \\ &= 9.57 \text{ cm} \end{aligned}$$

# Resolução

Fernando Soares

Como entender rapidamente o espaço que uma imagem ocupa numa folha de papel:

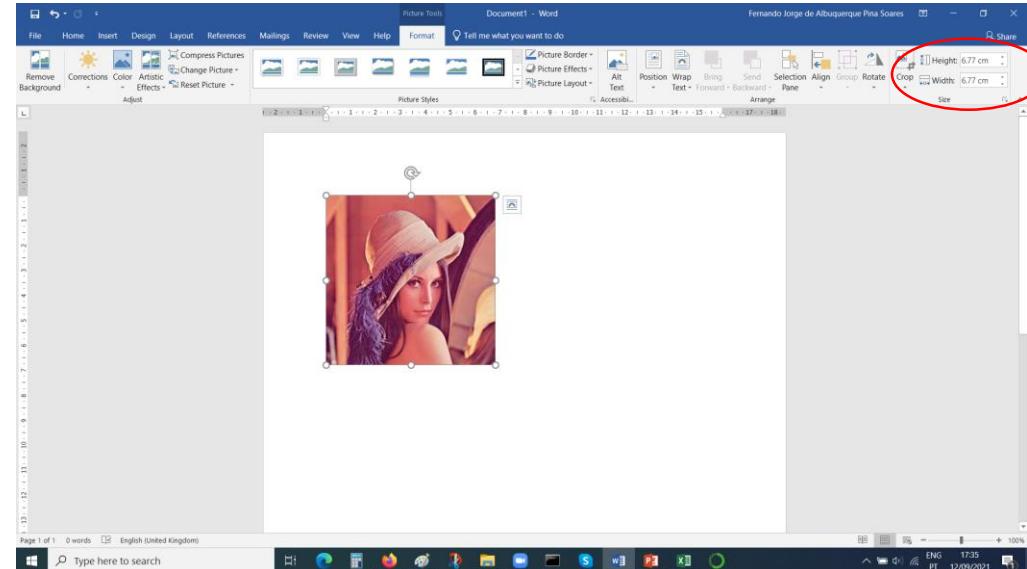
1. Abrir um novo documento MsWord (folha A4).
2. Importar imagem.
3. Verificar a suas dimensões



Imagen

256 x 256

96 PPI



$$\frac{\sqrt{256^2 + 256^2}}{96} \times 2.5400051 = \sqrt{6.77^2 + 6.77^2} \text{ cm}$$

# Resolução

A distância máxima de impressão, expressa em centímetros (1 polegada = 2.5400051 cm), de um comprimento de k pixels a n DPI, pode calcular-se da seguinte forma:

$$\text{distância (cm)} = \frac{k}{n} \times 2.5400051$$

Por exemplo, para uma imagem com resolução de  $1920 \times 1080$  pixels, a 72 DPI, tem-se:

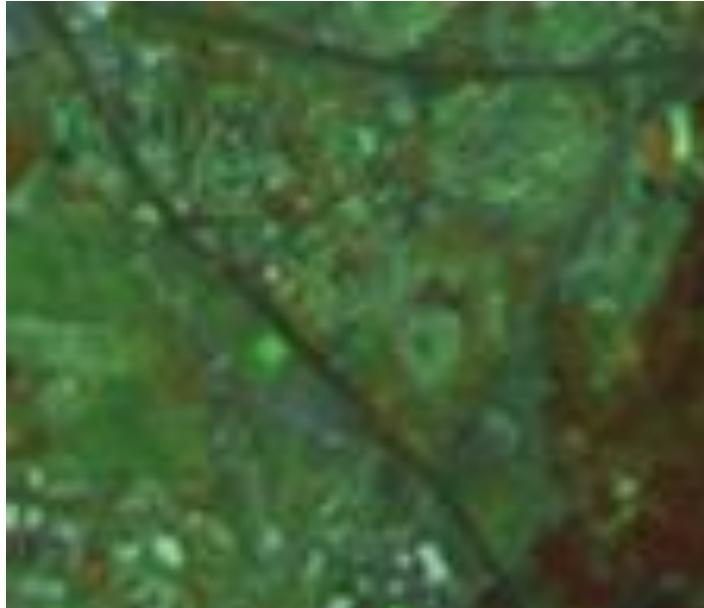
- Largura da imagem no papel =  $1920/72 \times 2.5400051 = 67,73$  cm
- Altura da imagem no papel =  $1080/72 \times 2.5400051 = 38,10$  cm

Para uma imagem com resolução de  $1920 \times 1080$  pixels tem-se, a 300 DPI, tem-se:

- Largura da imagem no papel =  $1920/300 \times 2.5400051 = 16,25$  cm
- Altura da imagem no papel =  $1080/300 \times 2.5400051 = 9,14$  cm

# Resolução

**Resolução espacial:** consiste na capacidade de distinção espacial, na imagem, entre dois objectos próximos (não é o tamanho do menor objecto que pode ser visto na imagem). Quantifica-se pela área real que um pixel da imagem representa.



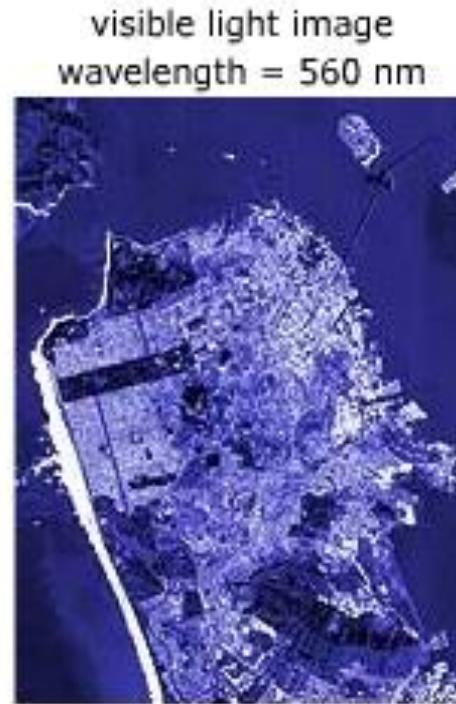
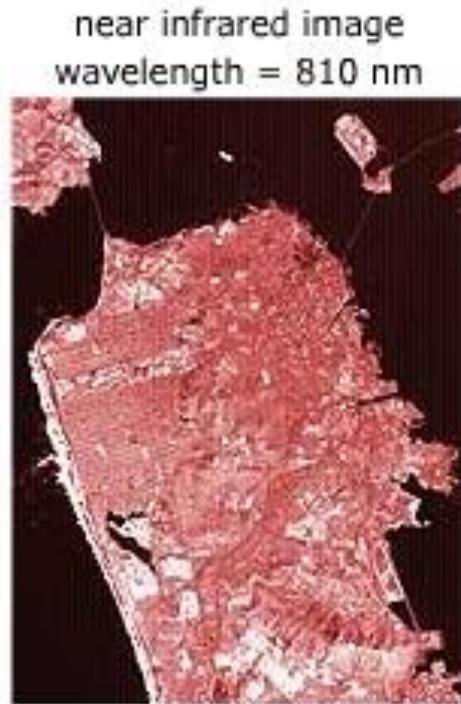
Landsat TM (30 m)



Ikonos (1 m)

# Resolução

**Resolução espectral:** poder de resolução do espectro de frequência, ou seja, medida do poder de resolução dos valores do comprimento de onda, dentro do espectro electromagnético.



# Resolução

Actualmente, a maioria dos satélites mede a energia electromagnética em muitos comprimentos de onda diferentes. Tal é designado como imagem multiespectral.

Imagens tiradas em diferentes comprimentos de onda podem ser combinadas entre si, criando imagens compostas.



Combinação cor-falsa

# Resolução

**Resolução radiométrica:** poder de distinção de diferenças de intensidade dos objectos. É geralmente expressa em níveis de cinzento, ou número de *bits*.

O número total de níveis de cinzento que uma imagem de *n bits* pode ter é igual a  $2^n$ . Por exemplo:

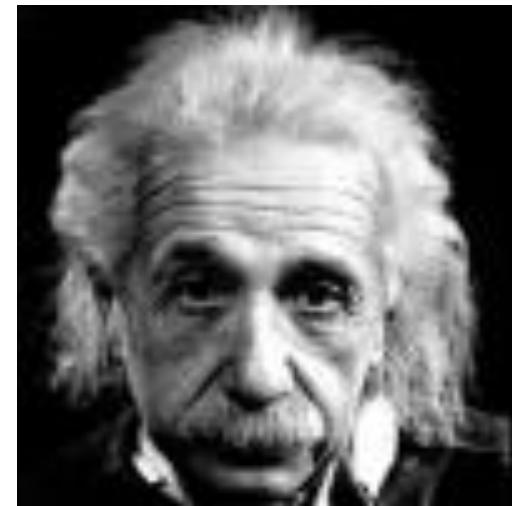


Imagen de 8 *bits*: número de níveis de cinzento =  $2^8 = 256$  (mais comum)

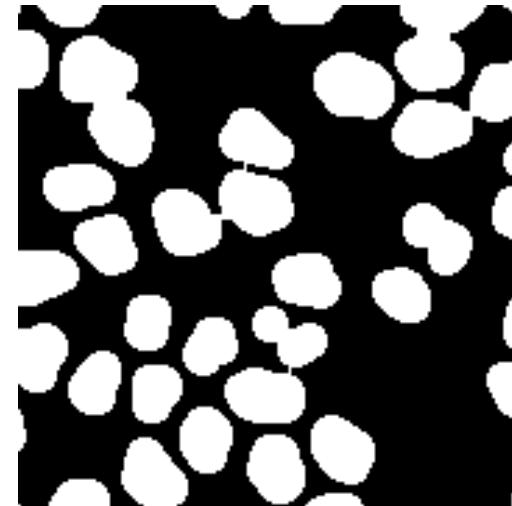


Imagen de 1 *bit*: número de níveis de cinzento =  $2^1 = 2$  (imagem binária)

# Resolução

No caso de uma imagem colorida (RGB), esta é constituída por uma composição de três componentes de 8 *bits* cada, ou seja, a resolução radiométrica é de 24 *bits*.

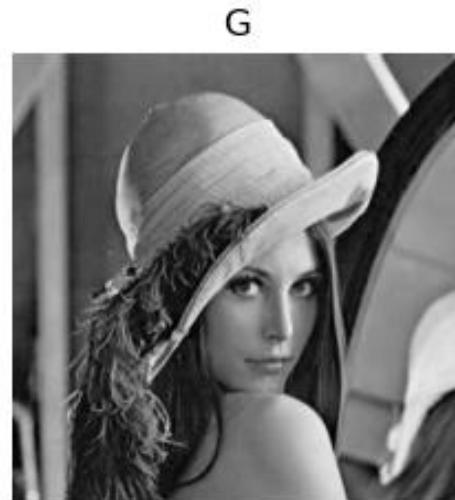
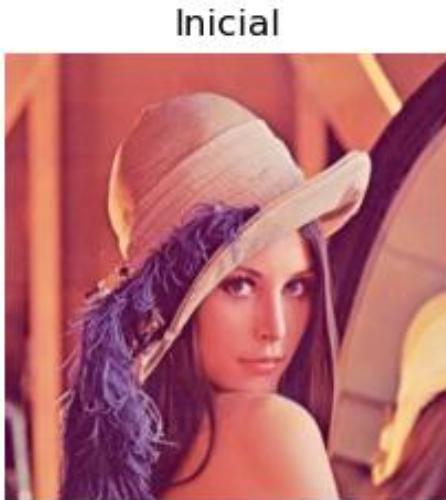
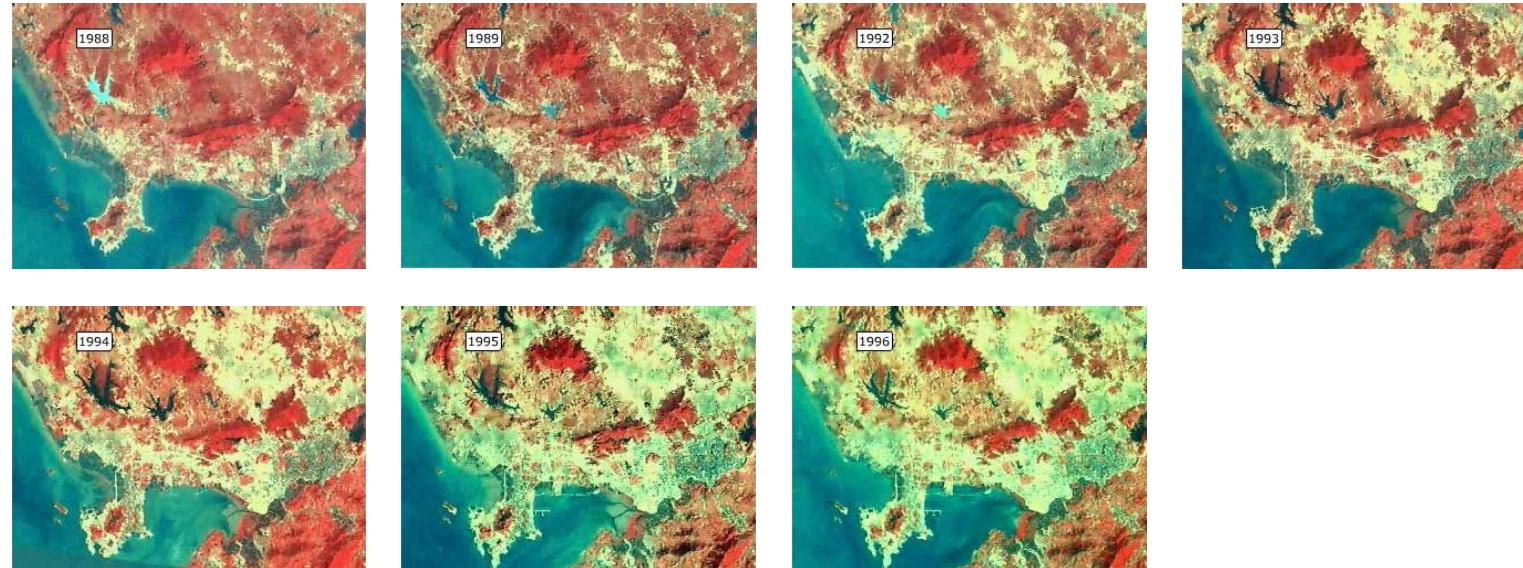


Imagen de 24 *bits*: 3 componentes de 8 *bits* ( $2^8 \times 2^8 \times 2^8 = 2^{24}$ )

# Resolução

**Resolução temporal:** intervalo de tempo depois do qual é adquirida informação acerca de um mesmo objecto.

Séries de imagens que mostram um mesmo objecto, tiradas em momentos diferentes (ao longo do tempo) são referidas como informação multi-temporal.



## 2. Filtragem espacial

Correlação e Convolução. Filtros passa-baixa, passa-alta e passa-banda. Exemplos de filtros digitais de suavização (média, filtro de *Gauss*, mediana, suavização conservativa, *Kuwahara*). Noção de gradiente. Filtros derivativos (*Roberts*, *Prewitt* e *Sobel*). *Unsharp*. Laplaciano. Laplaciano do gaussiano.

Seja  $H$  um filtro bidimensional, caracterizado por uma matriz. Por conveniência, considere-se que  $H$  respeita as seguintes condições:

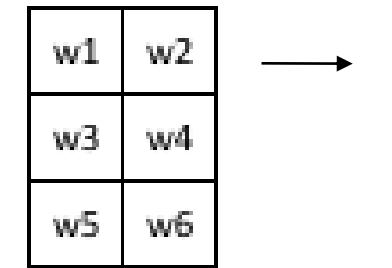
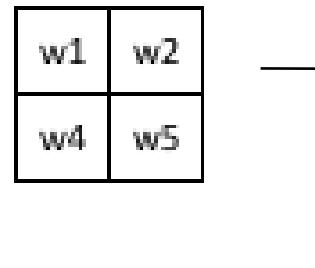
- $H$  é uma matriz quadrada.
- $H$  tem um número ímpar de elementos, ou seja, tem  $(2N+1) \times (2N+1)$  elementos, e que estes estão indexados desde  $-N$  até  $N$ , tal que o elemento central de  $H$  é  $H(0, 0)$ .
- Os valores numéricos ( $w_i$ ) de  $H$  designam-se por “coeficientes”.

Por exemplo:

	-1	0	1
-1	w1	w2	w3
0	w4	w5	w6
1	w7	w8	w9

# Filtro digital 2D

No caso de  $H$  não ter, à partida, um número ímpar de elementos, as anteriores condições podem ser estabelecidas, pois pode-se pegar em qualquer janela e preenchê-la com zeros, por forma a que passe a ser quadrada e com um número ímpar de elementos. Esta operação não muda o comportamento do filtro  $H$  original.



# Correlação / Convolução

Sendo  $f$  uma imagem matricial, define-se a operação de **correlação** (" $\otimes$ ") entre  $H$  e  $F$  por:

$$H \otimes f(x, y) = \sum_{i=-N}^N \sum_{j=-N}^N H(i, j) \times f(x + i, y + j)$$

Por exemplo, para um  $H$  de  $3 \times 3$  tem-se:

$$\begin{aligned} H \otimes f(x, y) &= \sum_{i=-1}^1 \sum_{j=-1}^1 H(i, j) \times f(x + i, y + j) = \\ &= H(-1, -1) \times f(x - 1, y - 1) + H(-1, 0) \times f(x - 1, y) + H(-1, 1) \times f(x - 1, y + 1) + \\ &\quad + H(0, -1) \times f(x, y - 1) + H(0, 0) \times f(x, y) + H(0, 1) \times f(x, y + 1) + \\ &\quad + H(1, -1) \times f(x + 1, y - 1) + H(1, 0) \times f(x + 1, y) + H(1, 1) \times f(x + 1, y + 1) \end{aligned}$$

# Correlação / Convolução

Fernando Soares

A operação de **convolução** ("\*") é semelhante à de correlação. A diferença consiste em primeiro rodar H de 180 graus, e só então aplicar a operação de correlação.

$$H * f(x, y) = \sum_{i=-N}^N \sum_{j=-N}^N H(i, j) \times f(x - i, y - j)$$

Note-se que a correlação e a convolução são operações idênticas se H for simétrico.

# Correlação / Convolução

Fernando Soares

A diferença entre a correlação e a convolução é a de que a última respeita a propriedade associativa. Ou seja, se  $G$  e  $H$  são filtros, então,

$$G * (H * f) = (G * H) * f$$

- A verificação desta propriedade torna-se bastante conveniente quando, por exemplo, se pretende aplicar mais do que um filtro a uma imagem.
- Como geralmente a dimensão da imagem é significativamente maior do que a do filtro, o esforço de cálculo é reduzido, executando-se a convolução entre os dois filtros, seguida da convolução entre o filtro resultante e a imagem.

# Correlação / Convolução

Em resumo, numa operação de filtragem espacial por convolução/correlação, uma das matrizes de entrada é geralmente uma imagem de níveis de cinzento ( $f$ ) e a outra, geralmente bastante mais pequena, é o filtro ( $H$ ). Dentro desta estabelece-se uma posição de referência (designado por “pixel central”, mas que não tem que ser necessariamente o que se encontra no seu centro).

z1	z2	z3	z4	z5	z6	z7	z8	z9	z10
z11	z12	z13	z14	z15	z16	z17	z18	z19	z20
z21	z22	z23	z24	z25	z26	z27	z28	z29	z30
z31	z32	z33	z34	z35	z36	z37	z38	z39	z40
z41	z42	z43	z44	z45	z46	z47	z48	z49	z50
z51	z52	z53	z54	z55	z56	z57	z58	z59	z60
z61	z62	z63	z64	z65	z66	z67	z68	z69	z70

$f$

h00	h01	h02
h10	h11	h12
h20	h21	h22

Filtro H

```
cr[i, j] = h[0, 0]*f[i-1, j-1] + h[0, 1]*f[i-1, j] + h[0, 2]*f[i-1, j+1] + \
           h[1, 0]*f[i, j-1] + h[1, 1]*f[i, j] + h[1, 2]*f[i, j+1] + \
           h[2, 0]*f[i+1, j-1] + h[2, 1]*f[i+1, j] + h[2, 2]*f[i+1, j+1]
```

# Correlação / Convolução

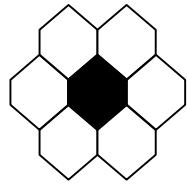
Para os pixels de fronteira da imagem há que fazer uma adaptação para a executar o processo de filtragem. Uma das quatro seguintes opções pode ser seguida:

1. Usa-se apenas a convolução que considere os subconjuntos de pixels de  $H$  que estejam dentro dos limites da imagem  $f$ .
2. São escolhidos valores iguais a zero para os pixels das regiões que estão fora da imagem, mas tal escolha pode distorcer a intensidade dos pixels de fronteira na imagem.
3. Acrescenta-se linhas e colunas à imagem. Cada pixel destas, terá um valor igual ao do pixel da imagem que dele estiver mais próximo.
4. Acrescenta-se linhas e colunas à imagem, por forma a que reflita uma continuidade de carácter periódico, do interior para o exterior da imagem.

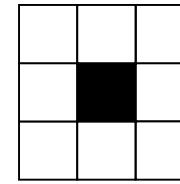
# Propriedades dos filtros digitais

Fernando Soares

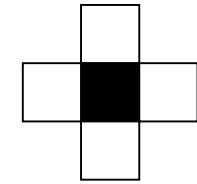
Para além dos valores dos coeficientes associados, a forma e a dimensão são características segundo as quais  $H$  também pode variar. Eis alguns exemplos de geometrias de  $H$ .



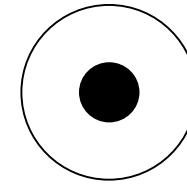
hexágono



quadrado



diamante



círculo



segmento



par de pontos

# Propriedades dos filtros digitais

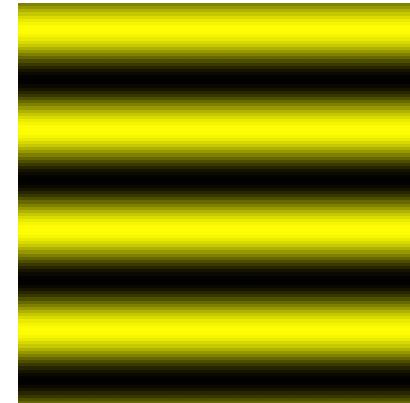
A classificação das janelas de convolução faz-se segundo duas propriedades: convexidade e a isotropia.

	Isotrópico	Anisotrópico
Convexo	Disco 	Segmento 
Não convexo	Contorno do disco 	Par de pontos 

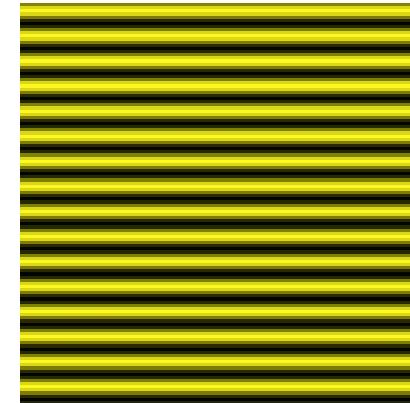
# Filtragem espacial

A frequência espacial de uma imagem é uma característica que pode ser definida pelo número de variações de níveis de cinzento por unidade de distância.

Se os valores numéricos de uma certa área oscilam pouco, diz-se que essa área tem variações de baixa frequência; se oscilam muito diz-se que essa zona é de alta frequência.



Sinal de baixa frequência



Sinal de alta frequência

Os **filtros espaciais** 2D são operadores que permitem alterar a frequência espacial de uma imagem, modificando o valor do tom de cinzento de cada pixel em função dos valores dos tons de cinzento dos pixels da sua vizinhança.

Os filtros podem ser lineares ou não-lineares. Nos filtros lineares cada pixel resulta de uma combinação linear entre os pixels da sua vizinhança, com coeficientes que correspondem aos pesos a atribuir às parcelas.

$$f_1 = H * f(x, y) = \sum_{i=-N}^N \sum_{j=-N}^N H(i, j) \times f(x - i, y - j)$$

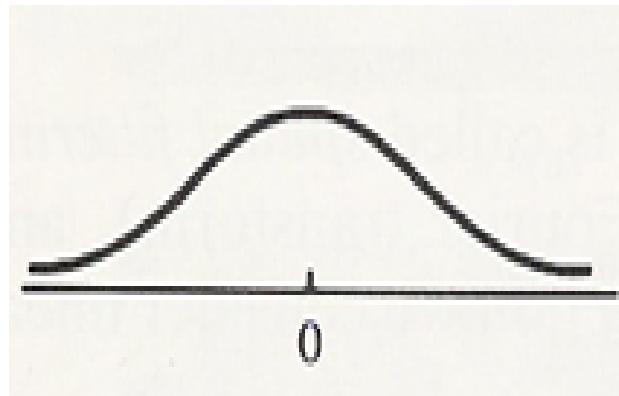
$f_1$ : imagem filtrada  
 $f$ : imagem inicial  
 $H$ : filtro

Quaisquer outros filtros são designados por filtros não lineares.

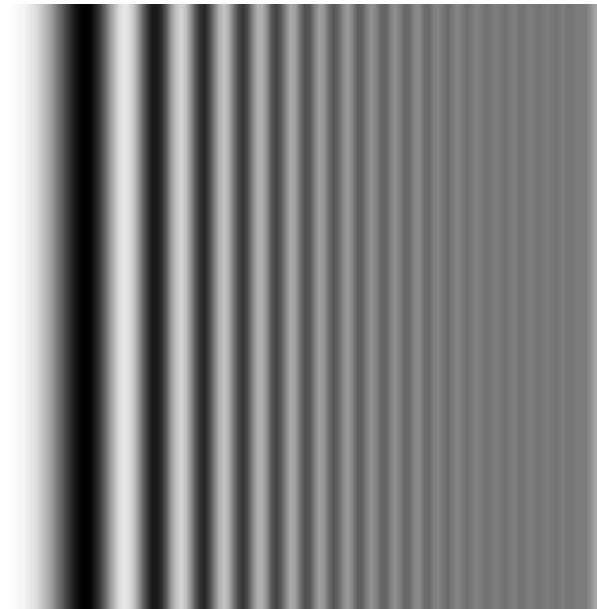
# Filtro passa-baixa

Filtro que suaviza o aspecto da imagem atenuando eventos de elevada frequência (transições abruptas), isto é, as zonas de fronteira radiométrica. Tende a minimizar ruídos e o resultado apresenta um efeito de desfocagem.

A forma da função resposta necessária para implementar um filtro passa-baixa indica que este deve ter todos os seus coeficientes positivos, devendo a respectiva soma ser igual a 1.



Função-resposta do filtro PB



Efeito

# Filtro passa-baixa

Por exemplo, no filtro passa-baixa da média aritmética  $3 \times 3$ , tem-se:

$f_{i-1,j-1}$	$f_{i-1,j}$	$f_{i-1,j+1}$
$f_{i,j-1}$	$f_{i,j}$	$f_{i,j+1}$
$f_{i+1,j-1}$	$f_{i+1,j}$	$f_{i+1,j+1}$

$1/9 \times$

1	1	1
1	1	1
1	1	1

=

$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$

Função

Coeficientes

	$PB_{i,j}$	

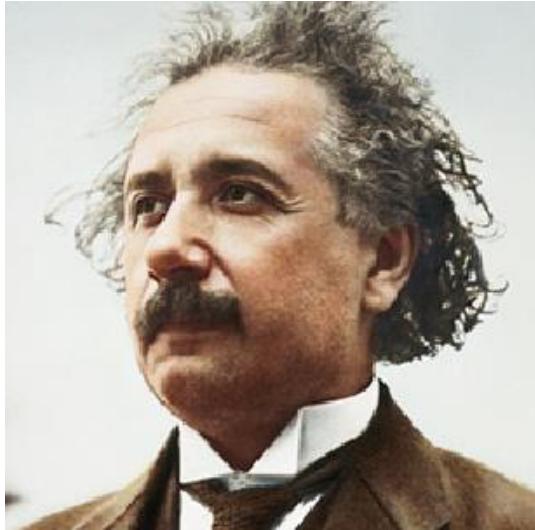
Valor calculado

Por convolução, tem-se:

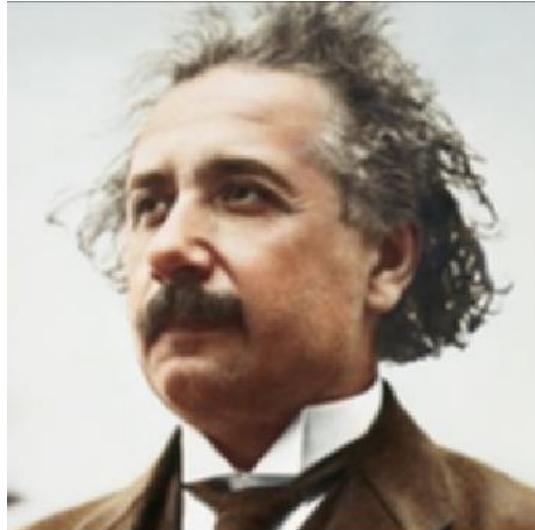
$$PB_{i,j} = \frac{1}{9} \times f_{i-1,j-1} + \frac{1}{9} \times f_{i-1,j} + \frac{1}{9} \times f_{i-1,j+1} + \frac{1}{9} \times f_{i,j-1} + \frac{1}{9} \times f_{i,j} + \frac{1}{9} \times f_{i,j+1} + \frac{1}{9} \times f_{i+1,j-1} + \frac{1}{9} \times f_{i+1,j} + \frac{1}{9} \times f_{i+1,j+1}$$

# Filtro passa-baixa

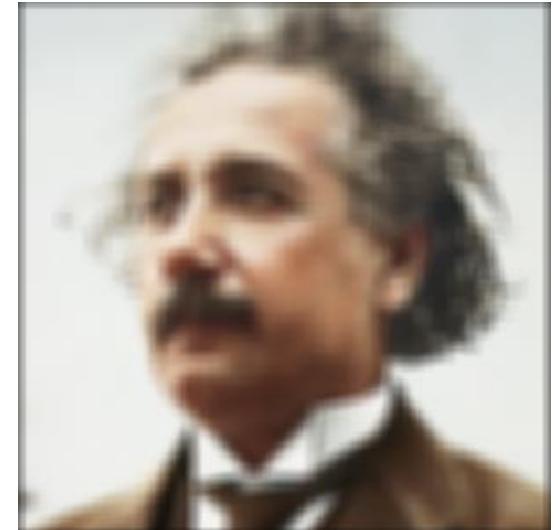
**Média:** É o mais simples filtro linear passa-baixa. Todos os coeficientes são iguais. Calcula-se a média dos tons de cinzento no interior da janela ( $H$ ) e substitui-se o pixel central da janela pelo valor resultante (por esta razão as janelas são normalmente quadradas com dimensão ímpar ( $3 \times 3$ ,  $5 \times 5$ , etc.).



Inicial



Média 3x3



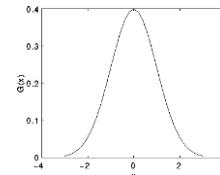
Média 9x9

$$H_{3 \times 3} = \frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

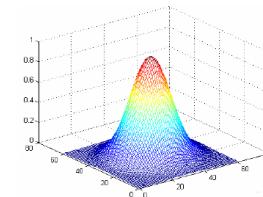
$$H_{9 \times 9} = \frac{1}{81} \times \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix}_{15}$$

# Filtro pass-baixa

**Gaussiano 2-D:** é um operador de convolução que utiliza um kernel representado sob a forma de uma “bossa” gaussiana (em forma de sino).



$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} \times e^{-\left(\frac{(x-x_0)^2}{2\sigma^2}\right)}$$



$$G(x, y) = \frac{1}{2\pi\sigma^2} \times e^{-\left(\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma^2}\right)}$$

```
x, y = np.meshgrid(np.linspace(-l, l, 2*l+1), np.linspace(-l, l, 2*l+1))
G = lambda x, y: 1/(2*np.pi*sigma**2)*np.e**(-(x**2+y**2)/(2*sigma**2))
```

Exemplo de  $7 \times 7$ , centrada em  $(0, 0)$ , com  $\sigma^2 = 1$ :

1.96413e-05	0.00023928	0.00107238	0.00176805	0.00107238	0.00023928	1.96413e-05
0.00023928	0.00291502	0.0130642	0.0215393	0.0130642	0.00291502	0.00023928
0.00107238	0.0130642	0.0585498	0.0965324	0.0585498	0.0130642	0.00107238
0.00176805	0.0215393	0.0965324	0.159155	0.0965324	0.0215393	0.00176805
0.00107238	0.0130642	0.0585498	0.0965324	0.0585498	0.0130642	0.00107238
0.00023928	0.00291502	0.0130642	0.0215393	0.0130642	0.00291502	0.00023928
1.96413e-05	0.00023928	0.00107238	0.00176805	0.00107238	0.00023928	1.96413e-05

0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0
0	1	3	5	3	1	0	0
0	1	5	8	5	1	0	0
0	1	3	5	3	1	0	0
0	0	1	1	1	0	0	0
0	0	0	0	0	0	0	0

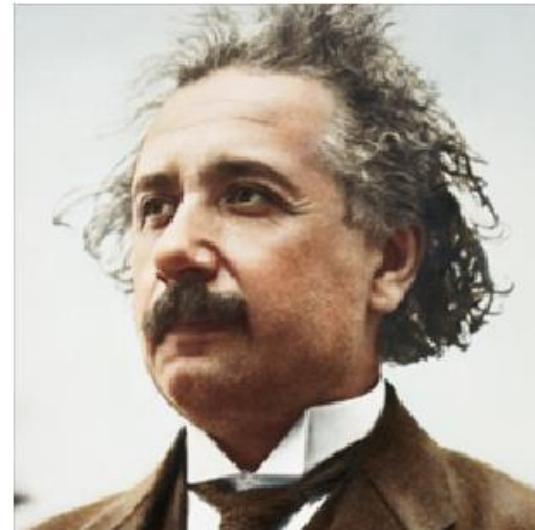
=

$$\times \frac{1}{49} =$$

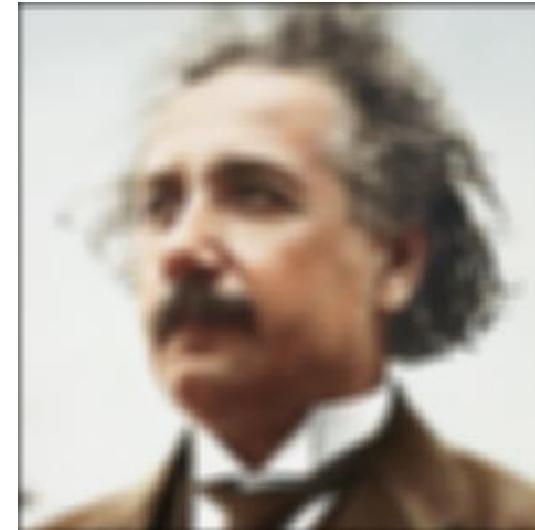
# Filtro passa-baixa

O filtro gaussiano é uma média ponderada, com maior peso aplicado ao pixel central, diminuindo progressivamente para o exterior. Como tal, proporciona uma suavização mais “delicada” que o da média, preservando melhor as fronteiras entre objectos.

O grau de suavização é determinado pelo valor da variância (ou do desvio-padrão) da função de Gauss (funções com desvios-padrão mais altos requerem janelas de convolução maiores no sentido de as funções ficarem mas bem representadas).



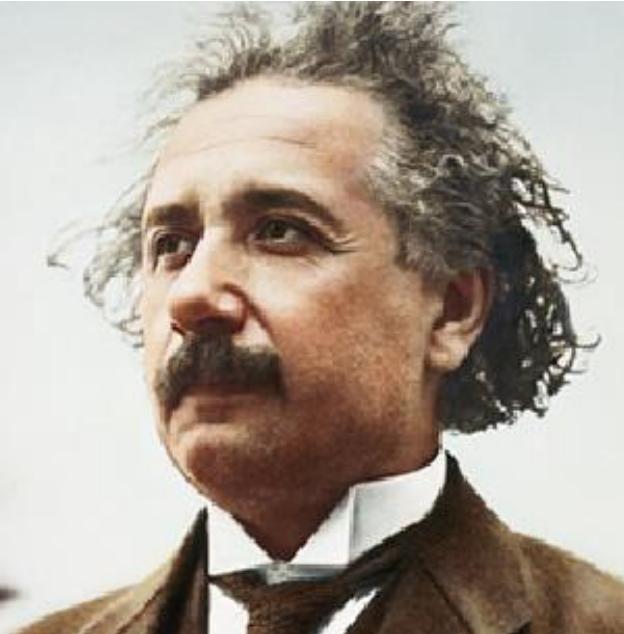
Gaussiana ( $\sigma^2 = 0.5$ )



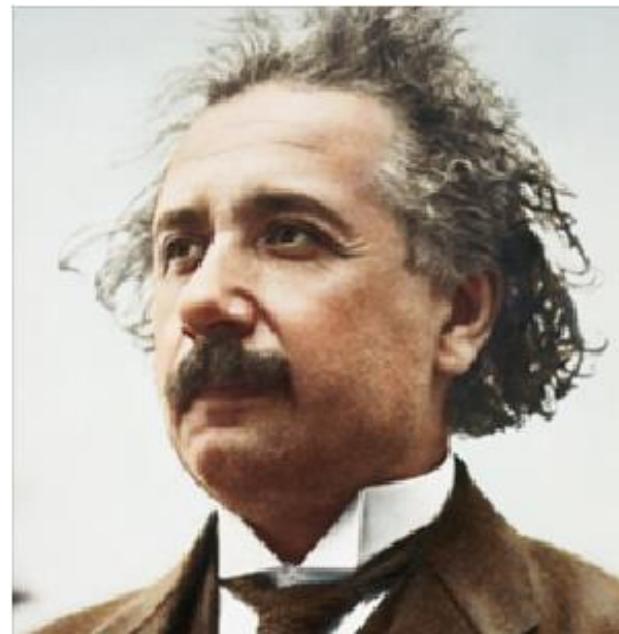
Gaussiana ( $\sigma^2 = 5$ )

# Filtro passa-baixa

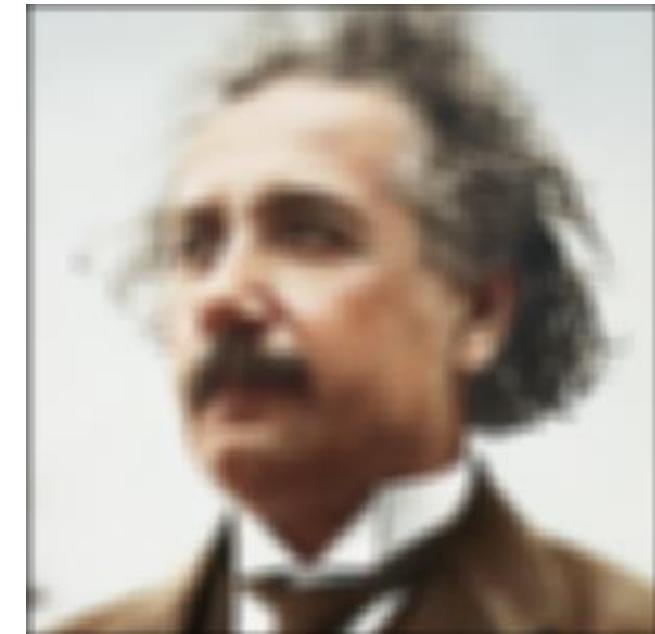
**Exemplo:** diferença entre os filtros da média e gaussiano.



inicial



Gaussiano 9 x 9



Média 9 x 9

# Filtro passa-baixa

**Mediana:** é um filtro não-linear de suavização. O resultado é geralmente melhor que o do filtro da média, quando usado em imagens com ruído do tipo Sal-e-Pimenta, ou Speckle. Ainda, tende a preserver melhor as fronteiras dos objectos.

123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

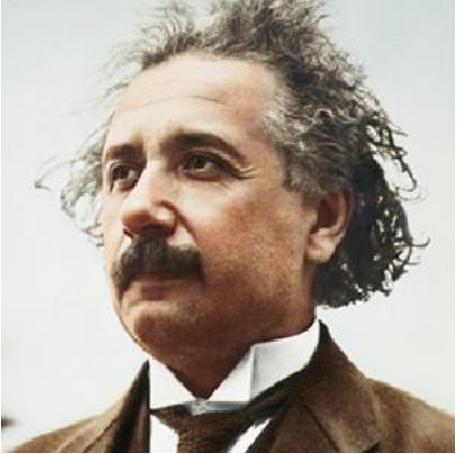
Valores da vizinhança

115, 119, 120, 123, 124,  
125, 126, 127, 150

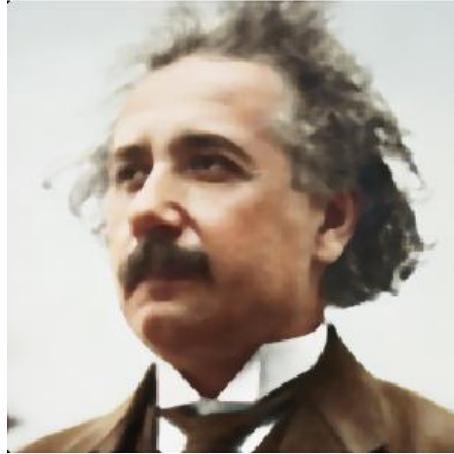
Mediana: 124

# Filtro passa-baixa

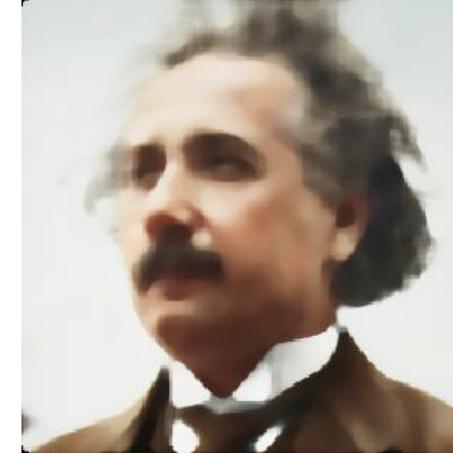
Exemplo: filtro da mediana.



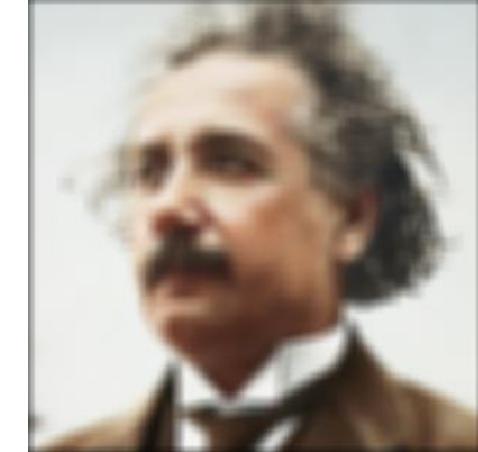
Inicial



Mediana 5 x 5



Mediana 9 x 9



Média 9 x 9

# Filtro passa-baixa

**Kuwahara:** Suaviza uma imagem sem perturbar a nitidez e a posição das fronteiras.

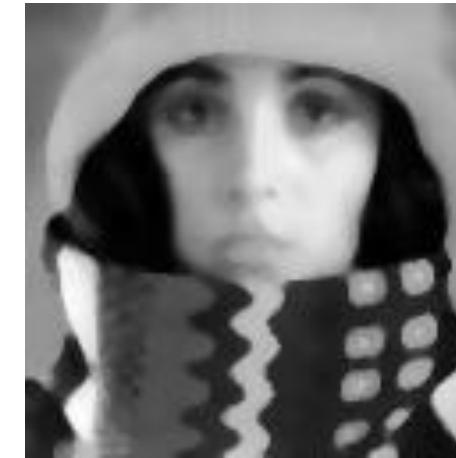
Embora possa ser implementado em janelas de formas diversas, considera-se aqui uma janela quadrada de dimensão ímpar. Esta janela é dividida em 4 regiões e em cada uma delas calcula-se a intensidade média  $m_i$  e a variância  $s_i^2$ , ( $i = 1, 2, 3, 4$ ). O valor atribuído ao pixel central da janela corresponde ao valor médio da janela que tem menor variância.



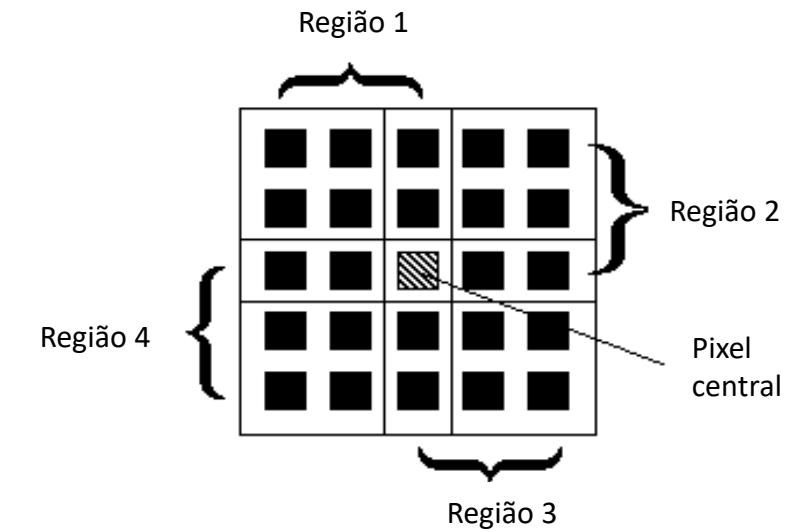
Inicial



Kuwahara 5 x 5



Mediana 5 x 5



# Filtro passa-alta

Ao contrário da filtragem passa-baixa, que esbate/elimina os eventos de detalhe contidos nas imagens, a diferenciação vai ter o efeito contrário, ou seja, evidenciar o detalhe. Definem-se assim outros filtros designados por filtros derivativos (operadores de gradiente).

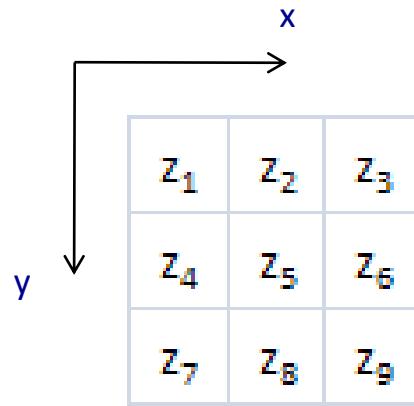
**Gradiente:** O gradiente de uma função  $f$ , no ponto  $(x,y)$ , define-se por  $\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$

A magnitude é dada por  $mag(\nabla f) = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$ .

Estes conceitos constituem a base de diversas abordagens de diferenciação da imagem.

# Filtro passa-alta

Considerando a janela da figura, pode aproximar-se a equação anterior no ponto  $z_5$  de diversas formas. A mais simples é utilizar a diferença  $(z_5 - z_6)$  para definir a derivada parcial na direcção x e a diferença  $(z_5 - z_8)$  para definir a derivada parcial na direcção y.



$$f(z_5) = \text{mag}(\nabla f) \approx \sqrt{(z_5 - z_6)^2 + (z_5 - z_8)^2}$$

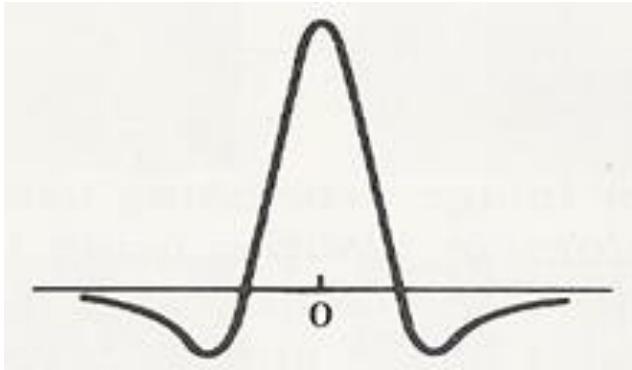
ou

$$f(z_5) = \text{mag}(\nabla f) \approx |z_5 - z_6| + |z_5 - z_8|$$

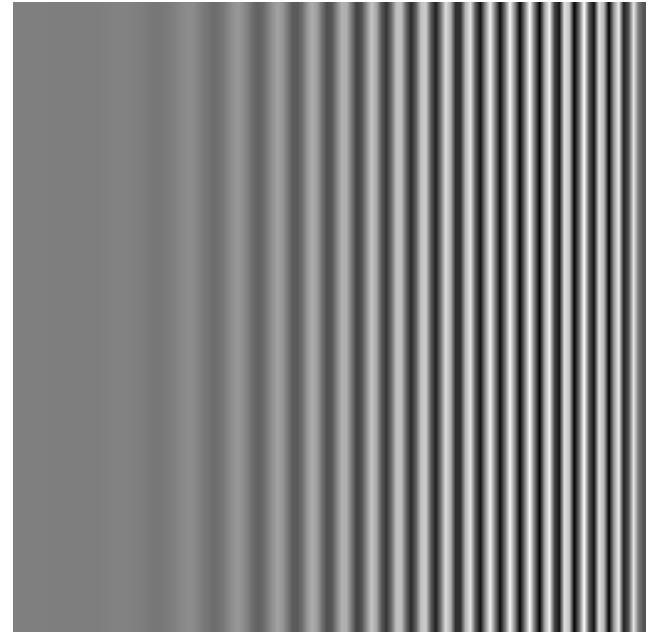
# Filtro passa-alta

Filtro que atenua, ou elimina, os eventos da imagem com baixa frequência, pelo que os filtros tornam mais nítidas as fronteiras radiométricas e os detalhes.

A forma da função resposta necessária para implementar um filtro passa-alta indica que este deve ter os coeficientes positivos na vizinhança do centro e negativos na periferia, devendo a respectiva soma ser igual a 0.



Função-resposta



Efeito

$$\text{Filtro Passa-Alta} = \text{Imagen original} - \text{Filtro Passa-Baixa}$$

Por exemplo, a partir do filtro da média do slide 14, tem-se:

$$\begin{aligned} PA_{i,j} &= f_{i,j} - PB_{i,j} = \\ &= f_{i,j} - \left( \frac{1}{9} \times f_{i-1,j-1} + \frac{1}{9} \times f_{i-1,j} + \frac{1}{9} \times f_{i-1,j+1} + \frac{1}{9} \times f_{i,j-1} + \frac{1}{9} \times f_{i,j} + \frac{1}{9} \times f_{i,j+1} + \frac{1}{9} \times f_{i+1,j-1} + \frac{1}{9} \times f_{i+1,j} + \frac{1}{9} \times f_{i+1,j+1} \right) = \\ &= -\frac{1}{9} \times f_{i-1,j-1} - \frac{1}{9} \times f_{i-1,j} - \frac{1}{9} \times f_{i-1,j+1} - \frac{1}{9} \times f_{i,j-1} + \frac{8}{9} \times f_{i,j} - \frac{1}{9} \times f_{i,j+1} - \frac{1}{9} \times f_{i+1,j-1} - \frac{1}{9} \times f_{i+1,j} - \frac{1}{9} \times f_{i+1,j+1} \end{aligned}$$

$f_{i-1,j-1}$	$f_{i-1,j}$	$f_{i-1,j+1}$
$f_{i,j-1}$	$f_{i,j}$	$f_{i,j+1}$
$f_{i+1,j-1}$	$f_{i+1,j}$	$f_{i+1,j+1}$

 $1/9 \times$ 

-1	-1	-1
-1	8	-1
-1	-1	-1

-1/9	-1/9	-1/9
-1/9	8/9	-1/9
-1/9	-1/9	-1/9

	$PA_{i,j}$	

Função

Coeficientes

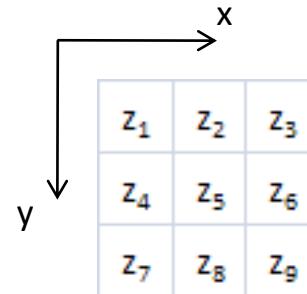
Valor a calcular

# Filtro passa-alta

**Roberts:** este filtro executa o gradiente cruzado, isto é, em vez de calcular as diferenças de valores de brilho na direção vertical e horizontal, fá-lo numa direção rodada de 45º, onde as janelas de convolução são as seguintes:

$$\begin{array}{|c|c|} \hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array} \quad \begin{array}{|c|c|} \hline 0 & 1 \\ \hline -1 & 0 \\ \hline \end{array}$$

$$f(z_5) = \text{mag}(\nabla f) \approx \sqrt{(z_5 - z_9)^2 + (z_6 - z_8)^2}$$

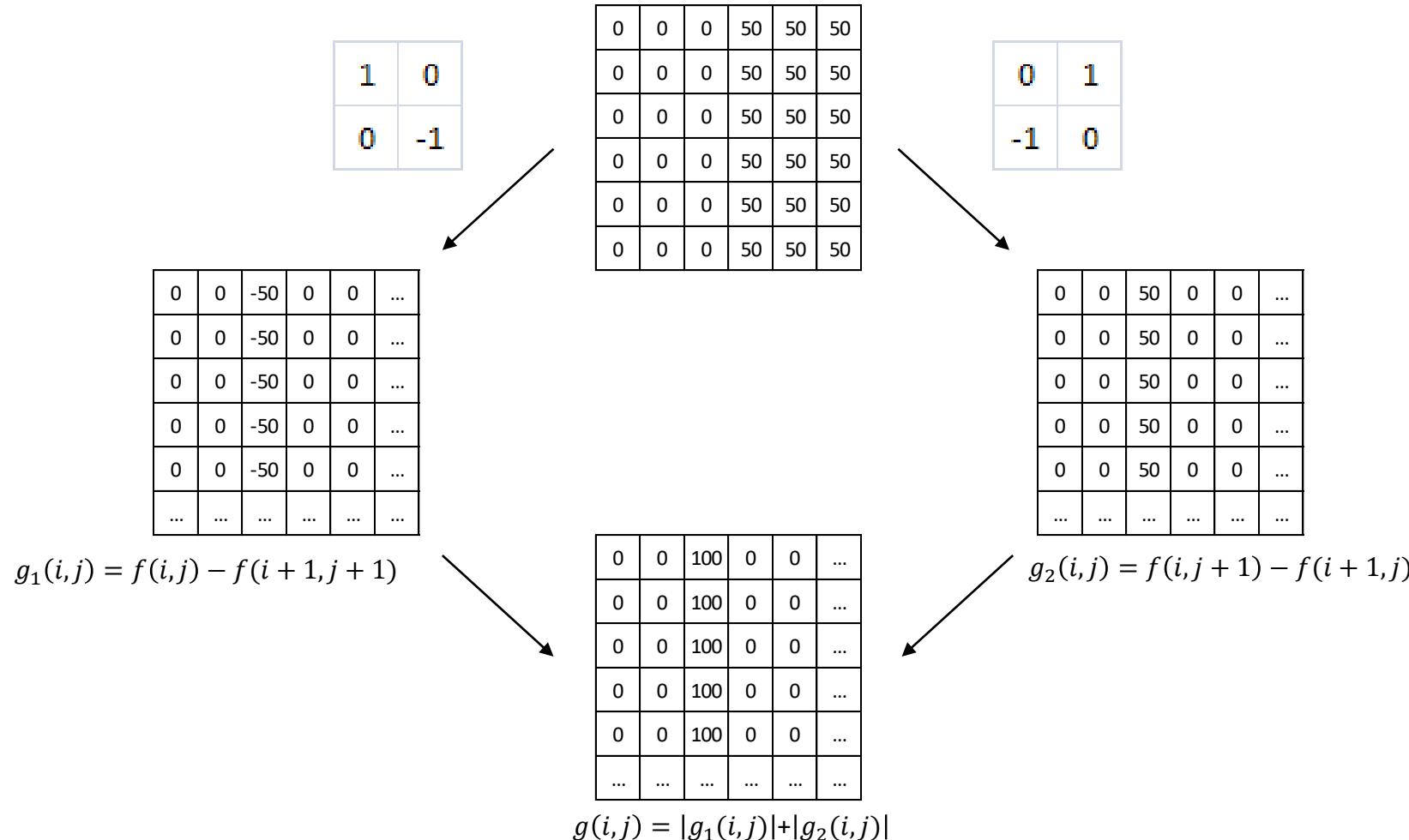


ou

$$f(z_5) = \text{mag}(\nabla f) \approx |z_5 - z_9| + |z_6 - z_8|$$

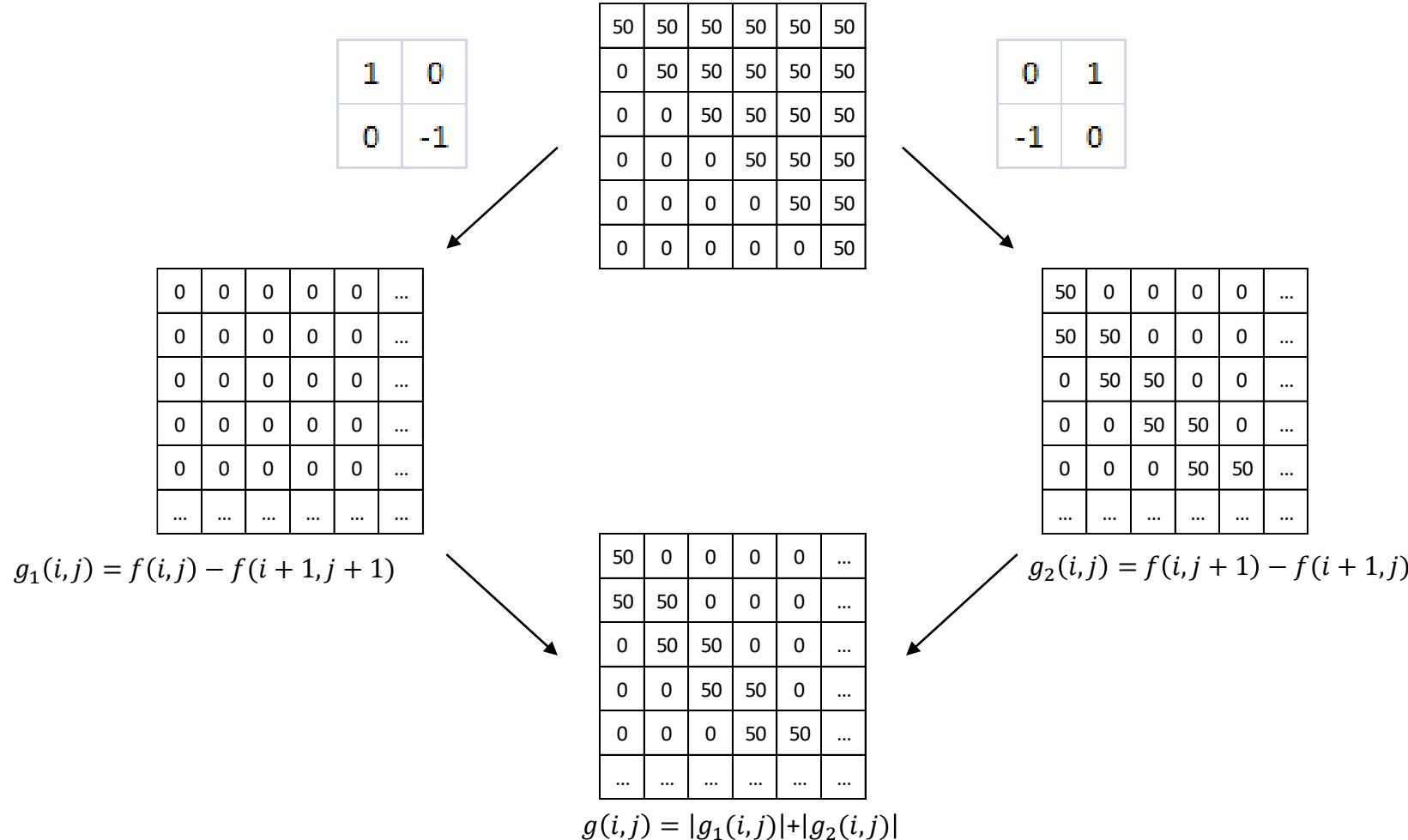
# Filtro passa-alta

Roberts (exemplo 1):



# Filtro passa-alta

**Roberts (exemplo 2):**



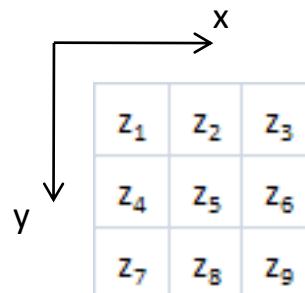
# Filtro passa-alta

**Sobel:** este operador realça linhas verticais e horizontais mais escuras que o fundo, sem realçar pontos isolados. Consiste na aplicação de duas máscaras, descritas a seguir, que compõem um resultado único.

$$\begin{array}{c} x \\ \begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \end{array} \quad \begin{array}{c} y \\ \begin{array}{|c|c|c|} \hline -1 & -2 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 2 & 1 \\ \hline \end{array} \end{array}$$

$$S_x = (z_1 + 2 \times z_4 + z_7) - (z_3 + 2 \times z_6 + z_9)$$

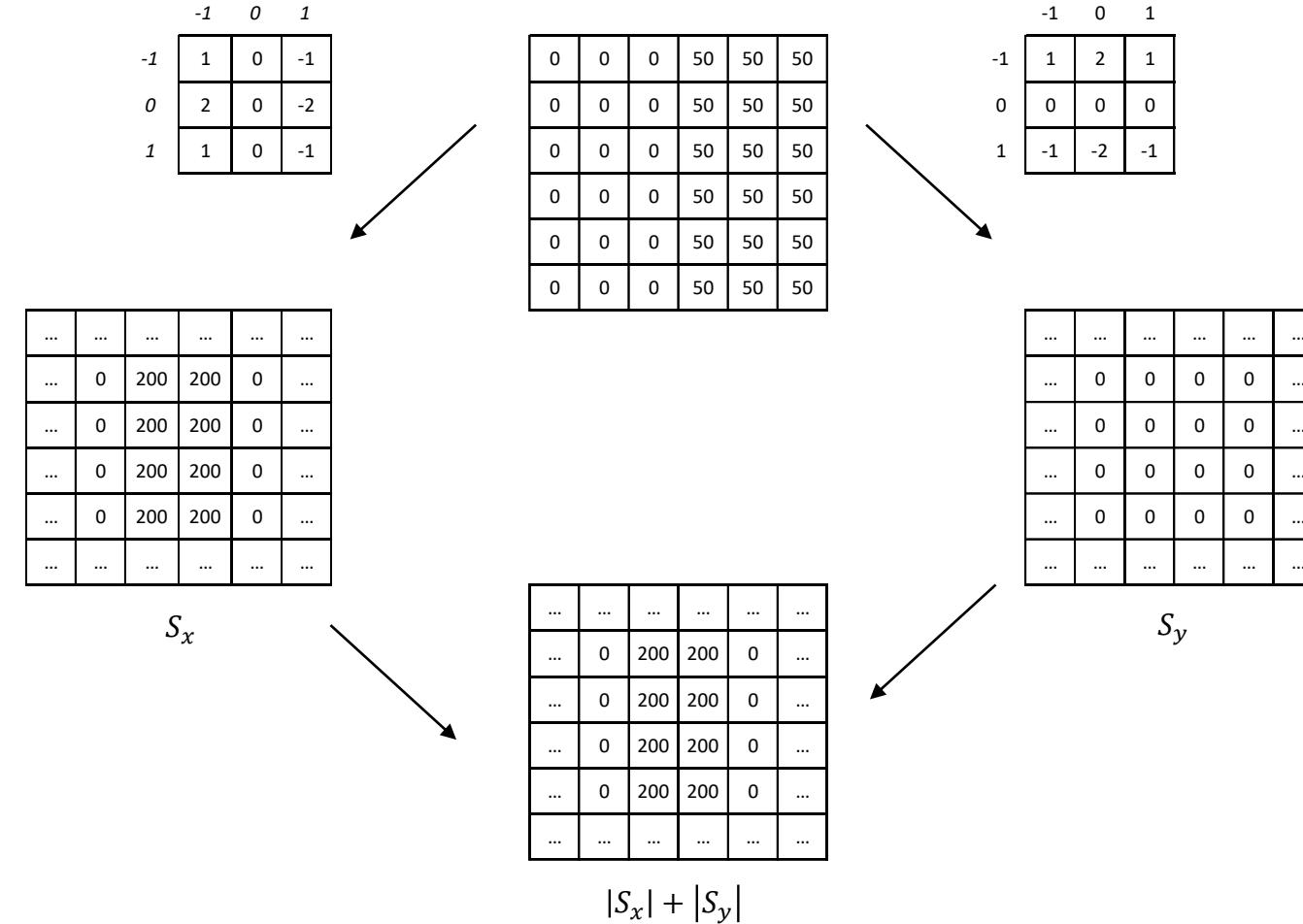
$$S_y = (z_1 + 2 \times z_2 + z_3) - (z_7 + 2 \times z_8 + z_9)$$



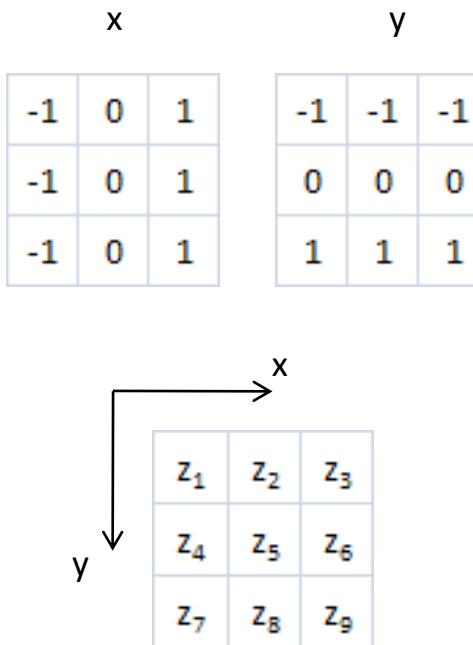
$$f(z_5) = |S_x| + |S_y|$$

# Filtro passa-alta

**Sobel (exemplo):**



**Prewitt:** este operador realça linhas verticais e horizontais mais escuras que o fundo, sem realçar pontos isolados. Consiste na aplicação de duas máscaras, descritas a seguir, que compõem um resultado único.



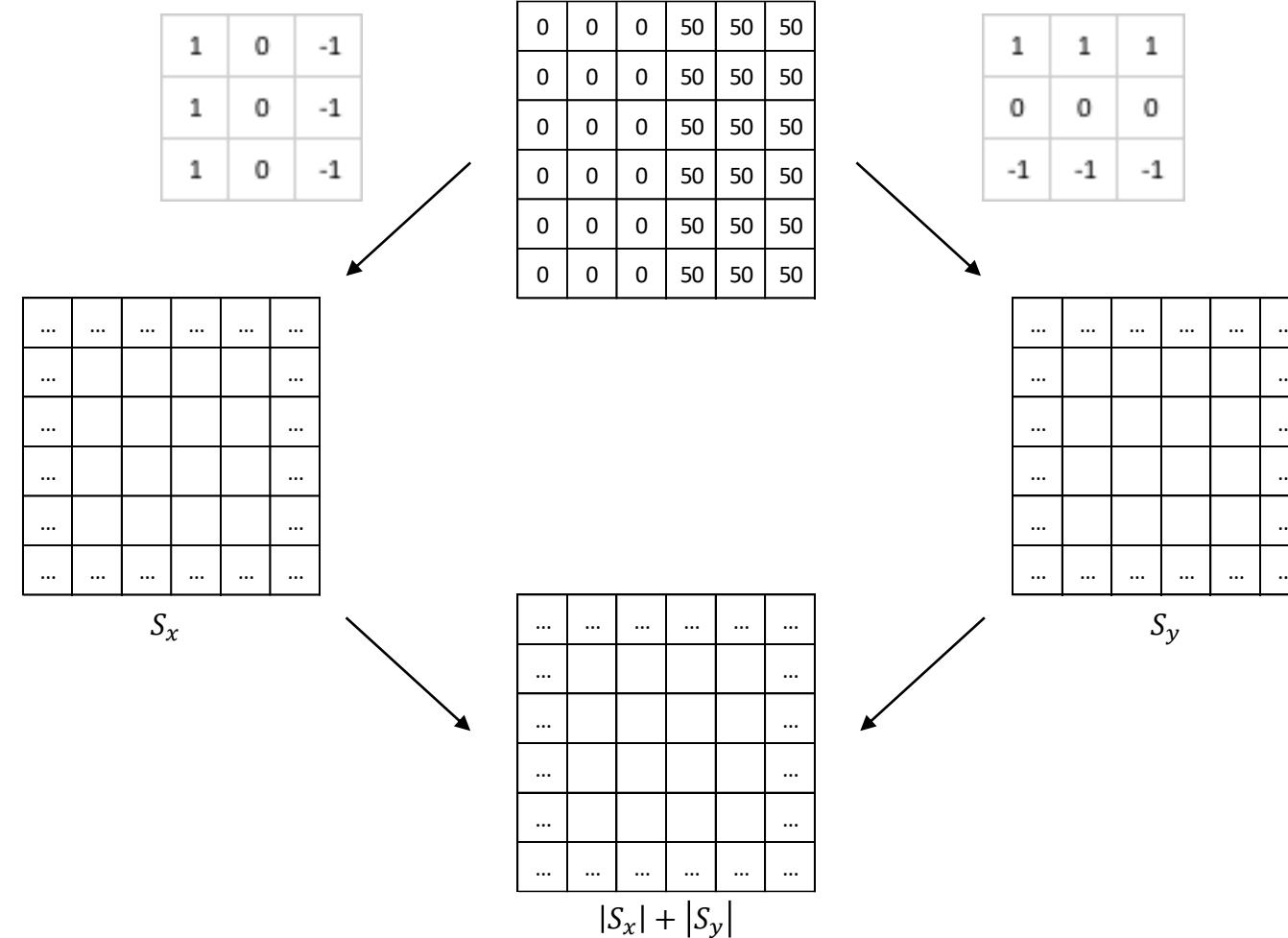
$$S_x = (z_1 + z_4 + z_7) - (z_3 + z_6 + z_9)$$

$$S_y = (z_1 + z_2 + z_3) - (z_7 + z_8 + z_9)$$

$$f(z_5) = |S_x| + |S_y|$$

# Filtro passa-alta

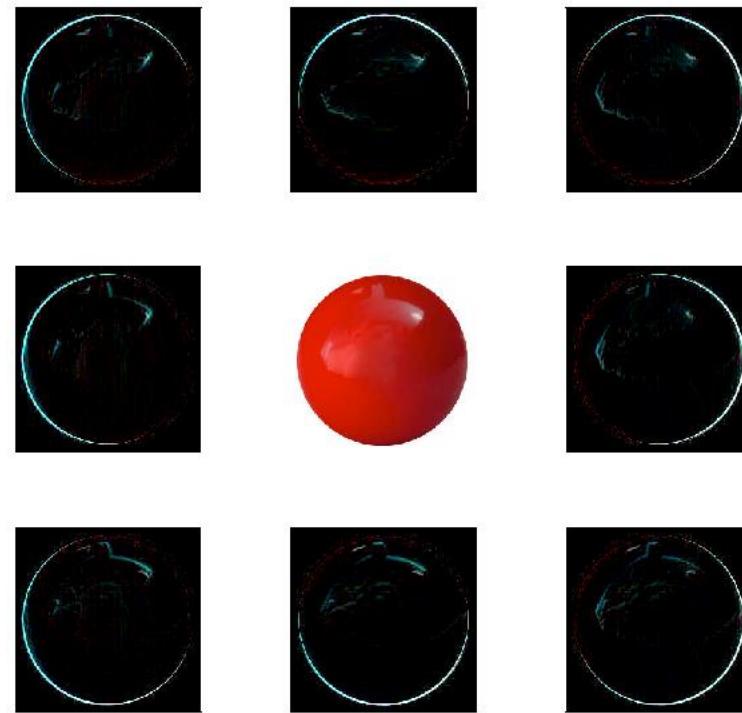
**Prewitt (completar):**



# Filtro passa-alta

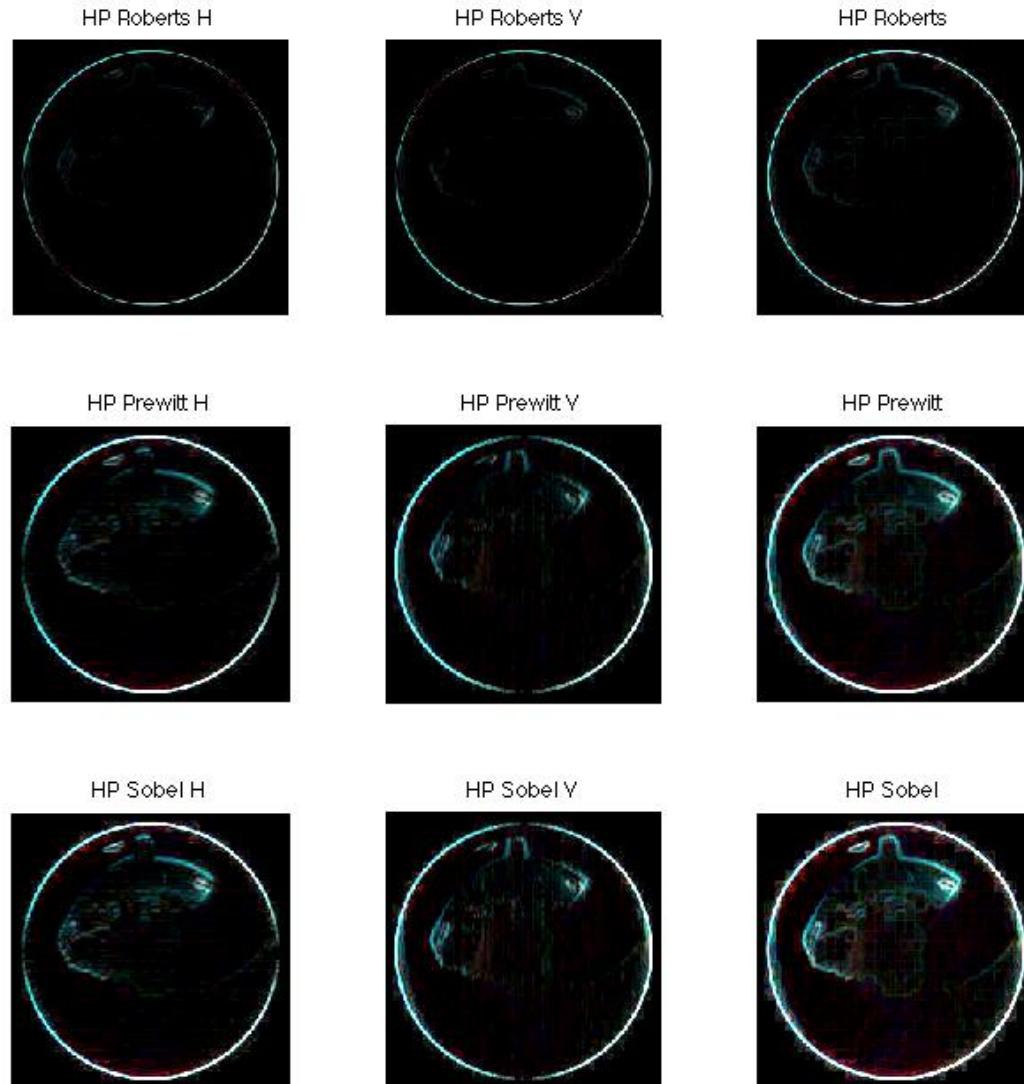
Os filtros passa-alta podem ser "desenhados" em função da direção. Neste caso o *kernel* contém coeficientes que variam em função da orientação que apresentam na imagem as fronteiras que se pretende realçar.

NW	N	NE
$\begin{matrix} 1 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & -1 \end{matrix}$	$\begin{matrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{matrix}$	$\begin{matrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{matrix}$
W		E
$\begin{matrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{matrix}$		$\begin{matrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{matrix}$
SW	S	SE
$\begin{matrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{matrix}$	$\begin{matrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{matrix}$	$\begin{matrix} -1 & -1 & 1 \\ -1 & -2 & 1 \\ 1 & 1 & 1 \end{matrix}$



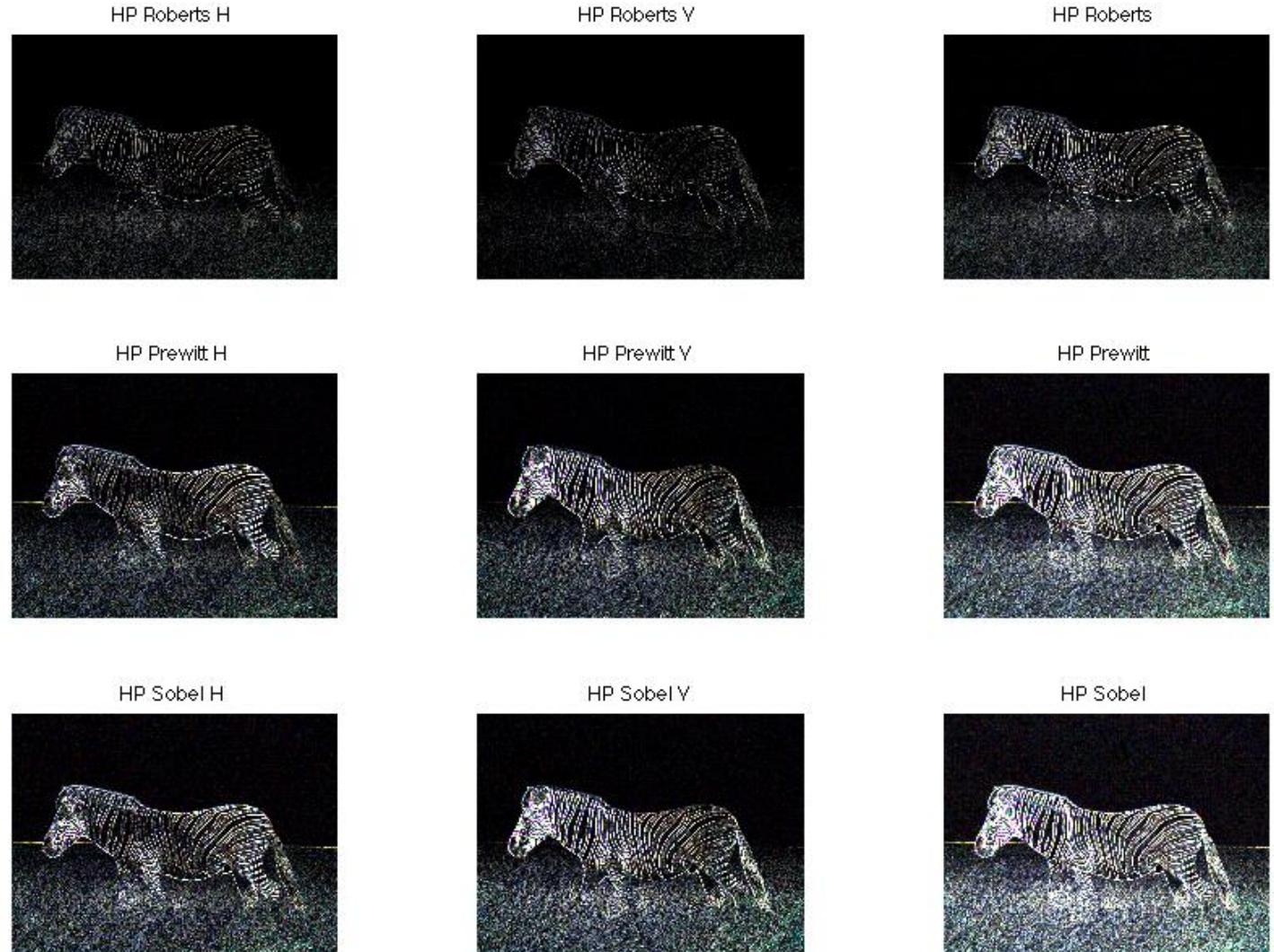
# Filtro passa-alta

Operadores de gradiente:



# Filtro passa-alta

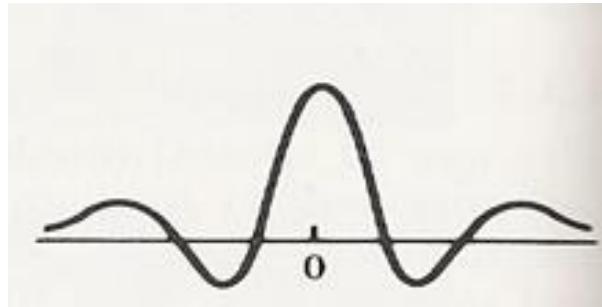
Operadores de gradiente:



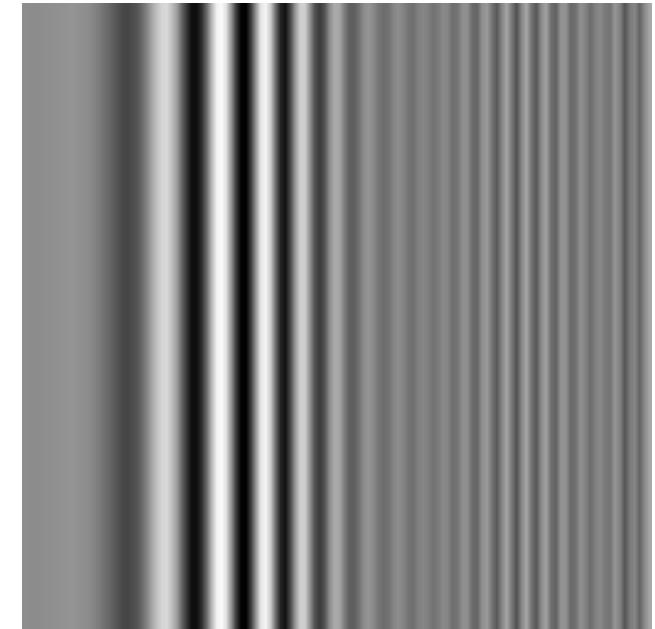
# Filtro passa-banda

Filtro que remove/atenua determinados intervalos de frequências.

A forma da função resposta necessária para implementar um filtro passa-banda indica que este deve ter os coeficientes positivos na vizinhança do centro e alternadamente negativos e positivos no sentido da periferia.



Função-resposta



Passa-banda

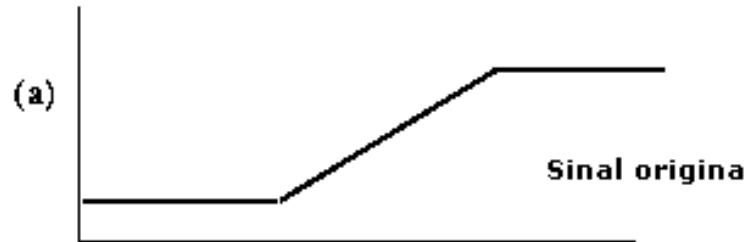
$$\text{Filtro Passa-Banda} = \text{Filtro Passa-Baixa 1} - \text{Filtro Passa-Baixa 2}$$

Tipicamente, os filtros PB1 e PB2 devem representar médias de curto-termo e de longo-termo.

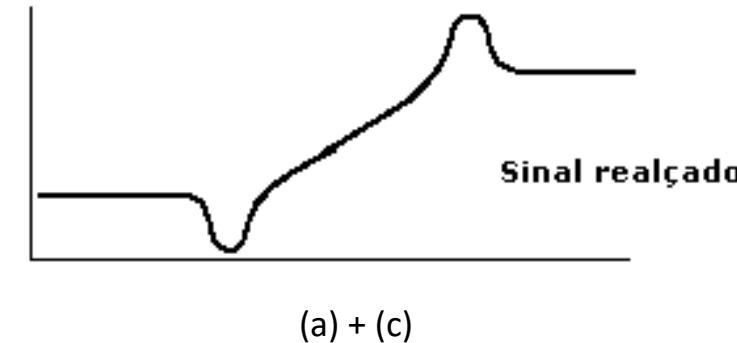
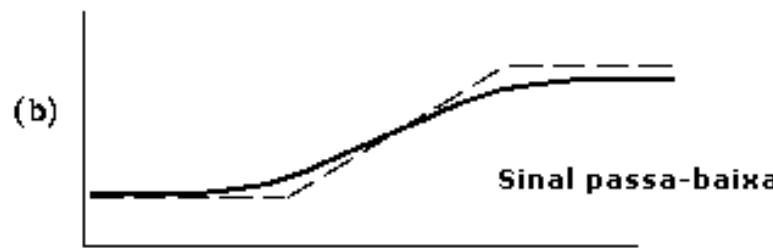
Os filtros Passa-Banda são geralmente usados para realçar as fronteiras e outras características de filtragem passa-alta na presença de ruído.

# Operação de unsharp

Permite fazer sobressair as fronteiras dos objectos de uma imagem, através da operação de subtracção entre a imagem original e a imagem suavizada com um filtro passa-baixa.

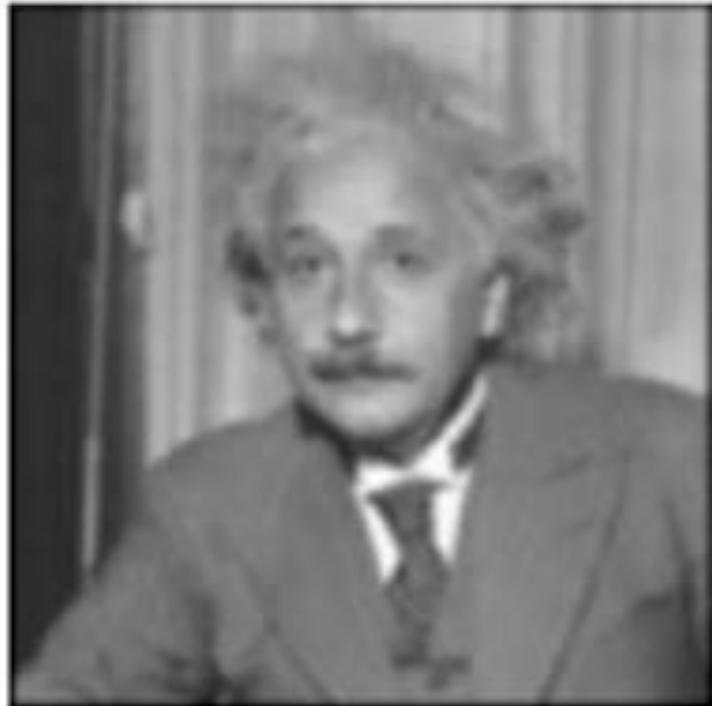


$$u(i,j) = f(i,j) + k \times [f(i,j) - PB(i,j)], \quad 0 \leq k \leq 1$$

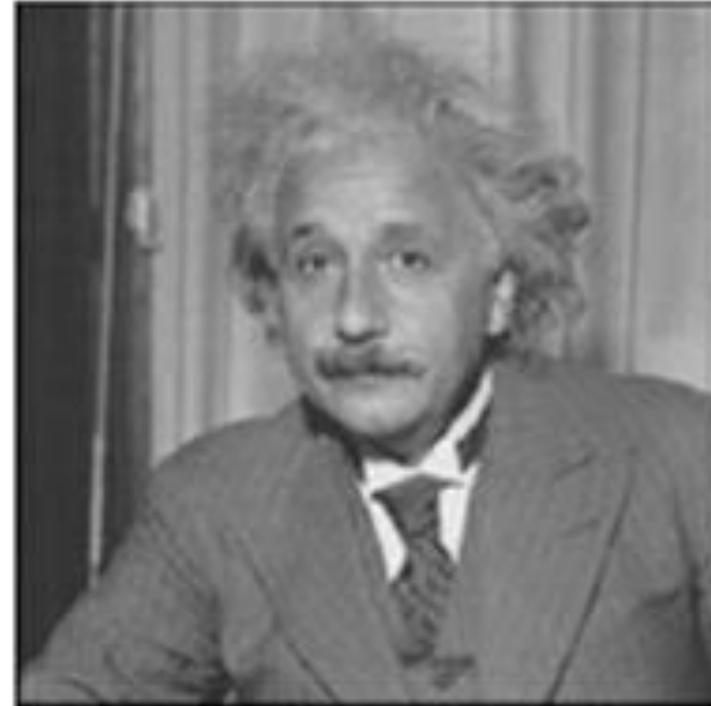


# Operação de unsharp

Exemplo:



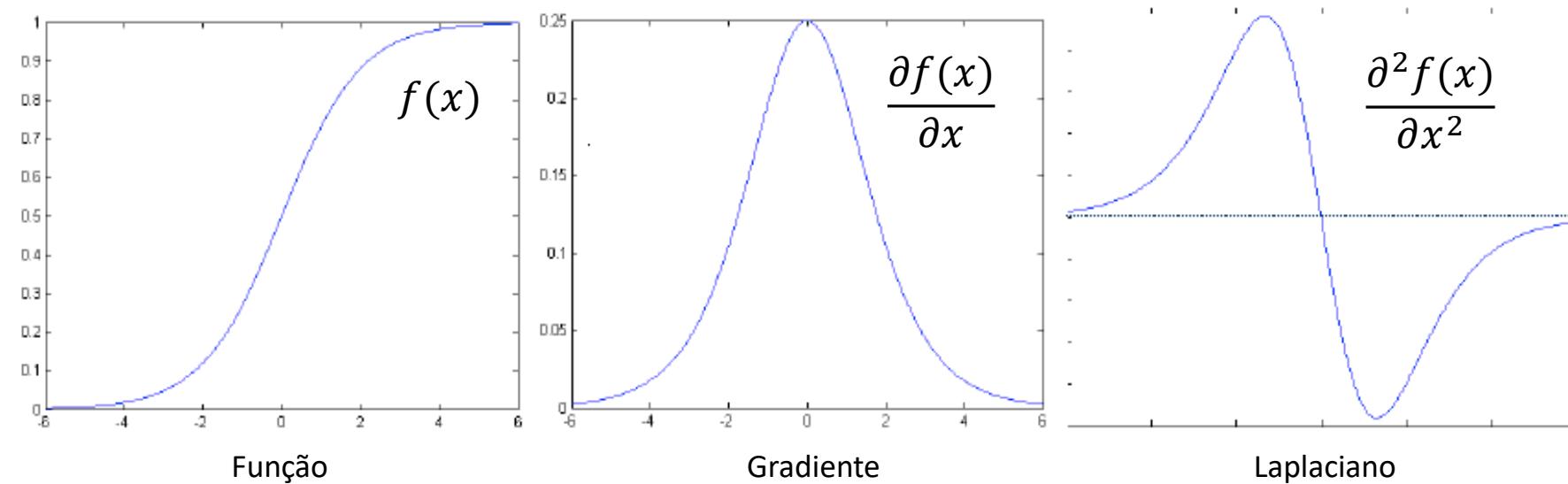
Antes



Depois

# Filtro laplaciano

O filtro laplaciano distingue-se dos restantes filtros de realce de fronteiras porque usa a informação de segundas derivadas relativa às variações de intensidade dos pixels.



No espaço 2D o laplaciano define-se como:

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}$$

As exigências para a definição do laplaciano digital são as de o coeficiente associado com o pixel central ser negativo e os coeficientes dos pixels externos serem positivos (laplaciano negativo), ou vice-versa (laplaciano positivo)

Como o laplaciano é uma derivada, a soma dos coeficientes tem que ser nula (toda a vez que o ponto em questão e seus vizinhos tiverem o mesmo valor, a resposta será nula).

# Filtro laplaciano

No caso discreto, para uma vizinhança de  $3\times 3$ , o laplaciano (negativo) pode ser aproximado por um operador de conectividade-4 ou um de conectividade-8:

$$\nabla^2 f(i,j) = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\nabla^2 f(i,j) = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 1 \\ 0 & -2 & 0 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Usando qualquer um dos anteriores *kernels*, a filtragem é ser calculada por convolução:

$$\nabla^2 f(i,j) = -f(i-1,j) - f(i+1,j) - f(i,j-1) - f(i,j+1) + 4 \times f(i,j)$$

$$\begin{aligned} \nabla^2 f(i,j) = & -f(i-1,j) - f(i+1,j) - f(i,j-1) - f(i,j+1) - \\ & -f(i-1,j-1) - f(i+1,j+1) - f(i-1,j+1) - f(i+1,j-1) + 8 \times f(i,j) \end{aligned}$$

Porque estes *kernels* são uma aproximação à segunda derivada, os operadores são bastante sensíveis à presença de ruído aleatório na imagem.

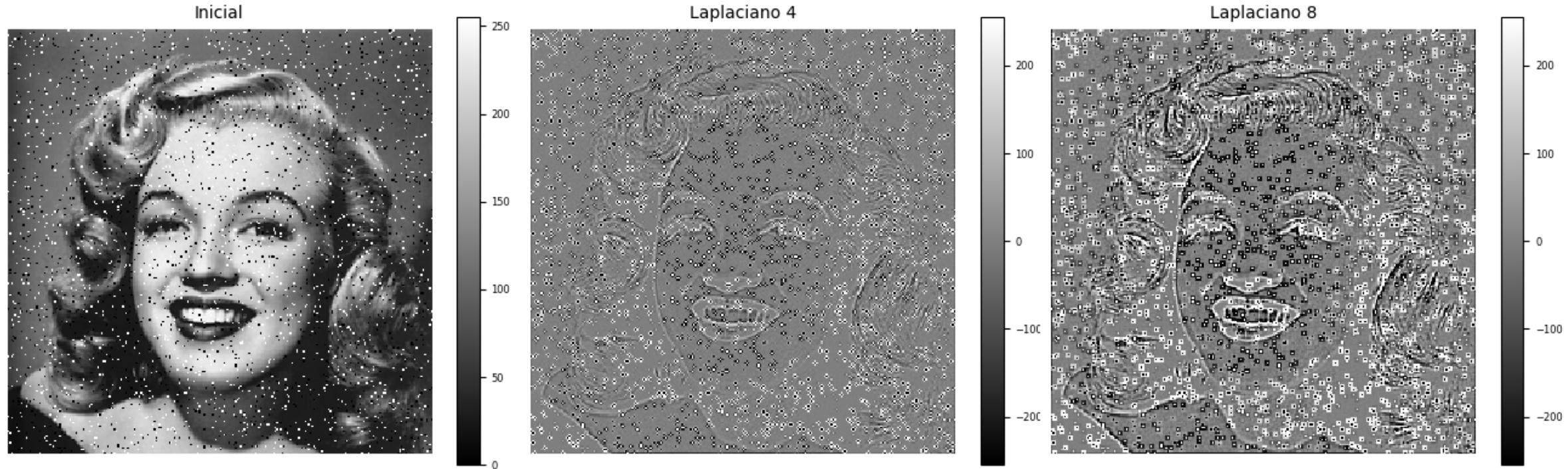
# Filtro laplaciano

Exemplo 1:



# Filtro laplaciano

Exemplo 2:



# Filtro laplaciano do gaussiano

Fernando Soares

Para atenuar o efeito da presença de ruído, a imagem é geralmente filtrada, primeiro com um filtro passa-baixa gaussiano, antes de aplicar o operador laplaciano. Esta tarefa reduz o ruído de alta frequência antes da diferenciação.

Como a operação de convolução é associativa, pode-se executar em primeiro lugar a convolução do filtro passa-baixa Gaussiano com operador laplaciano, e só depois executar a convolução da imagem com este operador híbrido (*LoG - Laplacian of Gaussian*). Desta forma têm-se as seguintes vantagens:

- Como ambos os *kernels* gaussiano e laplaciano são geralmente bastante menores que a imagem, este método requer de longe muito menos operações aritméticas.
- O filtro LoG pode ser pré-calculado antecipadamente e, como tal, executa-se apenas uma operação de convolução, em vez de duas.

# Filtro laplaciano do gaussiano

$$G(x, y) = \frac{1}{2\pi\sigma^2} \times e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}$$

Função gaussiana

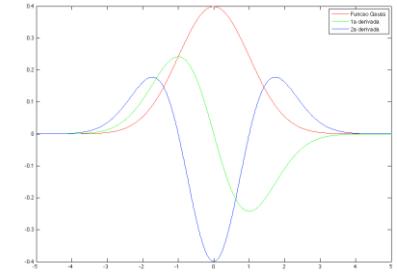
$\downarrow$                                    $\downarrow$

$$\frac{\partial^2}{\partial x^2} G(x, y) = \frac{x^2 - \sigma^2}{2\pi\sigma^6} \times e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}$$

2ª derivada em ordem a x

$$\frac{\partial^2}{\partial y^2} G(x, y) = \frac{y^2 - \sigma^2}{2\pi\sigma^6} \times e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}$$

2ª derivada em ordem a y



$$LoG(x, y) = \nabla^2(G(x, y)) = \frac{\partial^2}{\partial x^2} G(x, y) + \frac{\partial^2}{\partial y^2} G(x, y) = -\frac{1}{\pi\sigma^4} \times \left(1 - \frac{x^2 + y^2}{2\sigma^2}\right) \times e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}$$

Laplaciano do gaussiano

# Filtro laplaciano do gaussiano

A aproximação discreta do operador LoG pode ser obtida com,

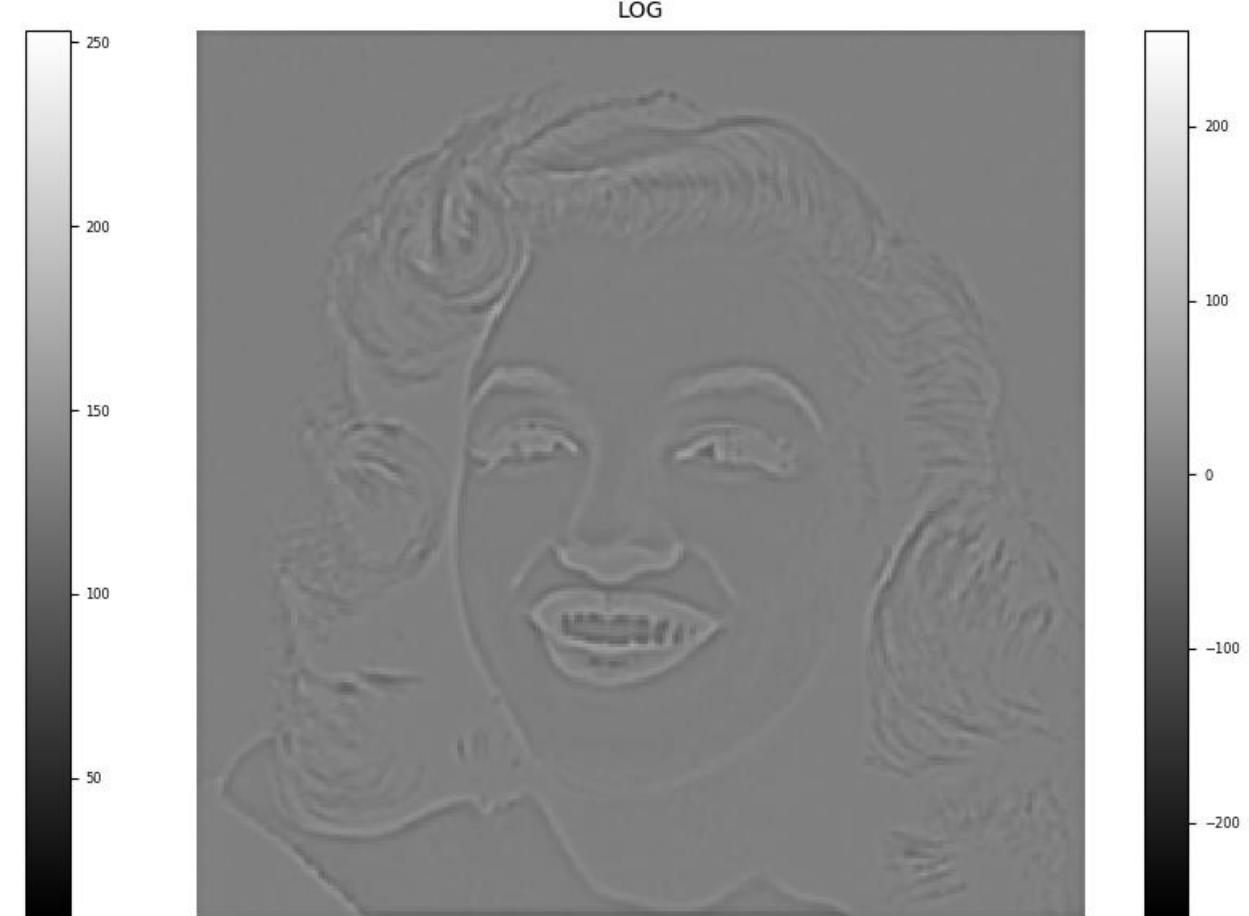
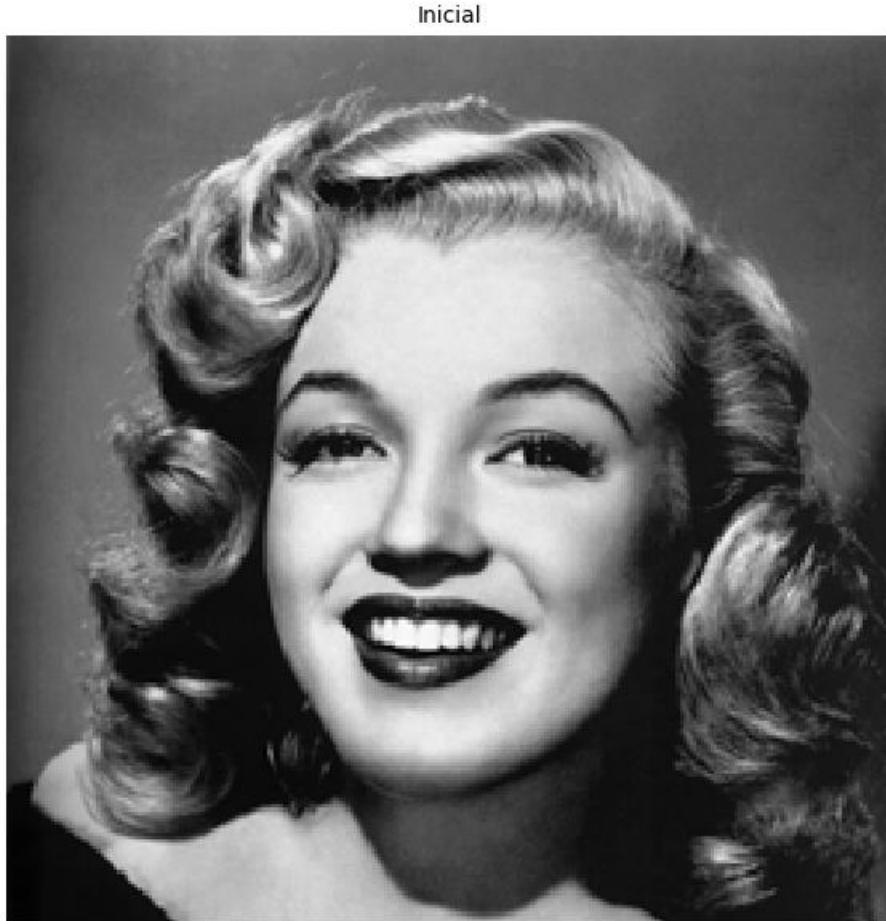
```
Lap0fGauss = lambda x, y: -1/(np.pi*sigma**4) * (1-((x-x0)**2+(y-y0)**2)/(2*sigma**2)) * np.e**(-((x-x0)**2+(y-y0)**2)/(2*sigma**2))
```

0.00031426	0.00263208	0.00857902	0.0123764	0.00857902	0.00263208	0.00031426
0.00263208	0.0174901	0.0391927	0.0430786	0.0391927	0.0174901	0.00263208
0.00857902	0.0391927	-0	-0.0965324	-0	0.0391927	0.00857902
0.0123764	0.0430786	-0.0965324	-0.31831	-0.0965324	0.0430786	0.0123764
0.00857902	0.0391927	-0	-0.0965324	-0	0.0391927	0.00857902
0.00263208	0.0174901	0.0391927	0.0430786	0.0391927	0.0174901	0.00263208
0.00031426	0.00263208	0.00857902	0.0123764	0.00857902	0.00263208	0.00031426

$$= \begin{matrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 2 & 2 & 1 & 0 \\ 0 & 2 & -0 & -5 & -0 & 2 & 0 \\ 1 & 2 & -5 & -16 & -5 & 2 & 1 \\ 0 & 2 & -0 & -5 & -0 & 2 & 0 \\ 0 & 1 & 2 & 2 & 2 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{matrix} \times \frac{1}{49} =$$

# Filtro laplaciano do gaussiano

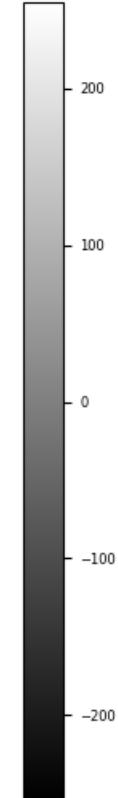
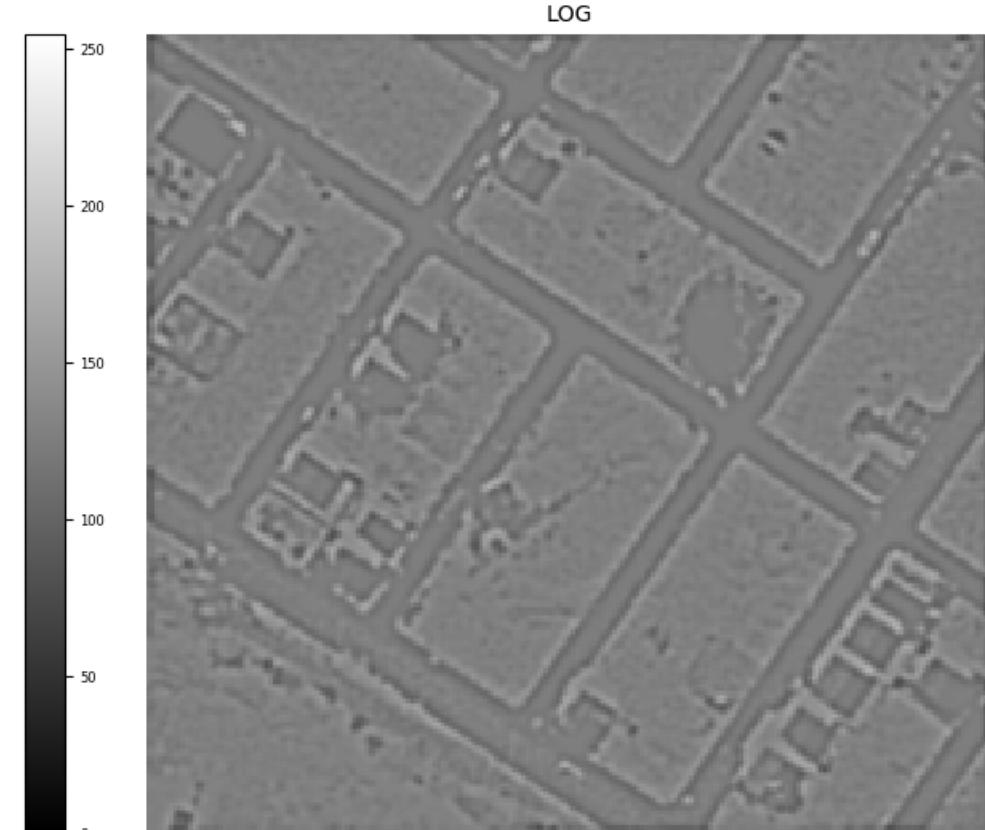
Exemplo 1:



# Filtro laplaciano do gaussiano

Fernando Soares

Exemplo 2:



# 3. Domínio das frequências

Sinal e série amostral. Transformada de *Fourier*. Espectro de frequências. Filtragem no espaço de *Fourier*. Teorema da convolução. Exemplos de aplicação.

# Sinal e série amostral

Uma onda periódica elementar como o seno fica definida por uma das seguintes expressões (em função do período T, ou da frequência f):

$$P(t) = A \times \sin\left(\frac{2\pi}{T}t + \varphi\right) + C$$

$$P(t) = A \times \sin(2\pi ft + \varphi) + C$$

A: amplitude

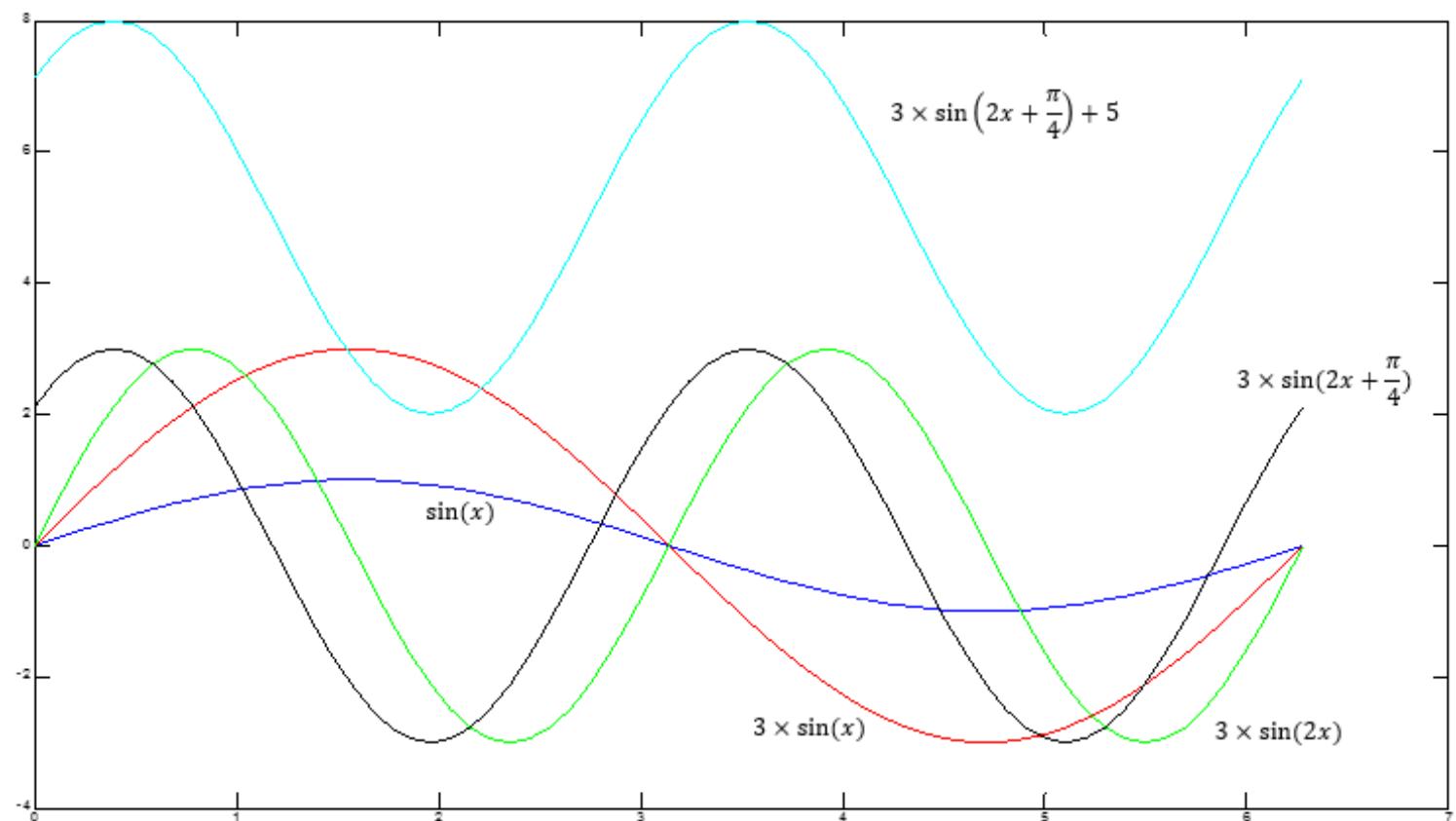
f: frequência linear

T: período

$\varphi$  : fase

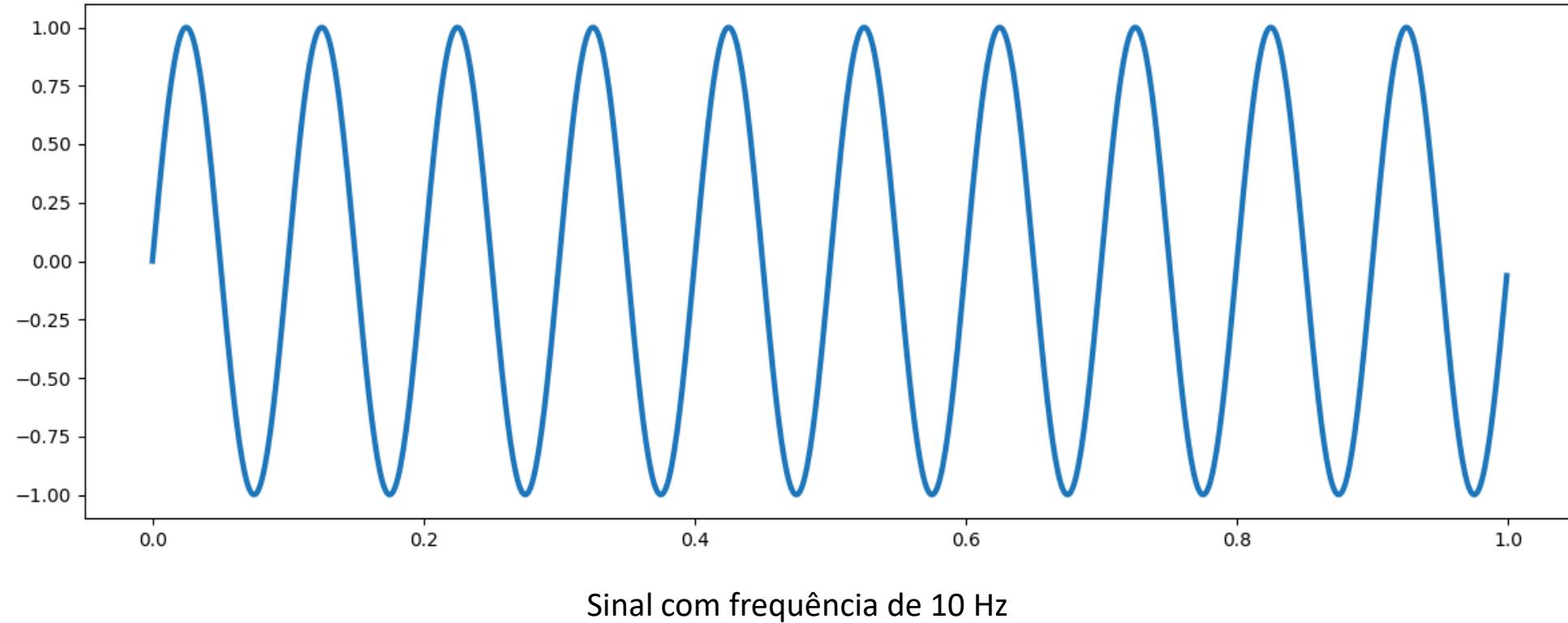
C: translação

$\omega = 2\pi f$ : frequência angular



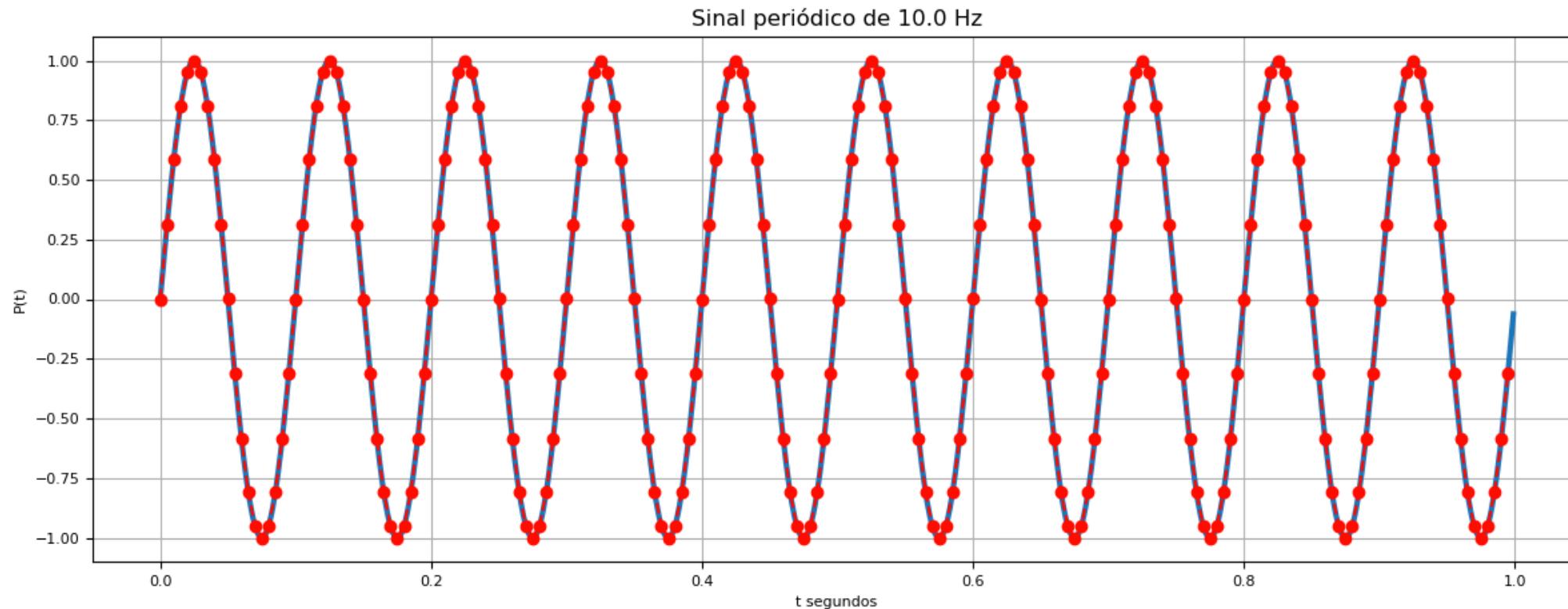
# Sinal e série amostral

Num sinal periódico  $P$ , o valor da frequência  $f$ , quando expressa em Hertz (Hz), corresponde ao número de ciclos  $m$  que ocorre no intervalo de tempo de um segundo.



# Sinal e série amostral

Num processo de digitalização de um sinal analógico, a amostragem de P consiste em gerar uma sequência discreta S de impulsos (série amostral) com uma frequência de amostragem  $f_s$ .



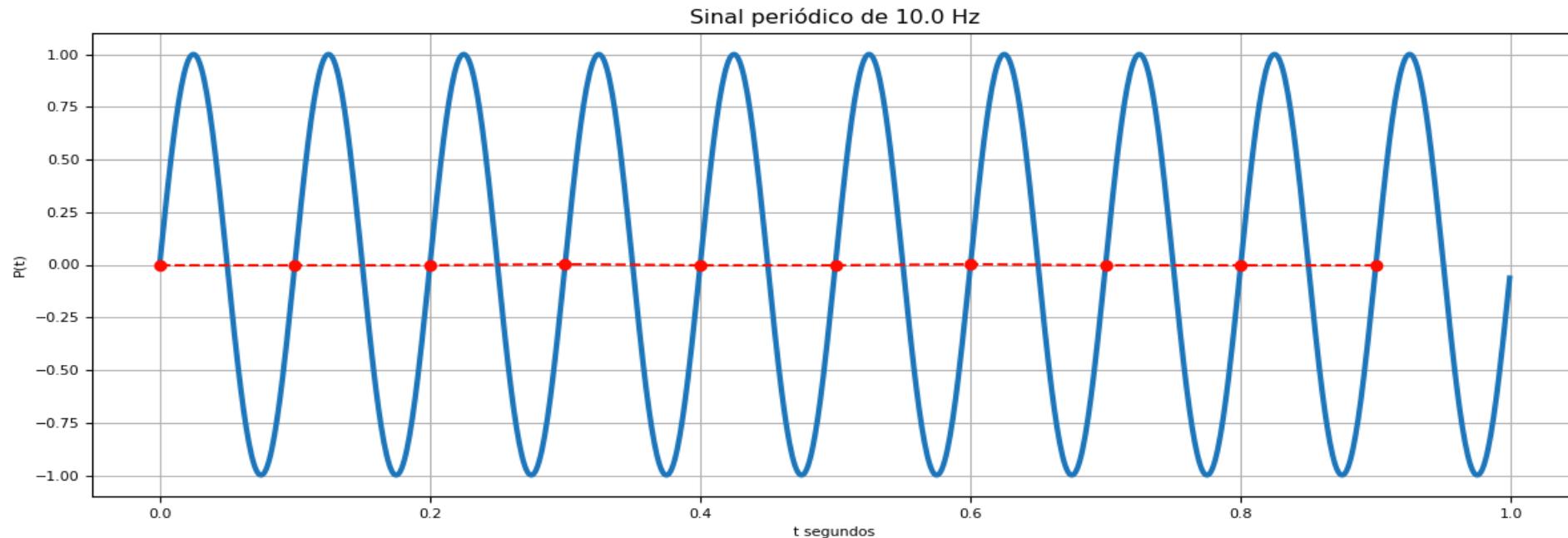
O teorema de amostragem de *Nyquist-Shannon* estabelece a condição que determina qual deve ser a menor frequência de amostragem que torna adequada a reconstrução do sinal:

$$f_s > 2 \times f_{max}$$

Por outras palavras, a frequência de amostragem deve ser maior que duas vezes a máxima frequência do sinal.

# Sinal e série amostral

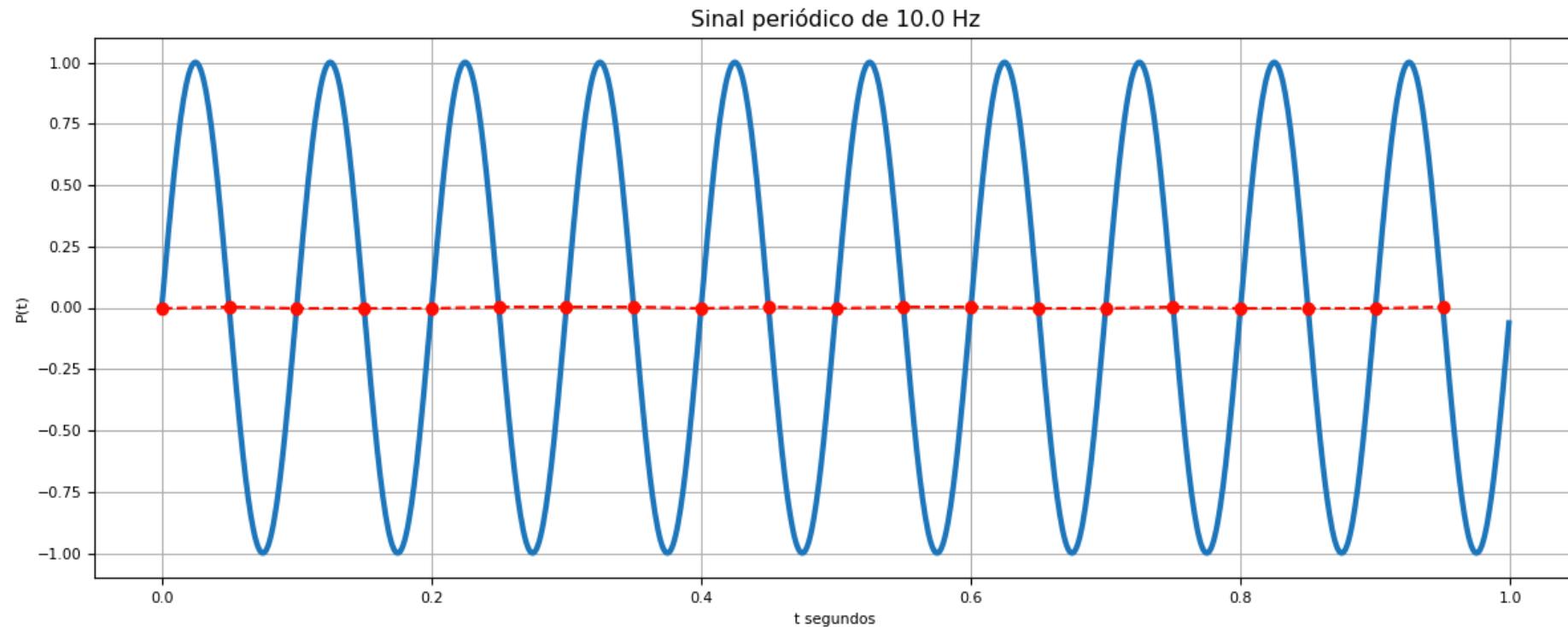
Considere-se, como exemplo, uma onda periódica P de um seno com  $f = 10$  Hz, ou seja, um sinal com uma frequência de dez ciclos por segundo (simplificada com  $A = 1$  e  $\phi = C = 0$ ). Uma série com  $f_s = 10$  Hz (que não respeita a condição referida anteriormente) tem uma amostra por período.



A ligação desses pontos resulta numa linha sem qualquer relação com a forma do sinal

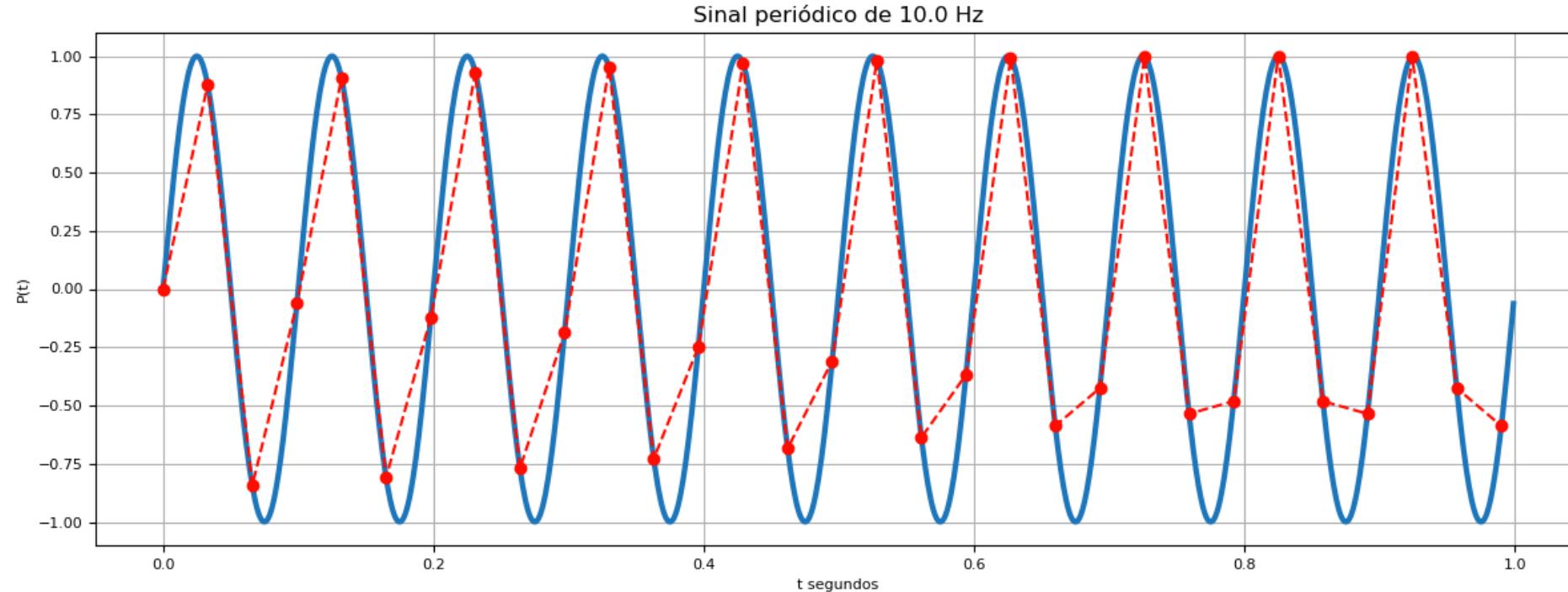
# Sinal e série amostral

Veja-se agora uma série com  $f_s = 2 \times 10 = 20$  Hz, que está no limite da condição de Nyquist (mas ainda sem a respeitar); neste caso têm-se duas amostras por período que, conectadas, formam uma linha semelhante à anterior.



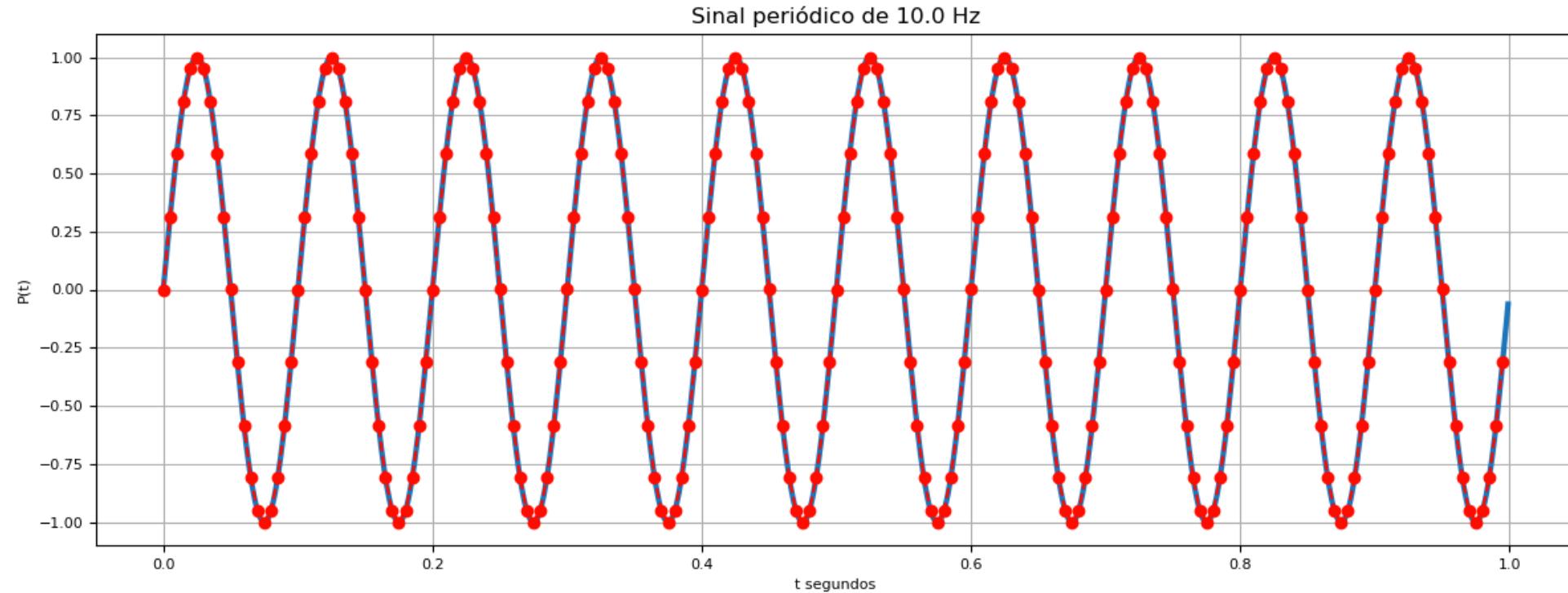
# Sinal e série amostral

Aumentando-se a frequência de amostragem para um valor que respeite a condição, por exemplo,  $f_s = 3 \times 10 = 30$  Hz, pode-se verificar que a série adquire um comportamento que possibilita já reconstituir o sinal de uma forma clara, podendo ser quase suficiente para os objetivos a cumprir.



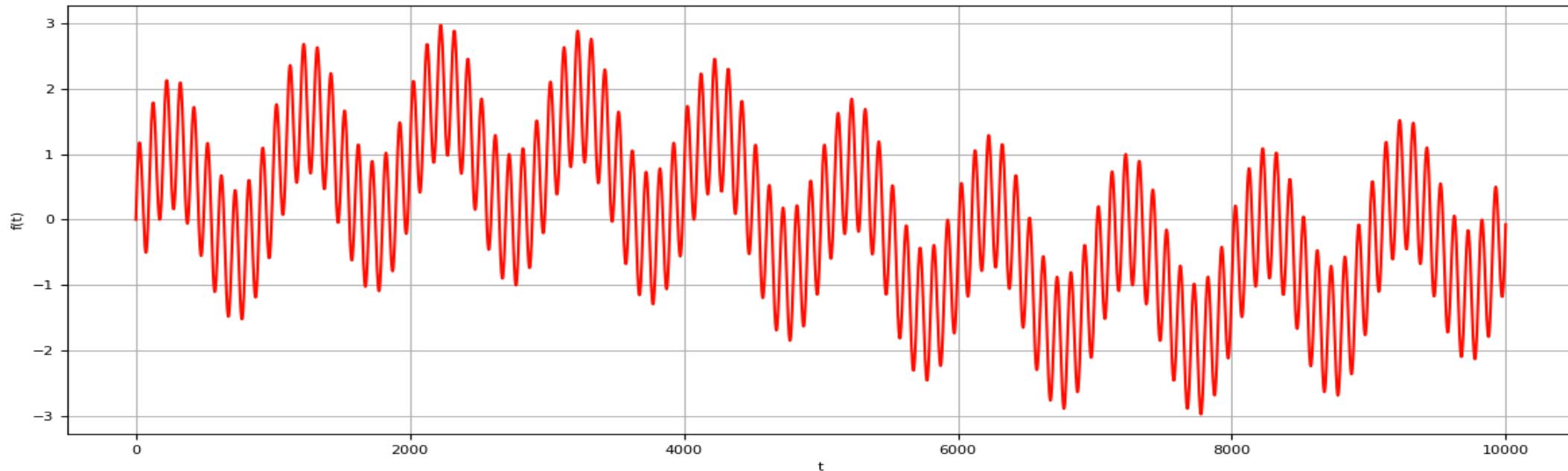
# Sinal e série amostral

Com um valor superior de, por exemplo,  $f_s = 20 \times 10 = 200$  Hz, tem-se uma série amostrada já bastante fiel ao sinal original.



Assim sendo é desnecessário amostrar um sinal com frequências de amostragem acima das que permitem reconstituí-lo sem ambiguidade.

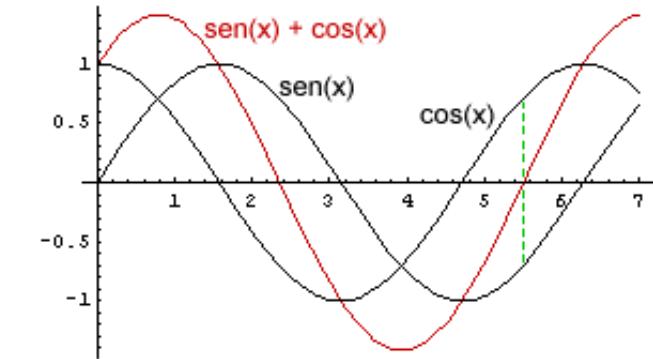
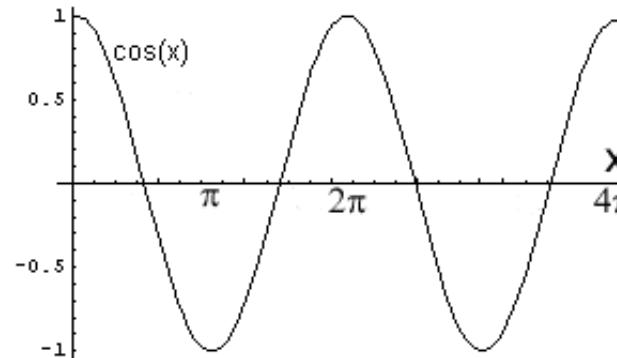
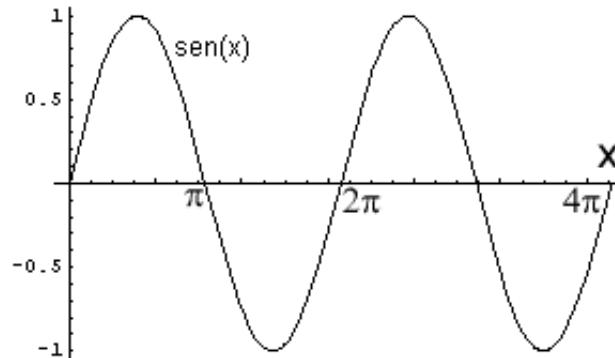
Considere-se um segundo exemplo sintético, produzido a partir da soma de três senoidais com diferentes frequências não conhecidas e que, apesar de igualmente periódica, tem um comportamento mais difícil de interpretar. O objetivo será o de saber as características das ondas elementares. Para tal recorre-se à análise espectral com a **transformada de Fourier**.



# Transformada de Fourier

A teoria de *Fourier* afirma que qualquer sinal pode ser expresso como uma soma de várias senoidais, ou seja, uma soma de funções seno e cosseno.

$$f(x) = a_0 + a_1 \sin(x) + a_2 \sin(2x) + a_3 \sin(3x) + \dots + b_1 \cos(x) + b_2 \cos(2x) + b_3 \cos(3x) + \dots$$



A transformada de *Fourier* (TF) permite realizar a decomposição de um sinal (definido no domínio espacial) nas suas componentes seno e cosseno, representando-o no chamado “domínio das frequências”, ou espaço de *Fourier*.

*Transformada Directa* (DTF):

$$F(u) = \sum_{x=0}^{N-1} \left[ f(x) \times e^{-j\frac{2\pi ux}{N}} \right] = \sum_{x=0}^{N-1} \left[ f(x) \times \left( \cos\left(-\frac{2\pi ux}{N}\right) + j \sin\left(-\frac{2\pi ux}{N}\right) \right) \right]$$

*Transformada Inversa* (IFT):

$$f(x) = \frac{1}{N} \times \sum_{u=0}^{N-1} \left[ F(u) \times e^{j\frac{2\pi ux}{N}} \right] = \frac{1}{N} \times \sum_{u=0}^{N-1} \left[ f(x) \times \left( \cos\left(\frac{2\pi ux}{N}\right) + j \sin\left(\frac{2\pi ux}{N}\right) \right) \right]$$

N = número de amostras que se tem

x = amostra corrente (0...N-1)

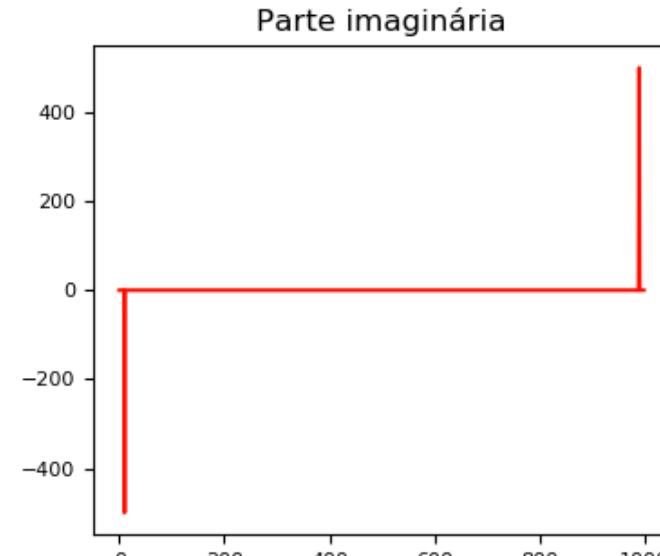
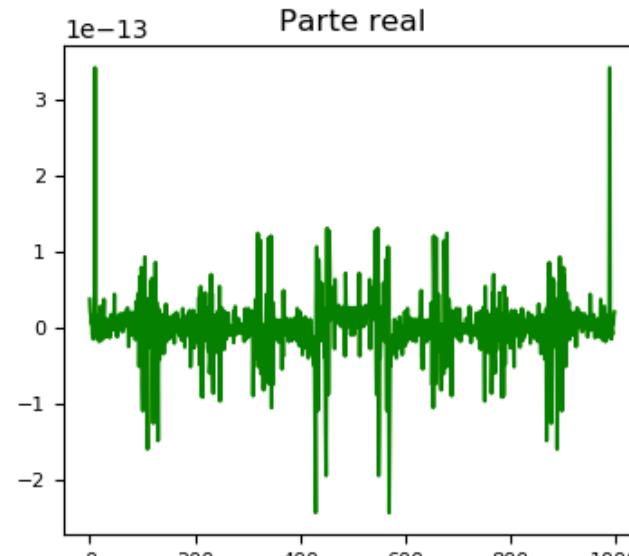
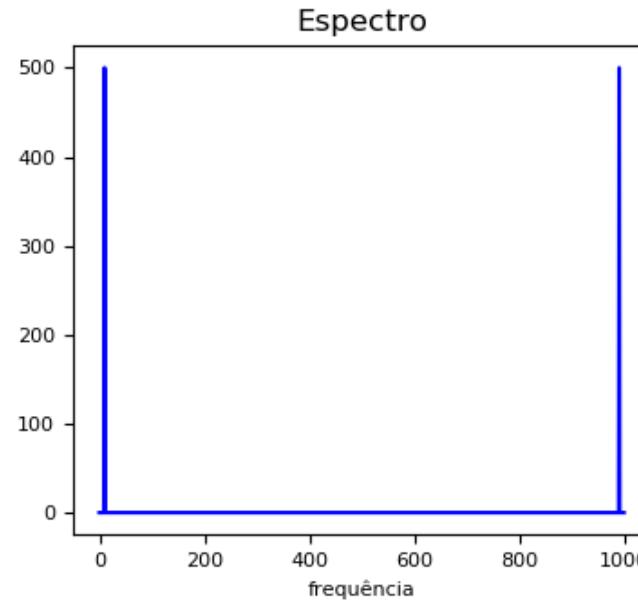
f(x) = valor do sinal em x

u = frequência corrente (0 Hertz até N-1 Hertz)

F(u) = quantidade de frequência u presente no sinal

# Transformada de Fourier 1D

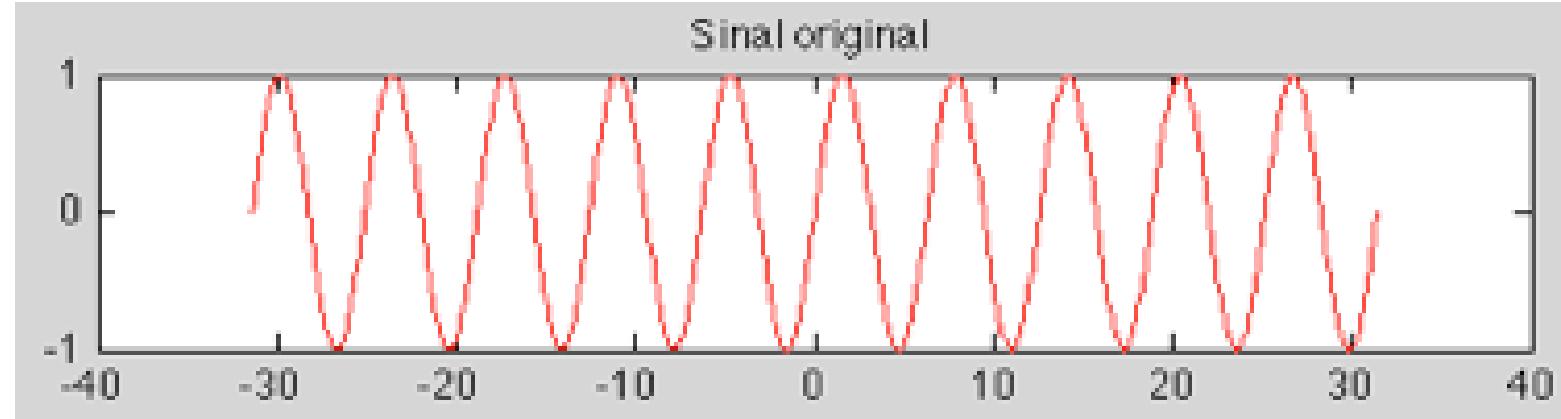
O espectro de frequências de uma função resulta da aplicação da DFT. Os valores resultantes desta transformação pertencem ao domínio dos números complexos ( $z = a+bj$ ), sendo constituídos por uma parte real ( $a$ ) e uma parte imaginária ( $b$ ). A magnitude do espectro é igual ao valor absoluto da DFT ( $|z|$ )



Magnitude

# Espectro de frequências

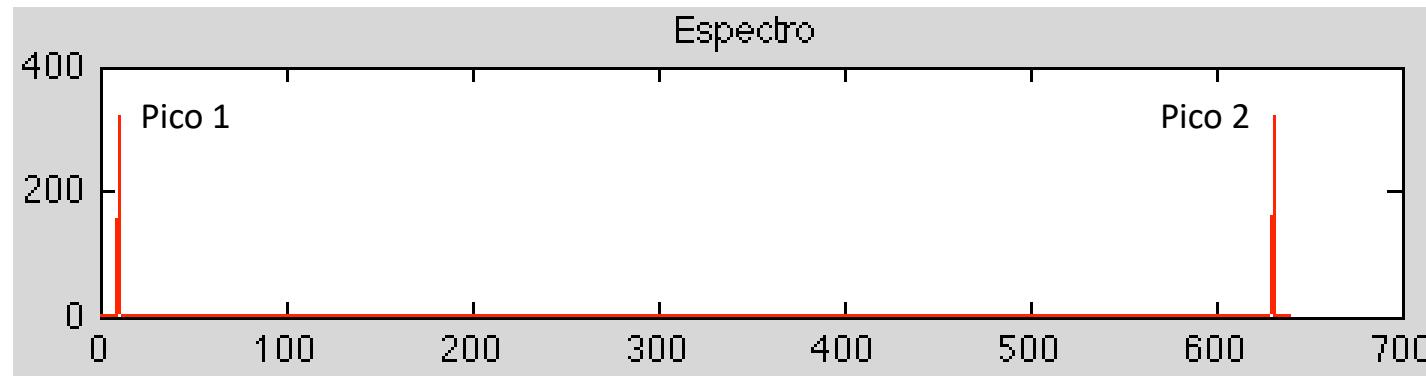
Considere-se o seguinte sinal periódico (seno).



O **espectro** de um sinal contém as magnitudes das frequências das ondas que o constituem. Neste exemplo, como o sinal representa um senoide (seno), há apenas uma frequência.

# Espectro de frequências

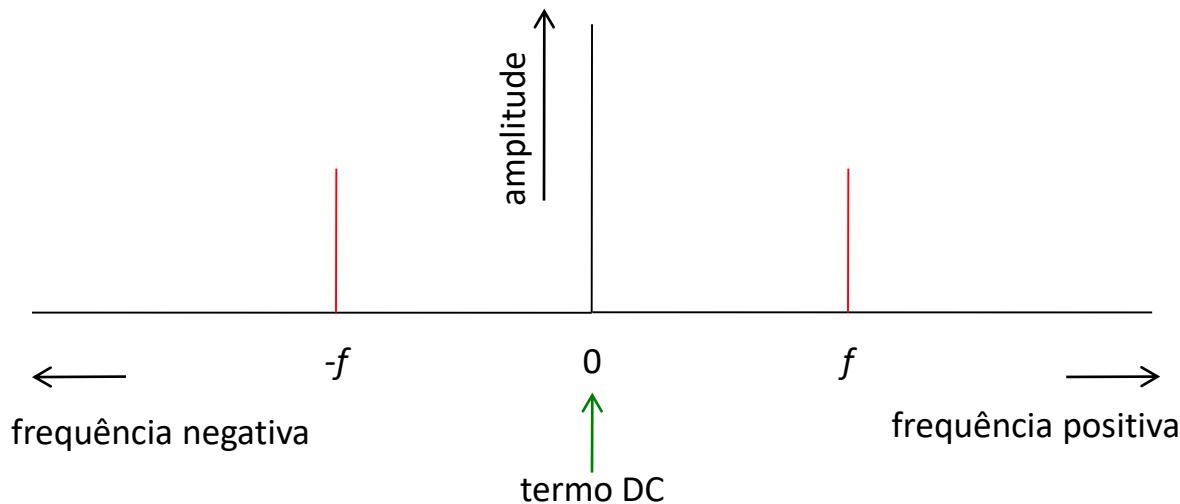
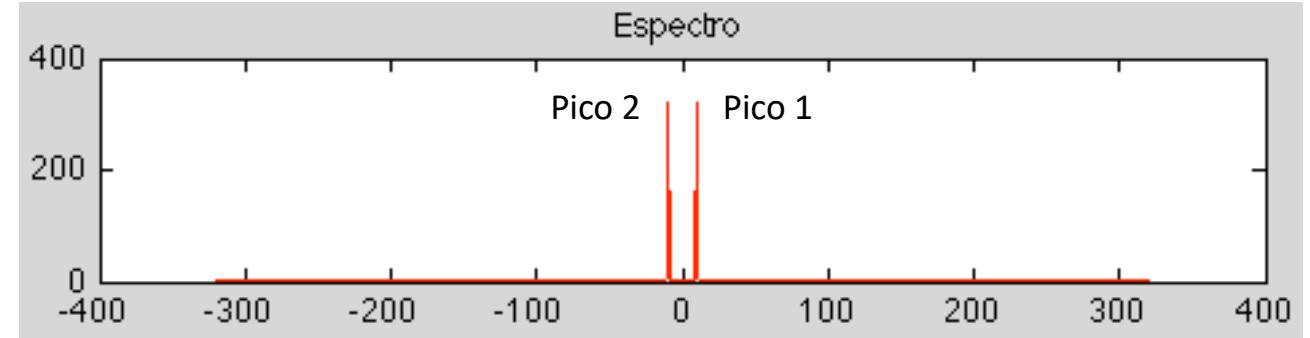
O espectro tem duas partes: uma parte negativa e uma parte positiva. Para sinais reais (sem parte imaginária), a parte negativa do espectro é sempre uma versão “espelhada” da parte positiva. Assim, neste exemplo, a parte positiva terá apenas um pico e a parte negativa terá um pico idêntico ao da parte positiva.



# Espectro de frequências

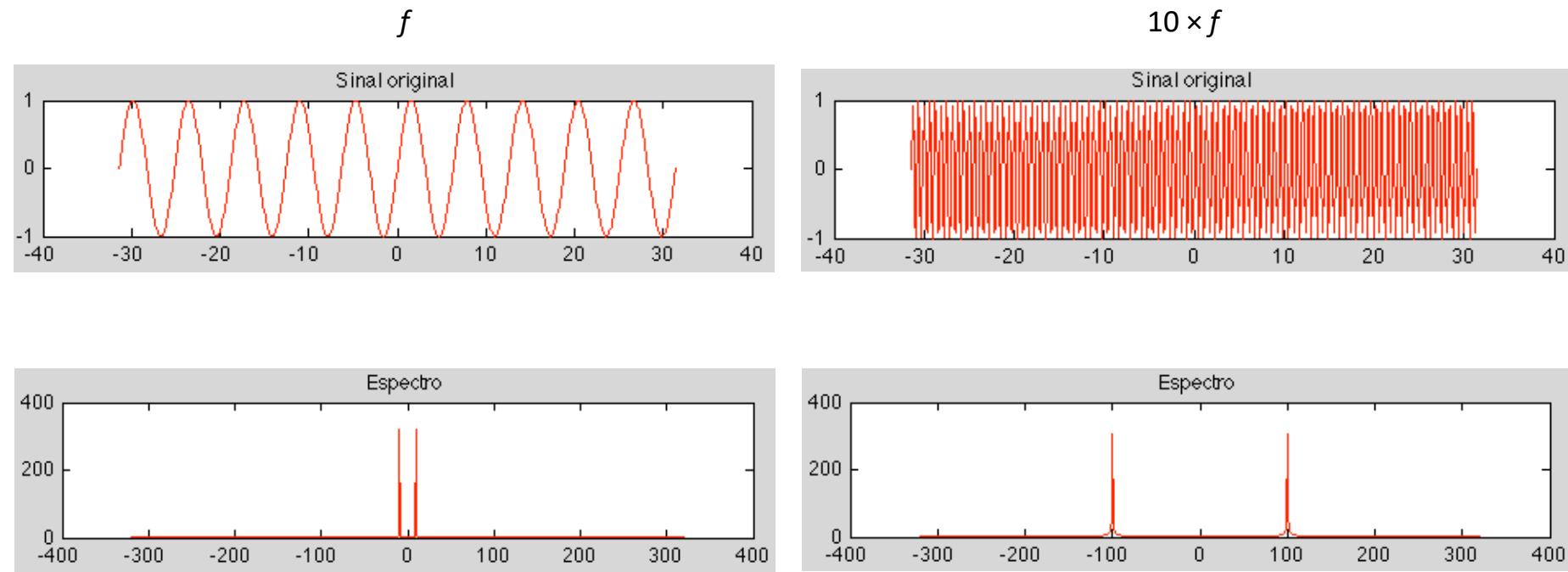
Fernando Soares

Representação centrada do espectro



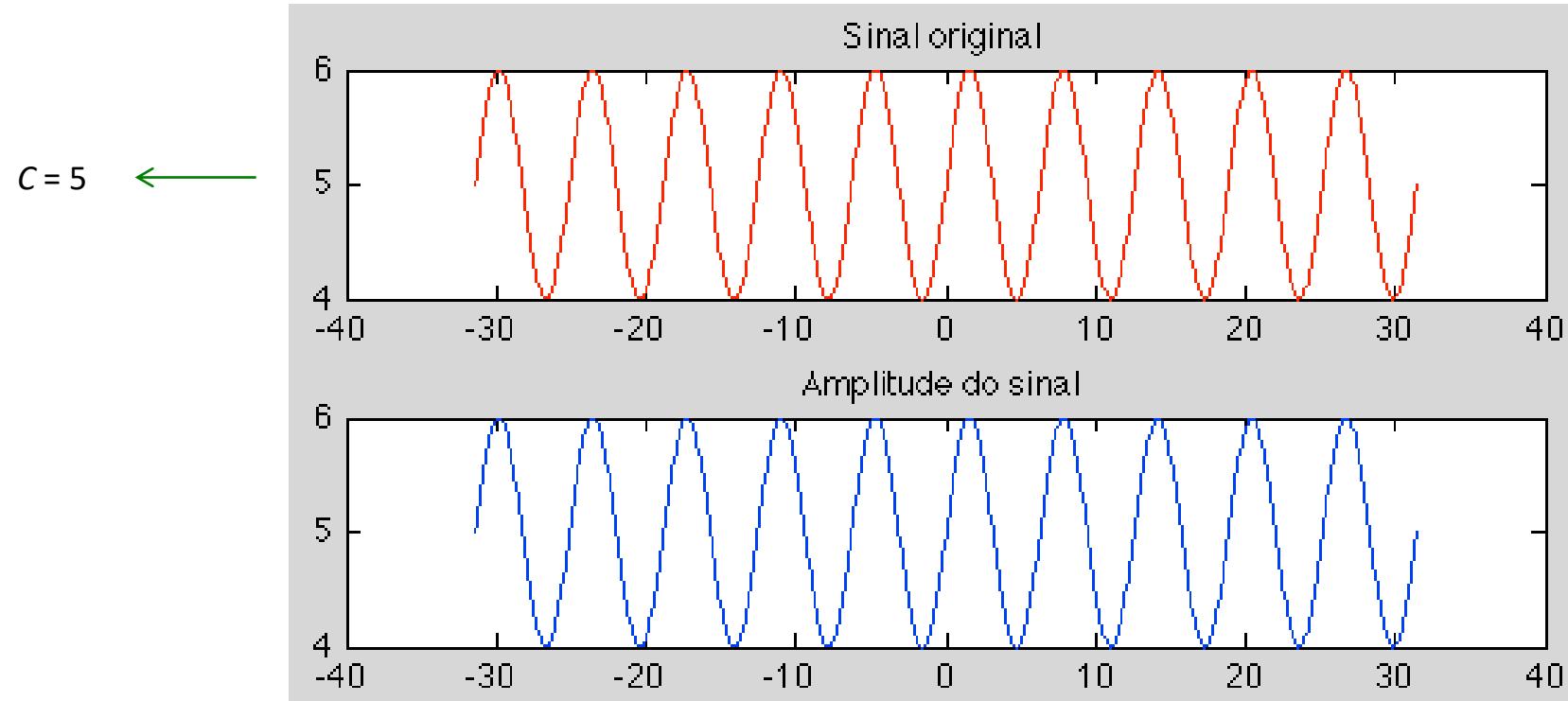
# Espectro de frequências

Quanto mais afastado para a esquerda (e para a direita) estiver um pico, maior é a frequência que o mesmo representa. Por outras palavras, um pico bastante afastado para a direita (e esquerda) significa que o sinal contém uma componente periódica de alta frequência.



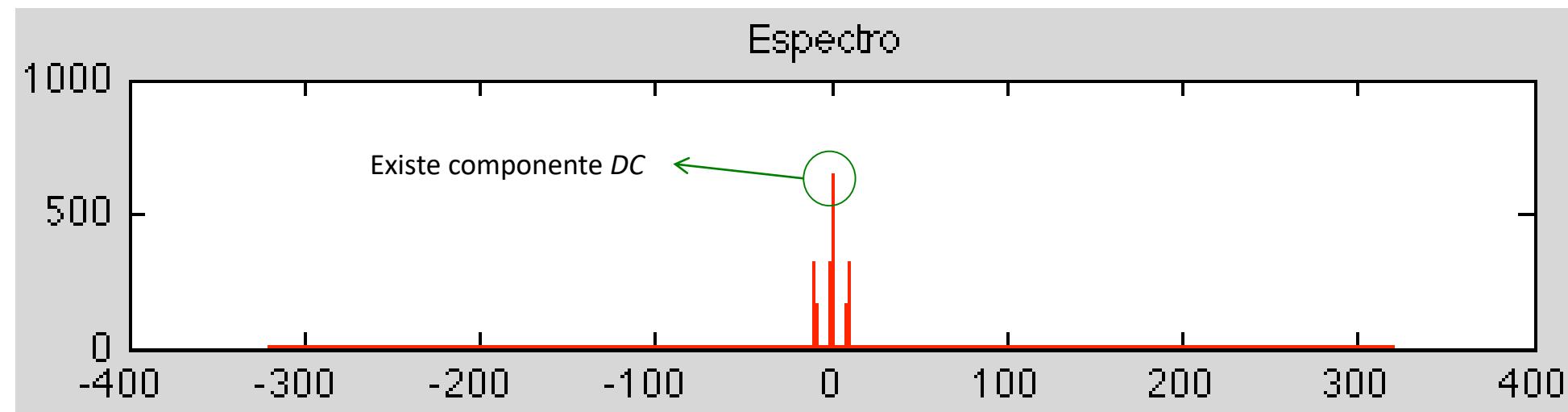
# Espectro de frequências

Considere-se um segundo exemplo de uma curva sinusoidal adicionada de um valor C:  $f(x) = \sin(x)+C$ . A acção desta soma é a de transladar o sinal segundo a direcção yy. O valor médio do sinal é igual ao valor de C (pois o valor médio de  $\sin(x) = 0$ ).



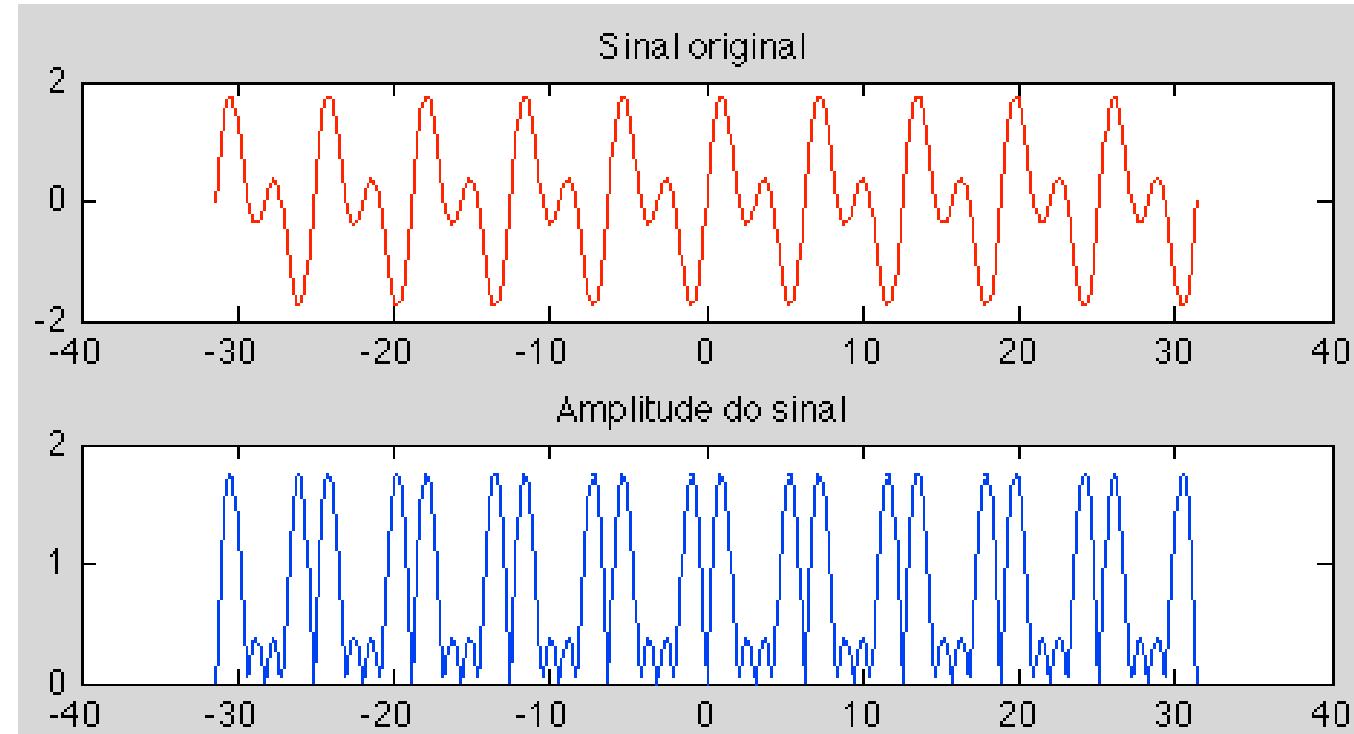
# Espectro de frequências

No espectro de frequências a frequência  $f=0$  corresponde à componente DC. Esta componente tem uma amplitude igual ao valor médio do sinal, para além dos picos de frequência das componentes sinusoidais. Assim, se o espectro de um sinal tiver um valor diferente de zero na origem, saber-se-á que o valor médio do sinal é diferente de zero.



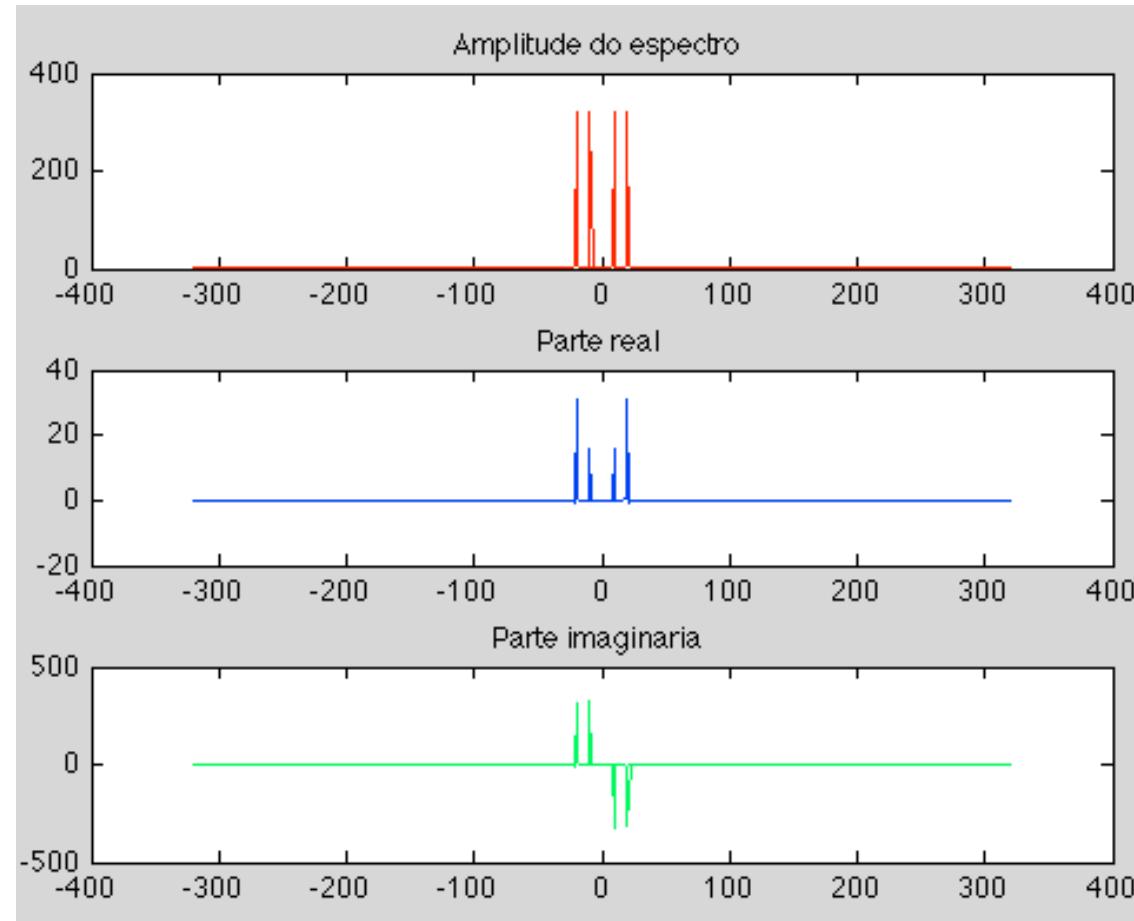
# Espectro de frequências

Considere-se um terceiro exemplo, como a soma de duas funções seno, em que a segunda tem uma frequência dupla da primeira:  $f(x) = \sin(kx)+\sin(2kx)$ . O sinal tem a seguinte forma:



# Espectro de frequências

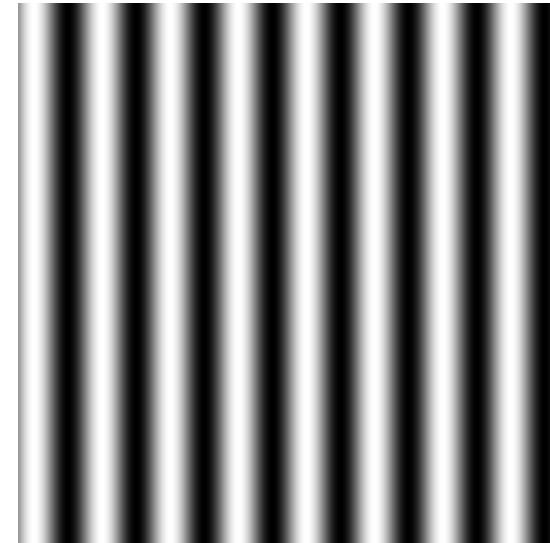
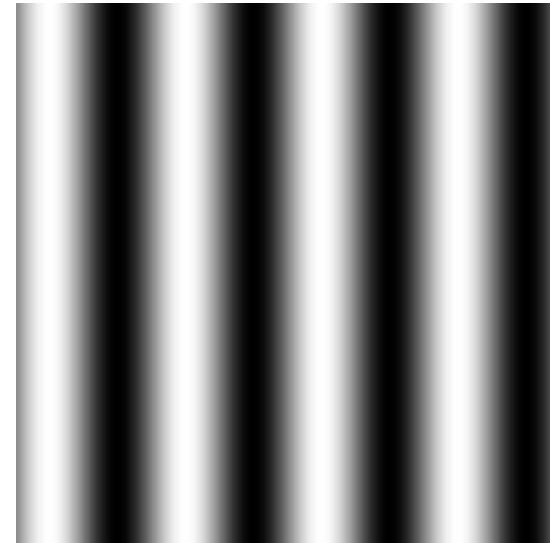
Como há duas funções seno com duas frequências diferentes, pode-se esperar dois picos no lado positivo do espectro (e dois no lado negativo).



# Transformada de Fourier 2D

Numa imagem as variações são representadas pelas variações espaciais dos tons de cinzento dos pixels.

A frequência espacial corresponde à frequência ao longo do espaço onde há modulação da intensidade (no caso das imagens abaixo, ao longo do eixo dos xx). A imagem da esquerda tem uma menor frequência espacial do que a da direita.



# Transformada de Fourier 2D

Fernando Soares

A transformação do domínio espacial para o domínio das frequências espaciais resulta numa função de valores complexos, ou seja, do tipo  $z = a+bj$ .

$$F(u) = re(F(u)) + im(F(u))j$$

*re*: parte real do número complexo

*im*: parte imaginária do número complexo

$$\text{Magnitude} = |F(u)| = \sqrt{(re(F(u)))^2 + (im(F(u)))^2}$$

$$\text{fase} = \text{atan}\left(\frac{im(F(u))}{re(F(u))}\right)$$

A visualização do espectro pode fazer-se com a determinação da função da Magnitude.

**Transformada Discreta de Fourier (2D):** Para uma imagem de dimensões M×N, a Transformada Directa Discreta de Fourier bidimensional é dada por:

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f(x, y) \times e^{-j2\pi(\frac{u \times x}{M} + \frac{v \times y}{N})}]$$

A função  $f(x,y)$  é a imagem no domínio espacial e o termo exponencial é a função-base que corresponde à representação de cada ponto  $F(u,v)$  do espaço de Fourier.

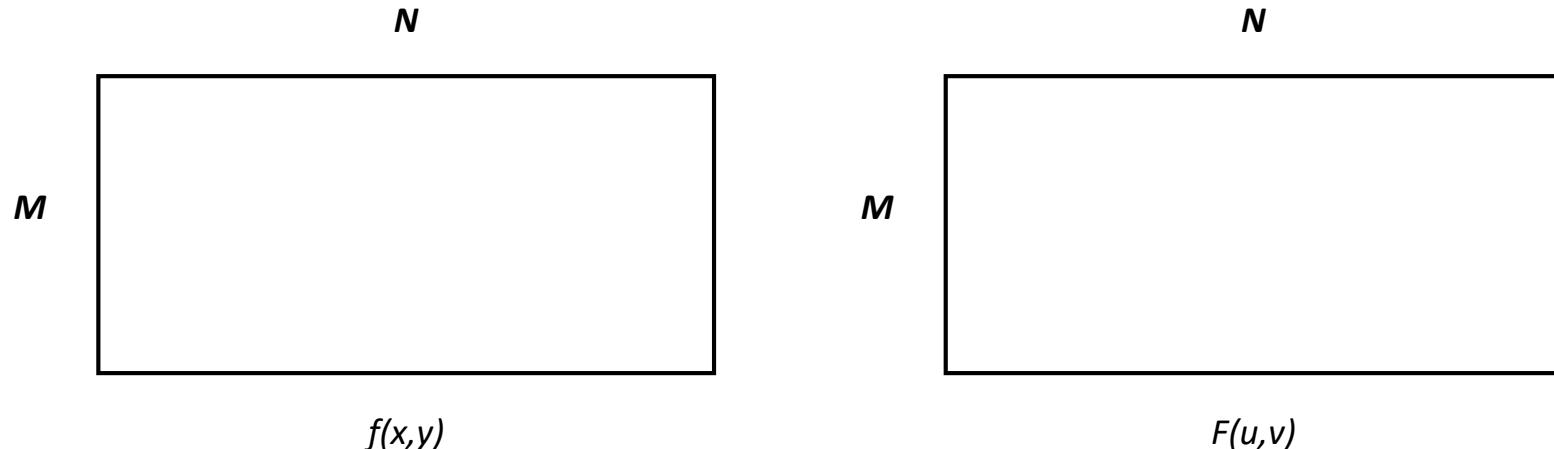
A magnitude do espectro calcula-se com a seguinte expressão:

$$Mag(u, v) = \frac{1}{(N \times M)} |F(u, v)|$$

# Transformada de Fourier 2D

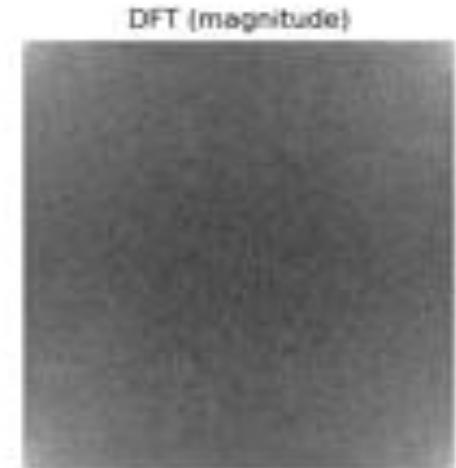
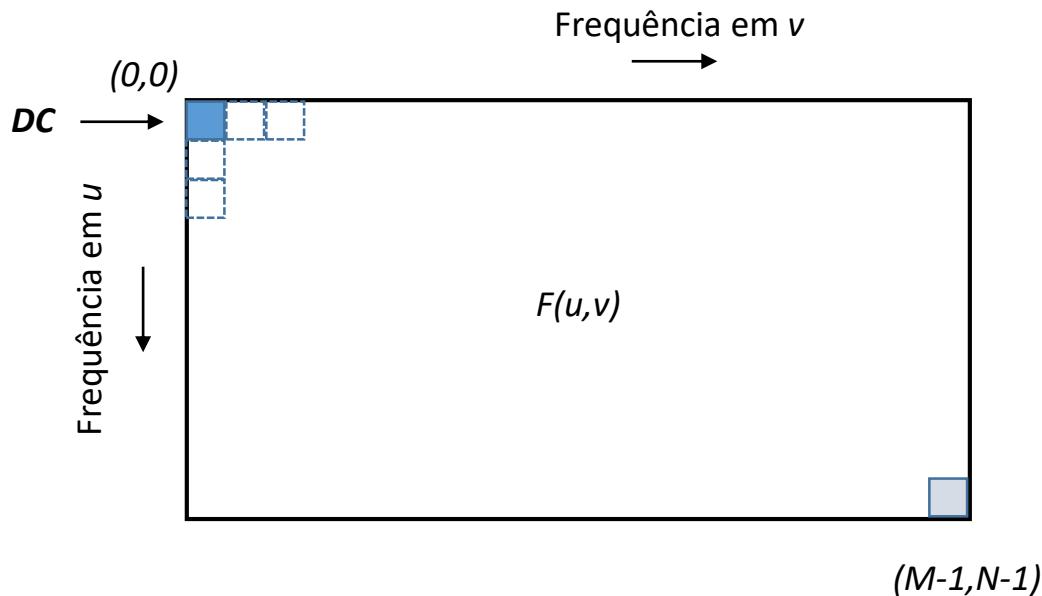
A Transformada Directa Discreta de *Fourier* (DDFT) corresponde à TF amostrada e, como tal, não contém todas as frequências que formam uma imagem, mas apenas um conjunto de amostras que é suficientemente grande para descrever o domínio espacial da imagem.

O número de frequências corresponde ao número de pixels da imagem do domínio espacial  $f(x,y)$ , ou seja, a imagem e o espectro têm as mesmas dimensões.



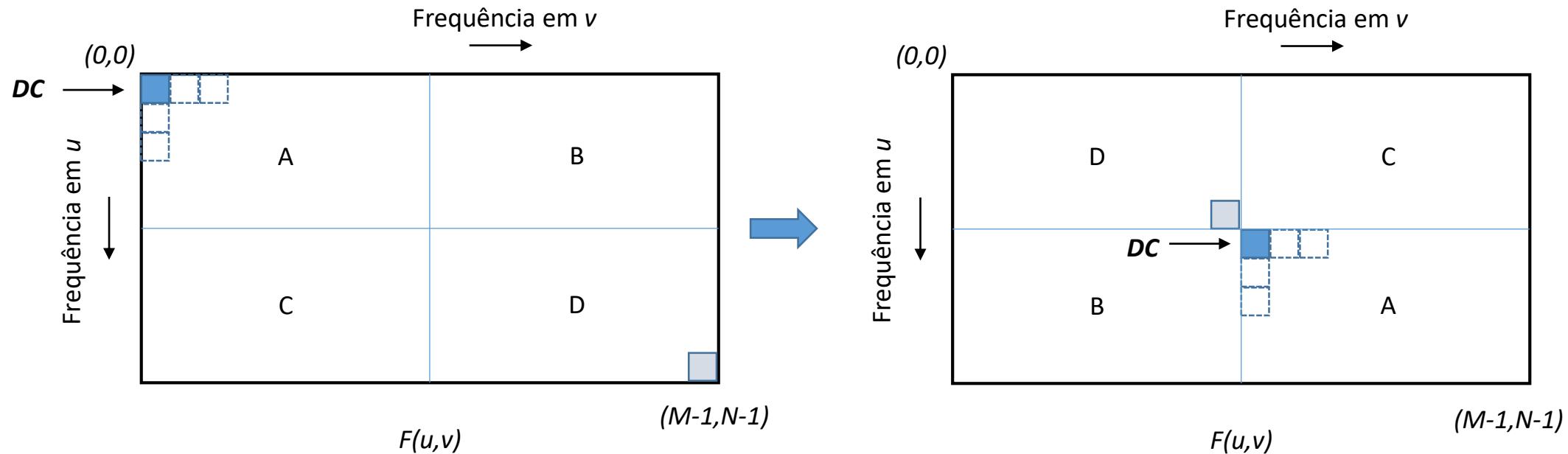
# Transformada de Fourier 2D

As funções de base são ondas seno e co-seno com frequências progressivas, isto, é,  $F(0, 0)$  representa a componente DC da imagem (que tem frequência zero), correspondente à intensidade média, e  $F(M-1, N-1)$  representa a frequência mais elevada em xx e em yy.



# Transformada de Fourier 2D

Por motivos de facilidade de interpretação/análise do espectro, a sua disposição é geometricamente alterada por forma a que o valor de  $F(0, 0)$  se localize ao centro da janela.

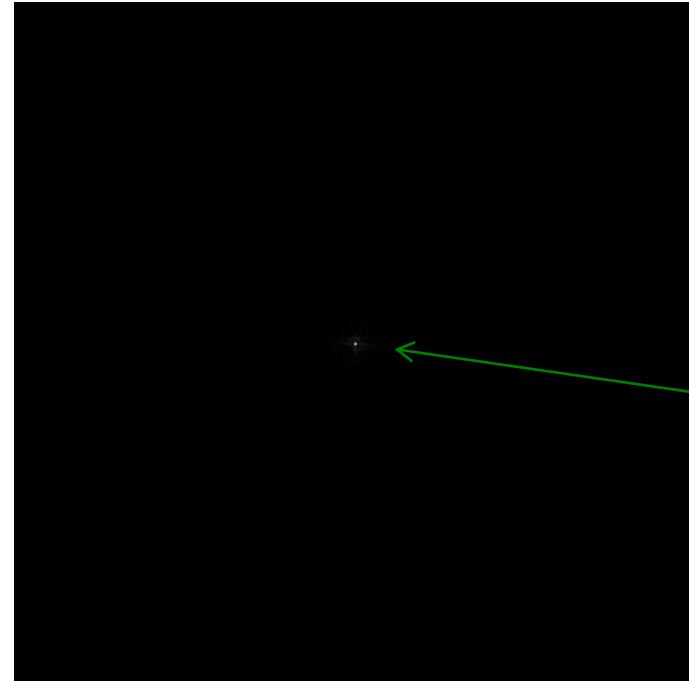


# Transformada de Fourier 2D

Quase sempre o valor de DC é, de longe, a maior componente do espectro de frequências. O intervalo numérico dos valores do espectro é bastante grande para ser visualizado no ecrã, o que faz com que a sua representação imediata não seja frequentemente viável.



Imagen inicial



Magnitude do espectro

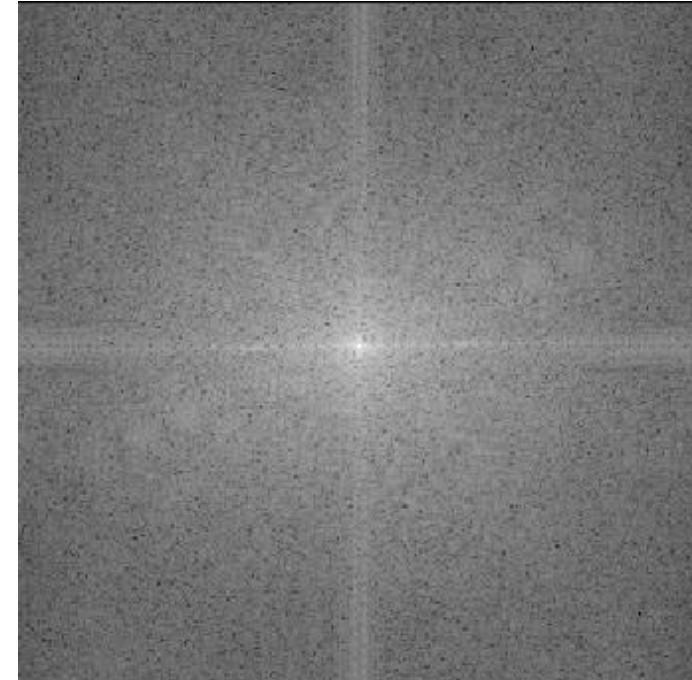
Min = 1.4514; Max = 6955093

# Transformada de Fourier 2D

Aplicando a operador logarítmico, à função da Magnitude, para uma representação de 8 bits, obtém-se uma representação apropriada do espectro das frequências (como alternativa pode-se, por vezes, dividir a matriz da magnitude pelo número total de pixels).



Imagen inicial



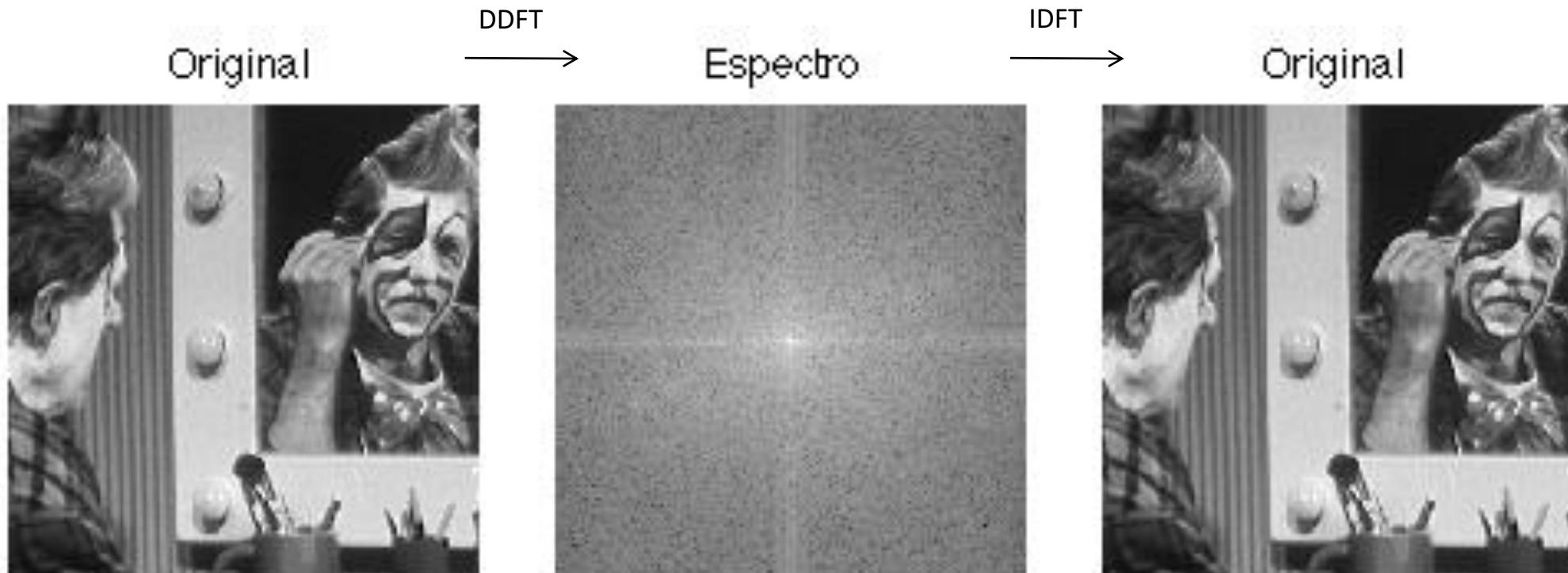
Magnitude do espectro centrado

Min = 0; Max = 255

# Transformada de Fourier 2D

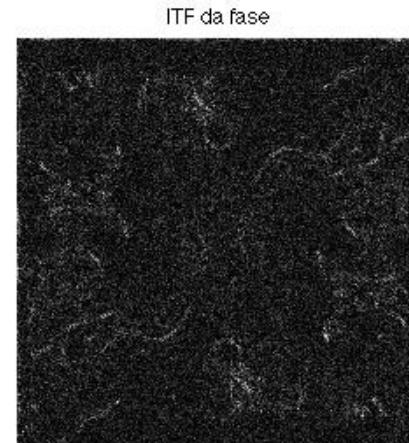
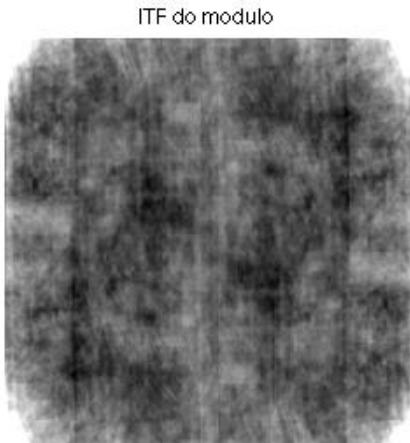
A Transformada Inversa Discreta de Fourier (IDFT) bidimensional é dada por:

$$f(x, y) = \left( \frac{1}{M \times N} \right) \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} [F(u, v) \times e^{j2\pi(\frac{u \times x}{M} + \frac{v \times y}{N})}]$$



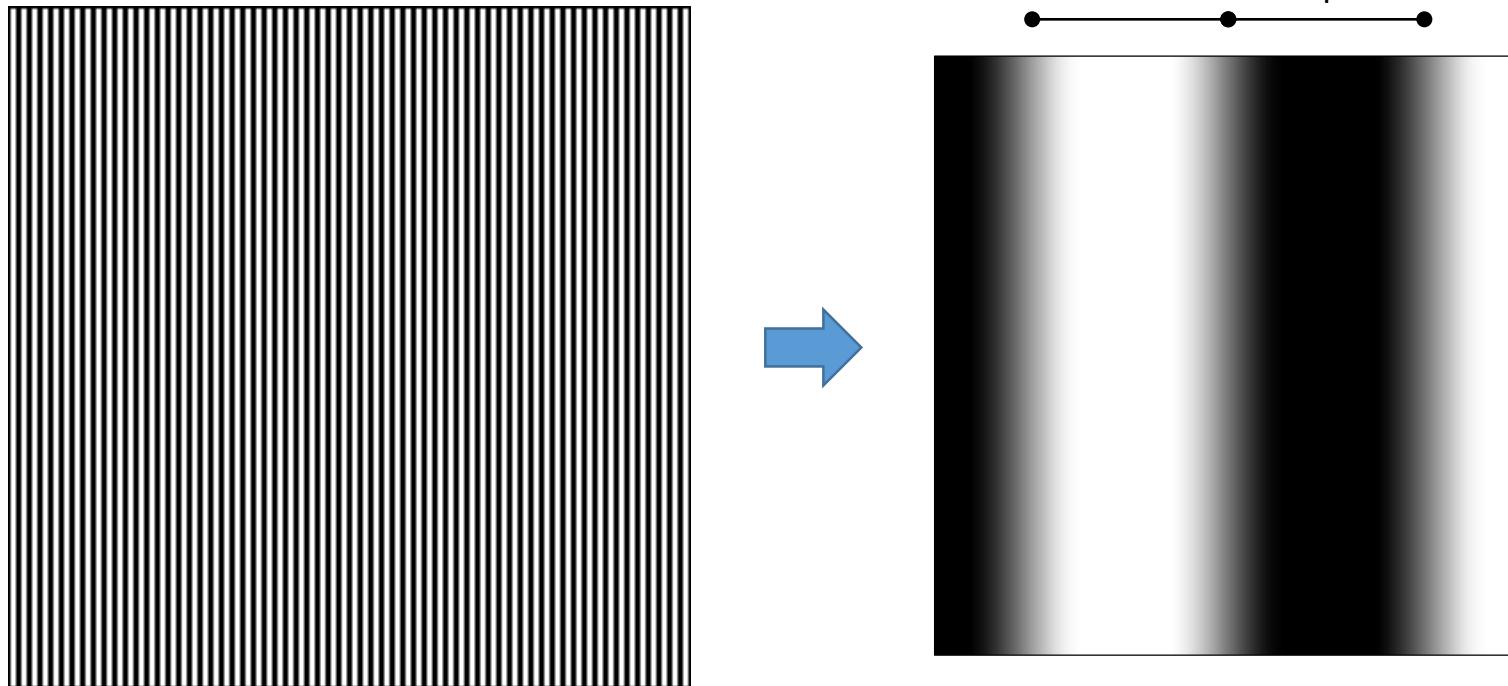
# Transformada de Fourier 2D

A IDFT necessita das partes real e imaginária obtidas com a DDFT.



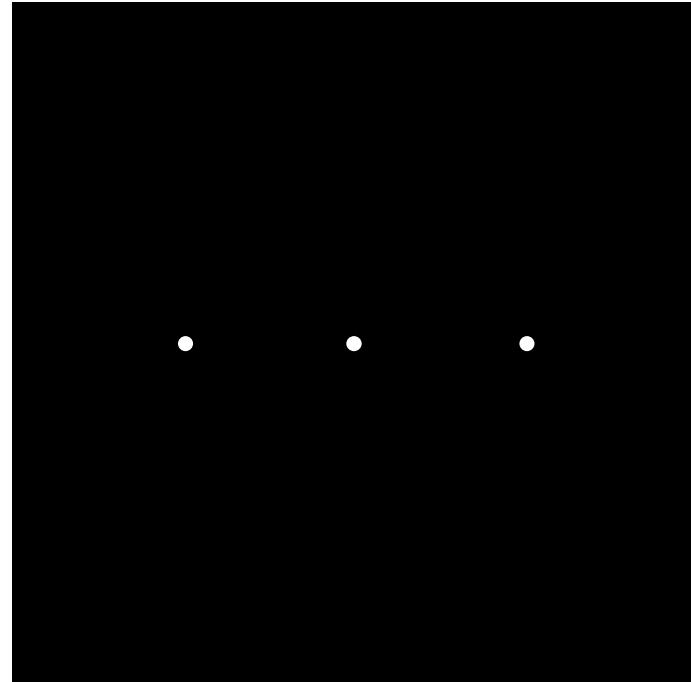
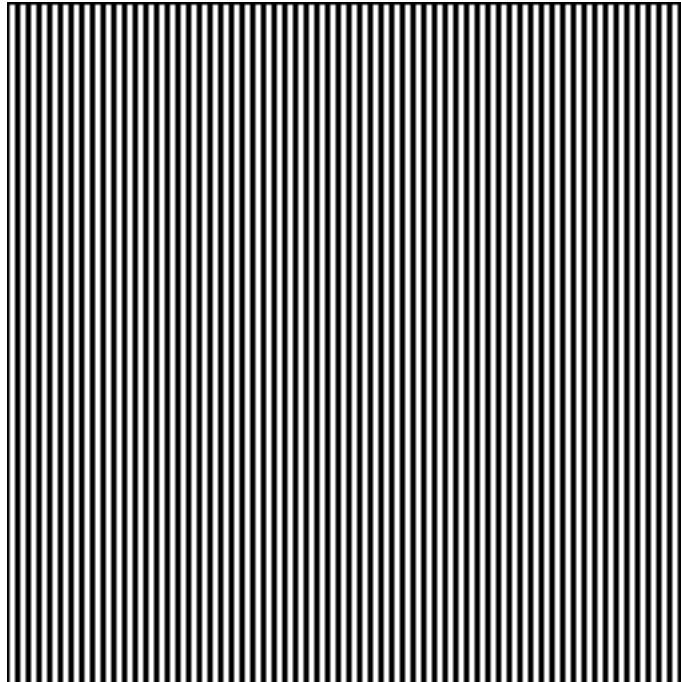
# Transformada de Fourier 2D

Interpretação: como exemplo considere-se a imagem abaixo, em que cada faixa vertical (branca ou preta) tem uma espessura de 2 pixels.



# Transformada de Fourier 2D

O respectivo espectro (à direita) tem três frequências: o valor de DC ao centro (frequência 0) e os dois picos correspondentes à frequência das linhas na imagem original. Como, no domínio espacial, a função se altera segundo a direcção horizontal, os pontos do espectro apresentam-se alinhados ao longo de uma recta horizontal que passa pelo centro.



# Transformada de Fourier 2D

Fernando Soares

A distância dos pontos ao centro pode ser explicada da seguinte forma: a frequência máxima ( $k_{\max}$ ), que pode ser representada na imagem original, corresponde a 2 pixels de distância (um preto e um branco):

$$k_{\max} = \frac{1}{2}$$

Sendo o sinal constituído por faixas verticais com espessura de 2 pixels (ciclo de 4 pixels) tem-se:

$$k = \frac{1}{4} = \frac{k_{\max}}{2}$$

Portanto, os pontos do espectro situam-se a meio caminho entre o centro e os limites da imagem, ou seja, a frequência representada é igual a metade da frequência máxima.

# Transformada de Fourier 2D

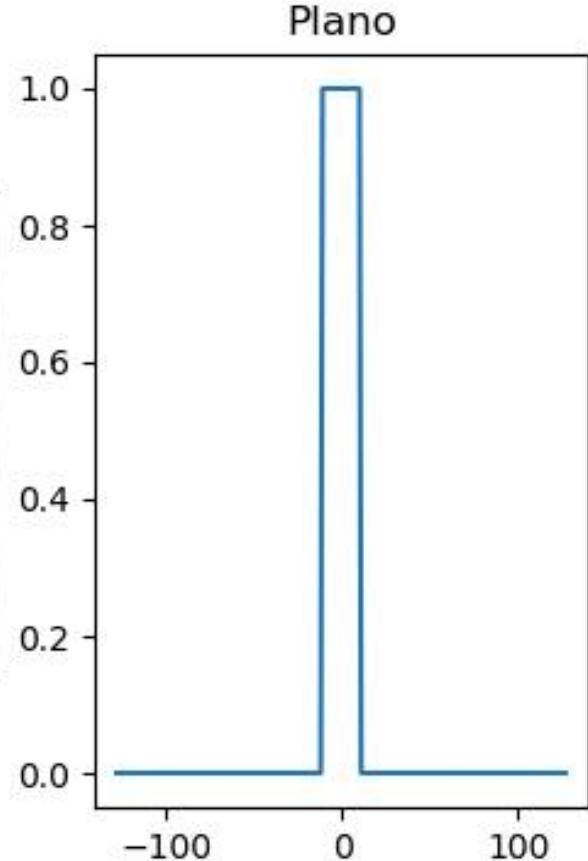
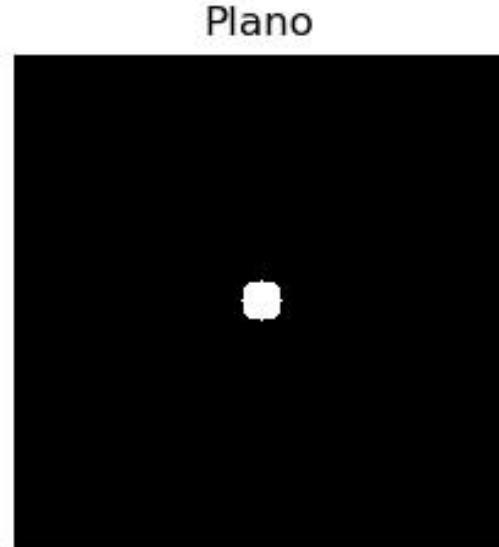
A TF está directamente relacionada com muitas operações, entre as quais as operações de filtragem.

Uma operação de filtragem, que no domínio espacial resulta de uma operação de convolução entre um filtro (kernel) e a imagem de cinzentos, executa-se com uma simples multiplicação no domínio das frequências (TF da imagem inicial).

Assim podem-se definir filtros passa-baixa, passa-alta e passa-banda, eliminando frequências do espectro de forma seleccionada.

# Filtragem no domínio das frequências

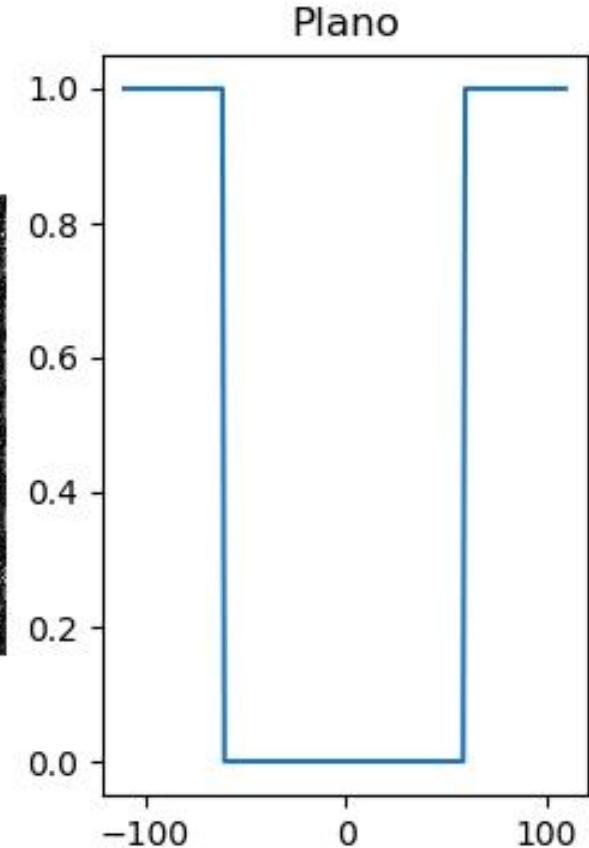
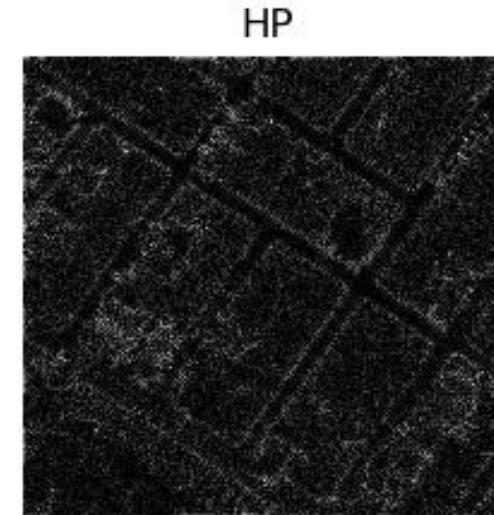
Filtro passa-baixa plano:



# Filtragem no domínio das frequências

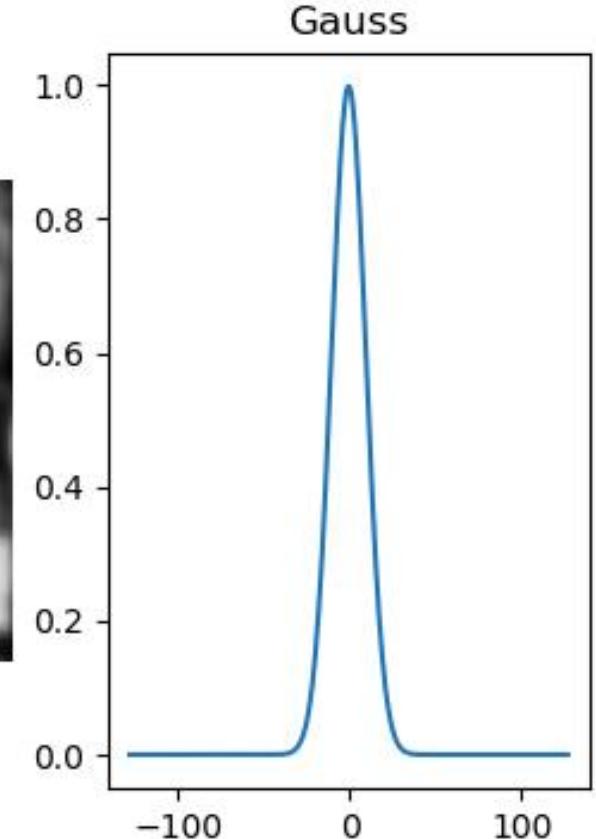
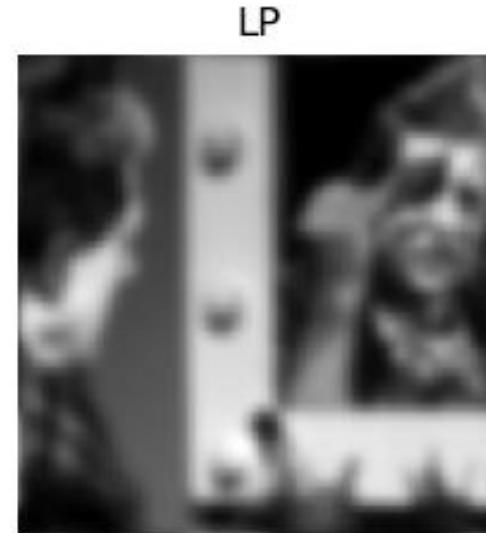
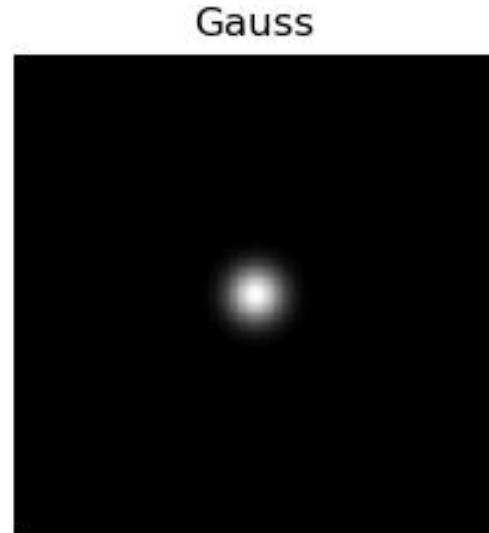
Fernando Soares

Filtro passa-alta plano:



# Filtragem no domínio das frequências

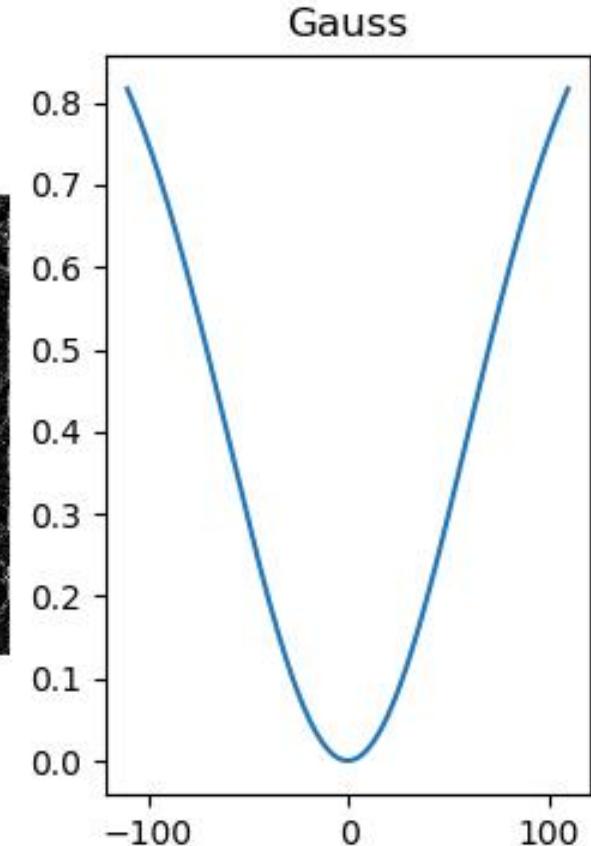
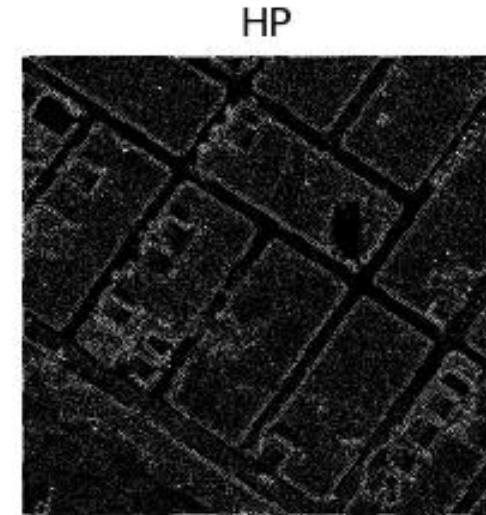
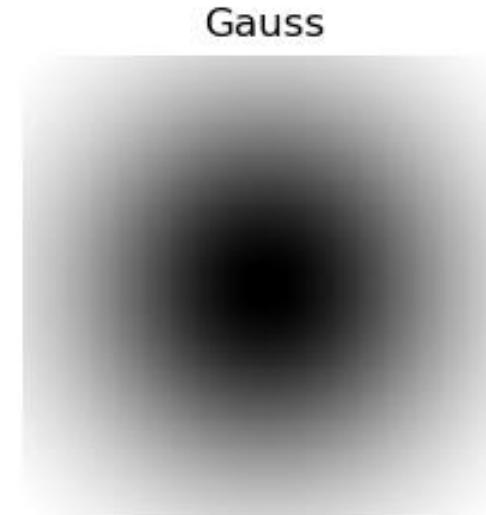
Filtro passa-baixa gaussiano:



# Filtragem no domínio das frequências

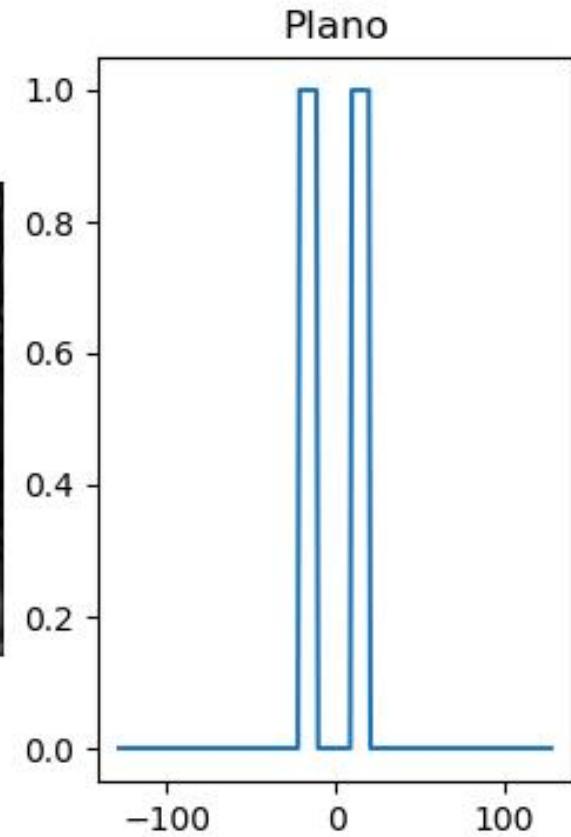
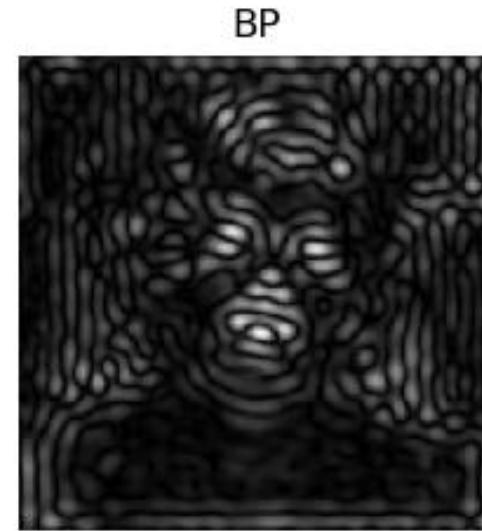
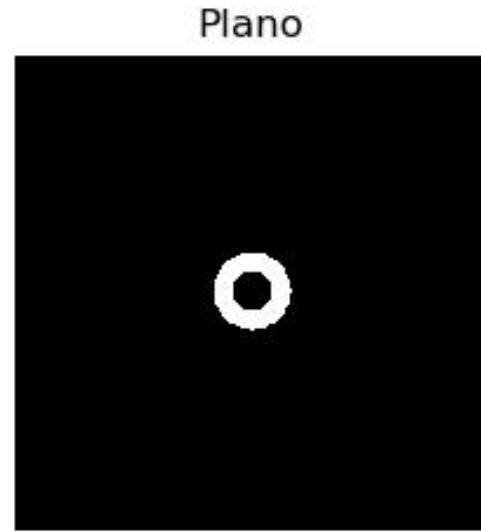
Fernando Soares

Filtro passa-alta gaussiano:



# Filtragem no domínio das frequências

Filtro passa-banda plano:



# Filtragem no domínio das frequências

O filtro **Butterworth** de ordem n e frequência de corte  $D_0$  é definido como:

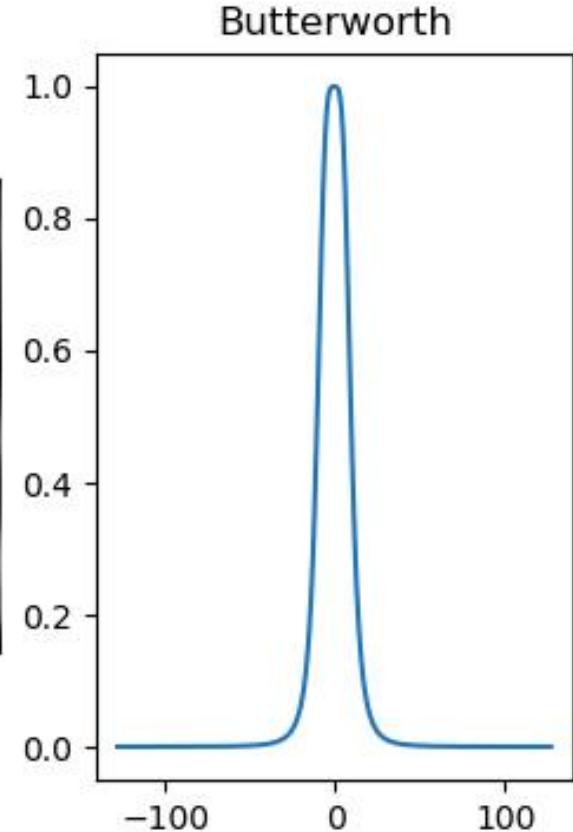
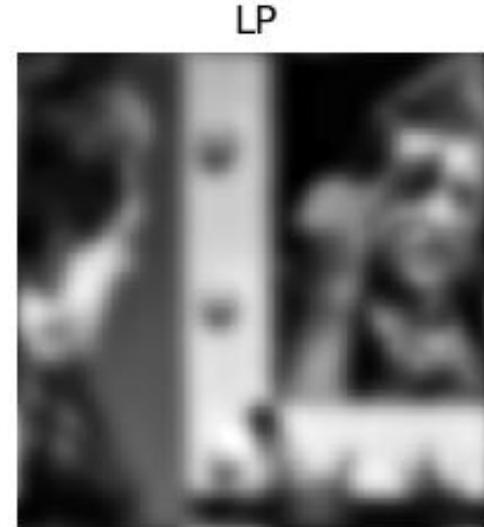
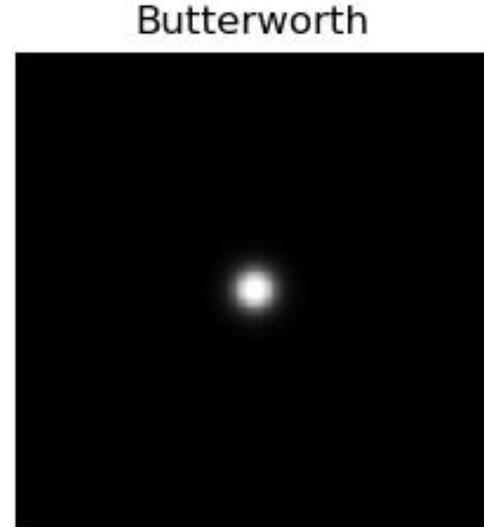
$$B(u, v) = \frac{1}{\left[ 1 + \left( \frac{\sqrt{u^2 + v^2}}{D_0} \right)^{2n} \right]}$$

Este filtro tem a vantagem de se poder controlar a nitidez com a ordem (n).

Um filtro passa-baixa *Butterworth* mantém frequências dentro do raio  $D_0$  e descarta as de fora. Introduz ainda uma transição gradual de 1 para 0 para reduzir o efeito de *ringing* que se verifica com o filtro plano.

# Filtragem no domínio das frequências

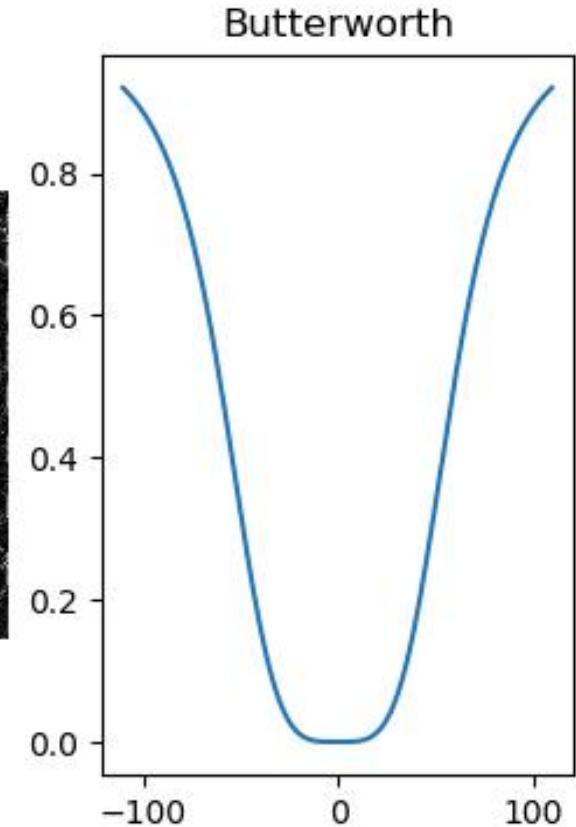
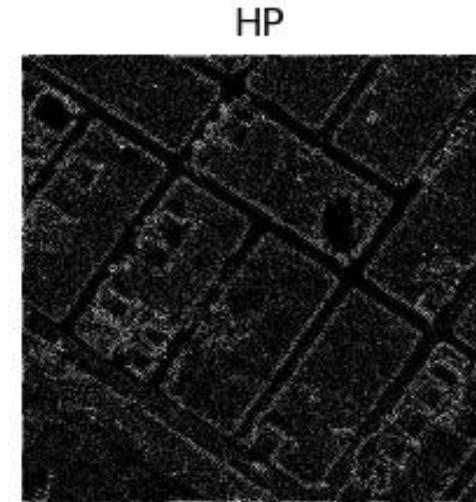
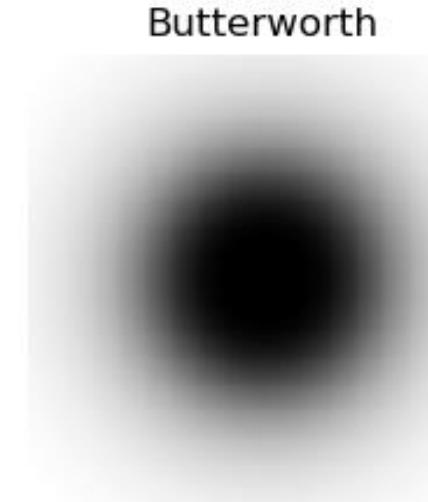
Filtro passa-baixa *Butterworth*:



# Filtragem no domínio das frequências

Fernando Soares

## Filtro passa-alta *Butterworth*



# Teorema da convolução

Fernando Soares

Em matemática, o **teorema da convolução** afirma que, em condições adequadas, a TF de uma convolução de dois sinais obtém-se multiplicando as suas transformadas individuais de *Fourier* e executando a inversa da TF desse produto.

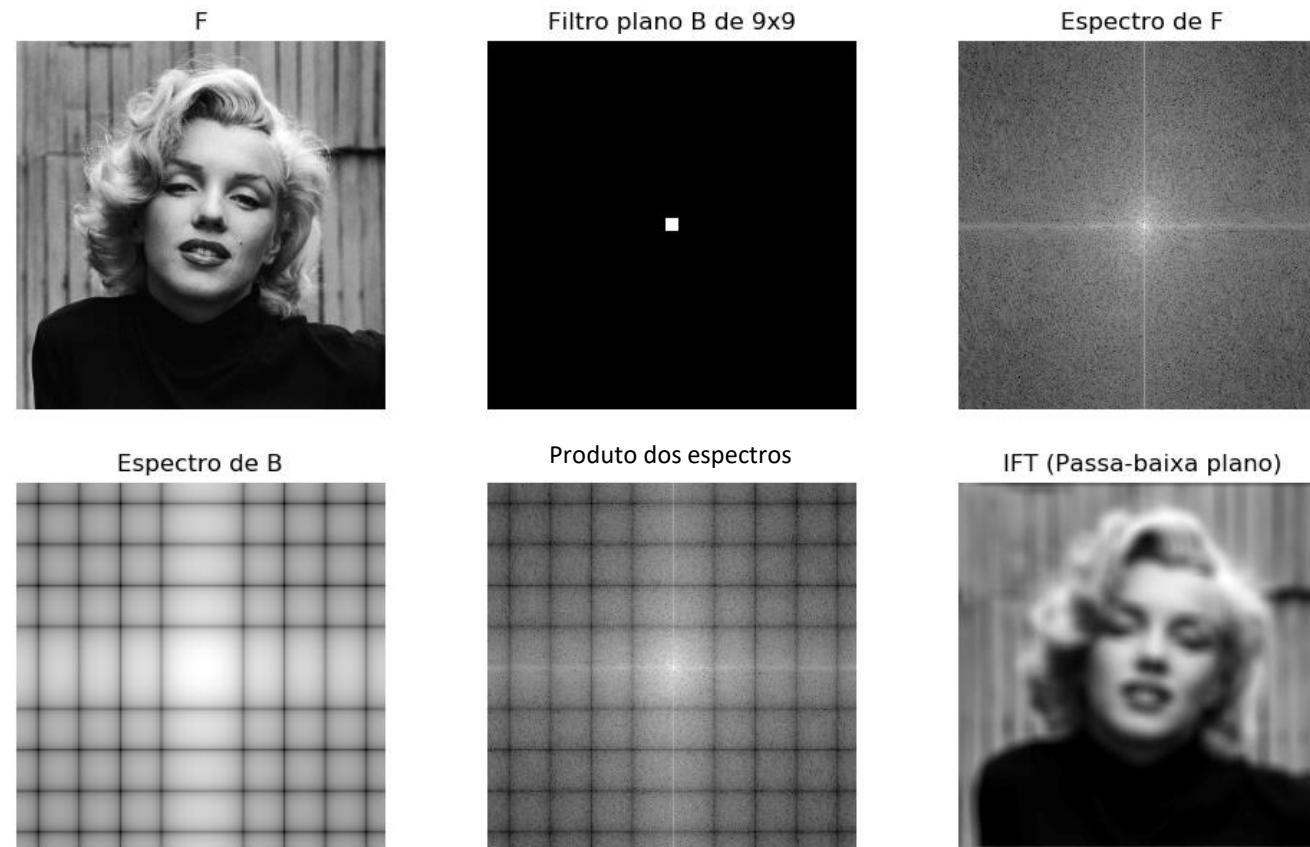
$$f(x, y) * h(x, y) = idft(F(u, v) \times H(u, v))$$

Este teorema é importante porque estabelece a ligação entre operações no domínio das frequências e a acção de filtros espaciais lineares.

As operações de filtragem linear, feitas por convolução no domínio espacial, podem ser realizadas por multiplicações simples no domínio de *Fourier*, tornando mais rápido o processo filtragem.

# Teorema da convolução

Multiplica-se matriz do filtro pretendido pela imagem resultante da DFT (espectro com ambas as partes real e imaginária) e aplica-se de seguida a IFT. Na figura pode-se ver a ilustração do processo de filtragem de uma imagem, por um filtro rectangular da média aritmética.

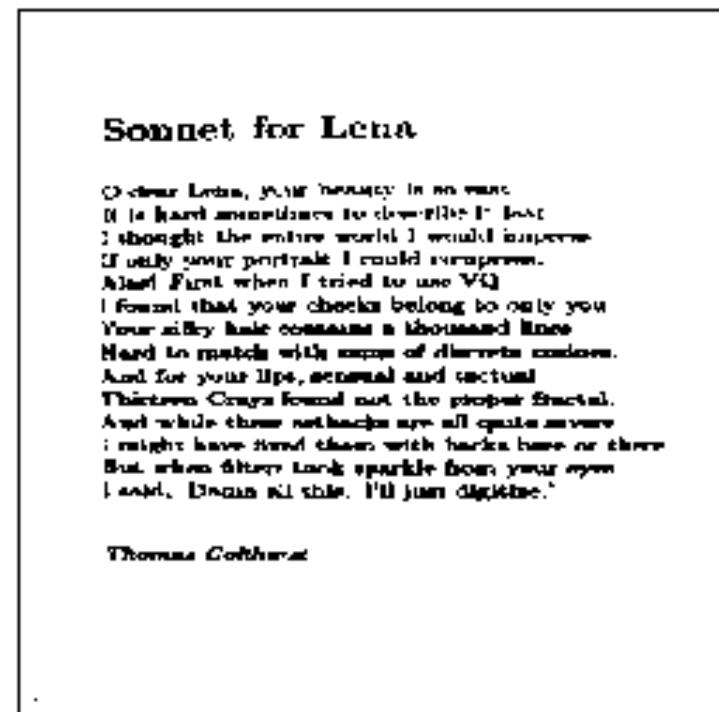


# Exemplos de aplicação

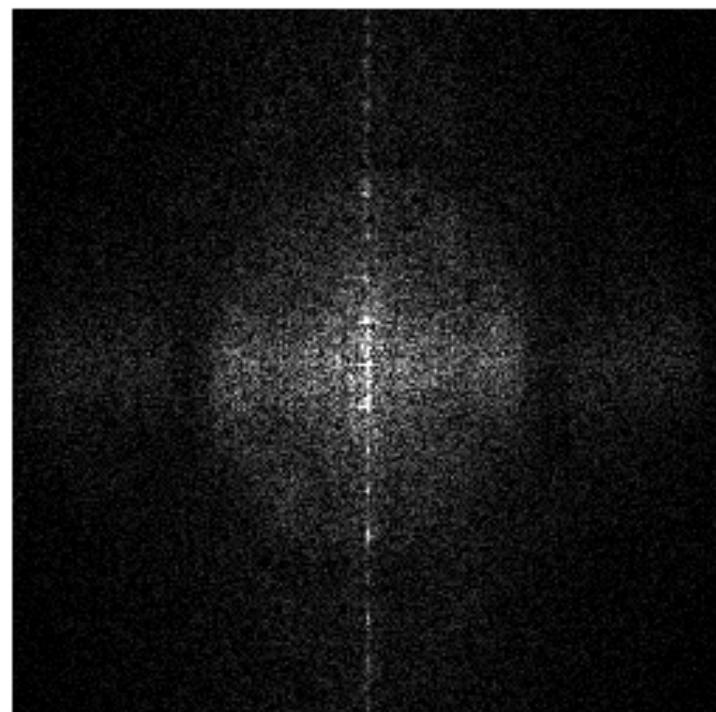
Fernando Soares

Detecção de orientação de texto: A TF é usada para adquirir informação acerca da estrutura geométrica do domínio espacial de uma imagem. O reconhecimento de texto, usando técnicas de processamento de imagem fica simplificado se for assumido que as linhas de um dado texto estão dispostas numa dada direção.

Original



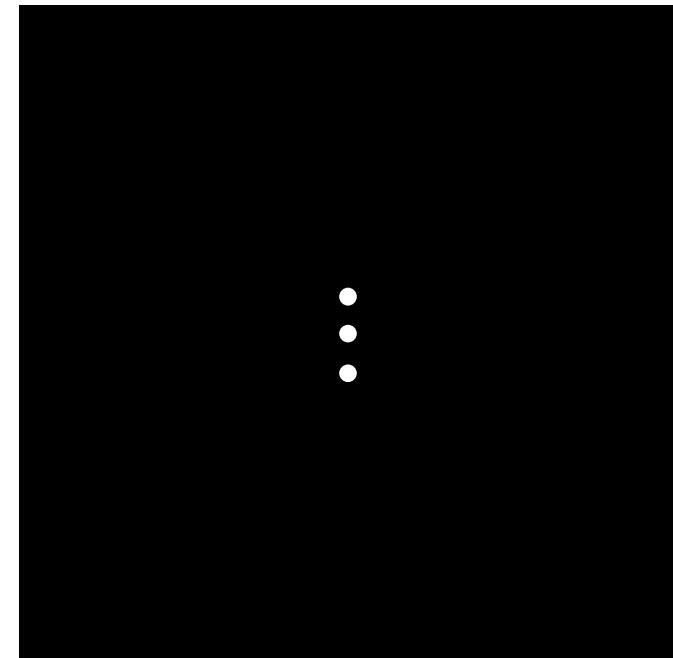
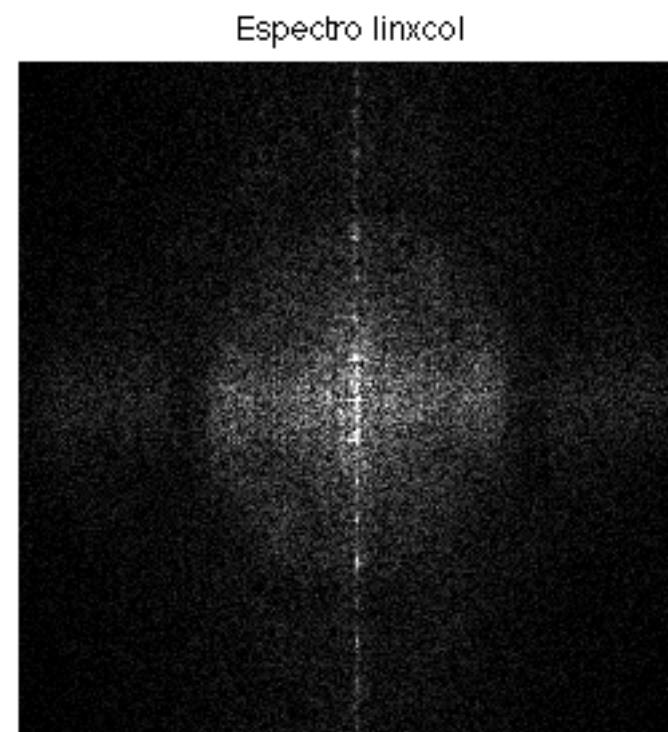
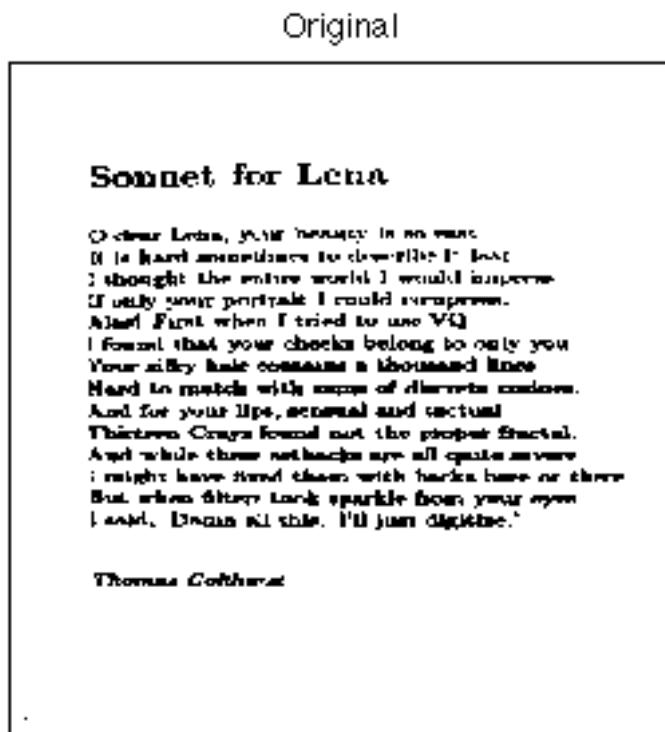
Espectro linxcol



# Exemplos de aplicação

Fernando Soares

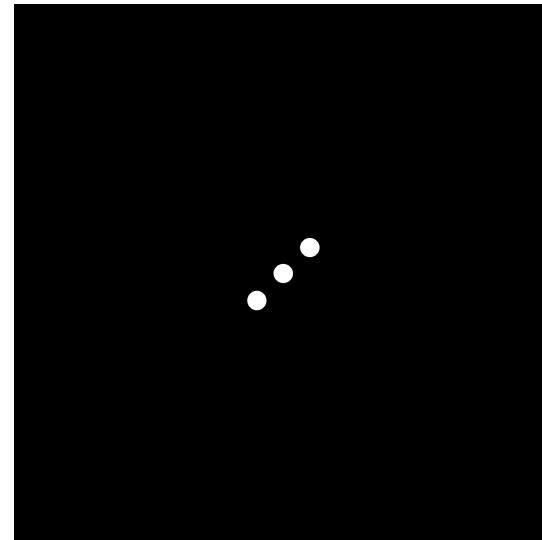
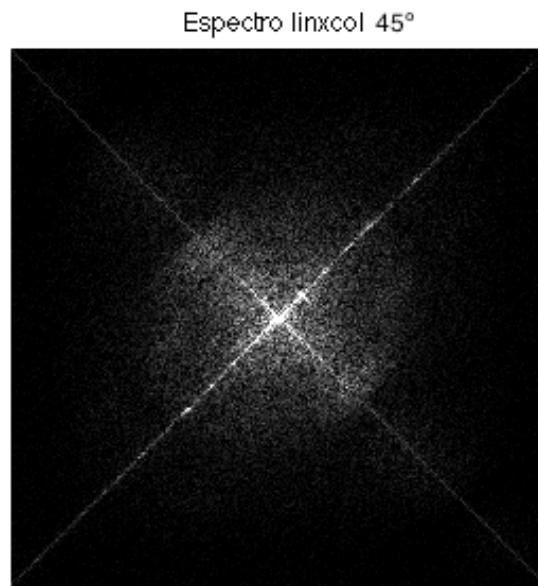
A limiarização dos picos da imagem do espectro permite identificar a orientação da disposição das linhas de texto.



# Exemplos de aplicação

Fernando Soares

No caso de uma imagem rodada de  $45^\circ$ , após a identificação dos picos de frequência sobre a linha principal, a imagem pode ser rodada usando para tal o conhecimento do ângulo que se obtém nesse procedimento.



A linha perpendicular à linha principal resulta da existência de cantos escuros na imagem rodada.

# Exemplos de aplicação

Fernando Soares

A Transformada de *Fourier* pode ser usada para executar operações de localização objectos numa imagem, por determinação do espectro de potência cruzada (que se verá no capítulo de segmentação de imagem).

Template

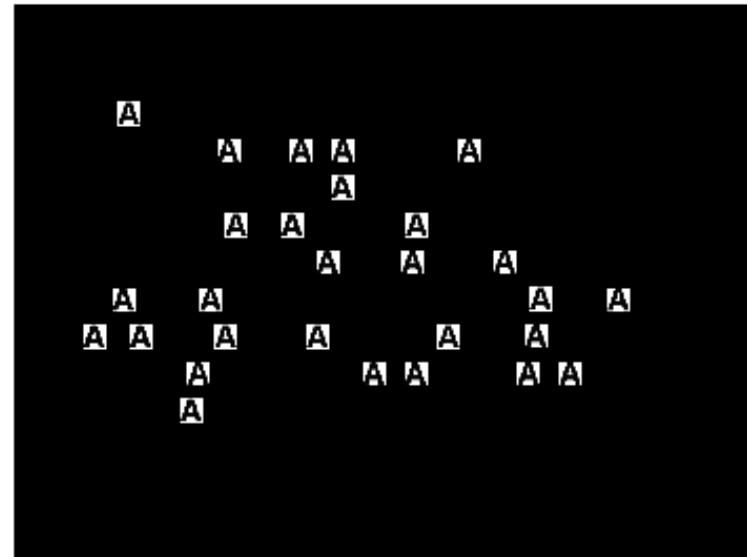


F

JOÃO RECEBEU UM TEXTO DE  
INSTRUÇÃO PARA MONTAR UM  
BRINQUEDO: BRACELETE DE SUPER  
HERÓI. MAS, A PESSOA QUE ESCREVEU,  
ESQUECEU DE FAZER A LISTA DOS  
MATERIAIS. VOCÊ CONSEGUE AJUDÁ-LO  
A FAZER A LISTA DOS MATERIAIS QUE  
PRECISAM SER SEPARADOS PARA  
MONTAR ESSE BRINQUEDO?

Imagen inicial

Resultado



Posições  
(máximos da correlação)

# 4. Morfologia matemática binária

Teoria elementar de conjuntos. Propriedades dos conjuntos. Propriedades das transformações morfológicas. Elemento estruturante. Transformação de vizinhança. Transformações morfológicas elementares. Transformações Tudo-ou-Nada. Adelgaçamento e Espessamento. Transformações geodésicas. Número de Euler (conectividade).

## Origens:

- Trabalhos pioneiros de *Georges Matheron* e *Jean Serra* (década de 60, França).
- Criação do *Centre de Morphologie Mathématique de Fontainebleau* (1968).

## Objectivo:

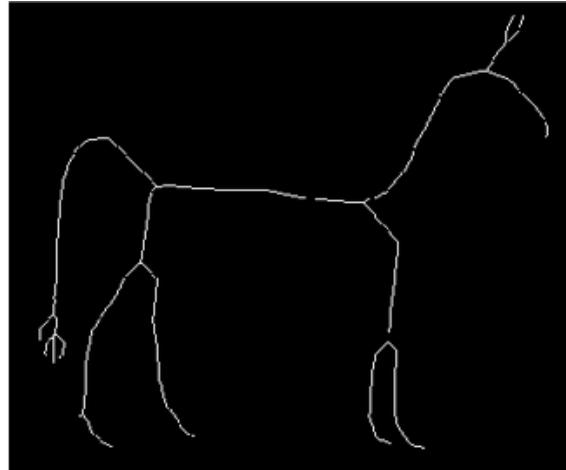
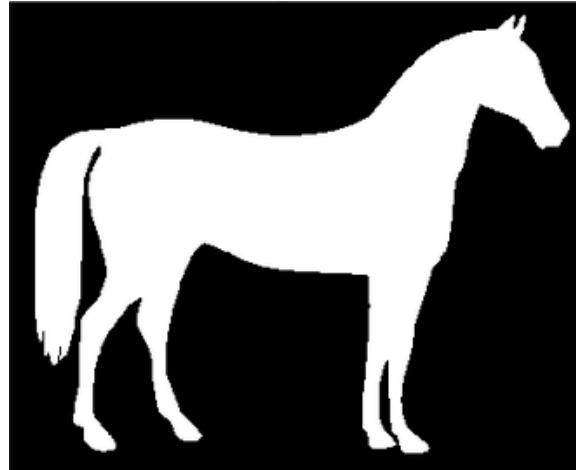
- A morfologia matemática baseia-se na teoria dos conjuntos, e pretende quantificar estruturas do ponto de vista geométrico.

## Metodologia:

1. Utiliza a noção de conjunto para representar estruturas.
2. Transformação dos conjuntos por forma a torná-los mensuráveis:
  - Interacção do conjunto de objectos em estudo, com outro objecto com forma conhecida (elemento estruturante).
  - A transformação do conjunto inicial, ao longo de sucessivas operações, evidencia as suas características estruturais, que são registadas ao longo dos novos conjuntos gerados, o que implica que o conjunto transformado é mais simples que o conjunto original.

## Metodologia:

3. Realização de medidas sobre os conjuntos transformados.



→

Transformação

→

Medida

### Medida:

- Superfície
- Forma
- Perímetro
- Etc.

O processamento morfológico de imagem pode ser aplicado nos seguintes contextos:

- Morfologia binária: as imagens são binárias (mais frequente).
- Morfologia numérica: as imagens podem ser, ou de níveis de cinzento (monocromáticas), ou coloridas (policromáticas).

Os termos «Morfologia» e «Matemática», associados (que constituem a designação desta teoria), referem-se à utilização de conceitos de lógica de conjuntos e operações numéricas.

## Intersecção

$$X \cap Y = \{x : x \in X \wedge x \in Y\}$$

- Comutativa:

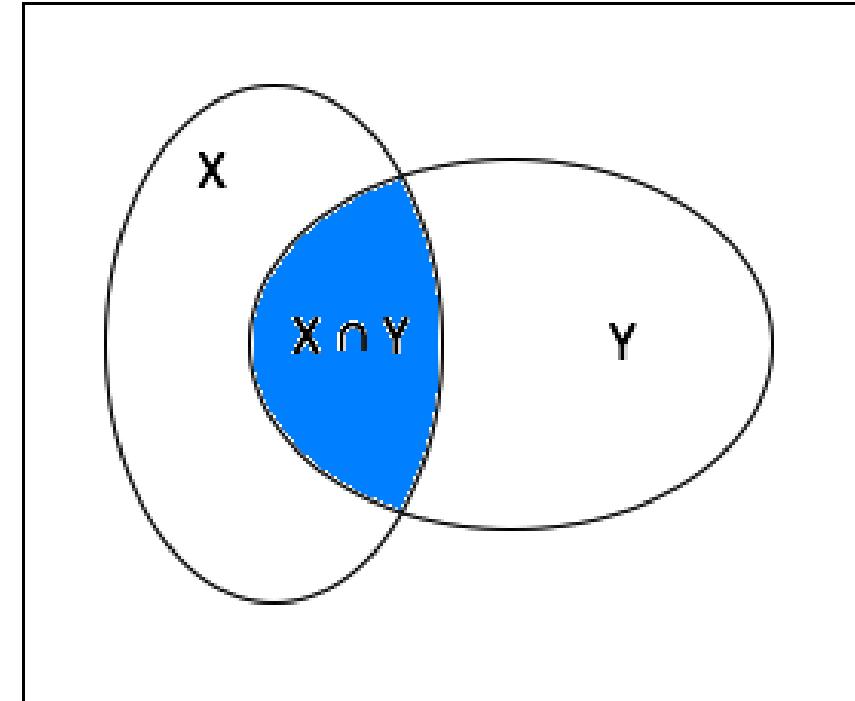
$$X \cap Y = Y \cap X$$

- Associativa:

$$X \cap (Y \cap Z) = (Y \cap X) \cap Z$$

- Idempotente:

$$X \cap X = X$$



## Reunião

$$X \cup Y = \{x : x \in X \vee x \in Y\}$$

- Comutativa:

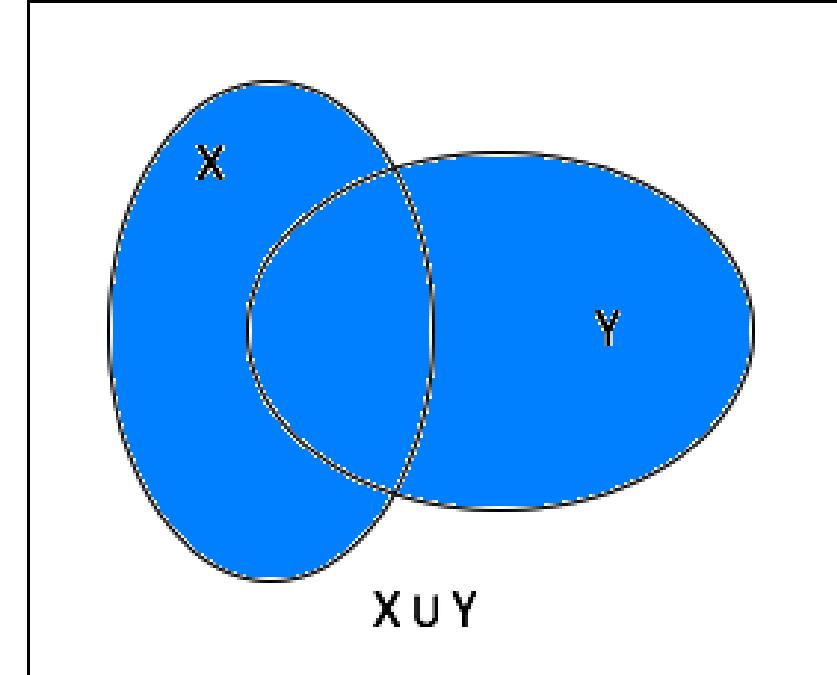
$$X \cup Y = Y \cup X$$

- Associativa:

$$X \cup (Y \cup Z) = (Y \cup X) \cup Z$$

- Idempotente:

$$X \cup X = X$$



# Teoria elementar acerca dos conjuntos

Fernando Soares

Relação entre a intersecção e a união (distributividade):

$$X \cap (Y \cup Z) = (X \cap Y) \cup (X \cap Z)$$

$$X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z)$$

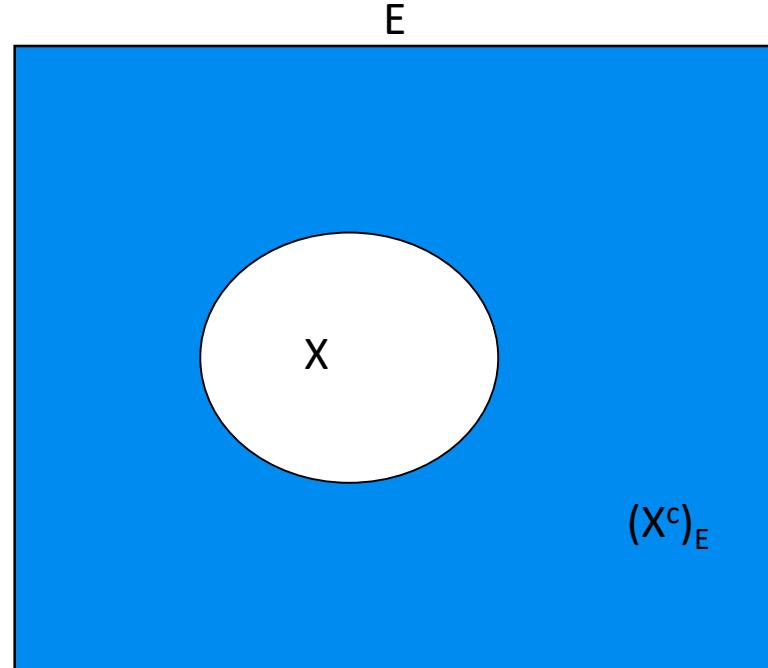
## Conjunto complementar

$$(X^c)_E = \{x : x \in E \wedge x \notin X\}$$

- Fórmulas de Morgan:

$$((X \cap Y)^c)_E = (X^c)_E \cup (Y^c)_E$$

$$((X \cup Y)^c)_E = (X^c)_E \cap (Y^c)_E$$

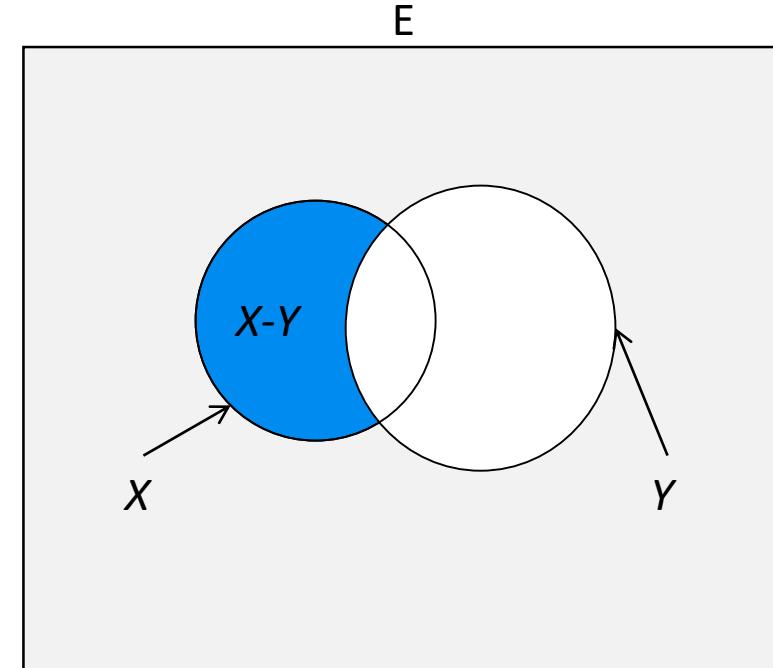


## Diferença lógica

$$X - Y = \{x : x \in X \wedge x \notin Y\}$$

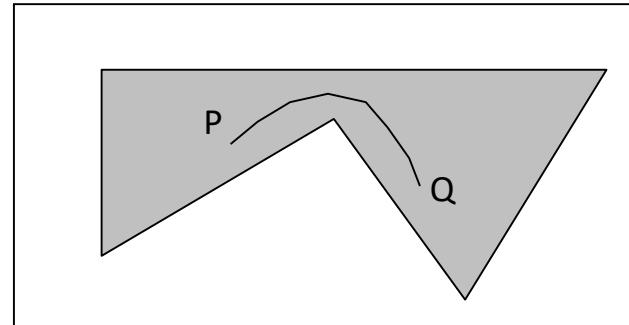
ou

$$X - Y = X \cap (X \cap Y)^c$$

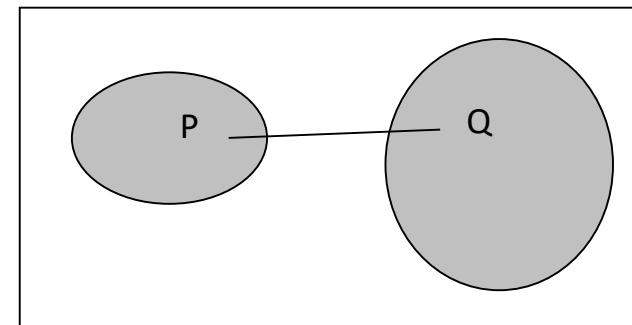


# Propriedades dos conjuntos

**Conexidade:** um conjunto  $X$  diz-se conexo se, para quaisquer dois pontos  $P(x_i, y_i)$  e  $Q(x_m, y_n)$  nele incluídos, existe pelo menos um caminho que os une e que está totalmente incluído em  $X$ .



X Conexo

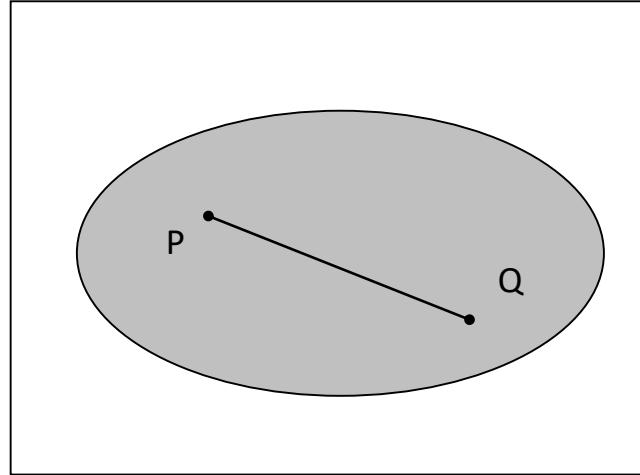


X não conexo

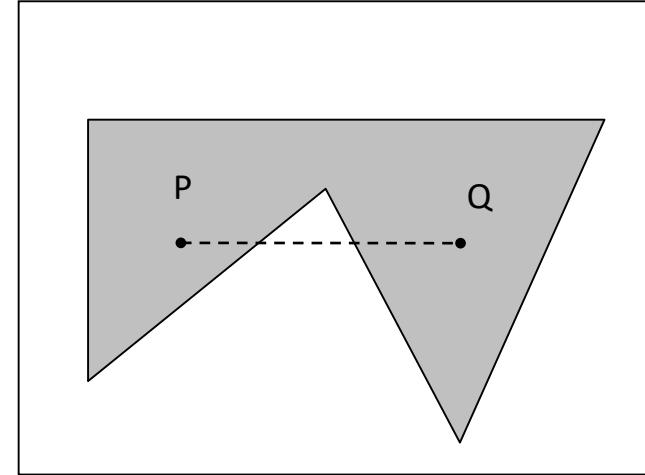
A conexidade de um conjunto depende, no entanto, da forma como se encontra definida a ligação entre os pixels numa malha digital (nas malhas digitais quadradas com conectividade-4 e conectividade-8, ou hexagonal com conectividade-6).

# Propriedades dos conjuntos

**Convexidade:** um conjunto  $X$  diz-se convexo se, qualquer que seja o par de pontos  $P(x_i, y_i)$  e  $Q(x_m, y_n)$  nele incluídos, o segmento de recta que os une está totalmente incluído em  $X$ .



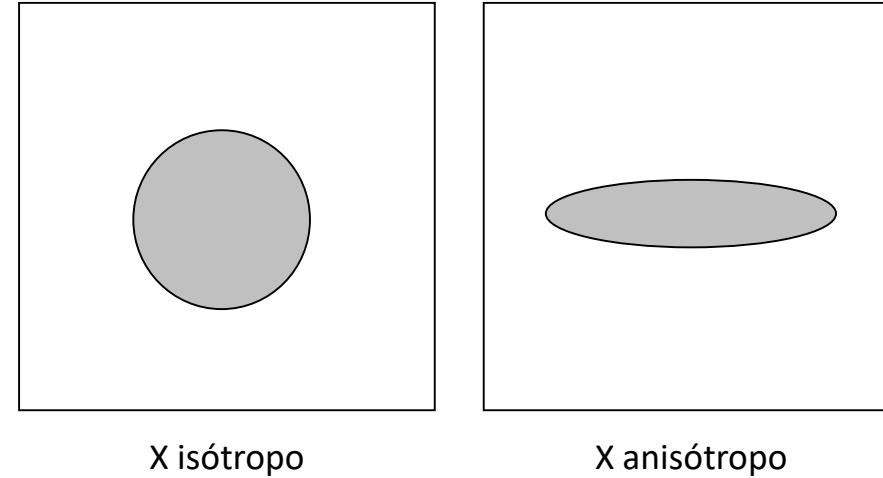
$X$  convexo



$X$  não convexo

# Propriedades dos conjuntos

**Isotropia:** Um conjunto  $X$  diz-se isotrópico se está uniformemente espalhado segundo todas as direções.



Predominando uma dada direcção, está-se perante um conjunto anisotrópico. Este conceito está associado à estrutura espacial dos objetos.

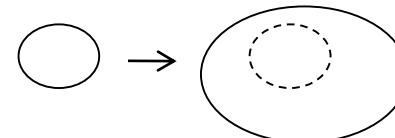
# Propriedades das transformações morfológicas

Fernando Soares

Se  $\varphi$  é uma transformação morfológica então obedece a uma ou mais do que uma das seguintes propriedades:

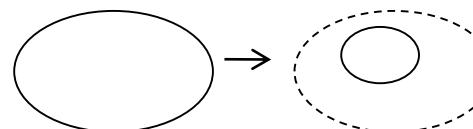
- $\varphi$  é **extensiva** se o conjunto resultante contém conjunto inicial X.

$$X \subset \varphi(X)$$



- $\varphi$  é **anti-extensiva** se o conjunto resultante está contido no conjunto inicial X.

$$\varphi(X) \subset X$$

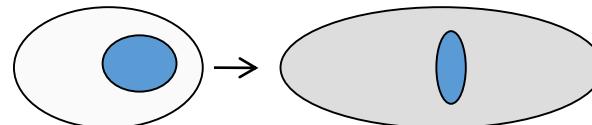


# Propriedades das transformações morfológicas

Fernando Soares

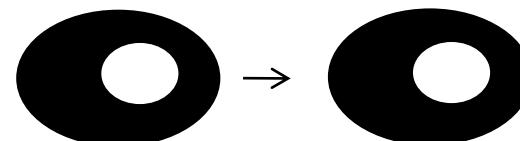
- $\varphi$  é **crescente** se a relação de inclusão entre os conjuntos inicial e resultante se mantém.

$$Y \subset X \Rightarrow \varphi(Y) \subset \varphi(X)$$



- $\varphi$  é **idempotente** se a sua aplicação sucessiva a  $X$  não o altera.

$$\varphi(\varphi(X)) \equiv X$$



# Propriedades das transformações morfológicas

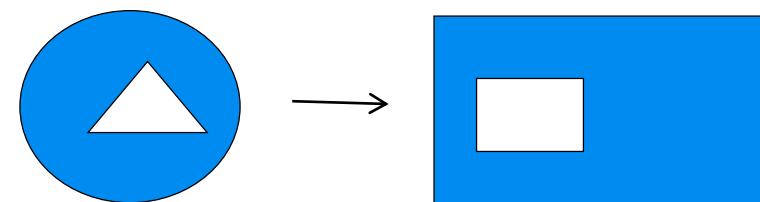
Fernando Soares

- Duas transformações morfológicas  $\varphi_1$  e  $\varphi_2$  são **duais** se:

$$\varphi_1(X) \equiv \varphi_2(X^c)$$

- Finalmente, uma transformação é dita **homotópica** se ela não modifica o número de conexidade  $E$  de um conjunto  $X$ , isto é:

$$E[\varphi(X)] = E[X]$$



# Elemento estruturante

**Elemento estruturante (B):** É um caso particular de imagem binária, sendo usualmente pequeno e simples (por exemplo  $B_{3 \times 3}$  com todos os valores iguais a um).

Uma transformação morfológica só pode ser realizada com a definição prévia de um elemento estruturante.

Como o próprio nome indica, quando associado a uma certa transformação morfológica, B percorre a imagem com o propósito de aferir se “encaixa” ou não nos objectos nela presentes. No processo, pode modificar a forma e as características topológicas desses objectos (por exemplo, a conexidade ou a convexidade).

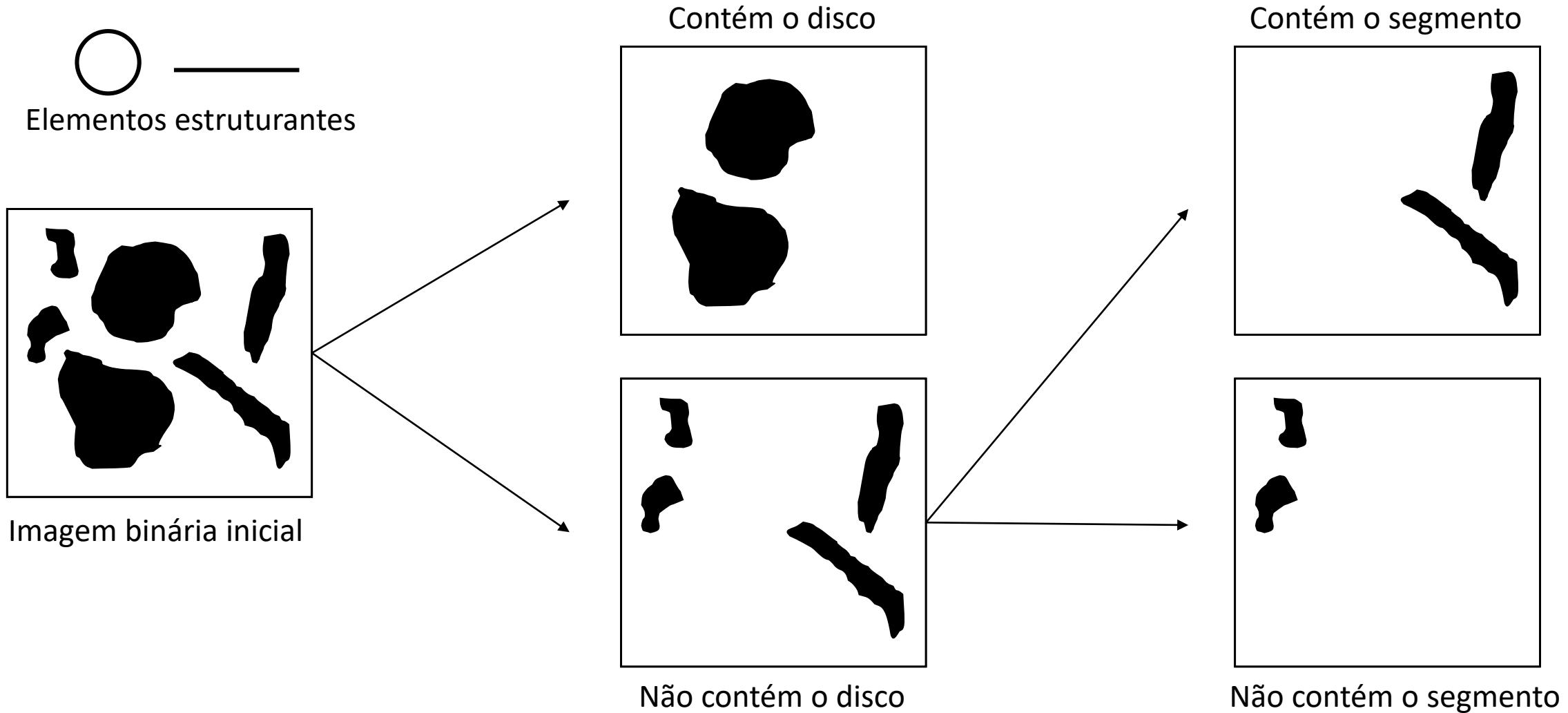
# Elemento estruturante

Fernando Soares

A forma de B pode ser qualquer, sendo as mais comuns o quadrado, o disco, o segmento, o círculo, o par de pontos e o hexágono (na malha digital hexagonal), que são escolhidos de acordo com os objectivos pretendidos. Por exemplo:

- DISCO: determinação da distribuição de tamanho dos objectos (granulometria).
- SEGMENTO: detecção de alinhamentos preferenciais.
- PAR DE PONTOS: caracterização do estado de dispersão (covariância).
- CIRCUNFERÊNCIA: estudo da vizinhança de um ponto (transformação de vizinhança).

# Elemento estruturante



# Elemento estruturante

Fernando Soares

O centro, ou pixel de referência de B é geralmente o seu centro geométrico, podendo contudo ser definido qualquer outro ponto para o efeito.

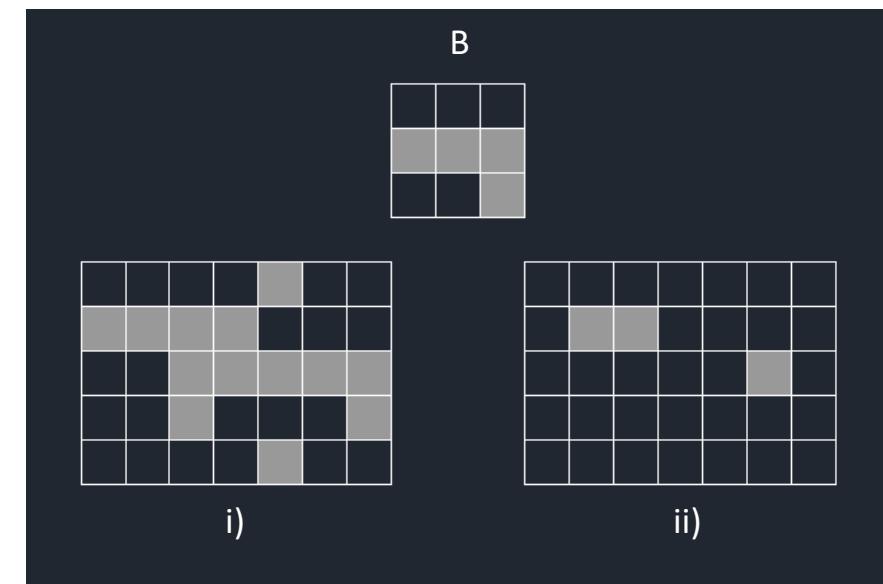
O centro de B marca a sua posição sobre a imagem inicial e, por conseguinte, a posição do pixel transformado.

A distribuição dos valores dos pixels no interior de B designa-se por configuração de vizinhança (V).

# Transformação de vizinhança

Uma **transformação de vizinhança** consiste na identificação/alteração de um pixel de uma imagem no caso de se verificar uma dada configuração de vizinhança V em redor desse pixel.

- Na figura, tem-se um quadrado elementar B que percorrerá todos os pixels da imagem i).
- Em ii) estão marcadas as posições em que a configuração de vizinhança  $V(B_x)$  é idêntica à de B.
- Como transformações de vizinhança mais comuns têm-se o adelgaçamento e o espessamento.

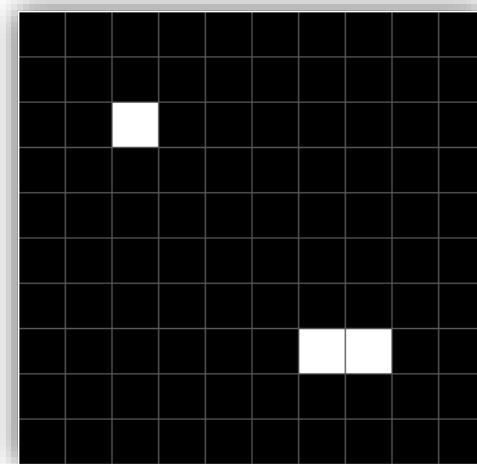
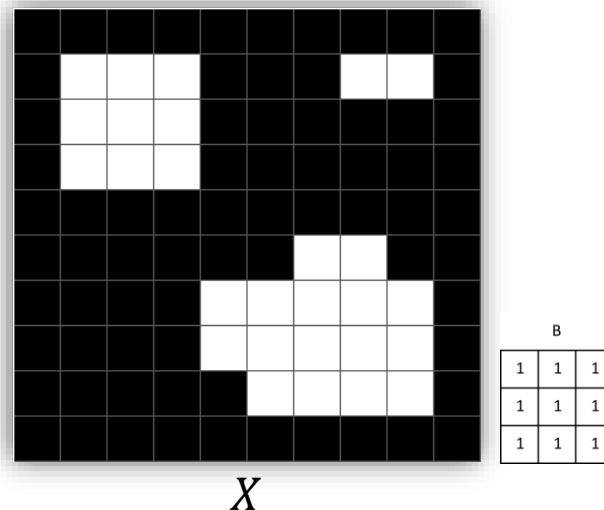


# Transformações morfológicas elementares

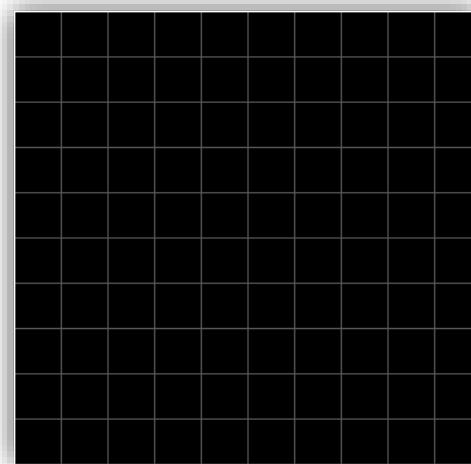
Fernando Soares

A transformação morfológica de **erosão** ( $\varepsilon$ ) de um dado conjunto  $X$ , por um elemento estruturante com a sua origem em  $x$  ( $B_x$ ), define-se pela expressão seguinte:

$$\varepsilon_B(X) = \{x: B_x \subset X\}$$



$$\varepsilon_{B_{x(3 \times 3)}}(X)$$

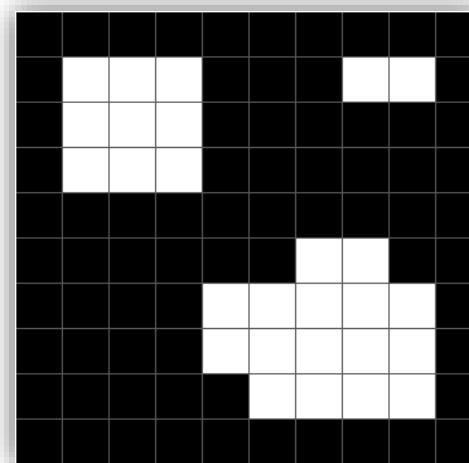


$$\varepsilon_{B_{x(5 \times 5)}}(X)$$

# Transformações morfológicas elementares

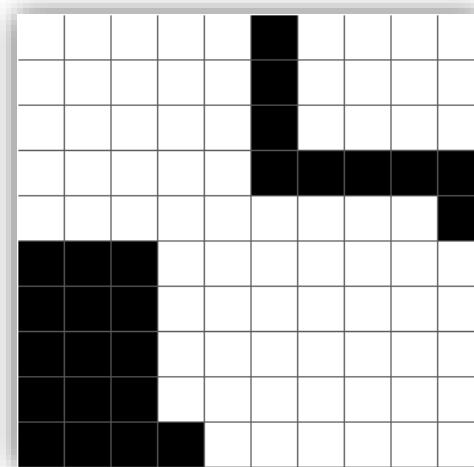
A transformação morfológica de **dilatação** ( $\delta$ ) de um dado conjunto  $X$ , por um elemento estruturante com a sua origem em  $x$  ( $B_x$ ), define-se pela expressão seguinte:

$$\delta_B(X) = \{x: B_x \cap X \neq \emptyset\}$$

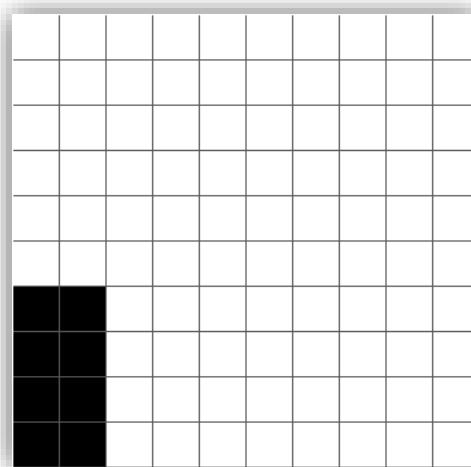


B
1 1 1
1 1 1
1 1 1

$X$



$\delta_{B_{x(3\times 3)}}(X)$

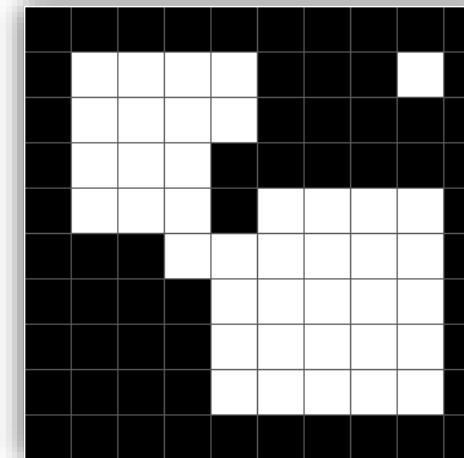


$\delta_{B_{x(5\times 5)}}(X)$

# Transformações morfológicas elementares

A **abertura** ( $\gamma$ ) de  $X$  consiste em executar a dilatação do resultado da erosão do conjunto  $X$ .

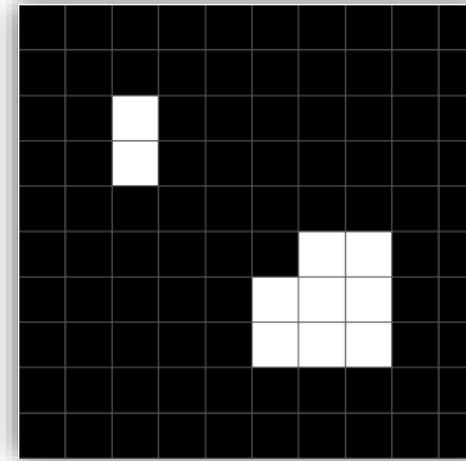
$$\gamma_B(X) = \delta_B(\varepsilon_B(X))$$



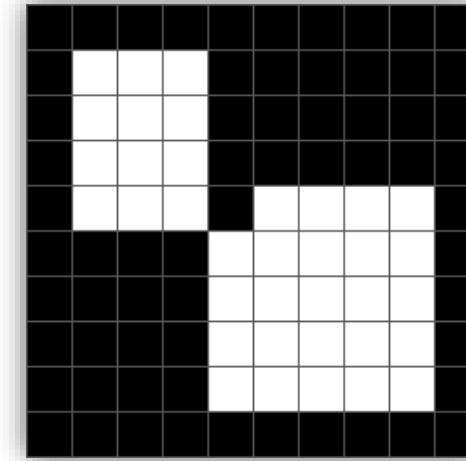
B

1	1	1
1	1	1
1	1	1

$X$



$$\varepsilon_{B_{X(3 \times 3)}}(X) = Y$$



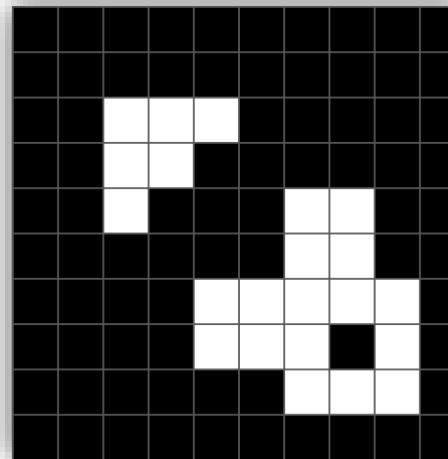
$$\delta_{B_{X(3 \times 3)}}(Y) = \gamma_B(X)$$

# Transformações morfológicas elementares

Fernando Soares

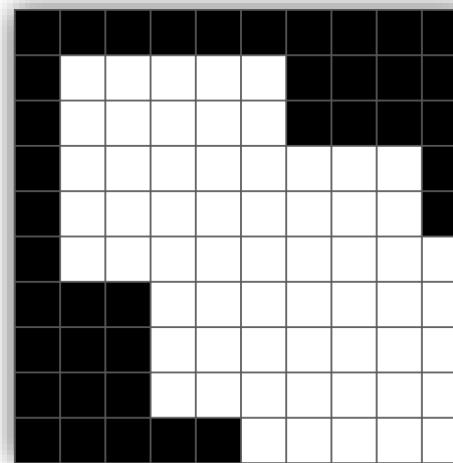
O fecho ( $\phi$ ) de  $X$  consiste em executar a erosão do resultado da dilatação do conjunto  $X$ .

$$\phi_B(X) = \varepsilon_B(\delta_B(X))$$

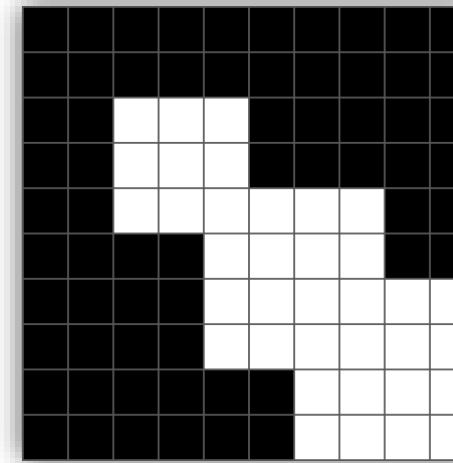


B
1 1 1
1 1 1
1 1 1

$X$



$$\delta_{B_{x(3 \times 3)}}(X) = Y$$



$$\varepsilon_{B_{x(3 \times 3)}}(Y) = \phi_B(X)$$

# Transformações morfológicas elementares

Fernando Soares

Síntese das propriedades das transformações morfológicas elementares:

	Extensiva	Anti-extensiva	Crescente	Idempotente	Homotópica	Transformação dual
Erosão	-	X	X	-	-	<i>Dilatação</i>
Dilatação	X	-	X	-	-	<i>Erosão</i>
Abertura	-	X	X	X	-	<i>Fecho</i>
Fecho	X	-	X	X	-	<i>Abertura</i>

# Transformação tudo-ou-nada (*Hit-or-Miss*)

Fernando Soares

A **transformação “Tudo-ou-Nada”** (*Hit-or-Miss transformation - HMT*) aplicada a  $X$  consiste numa transformação de vizinhança que recorre a um elemento estruturante composto  $B = (B_1, B_2)$ , com  $B_1 \cap B_2 = \emptyset$ , e que resulta da verificação simultânea das seguintes condições:  $B_1$  coincide com  $X$  e  $B_2$  coincide com  $X^c$ .

$$HMT_B(X) = \{x: (B_1)_x \subseteq X \wedge (B_2)_x \subseteq X^c\}$$

$$HMT_B(X) = \varepsilon_{B_1}(X) \cap \varepsilon_{B_2}(X^c)$$

- Os índices do elemento estruturante composto são geralmente três: “1” ( $\in$  domínio de  $B_1$ ), “-1” ( $\in$  domínio de  $B_2$ ) e “0” (é indiferente).
- A *HMT* é geralmente usada para encontrar configurações específicas em grupos de pixels ou objectos e determina-se pela intersecção entre as erosões de  $X$  por  $B_1$  e de  $X^c$  por  $B_2$  (como se verá em alguns exemplos a seguir).

# Exemplos de transformações tudo-ou-nada

**Pontos isolados** (pixels sem quaisquer outros pixels na sua vizinhança).

Desenham-se duas configurações (EE)  $B_1$  e  $B_2$ , tal que,  $B_1 \subseteq X$  e  $B_2 \subseteq X^c$ .

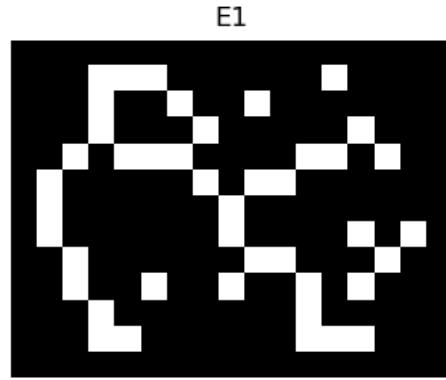
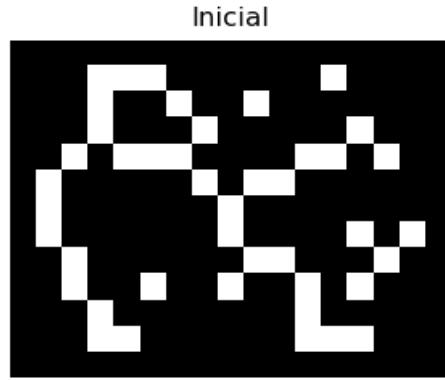
B1	B2
0   0   0	1   1   1
0   1   0	1   0   1
0   0   0	1   1   1

Optando por um B composto tem-se a seguinte configuração:

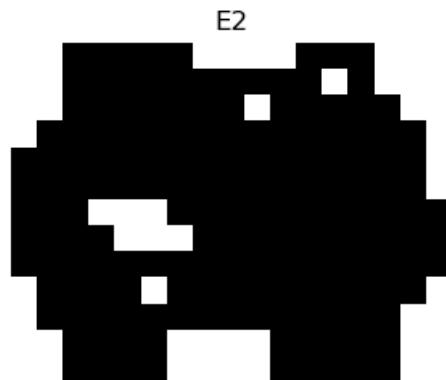
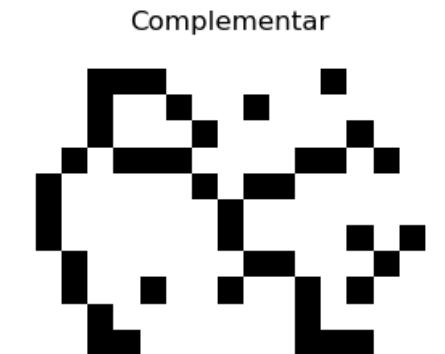
B
-1   -1   -1
-1   1   -1
-1   -1   -1

1 = pertence  
-1 = pertence ao complementar  
0 = indiferente.

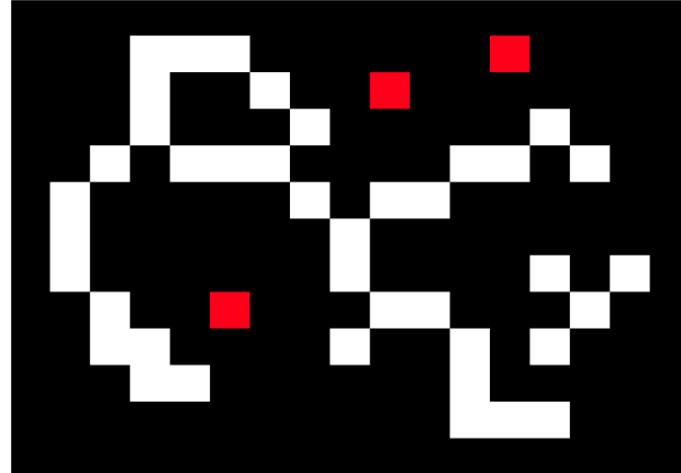
# Exemplos de transformações tudo-ou-nada



$$\varepsilon_{B_1}(X) = \{x: B_1 \subset X\}$$



$$\varepsilon_{B_2}(X^c) = \{x: B_2 \subset X^c\}$$



$$\text{Pontos isolados} = \varepsilon_{B_1} \cap \varepsilon_{B_2}$$

# Exemplos de transformações tudo-ou-nada

**Pontos extremos** (pixels com um pixel no máximo na sua vizinhança próxima).

Com  $B_1$  e  $B_2$ :

$B1(90^\circ)$

-1	1	-1
-1	1	-1
-1	-1	-1

$B2(90^\circ)$

-1	-1	-1
1	-1	1
1	1	1

$B1(45^\circ)$

1	-1	-1
-1	1	-1
-1	-1	-1

$B2(45^\circ)$

-1	1	1
1	-1	1
1	1	1

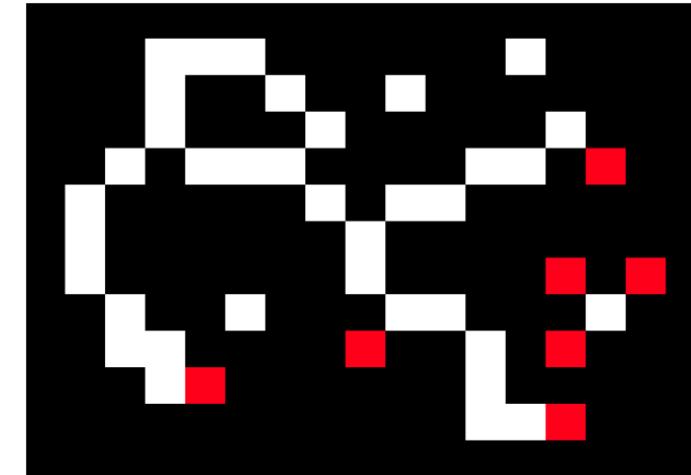
Optando por B composto:

$B(90^\circ)$

0	1	0
-1	1	-1
-1	-1	-1

$B(45^\circ)$

1	-1	-1
-1	1	-1
-1	-1	-1



*Extremidades*

# Exemplos de transformações tudo-ou-nada

**Pontos múltiplos** (pixels com mais do que dois pixels na sua vizinhança próxima).

Com  $B_1$  e  $B_2$ :

$B1(90^\circ)$

-1	1	-1
-1	1	-1
1	-1	1

$B2(90^\circ)$

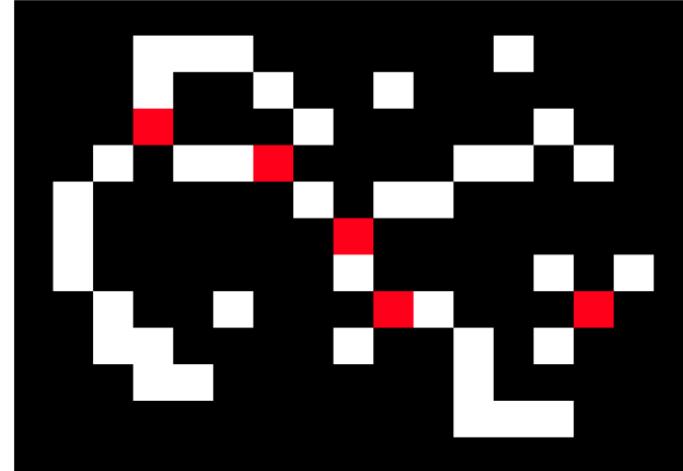
-1	-1	-1
1	-1	1
-1	1	-1

$B1(45^\circ)$

1	-1	1
-1	1	-1
1	-1	-1

$B2(45^\circ)$

-1	-1	-1
-1	-1	1
-1	1	1



Optando por B composto:

$B(90^\circ)$

0	1	0
-1	1	-1
1	-1	1

$B(45^\circ)$

1	0	1
0	1	-1
1	-1	-1

*Pontos múltiplos*

# Exemplos de transformações tudo-ou-nada

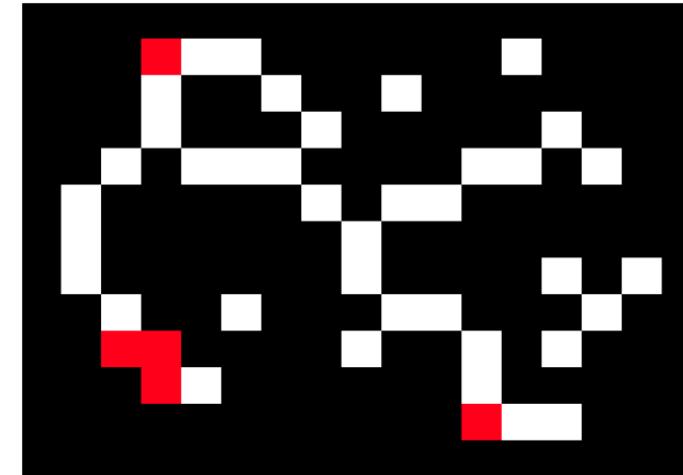
**Cantos rectos** (pixels que formam um ângulo recto convexo).

Com  $B_1$  e  $B_2$ :

$$\begin{array}{c} B1 \\ \hline \begin{matrix} -1 & 1 & -1 \\ -1 & 1 & 1 \\ -1 & -1 & -1 \end{matrix} \end{array} \quad \begin{array}{c} B2 \\ \hline \begin{matrix} -1 & -1 & -1 \\ 1 & -1 & -1 \\ 1 & 1 & -1 \end{matrix} \end{array}$$

Optando por B composto:

$$\begin{array}{c} B \\ \hline \begin{matrix} 0 & 1 & 0 \\ -1 & 1 & 1 \\ -1 & -1 & 0 \end{matrix} \end{array}$$



*Cantos rectos*

# Exemplos de transformações tudo-ou-nada

Fernando Soares

**Contornos** (pixels com pelo menos um pixel pertencente ao conjunto complementar na sua vizinhança próxima).

Com  $B_1$  e  $B_2$ :

B1(90º)

-1	-1	-1
-1	1	-1
-1	-1	-1

B2(90º)

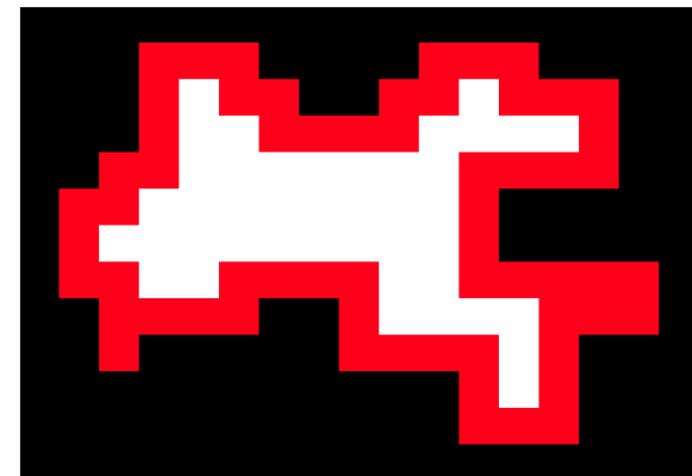
-1	1	-1
-1	-1	-1
-1	-1	-1

B1(45º)

-1	1	1
1	1	1
1	1	1

B2(45º)

1	-1	-1
-1	-1	-1
-1	-1	-1



Pontos de fronteira

Optando por B composto:

B(90º)

0	-1	0
0	1	0
0	0	0

B(45º)

-1	0	0
0	1	0
0	0	0

# Transformação de adelgaçamento

Fernando Soares

**Adelgaçamento (THIN)** de um conjunto X: consiste numa transformação de vizinhança que retira a X todos pontos que correspondam a uma dada configuração de vizinhança  $V(B_x)$ .

$$THIN(X, B) = X \cap NOT[HMT(X, B)]$$

É aplicada apenas a imagens binárias e produz uma imagem binária como resultado.

De forma geral, a operação de adelgaçamento é determinada por translação do elemento estruturante B, por todos os pixels da imagem, comparando em cada um a sua configuração de vizinhança com a configuração dos correspondentes pixels na imagem.

Havendo uma coincidência entre ambas as configurações, então, ao pixel da imagem correspondente à posição do centro de B, é atribuído o valor 0; caso contrário mantém-se inalterável.

# Exemplos de transformações de adelgaçamento

Fernando Soares

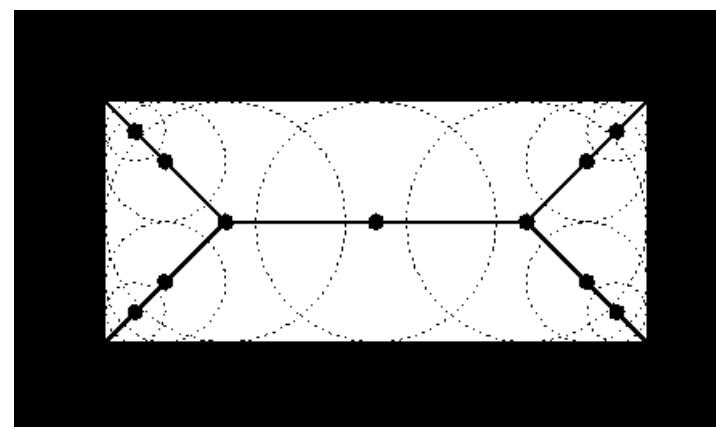
A **erosão** e a **abertura** morfológicas são exemplos de transformações de adelgaçamento.

# Exemplos de transformações de adelgaçamento

Fernando Soares

**Esqueletização:** é um exemplo de adelgaçamento morfológico que reduz as regiões de uma imagem binária a uma estrutura mínima que preserve a extensão e conectividade das regiões originais. É uma transformação usada frequentemente para “estreitar” resultados de detecção de fronteiras, reduzindo a espessura das linhas a outras com apenas um pixel de espessura. Note-se que o esqueleto resultante é um conjunto conexo é igualmente um conjunto conexo.

O esqueleto de um conjunto pode ser determinado de diversas formas: a) Localização dos centros de circunferências máximas bi-tangentes aos limites da região considerada.



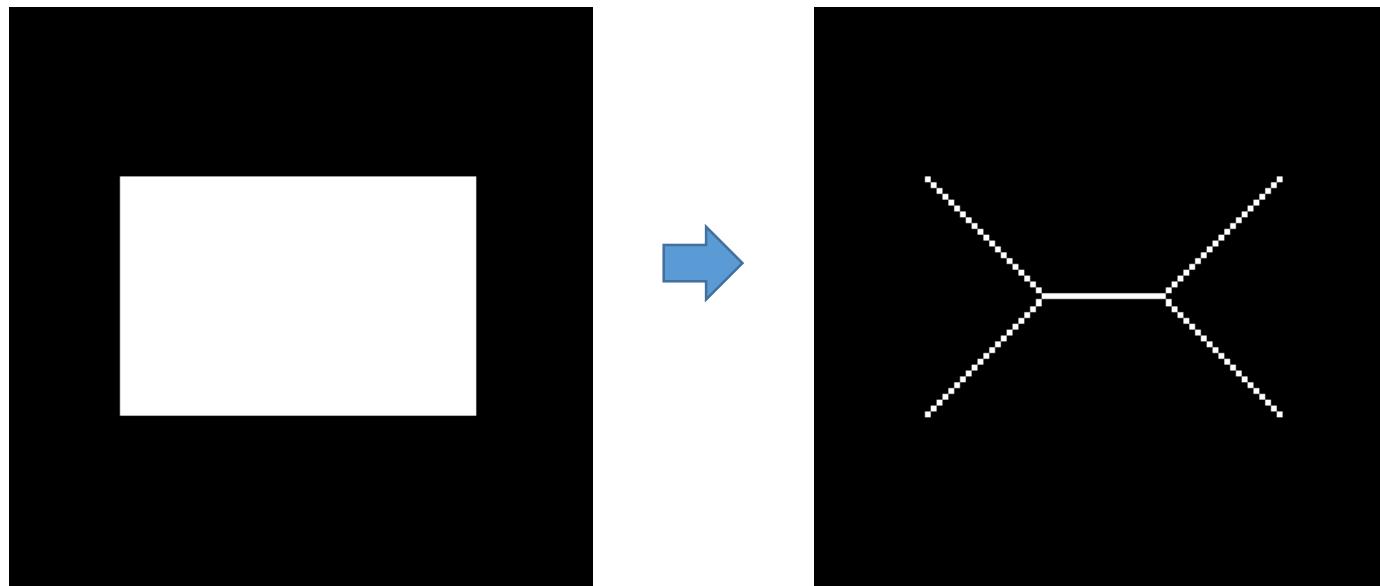
# Exemplos de transformações de adelgaçamento

Fernando Soares

b) Fórmula de Lantuéjoul: para uma imagem binária discreta  $X$ , o esqueleto  $S(X)$  é a união de todos os subconjuntos  $S_k(X)$ , com um elemento estruturante  $B$  de dimensão  $k$ .

$$S(X) = \bigcup_{k=1}^K S_k(X)$$

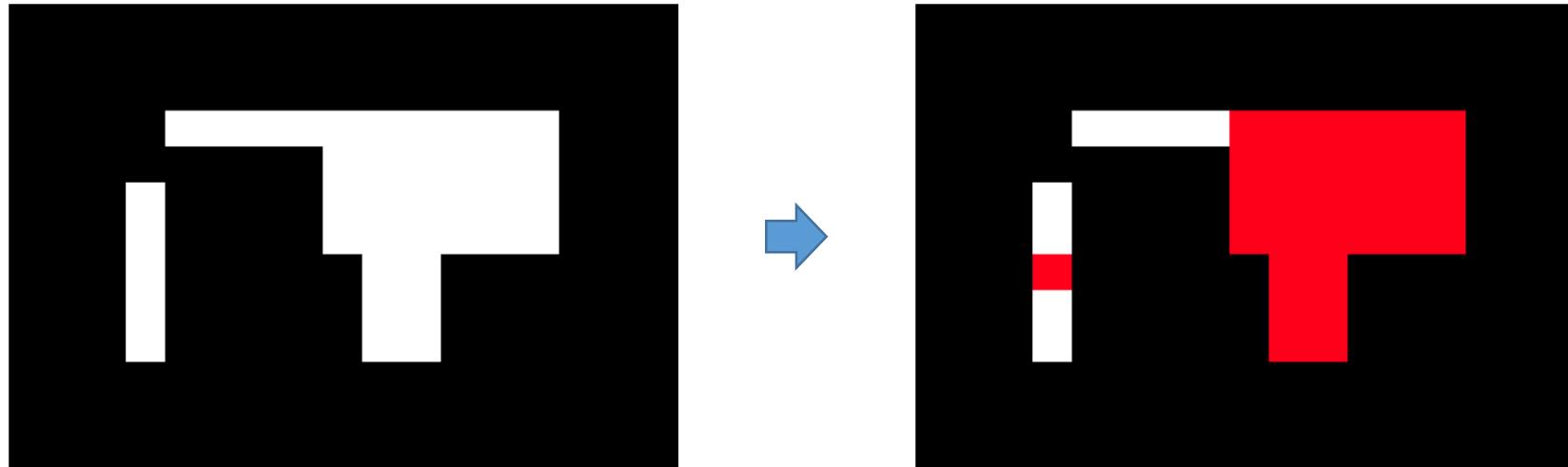
$$S_k(X) = \varepsilon_{kB}(X) \cap \text{NOT}[\gamma_B(\varepsilon_{kB}(X))]$$



# Exemplos de transformações de adelgaçamento

Fernando Soares

**Escanhoamento (prune):** é também uma operação de adelgaçamento que visa suprimir sucessivamente os pontos extremos de um conjunto binário, até se verificar a condição de idempotência.



# Transformação de espessamento

Fernando Soares

**Espessamento (THICK)** de um conjunto  $X$ : consiste numa transformação de vizinhança que adiciona a  $X$  todos pontos que correspondam a uma determinada configuração de vizinhança  $V(B_x)$ .

$$\text{THICK}(X, B) = X \cup HMT(X, B)$$

É aplicada apenas a imagens binárias e produz uma imagem binária como resultado.

De forma geral, a operação de adelgaçamento é determinada por translação do elemento estruturante  $B$ , por todos os pixels da imagem, comparando em cada um a sua configuração de vizinhança com a configuração dos correspondentes pixels na imagem.

Se se verificar uma coincidência entre ambas as configurações, então no pixel correspondente à posição do centro de  $B$  é atribuído o valor 1; caso contrário mantém-se inalterável.

# Exemplos de transformações de espessamento

Fernando Soares

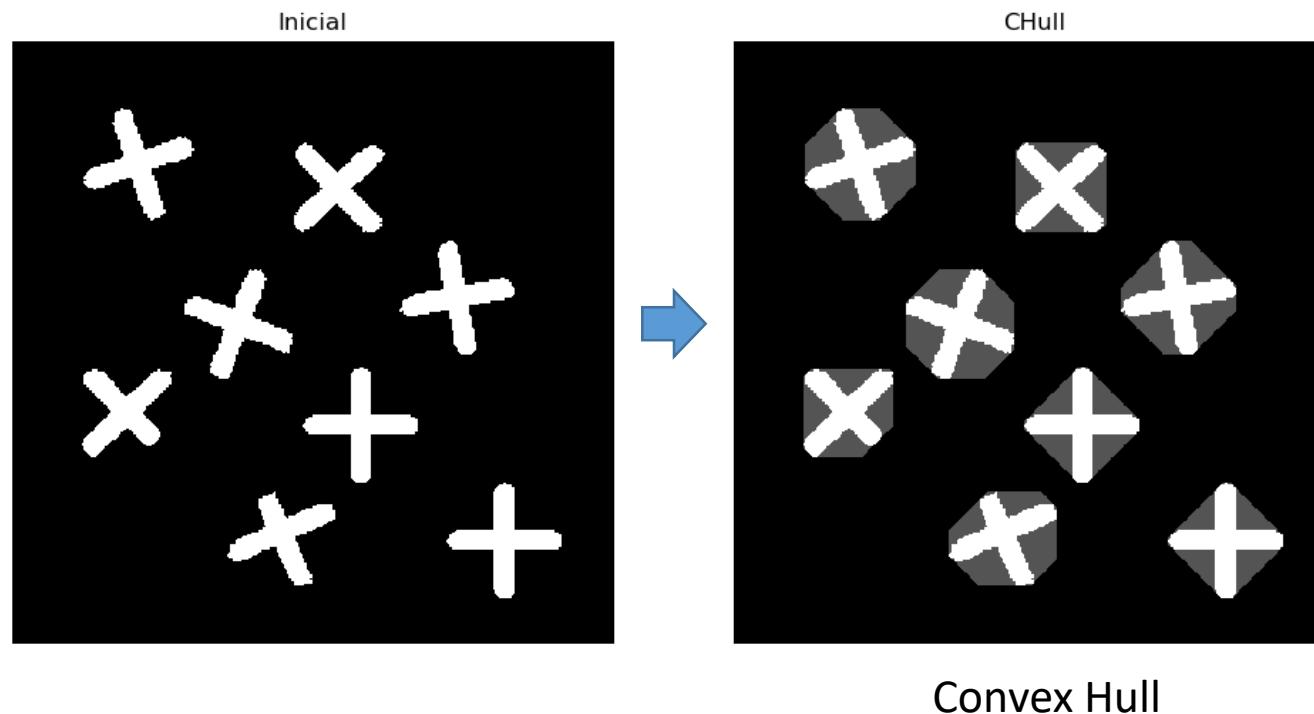
A **dilatação** e o **fecho** morfológicos são exemplos de transformações de espessamento.

# Exemplos de transformações de espessamento

**Envelope convexo (convex hull):** determina-se por execução da transformação HMT, para determinar concavidades nos objectos e consequente preenchimento. A operação é iterativa e continuará até atingir a idempotência.

$$\begin{array}{c} \text{B1}(90^\circ) \\ \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & -1 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array} \end{array} \quad \begin{array}{c} \text{B2}(90^\circ) \\ \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 1 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array} \end{array}$$

$$\begin{array}{c} \text{B1}(90^\circ) \\ \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline -1 & -1 & 1 \\ \hline -1 & -1 & -1 \\ \hline \end{array} \end{array} \quad \begin{array}{c} \text{B2}(90^\circ) \\ \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 1 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array} \end{array}$$



# Exemplos de transformações de espessamento

Fernando Soares

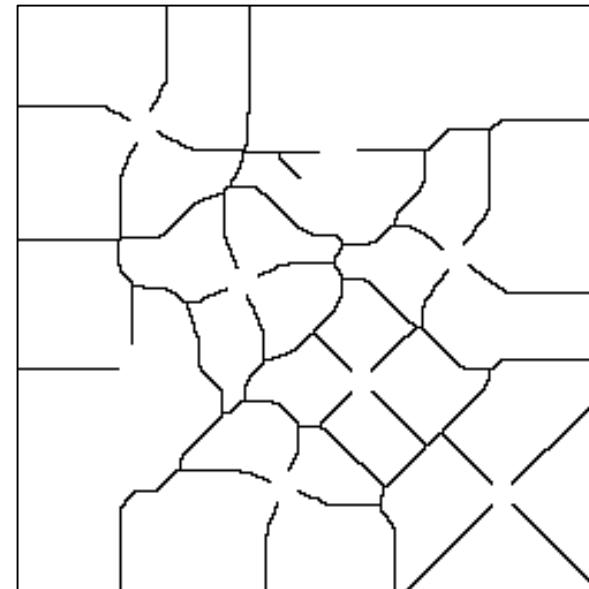
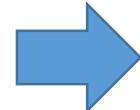
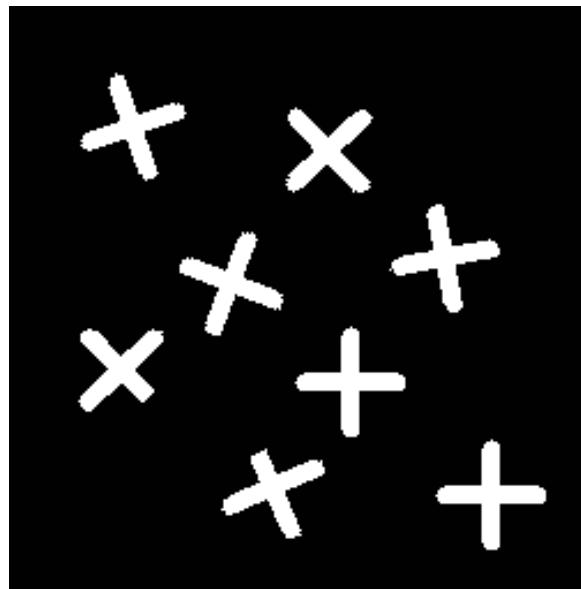
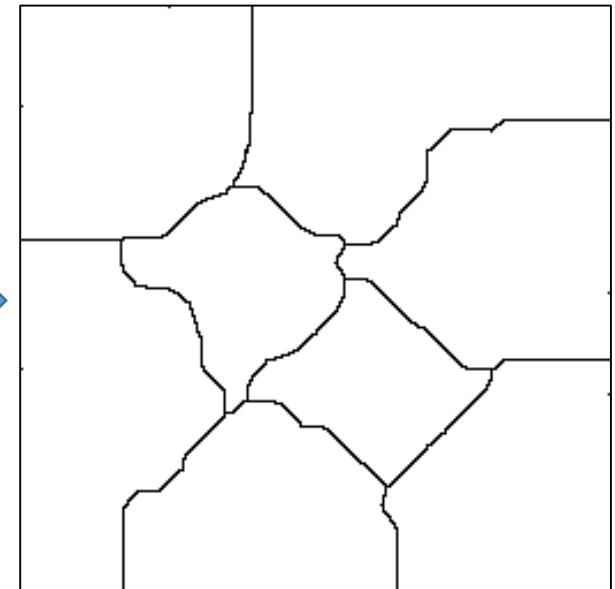
**Esqueleto por zonas de influência (*Skeleton by Influence Zones* - SKIZ):** é uma estrutura de esqueleto que divide uma imagem em regiões, cada qual contendo um objecto distinto da imagem.

As fronteiras são definidas por forma a que todos os pontos interiores a cada área estejam mais próximos do correspondente objecto interior a essa área. É por vezes designada por *Diagrama de Voronoi*. A operação é iterativa e continuará até atingir a idempotência.

# Exemplos de transformações de espessamento

Fernando Soares

O SKIZ pode ser obtido por um processo métrico, calculando distâncias euclidianas, ou por processos morfológicos, envolvendo dilatações com elementos estruturantes de diferentes tamanhos.

Esqueleto do *background*

Escanhoamento do esqueleto

# Transformações geodésicas binárias

Fernando Soares

As **transformações geodésicas** binárias são transformações morfológicas sobre uma imagem binária Y, condicionadas por uma determinada geodesia binária X. Destacam-se as seguintes:

1 - Dilatação geodésica

2 - Erosão geodésica

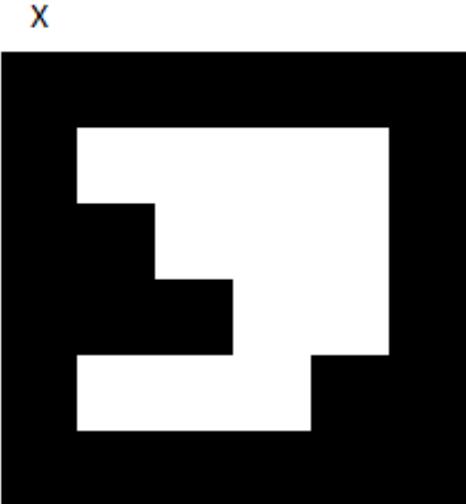
3 - Reconstrução geodésica binária por dilatações geodésicas sucessivas.

4 - Reconstrução geodésica binária por erosões geodésicas sucessivas.

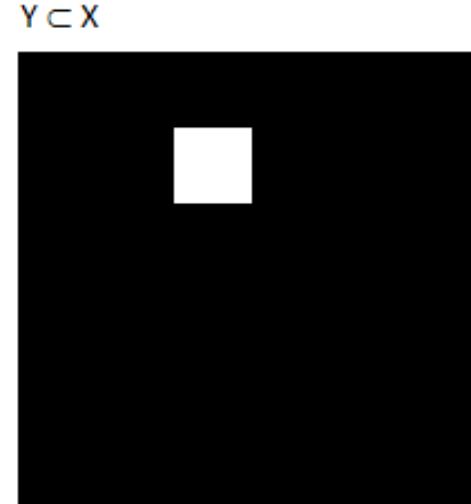
# Transformações geodésicas binárias

**Dilatação geodésica:** dilatação morfológica de um conjunto  $Y$  condicionada à geodesia  $X$ .

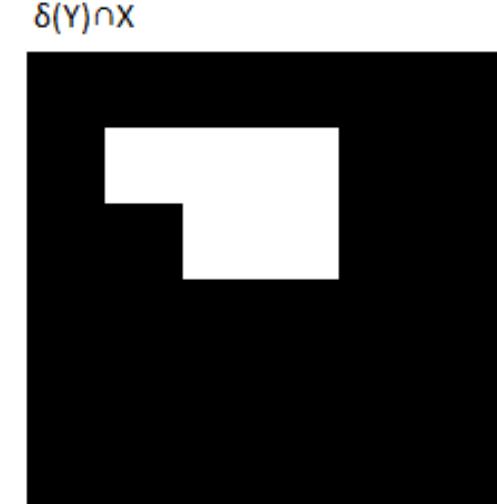
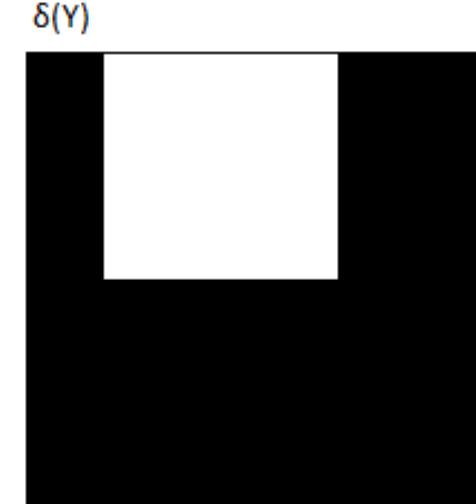
$$\delta_X(Y) = \delta(Y) \cap X$$



Geodesia



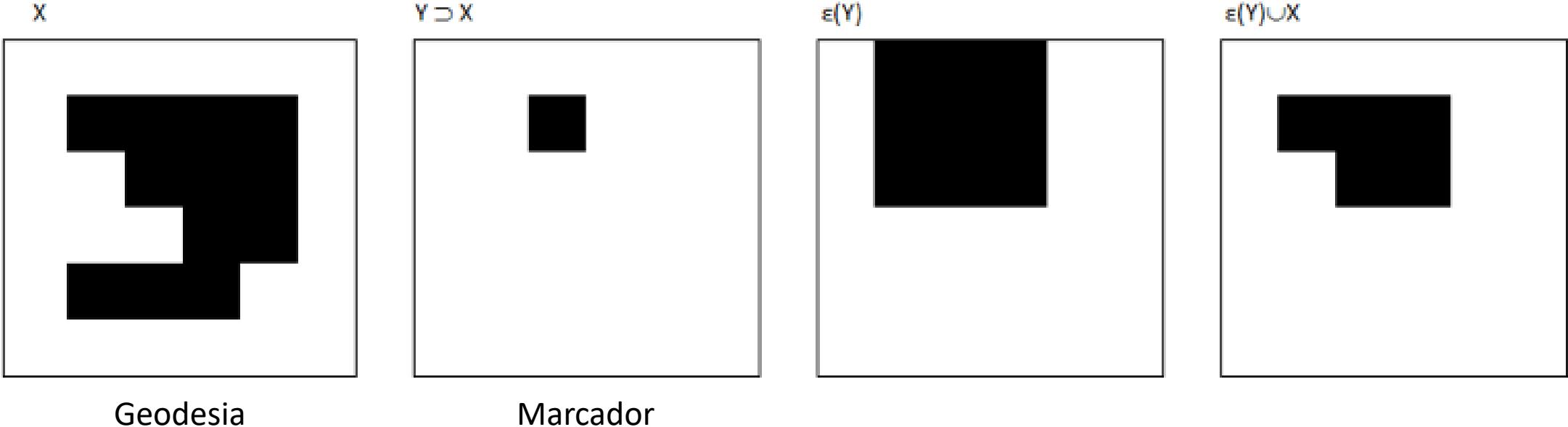
Marcador



# Transformações geodésicas binárias

**Erosão geodésica:** erosão morfológica de um conjunto  $Y$  condicionada à geodesia  $X$ .

$$\varepsilon_X(Y) = \varepsilon(Y) \cup X$$

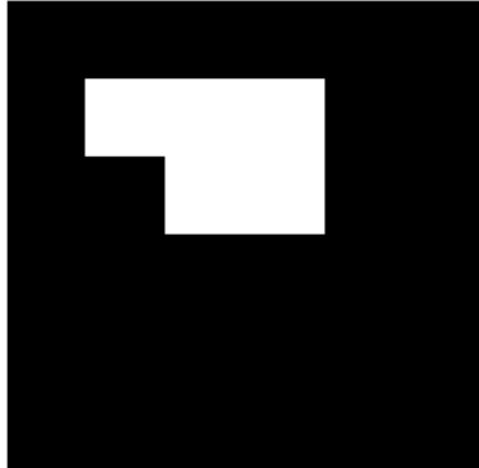


# Transformações geodésicas binárias

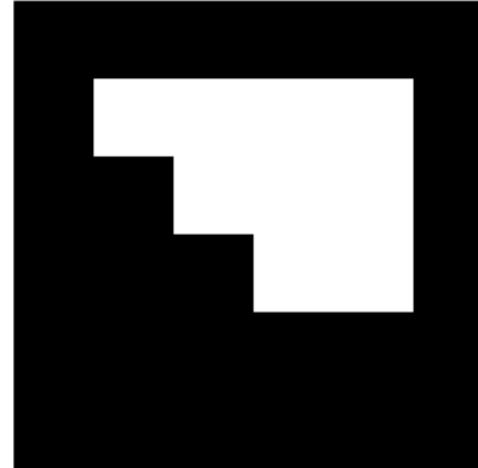
**Reconstrução geodésica binária por dilatações geodésicas sucessivas:**

$$R_X(Y) = \delta_X^\infty(Y) = \lim_{n \rightarrow \infty} (\delta_X \circ \dots \circ \delta_X)(Y) = \delta_X^n \left( \delta_X^{n-1} \left( \delta_X^{n-2} \left( \dots \left( \delta_X^1(Y) \right) \right) \right) \right)$$

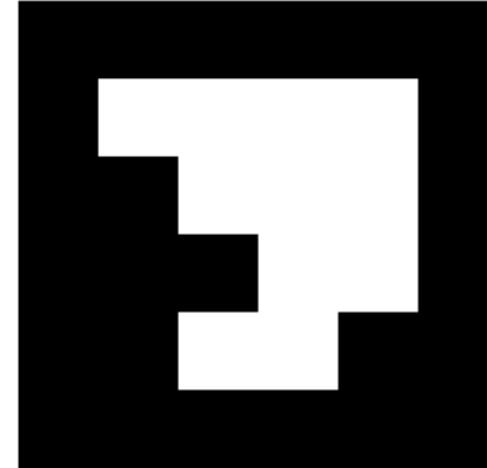
$\delta_1(Y) \cap X$



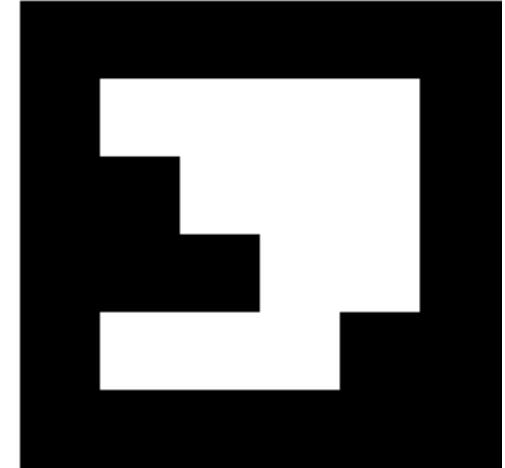
$\delta_2(Y) \cap X$



$\delta_3(Y) \cap X$



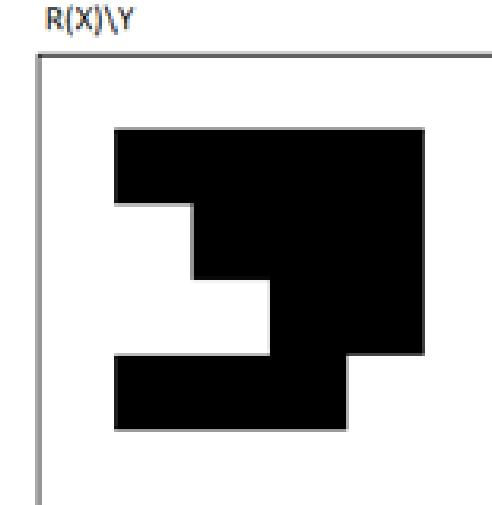
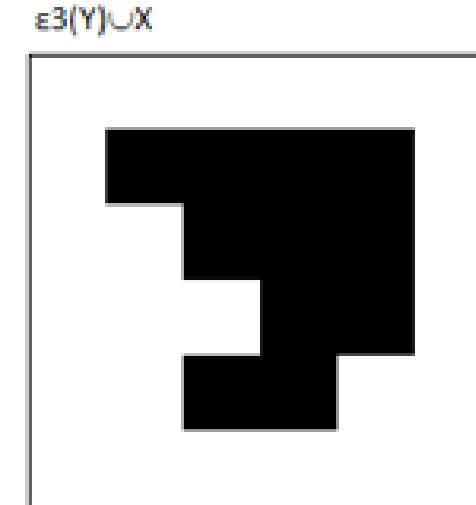
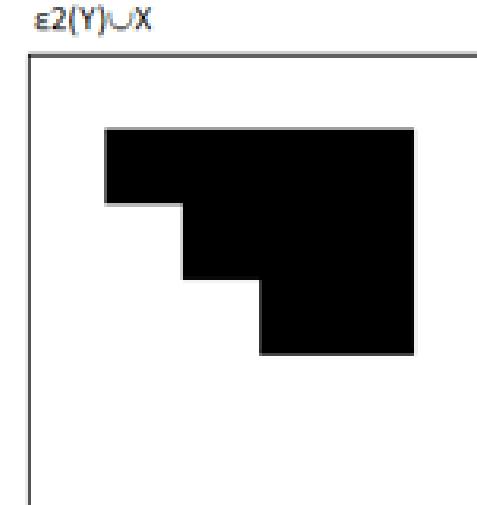
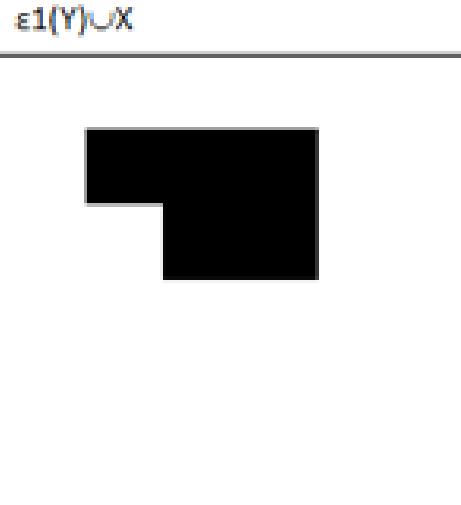
$R(X) \setminus Y$



# Transformações geodésicas binárias

**Reconstrução geodésica binária por erosões geodésicas sucessivas:**

$$R_X(Y) = \varepsilon_X^\infty(Y) = \lim_{n \rightarrow \infty} (\varepsilon_X \circ \dots \circ \varepsilon_X)(Y) = \varepsilon_X^n \left( \varepsilon_X^{n-1} \left( \varepsilon_X^{n-2} \left( \dots \left( \varepsilon_X^1(Y) \right) \right) \right) \right)$$

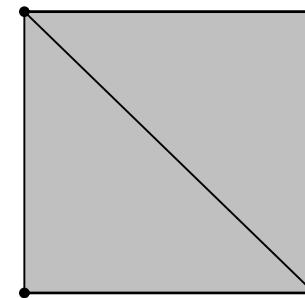
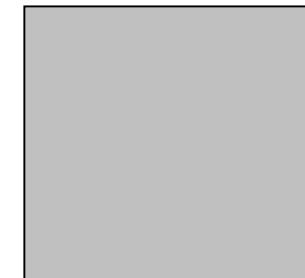


# Número de conectividade (número de Euler)

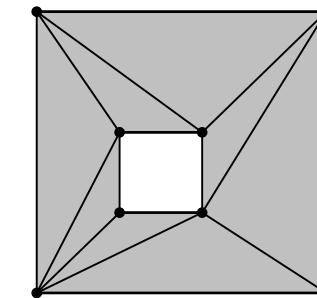
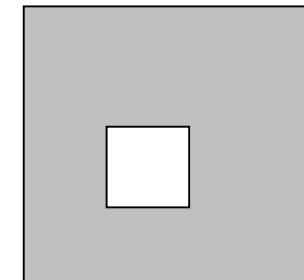
Fernando Soares

De forma geral, o **número de conexidade** de uma superfície ou conjunto, ou também chamado número de Euler ( $E$ ), é igual ao número de vértices  $v$ , menos o número de arestas  $a$ , mais o número de polígonos  $p$ , quando dividida a dita superfície em polígonos planos definidos pelas arestas e pelos vértices.

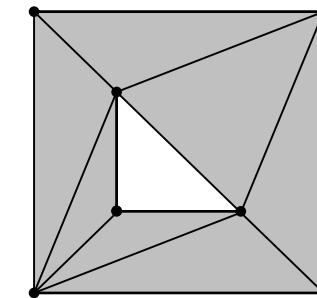
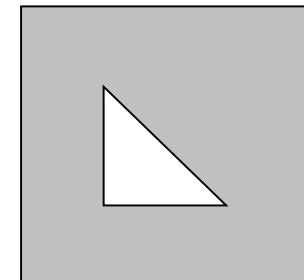
$$E = v - a + p$$



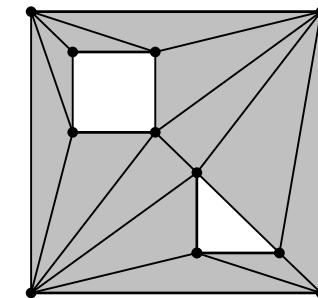
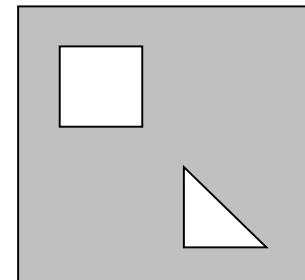
$$E = 4 - 5 + 2 = 1$$



$$E = 8 - 16 + 8 = 0$$



$$E = 7 - 14 + 7 = 0$$



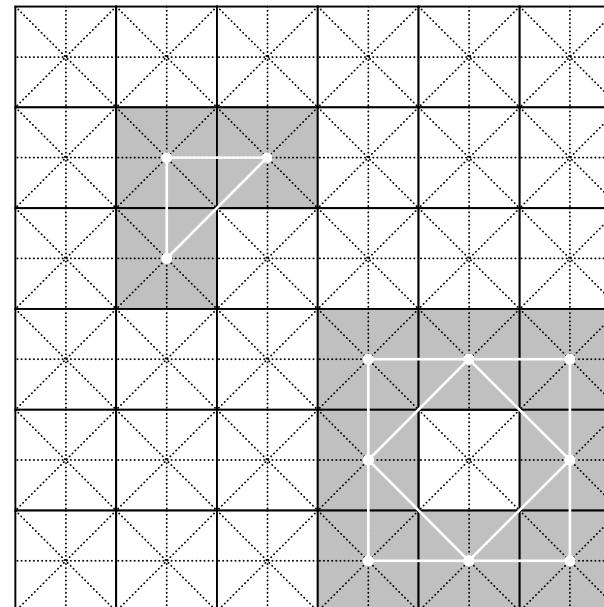
$$E = 11 - 25 + 13 = -1$$

# Número de conectividade (número de Euler)

Fernando Soares

No domínio de representação de uma imagem digital, o valor de E determina-se a partir do grafo de representação dos pixels. A seguir exemplifica-se a determinação de E para o objecto binário sombreado.

$$E = v - a + p$$



Número de pixels = 11

Número de arestas do grafo = 15

Número de triângulos = 5

$$E = 11 - 15 + 5 = 1$$

Número de objectos = 2

Número de buracos = 1

$$E = 2 - 1 = 1$$

# 5. Morfologia matemática numérica

Operações morfológicas elementares (erosão e dilatação). Operações morfológicas de abertura e fecho. Gradiente morfológico. Semi-gradientes. Operações de Chapéu-Alto e Chapéu-Baixo. Reconstrução geodésica numérica. Extremos regionais. Transformação Watershed.

# Introdução

Fernando Soares

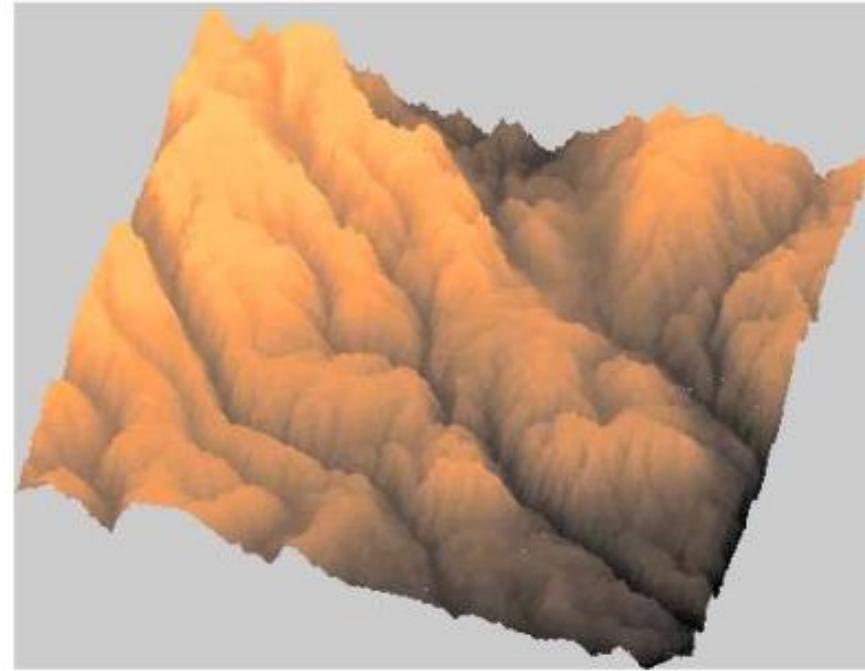
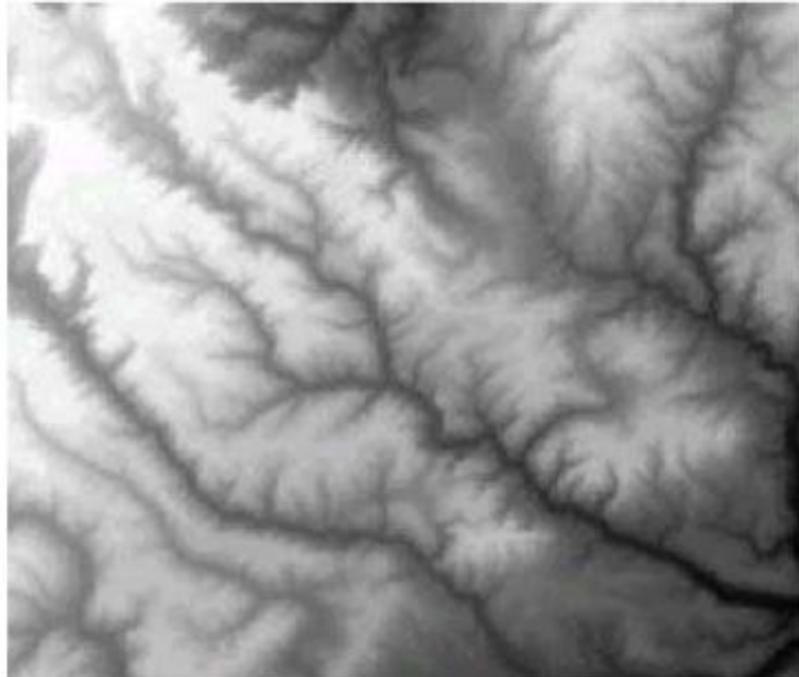
As operações elementares da morfologia matemática binária podem ser estendidas para as imagens de escala de cinzentos através do uso das operações de “**mínimo**” e “**máximo**”, que estabelecem um paralelo com as operações de erosão e dilatação binárias.

Estas operações atribuem a cada pixel da imagem novos valores correspondentes ao mínimo ou ao máximo valor de uma dada vizinhança em torno desse pixel. A vizinhança fica definida de acordo com a forma do elemento estruturante.

A morfologia matemática numérica tem aplicação em processos de contraste de imagem, descrição de texturas, detecção de fronteiras e limiarização, entre outras.

# Introdução

Na morfologia matemática as intensidades dos pixels das imagens numéricas são consideradas como elevações topográficas.

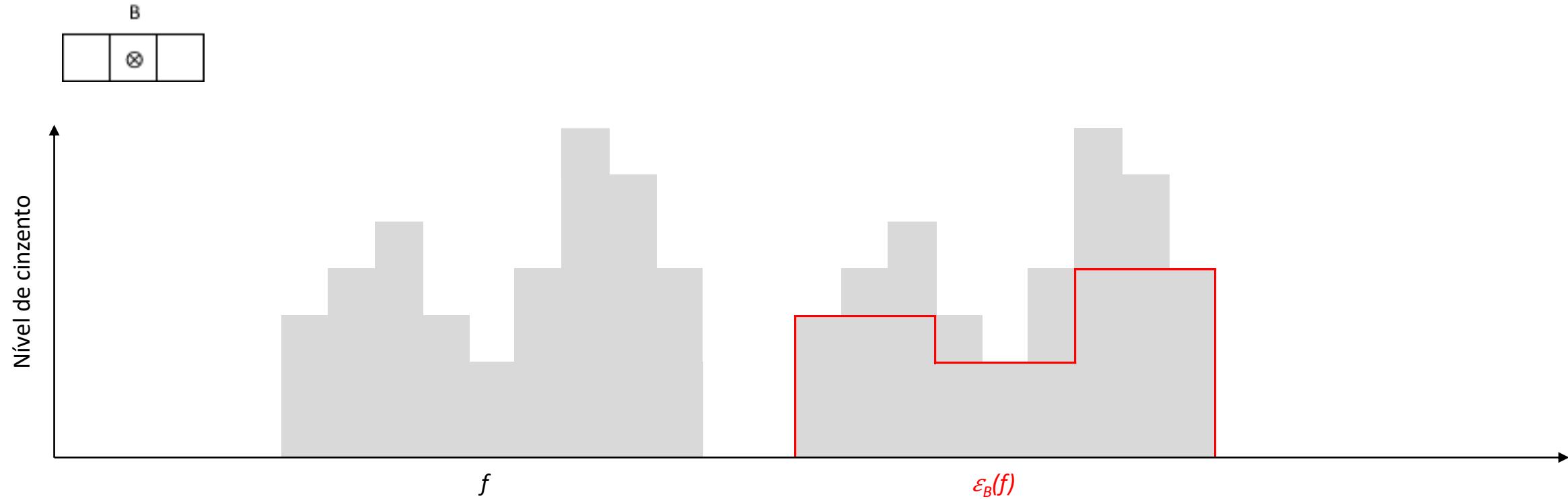


# Transformações morfológicas elementares

Fernando Soares

A operação de **erosão** ε de uma dada função  $f$ , por um elemento estruturante  $B$ , posicionado com a sua origem em  $x$  ( $B_x$ ), é dada pela expressão:

$$\varepsilon_B(f(x)) = \min_{b \in B} f(x + b)$$



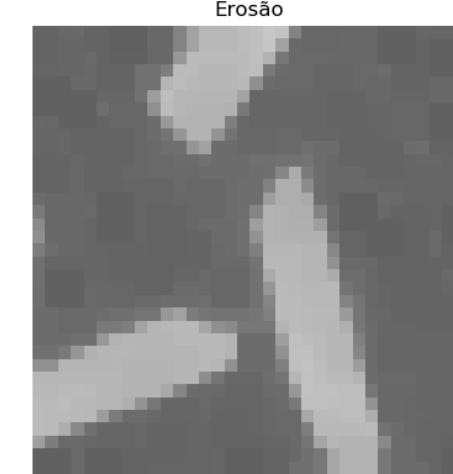
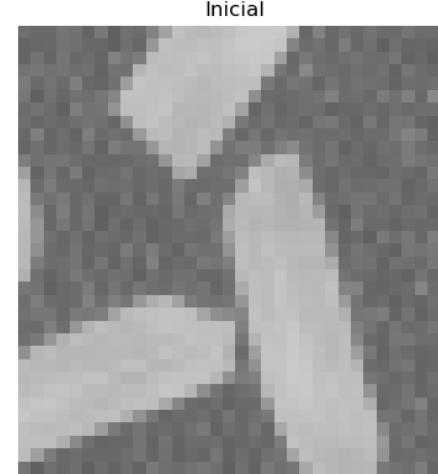
# Transformações morfológicas elementares

Fernando Soares

Exemplo 2D da erosão:

B

1	1	1
1	1	1
1	1	1



imagem

:	:	:	:	:		
..	15	8	18	6	11	..
..	16	5	21	2	0	..
..	22	14	3	20	19	..
..	4	10	7	1	24	..
..	13	12	17	23	9	..
:	:	:	:	:	:	

erosão

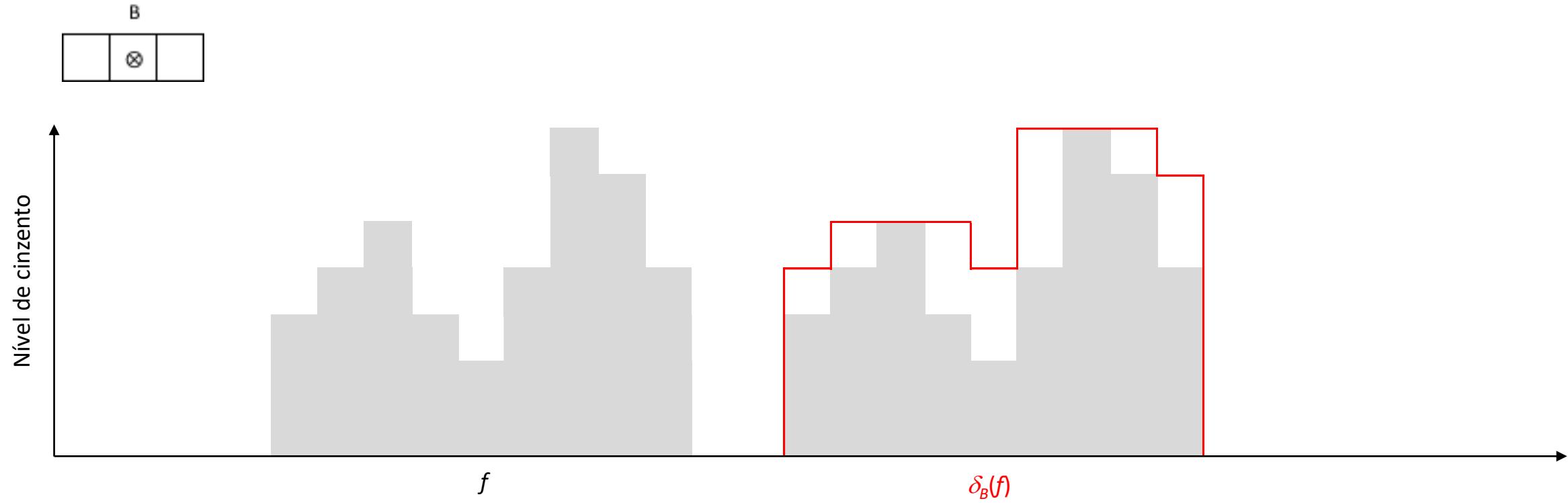
..	..	..	..	
..	3	2	0	..
..	3	1	0	..
..	3	1	1	..
..	..	..	..	..



# Transformações morfológicas elementares

A operação de **dilatação**  $\delta$  de uma dada função  $f$ , por um elemento estruturante  $B$ , posicionado com a sua origem em  $x$  ( $B_x$ ), é dada pela expressão:

$$\delta_B(f(x)) = \max_{b \in B} f(x + b)$$

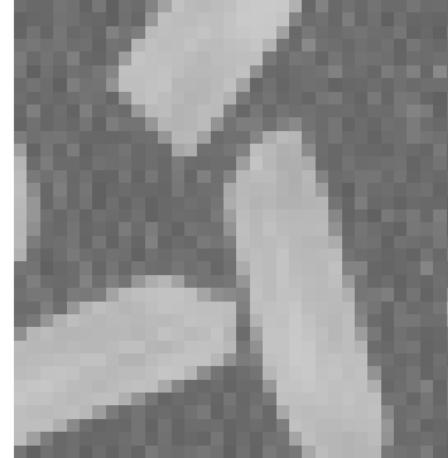


# Transformações morfológicas elementares

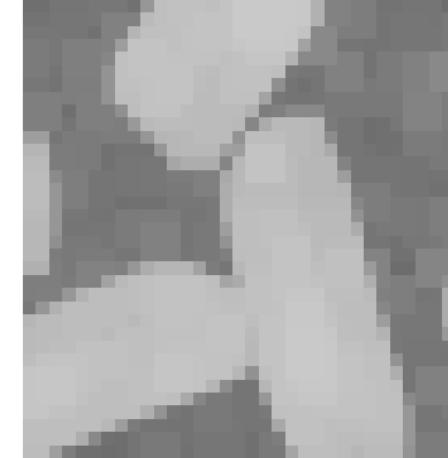
Exemplo 2D da dilatação:

B
1 1 1
1 1 1
1 1 1

Inicial



Dilatação



imagem

:	:	:	:	:	:	
..	15	8	18	6	11	..
..	16	5	21	2	0	..
..	22	14	3	20	19	..
..	4	10	7	1	24	..
..	13	12	17	23	9	..
:	:	:	:	:	:	

dilatação

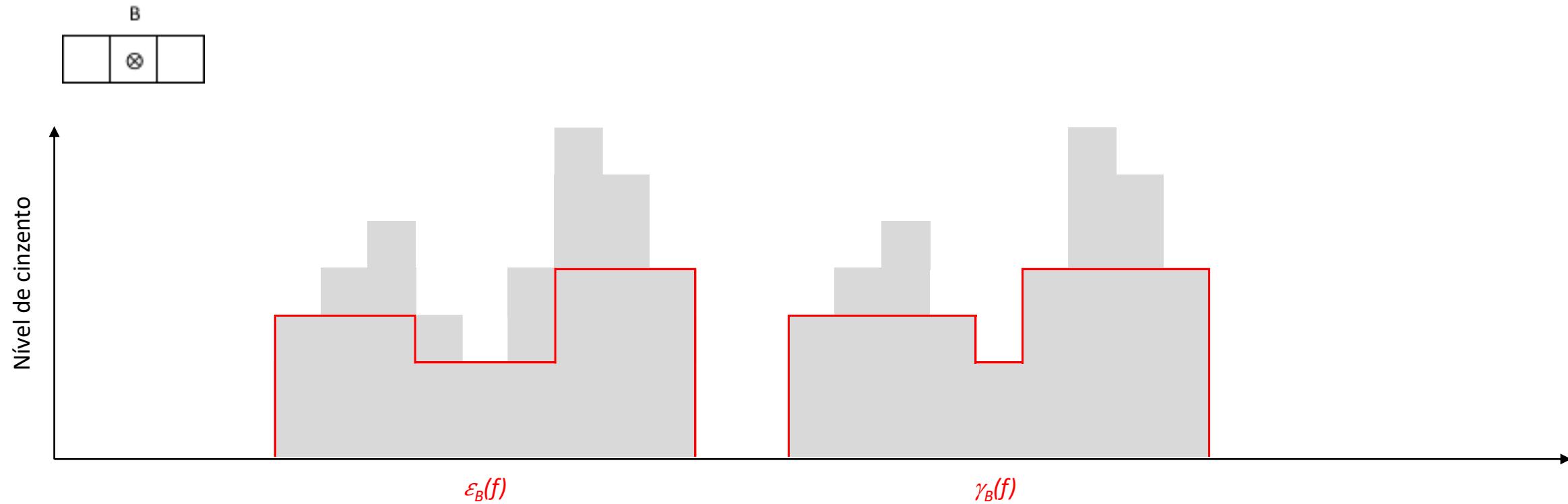
:	:	:	:	:	:	
..						..
..	24	21	21			..
..	22	21	24			..
..	22	23	24			..
:	:	:	:	:	:	



# Transformações morfológicas elementares

A operação de **abertura**  $\gamma$  de uma dada função  $f$ , por um elemento estruturante  $B$ , posicionado com a sua origem em  $x$  ( $B_x$ ), é dada pela expressão:

$$\gamma_B(f(x)) = \delta_B(\varepsilon_B(f(x)))$$



# Transformações morfológicas elementares

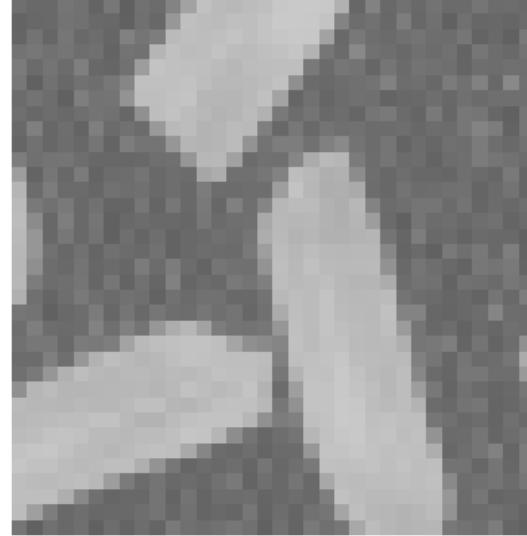
Fernando Soares

Exemplo 2D da abertura:

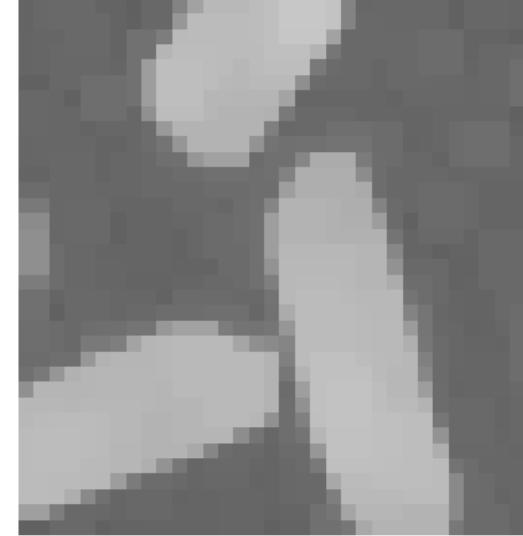
B

1	1	1
1	1	1
1	1	1

Inicial



Abertura



imagem

:	:	:	:	:
..	15	8	18	6
..	16	5	21	2
..	22	14	3	20
..	4	10	7	1
..	13	12	17	23
:	:	:	:	:

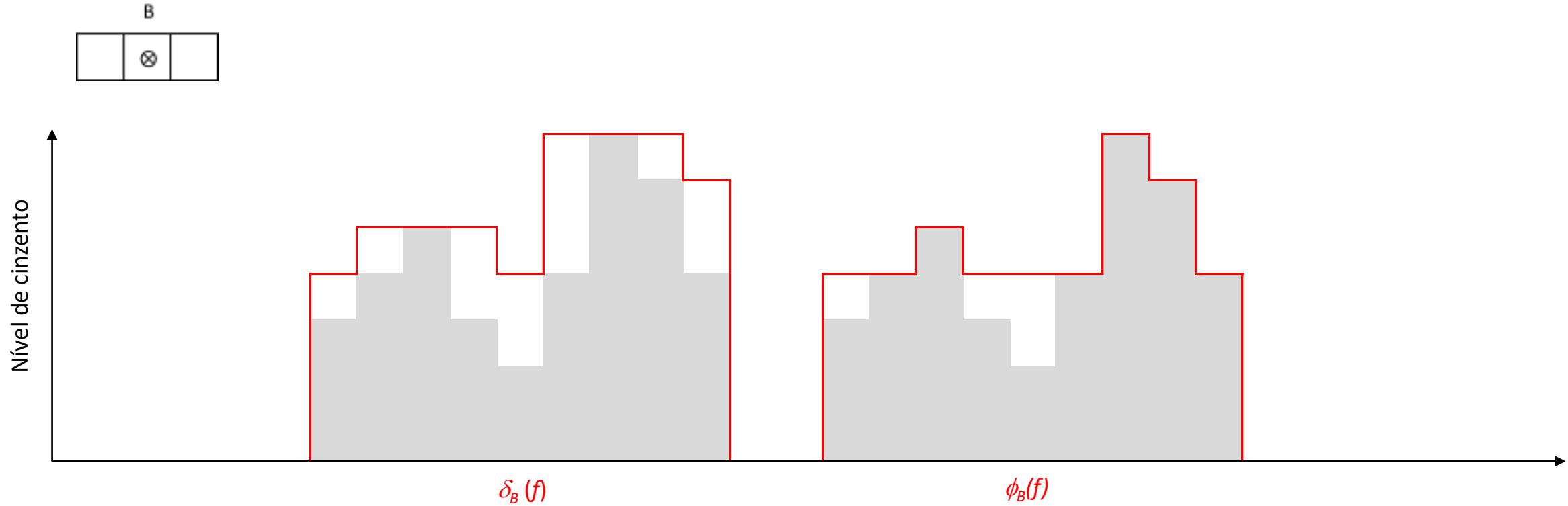
abertura

:	:	:	:	:
..				..
..		3		..
..				..
..				..
:	:	:	:	:

# Transformações morfológicas elementares

A operação de **fecho**  $\phi$  de uma dada função  $f$ , por um elemento estruturante  $B$ , posicionado com a sua origem em  $x$  ( $B_x$ ), é dada pela expressão:

$$\phi_B(f(x)) = \varepsilon_B(\delta_B(f(x)))$$



# Transformações morfológicas elementares

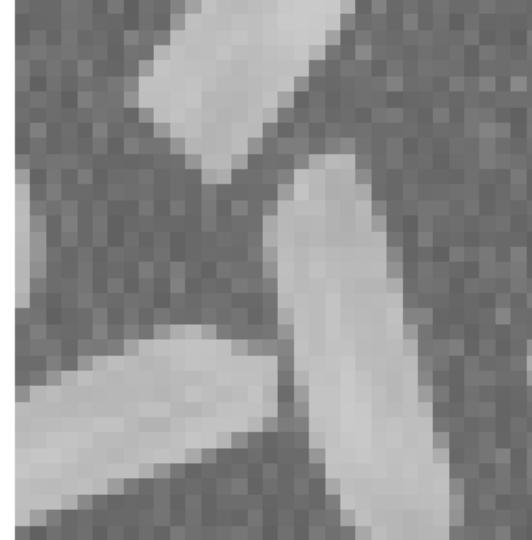
Fernando Soares

Exemplo 2D do fecho:

B

1	1	1
1	1	1
1	1	1

Inicial



imagem

:	:	:	:	:	:	
..	15	8	18	6	11	..
..	16	5	21	2	0	..
..	22	14	3	20	19	..
..	4	10	7	1	24	..
..	13	12	17	23	9	..
:	:	:	:	:	:	

Fecho



fecho

:	:	:	:	:	:
..					..
..					..
..		21			..
..					..
..					..
:	:	:	:	:	:

# Suavização morfológica

Fernando Soares

A suavização de uma imagem por métodos morfológicos pode ser obtida com várias abordagens. Uma delas consiste em executar uma operação de abertura, seguida de um fecho. Desta forma removem-se artefactos claros e escuros com tamanho igual ou abaixo do tamanho do elemento estruturante.

Inicial

 $f$ 

Suavização 1

 $\phi_B(\gamma_B(f))$

# Suavização morfológica

Fernando Soares

Uma segunda abordagem de suavização consiste em executar a média entre as operações de erosão e dilatação de uma imagem.

Inicial

 $f$ 

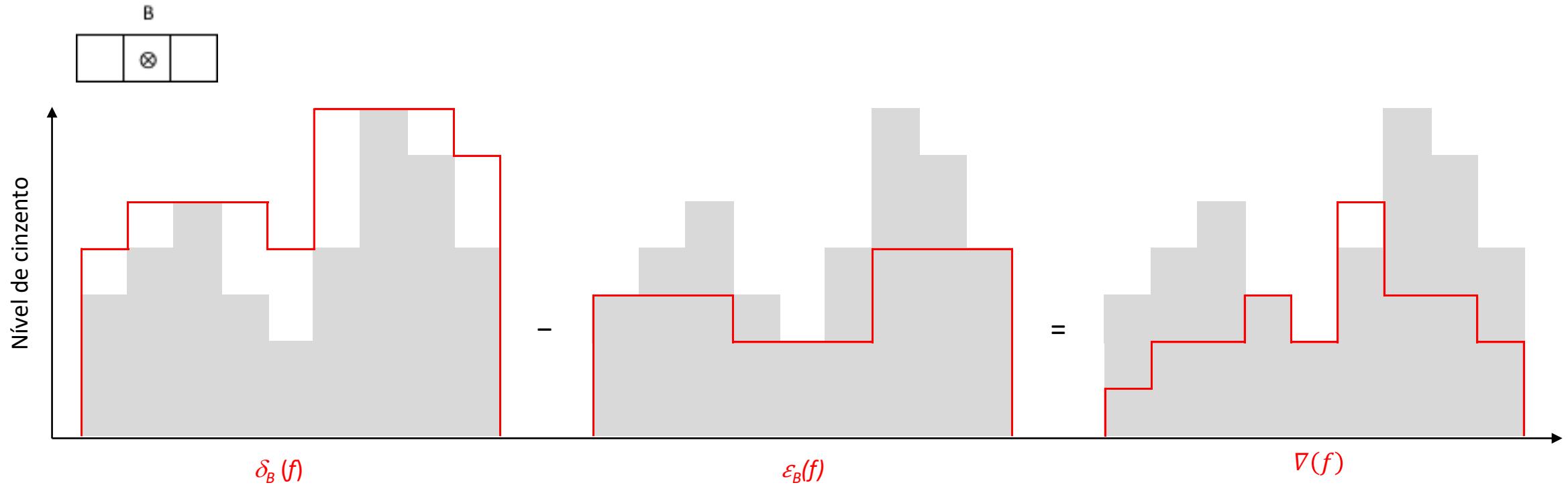
Suavização 2

 $0.5 \times [\varepsilon_B(f) + \delta_B(f)]$

# Gradiente morfológico

O **gradiente morfológico**  $\nabla$  (também designado por gradiente de Beucher) determina-se, em cada pixel, pela diferença algébrica entre a dilatação  $\delta(f)$  e a erosão  $\varepsilon(f)$ .

$$\nabla(f) = \delta_B(f) - \varepsilon_B(f)$$



# Gradiente morfológico

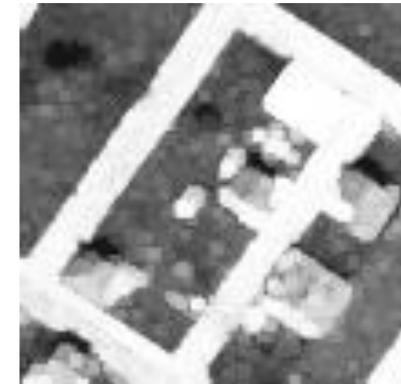
Exemplo 2D do gradiente morfológico:

B

1	1	1
1	1	1
1	1	1

imagem

:	:	:	:	:	:	
..	15	8	18	6	11	..
..	16	5	21	2	0	..
..	22	14	3	20	19	..
..	4	10	7	1	24	..
..	13	12	17	23	9	..
:	:	:	:	:	:	

 $f$  $e(f)$  $\delta(f)$  $\nabla(f) \equiv \delta(f) - e(f)$ 

gradiente morfológico

:	:	:	:	:	:
..					..
..	21	19	21		..
..	19	20	24		..
..	21	22	23		..
..					..
:	:	:	:	:	:

# Gradiente morfológico

Fernando Soares

A assunção comum relativa ao gradiente é a de que as fronteiras dos objectos, ou arestas estão localizadas onde se verificam diferenças elevadas entre os valores de pixels vizinhos. Os operadores de gradiente são usados para evidenciar essas variações.

A espessura de uma aresta detectada por um gradiente morfológico é igual a dois pixels (considerando um elemento estruturante elementar, ou seja, 3x3): um pixel em cada lado da fronteira.

Havendo ruído aleatório, a imagem deve ser filtrada antes de aplicar o operador de gradiente, para evitar realçar também o ruído. Ao contrário dos operadores de gradiente lineares de *Sobel*, *Prewitt*, ou *Roberts*, os gradientes morfológicos obtidos com elementos estruturantes simétricos tendem a depender menos da direccionalidade das arestas dos objectos.

Os **semi-gradientes** podem ser usados para detectar os limites interno ou externo de uma fronteira. Os semi-gradientes têm só um pixel de espessura.

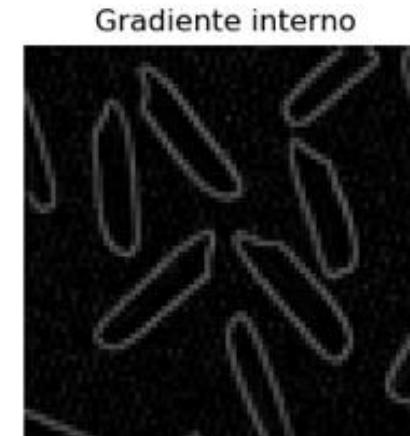
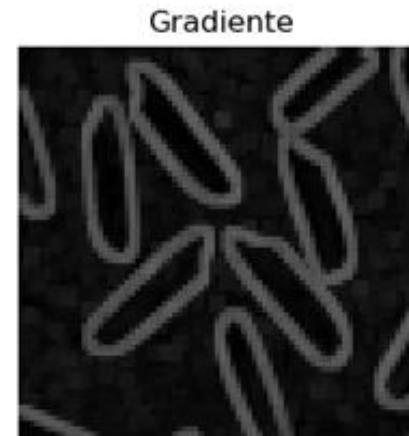
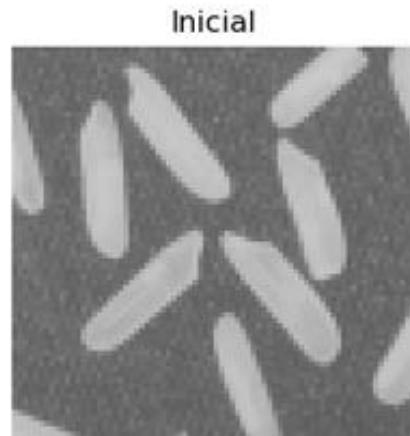
# Semi-gradientes

O **semi-gradiente por erosão**, ou **gradiente interno**  $\nabla^-$ , define-se como sendo a diferença entre a imagem original e a sua erosão numérica.

$$\nabla_B^-(f) = f - \varepsilon_B(f)$$

O gradiente interno evidencia:

- as fronteiras internas dos objectos que são mais claros que o fundo.
- as fronteiras externas dos objectos que são mais escuros que o fundo.



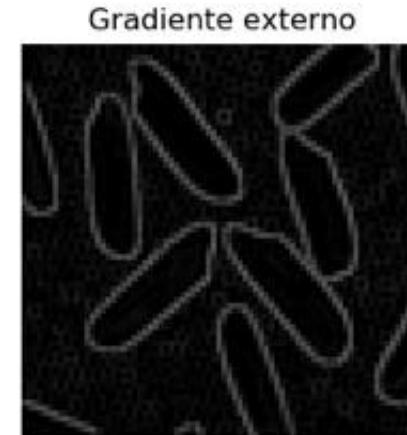
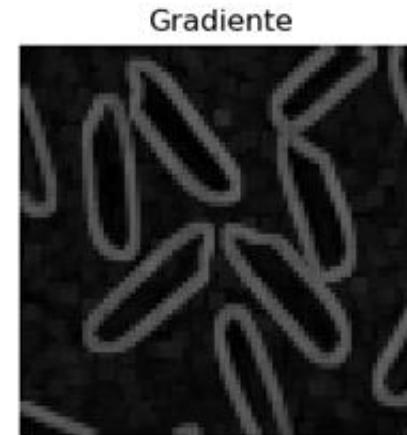
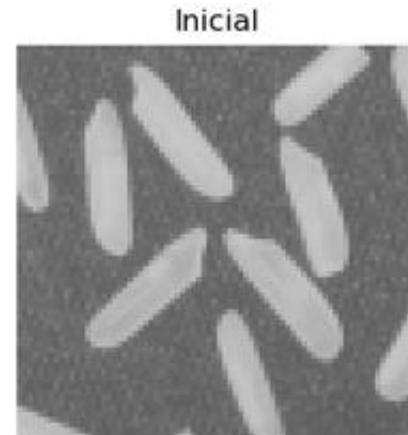
# Semi-gradientes

O **semi-gradiente por dilatação**, ou **gradiente externo**  $\nabla^+$ , define-se como sendo a diferença entre a dilatação numérica de uma imagem e a sua representação original.

$$\nabla_B^+(f) = \delta_B(f) - f$$

O gradiente externo evidencia.

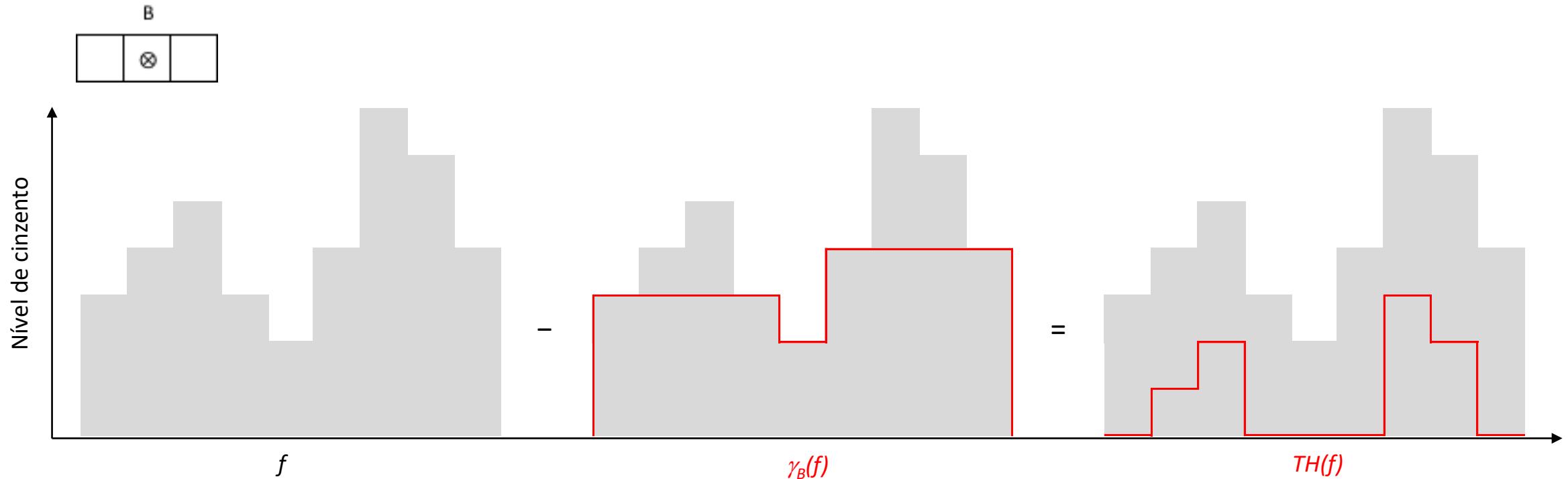
- as fronteiras internas dos objectos que são mais escuros que o fundo.
- as fronteiras externas dos objectos que são mais claros que o fundo.



# Transformação Chapéu-Alto

**Chapéu-Alto (Top-Hat):** consiste na diferença algébrica entre a função (imagem) e a sua abertura.

$$TH(f) = f - \gamma_B(f)$$



# Transformação Chapéu-Alto

São extraídos os picos de intensidade da imagem. Todas as estruturas “em pico” que não estão contidas no elemento estruturante são removidas da imagem. Realça os detalhes da imagem. É um exemplo de como é mais simples agir sobre as estruturas relevantes, em vez de suprimir directamente os objectos mais irrelevantes.

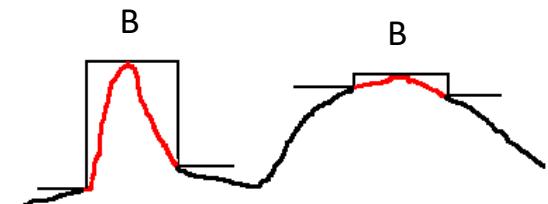
Inicial

 $f$ 

Top-Hat



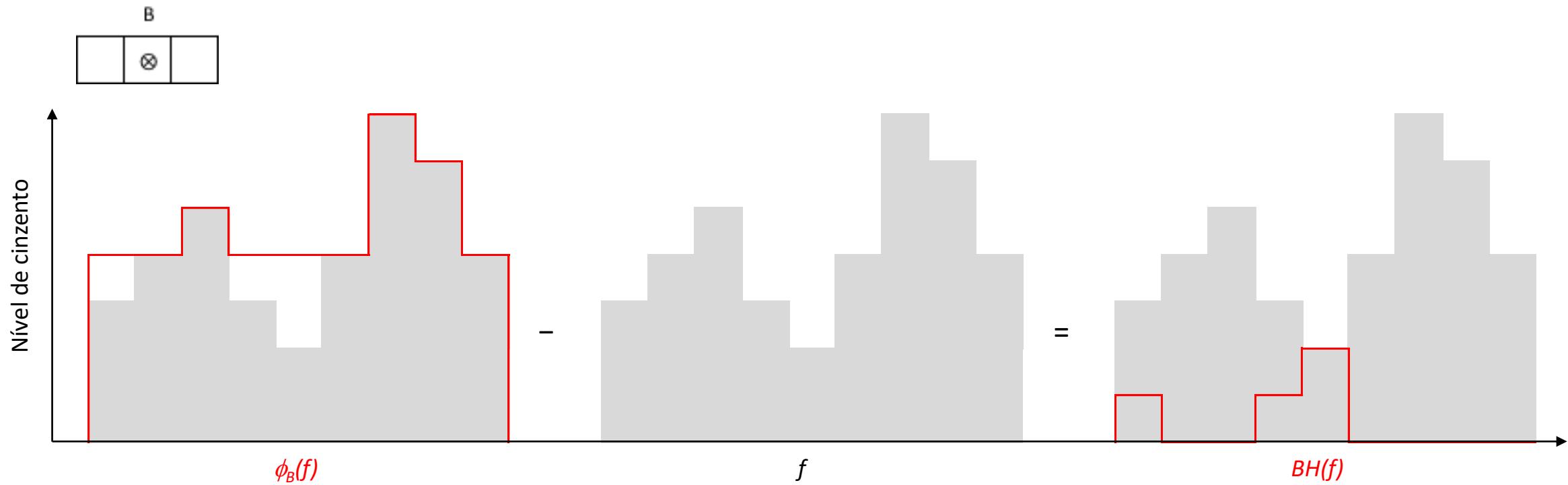
$$TH(f) = f - \gamma_B(f)$$



# Transformação Chapéu-Baixo

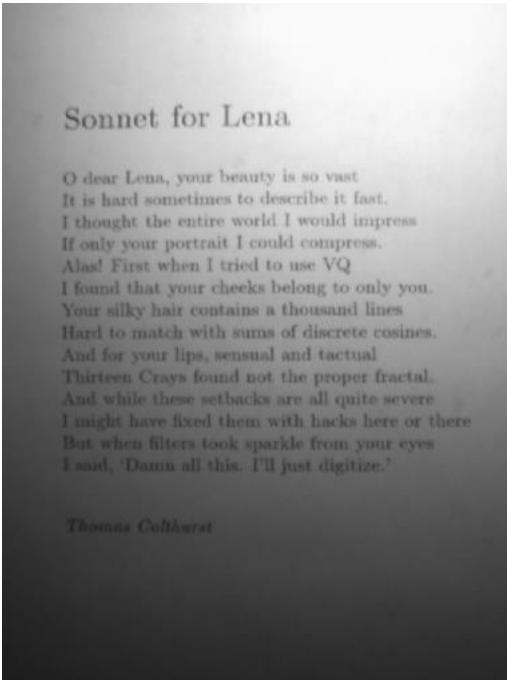
**Chapéu-Baixo (*Bottom-Hat*):** consiste na diferença algébrica entre o fecho da imagem e a imagem inicial.

$$BH(f) = \phi_B(f) - f$$

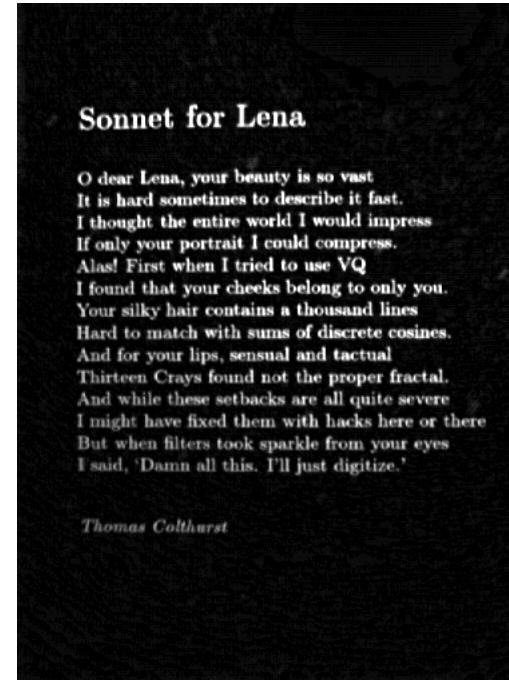


# Transformação Chapéu-Baixo

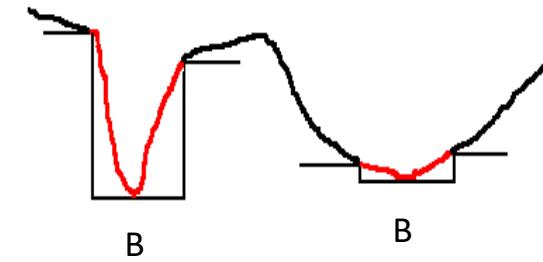
São extraídos os “vales” de intensidade da imagem. Todas as estruturas “em vale” que não estão contidas no elemento estruturante são removidas da imagem.



$f$



$$BH(f) = \phi_B(f) - f$$



# Chapéu-Alto auto complementar

Fernando Soares

A soma do *Top-Hat* com o *Bottom-Hat* extrai todos os objectos da imagem que não estão contidas no elemento estruturante dado, quaisquer que sejam os seus contrastes relativos (ou seja, picos ou vales).

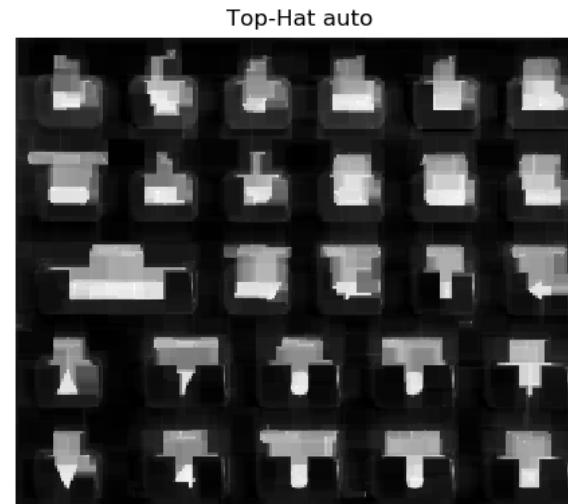
A partir das respectivas expressões deduz-se facilmente que,

$$TH(f) + BH(f) = f - \gamma_B(f) + \phi_B(f) - f = \phi_B(f) - \gamma_B(f)$$

Ou seja, o *Top-Hat* auto-complementar é dado pela **diferença entre o fecho e a abertura**.

# Chapéu-Alto auto complementar

Exemplo com elemento estruturante  $3 \times 3$  e  $9 \times 9$ :



# Chapéu-Alto e contraste de imagem

Fernando Soares

Um simples operador morfológico de contraste pode ser obtido com a adição, à imagem inicial, da diferença de ambos os operadores de *Top-Hat* e *Bottom-Hat*. Este procedimento consiste em começar por realçar os objectos mais claros (adição do *Top-Hat*), seguido do realce dos objectos mais escuros (subtracção do *Bottom-Hat*).

$$f + TH(f) - BH(f) = f + f - \gamma_B(f) - \phi_B(f) + f$$

Os valores resultantes que fiquem de fora do intervalo dinâmico da imagem inicial [ $z_{\min}$ ;  $z_{\max}$ ] terão os valores  $z_{\min}$ , ou  $z_{\max}$ , consoante fiquem abaixo ou acima dos extremos do intervalo ( $z_{\min} = 0$  e  $z_{\max} = 255$ , para imagens de 8 bits).

# Chapéu-Alto e contraste de imagem

Fernando Soares

Exemplo:

Inicial



Top-Hat realce



# Filtragem sequencial alternada

Fernando Soares

A filtragem de uma imagem que contenha ruídos escuro e claro pode ser obtida através da aplicação de uma sequência de operações de fecho-abertura ou abertura-fecho.

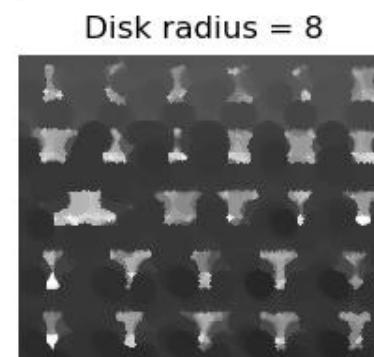
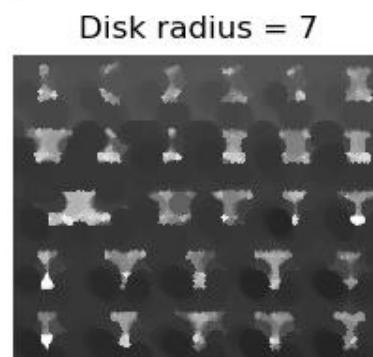
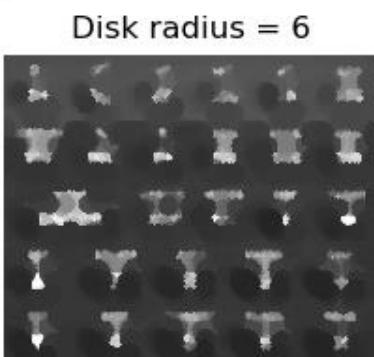
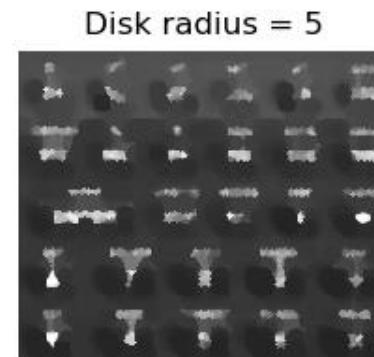
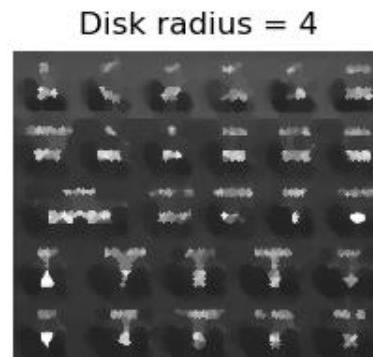
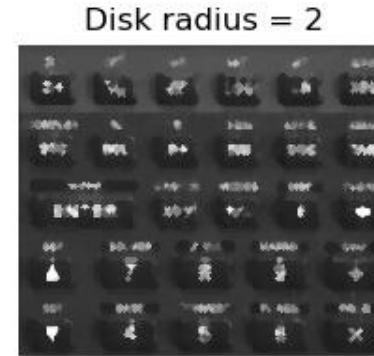
Uma só operação sequencial de fecho-abertura, ou abertura-fecho, com um elemento estruturante grande não produz resultados aceitáveis

Uma solução para este problema consiste em aplicar fechos e aberturas alternados, começando com elementos estruturantes de baixas dimensões e ir aumentando progressivamente o seu tamanho até um dado tamanho final.

A esta processo de filtragem por aplicação sequencial de fecho-abertura, ou abertura-fecho, dá-se o nome de **filtragem sequencial alternada**.

# Filtragem sequencial alternada

Exemplo:



# Transformações geodésicas numéricas

Fernando Soares

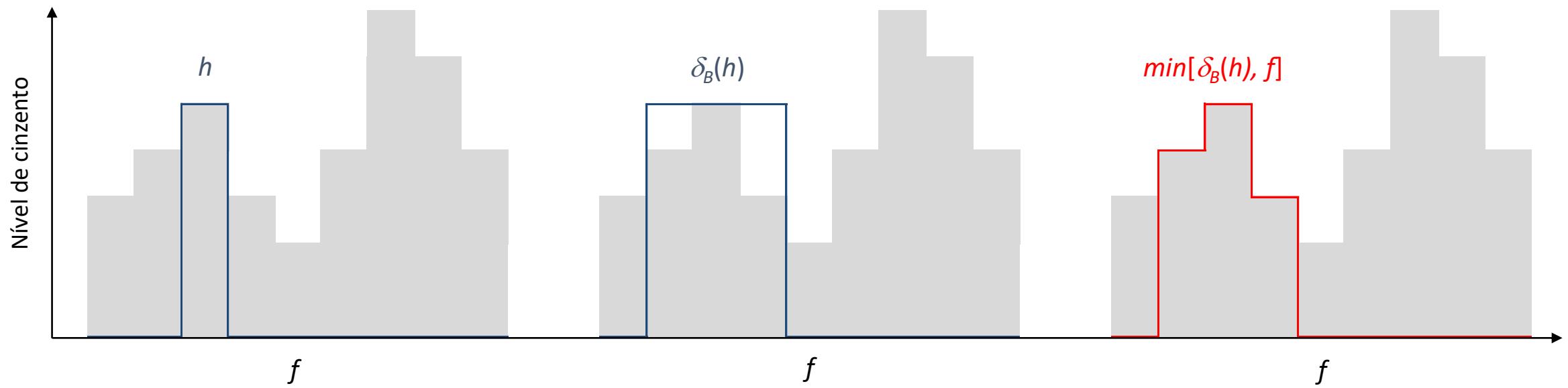
**Transformações geodésicas numéricas** são transformações morfológicas aplicadas a uma imagem numérica  $h$ , mas condicionadas pela morfologia de uma outra imagem numérica  $f$ . Destacam-se as seguintes:

- 1 - Dilatação geodésica
- 2 - Erosão geodésica
- 3 - Reconstrução geodésica numérica por dilatações geodésicas sucessivas.
- 4 - Reconstrução geodésica numérica por erosões geodésicas sucessivas (reconstrução dual)

# Transformações geodésicas numéricas

**Dilatação geodésica:** consiste em determinar o valor mínimo entre a dilatação da imagem marcadora  $h$  ( $\leq f$ ) e a função  $f$ .

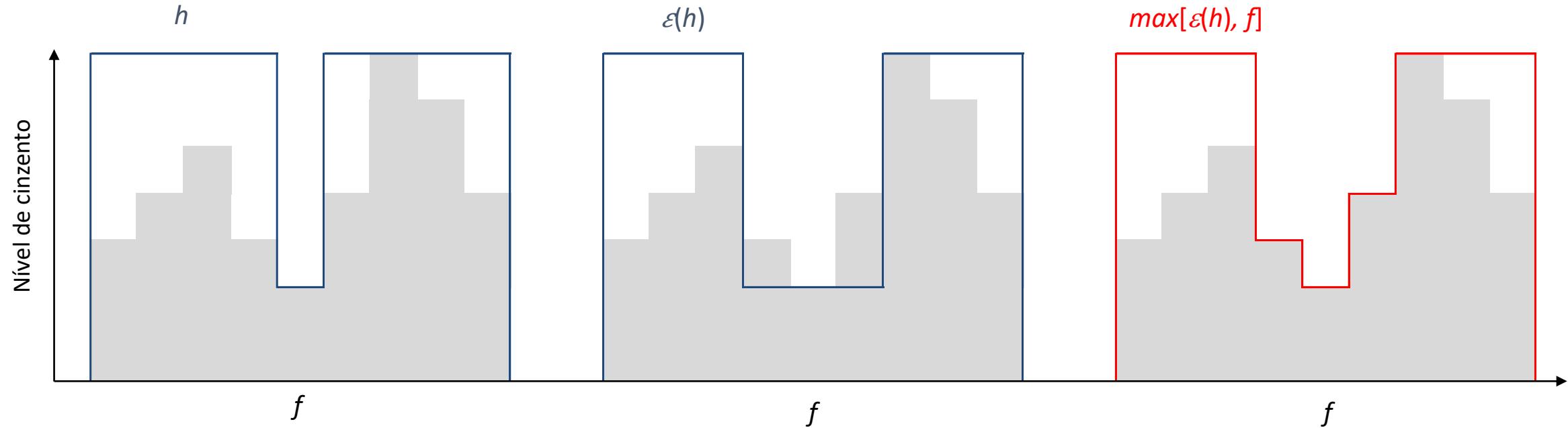
$$\delta_f(h) = \min[\delta_B(h(x, y)); f(x, y)]$$



# Transformações geodésicas numéricas

**Erosão geodésica:** consiste em determinar o máximo entre a erosão da imagem marcadora  $h$  ( $\geq f$ ) e a função  $f$ .

$$\varepsilon_f(h) = \max[\varepsilon_B(h(x, y)); f(x, y)]$$



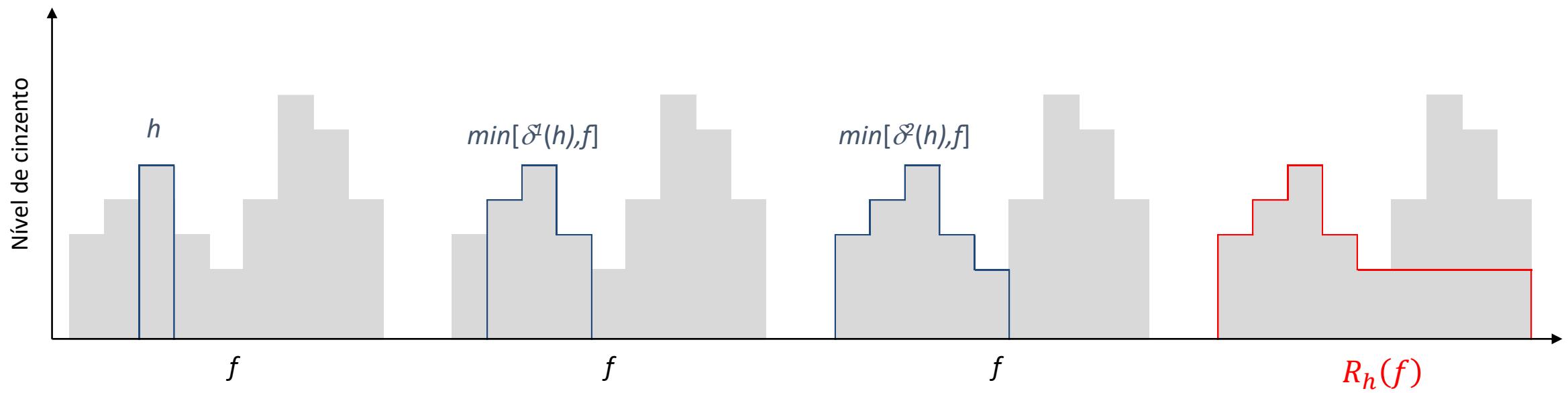
# Transformações geodésicas numéricas

Reconstrução geodésica por dilatações geodésicas sucessivas.

$$R_h(f) = \delta_f^{(i)}(h)$$

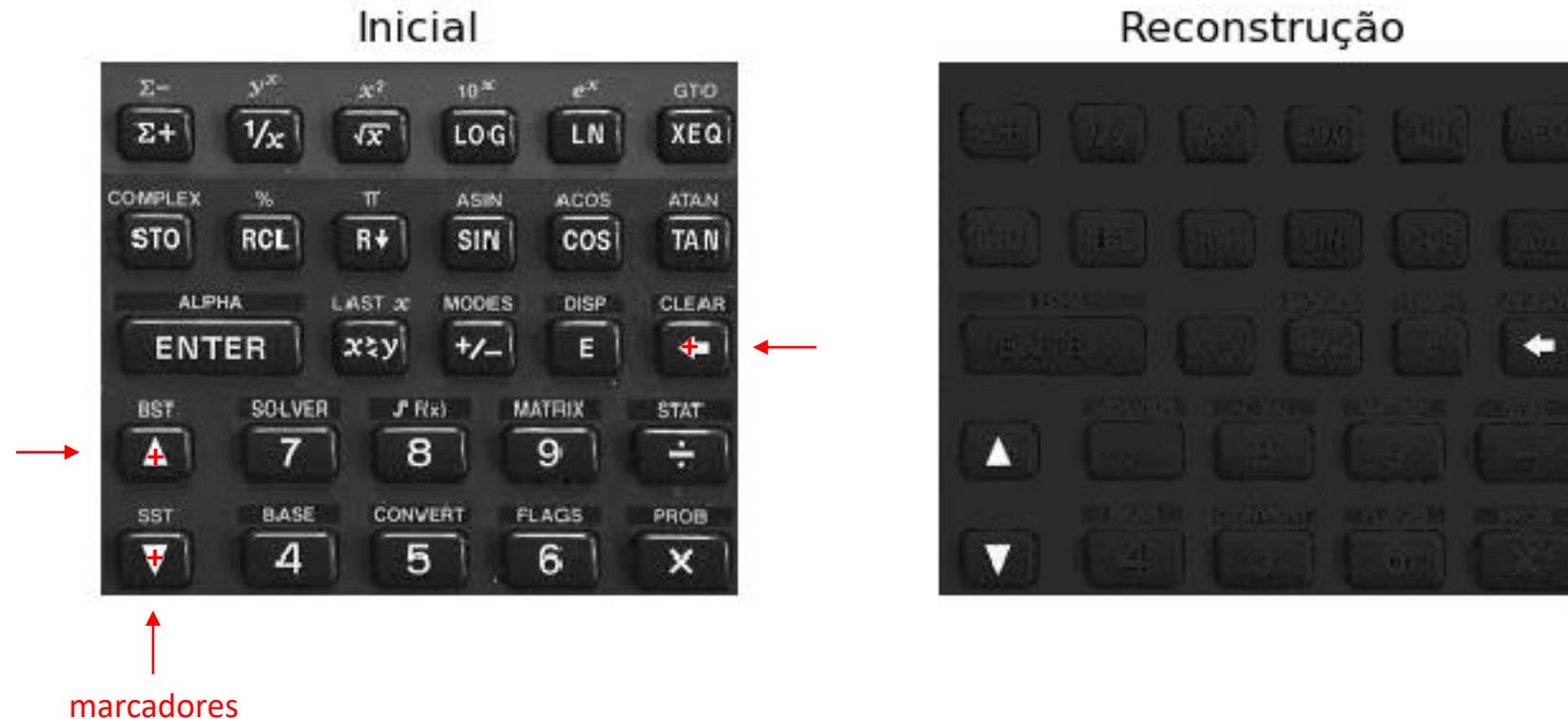
, até se verificar a condição

$$\delta_f^{(i+1)}(h) = \delta_f^{(i)}(h)$$



# Transformações geodésicas numéricas

Exemplo da reconstrução geodésica por dilatações geodésicas sucessivas.



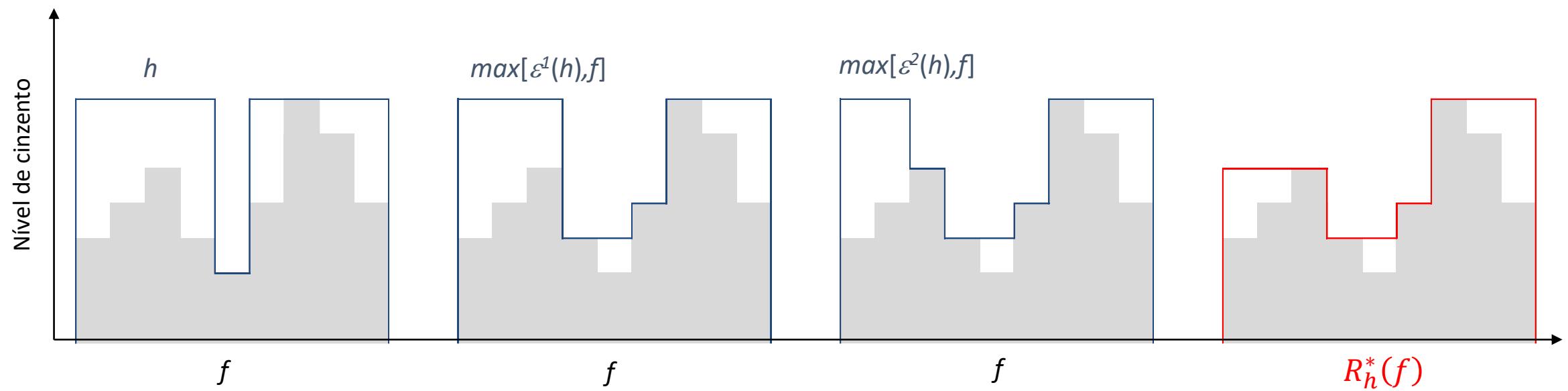
# Transformações geodésicas numéricas

**Reconstrução geodésica** numérica por erosões geodésicas sucessivas (Reconstrução Dual).

$$R_h^*(f) = \varepsilon_f^{(i)}(h)$$

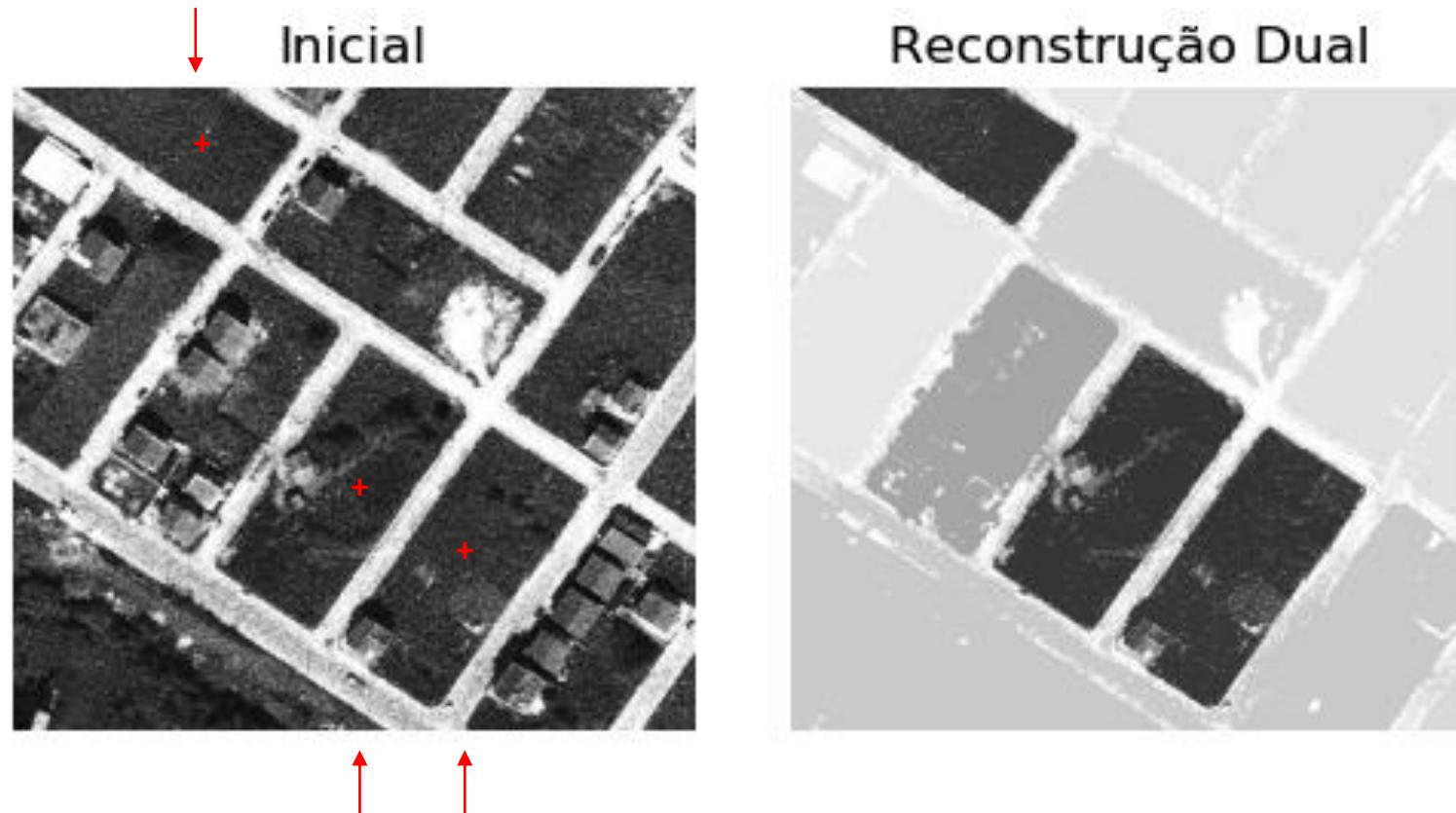
, até se verificar a condição

$$\varepsilon_f^{(i+1)}(h) = \varepsilon_f^{(i)}(h)$$



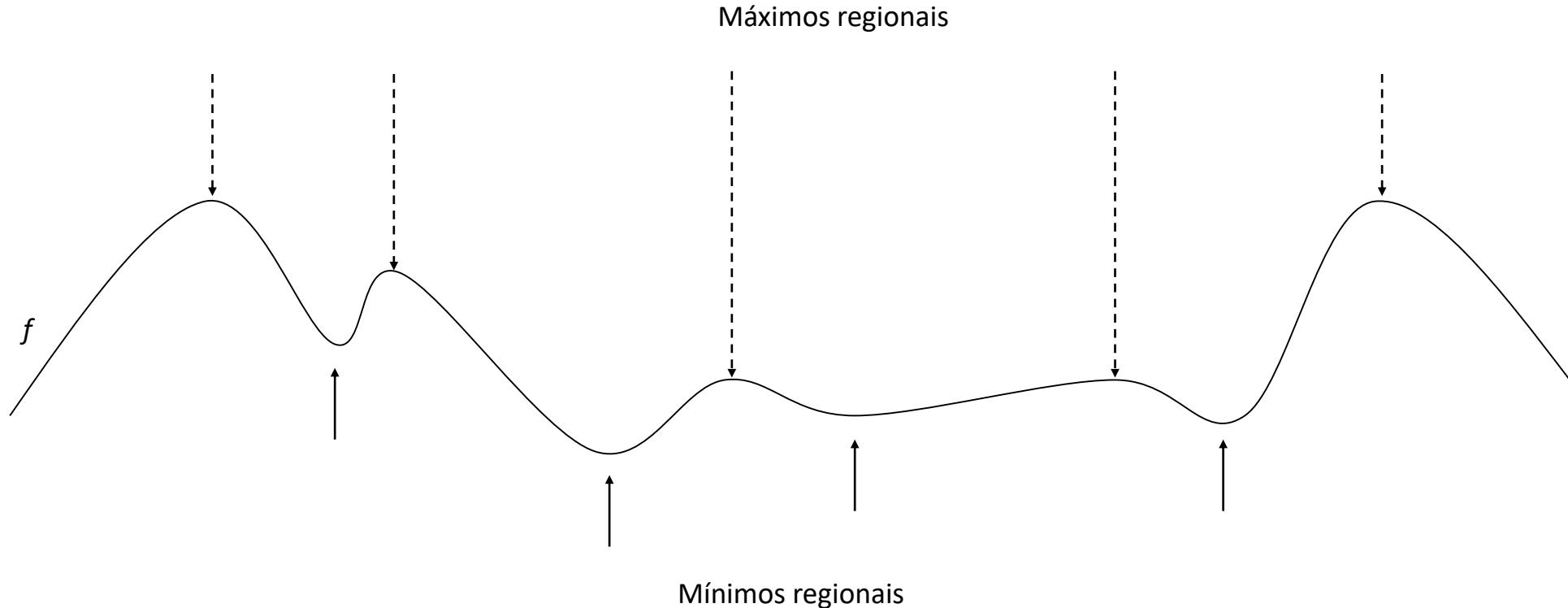
# Transformações geodésicas numéricas

Exemplo da reconstrução geodésica numérica por erosões geodésicas sucessivas (Reconstrução Dual).



# Extremos regionais

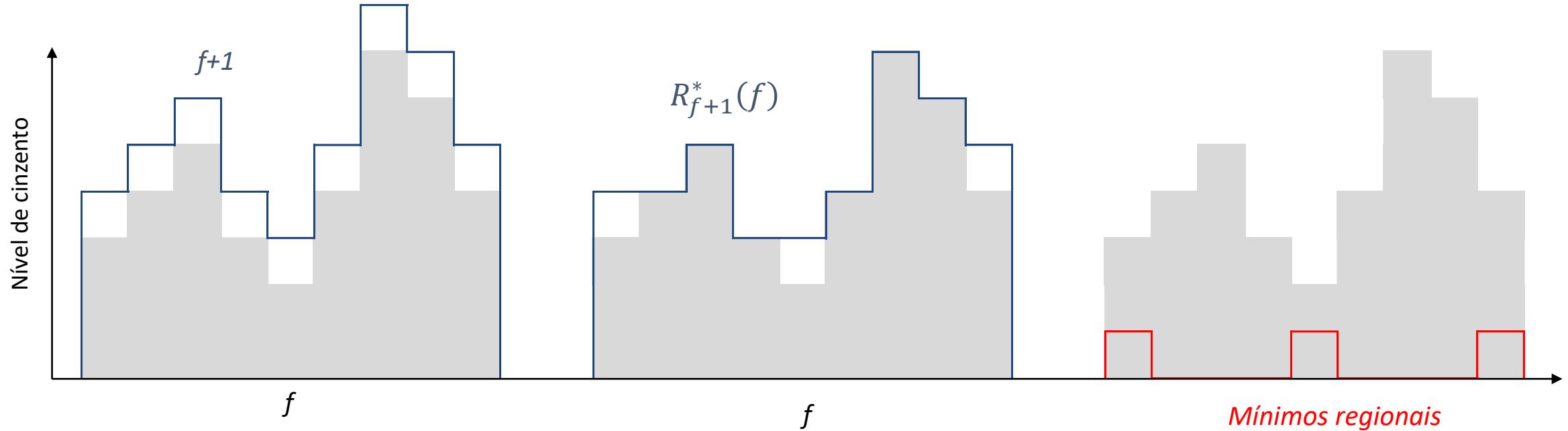
Uma imagem de mínimos/máximos regionais é, geralmente, uma imagem binária.



# Extremos regionais

Um conjunto binário  $M_h$  de uma imagem numérica  $f$  é um **mínimo regional** de elevação  $h$  se e só se  $M_h$  for uma superfície conexa de igual altitude  $h$ , a partir da qual seja impossível alcançar um ponto de elevação inferior sem ter que antes ascender na função.

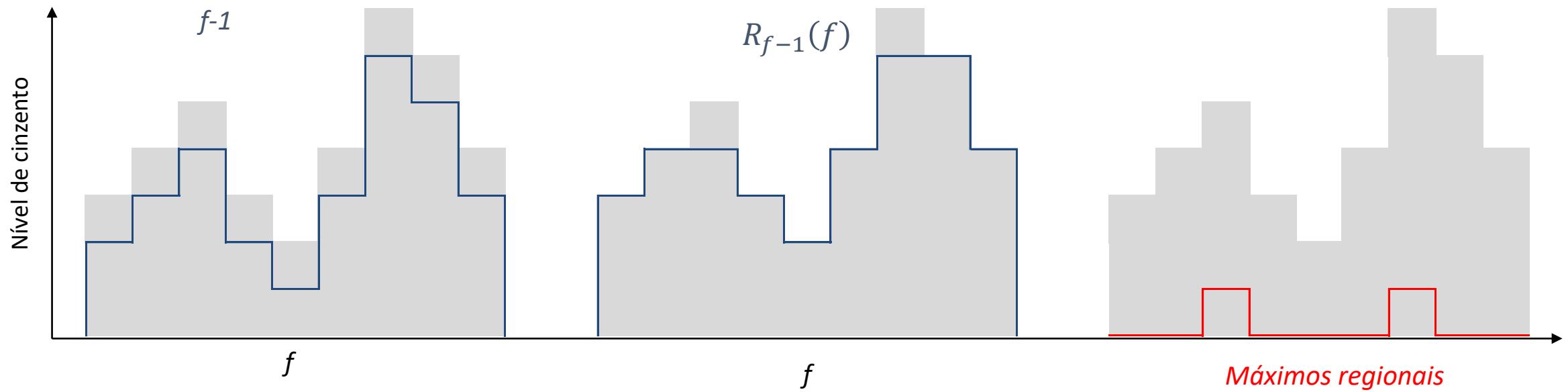
$$k_{\min(f)} = R_{f+1}^*(f) - f$$



# Extremos regionais

O mesmo se aplica ao conceito de **máximo regional** de elevação  $h$ , sendo que se tem que descer na função antes de alcançar outro máximo regional.

$$k_{\max(f)} = f - R_{f-1}(f)$$



# Extremos regionais

Exemplo:

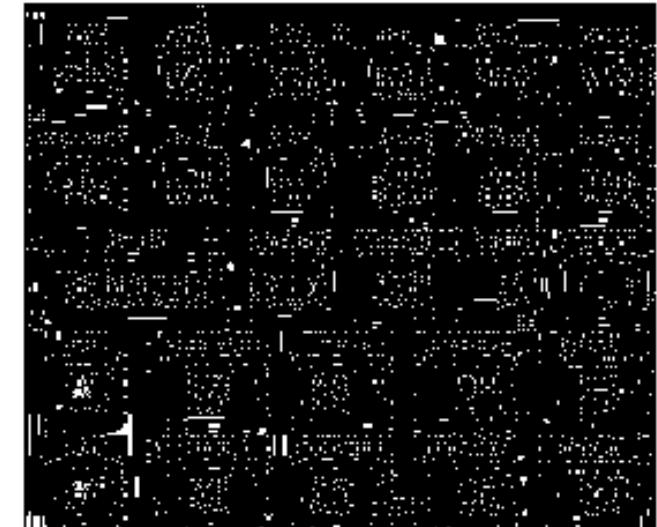
Inicial



Minimos



Máximos



# Transformação watershed

Fernando Soares

Transformação que visa **segmentar uma imagem de cinzentos** em regiões distintas, a partir de uma imagem binária de marcadores:

- Imagen 1 de entrada: imagem de cinzentos (superfície topográfica definida pelos valores de cinzento dos pixels).
- Imagen 2 de entrada: imagem de marcadores (pixels isolados ou conjuntos de pixels).
- Imagen de saída: Imagem binária, ou do tipo *label*, com as bacias de escoamento resultantes da transformação (ou, em alternativa, as linhas de *watershed*).

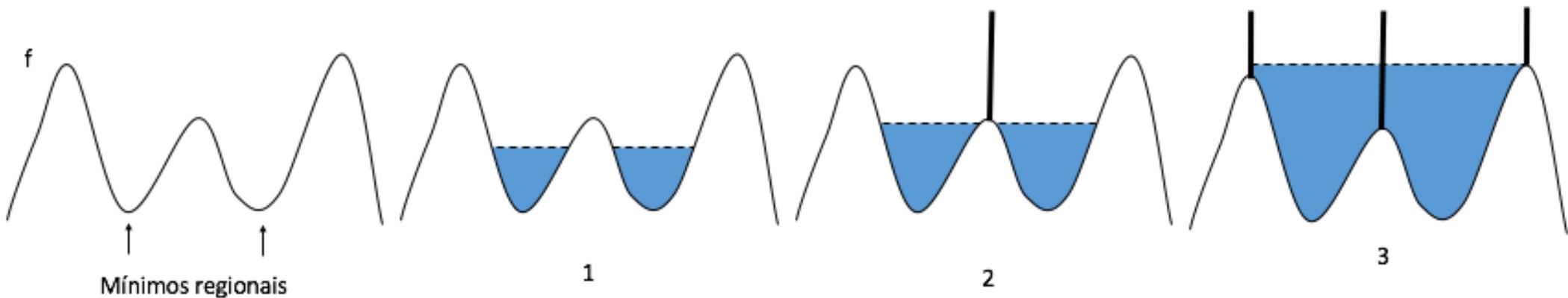
A transformação resulta da execução de um processo análogo ao processo físico de imersão vertical da superfície em água, com velocidade constante, a partir de “furos” feitos nessa superfície (imagem binária de marcadores).

# Transformação watershed

Fernando Soares

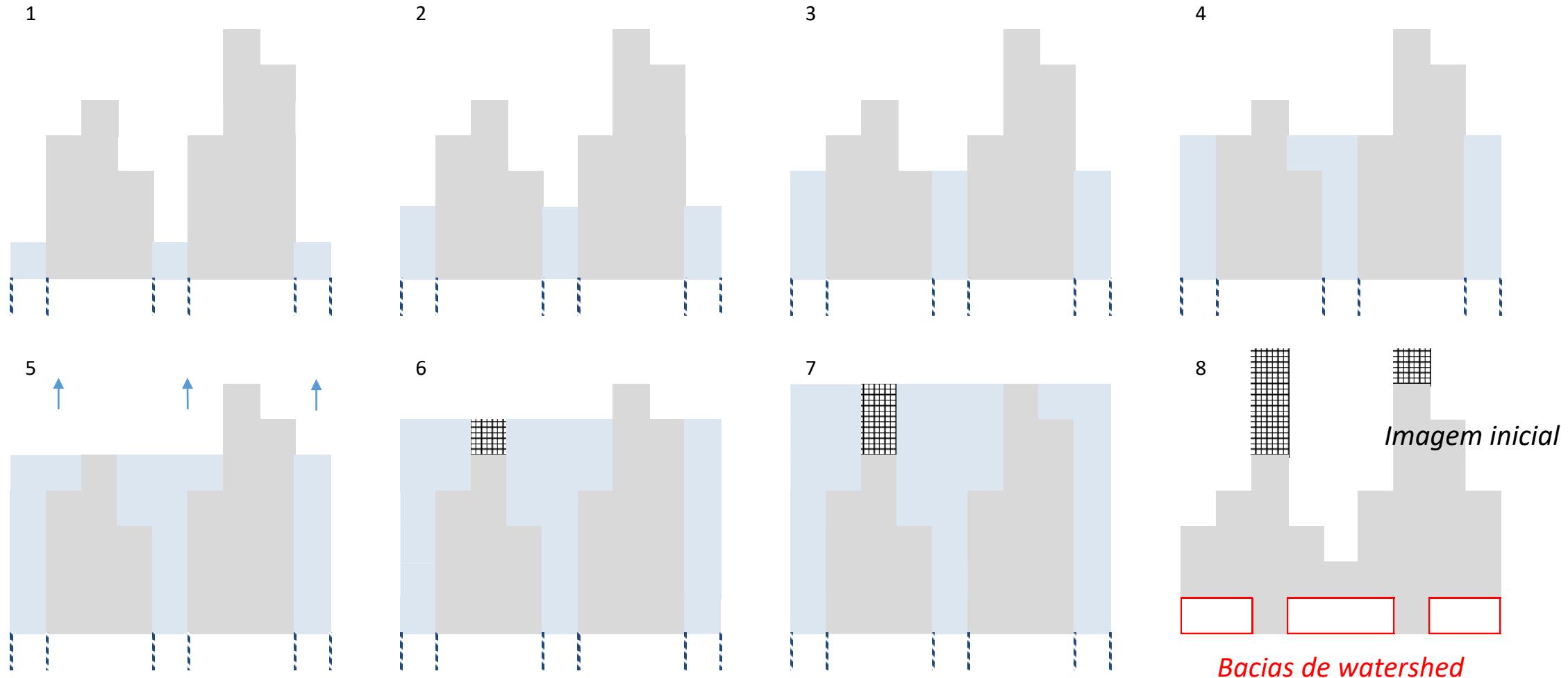
Por analogia com um processo de imersão, em água, de uma superfície com furos nos seus mínimos regionais, em certa altura dois ou mais vales inundados acabarão por se fundir. Para evitar que tal suceda, erguem-se barreiras em todos os pontos da superfície onde a fusão acontece.

No final do processo, apenas as barreiras são representadas. Estas barreiras, as linhas de *watershed* da superfície, separam as bacias de escoamento umas das outras.



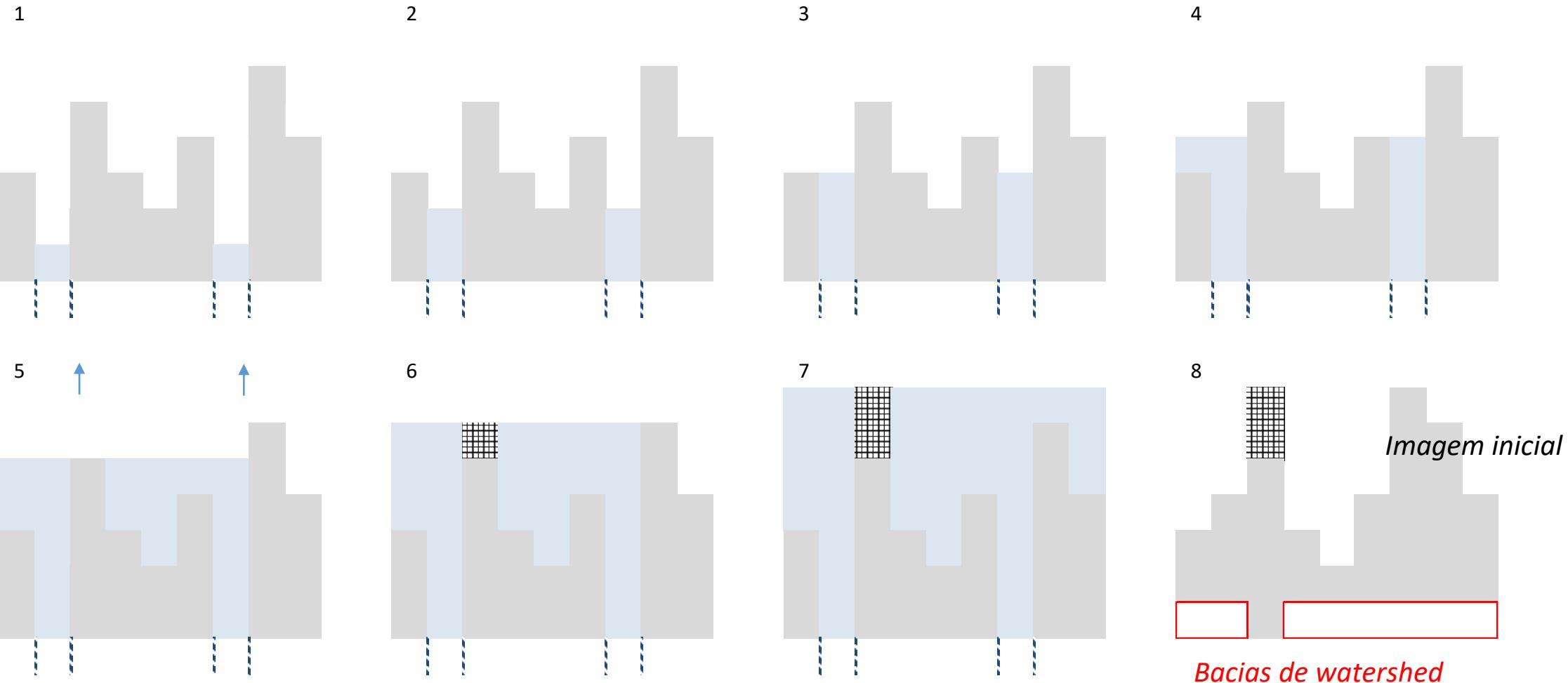
# Transformação watershed

Ilustração do processo de inundação a partir dos mínimos regionais.



# Transformação watershed

Ilustração do processo genérico de inundação a partir de um qualquer conjunto de marcadores.



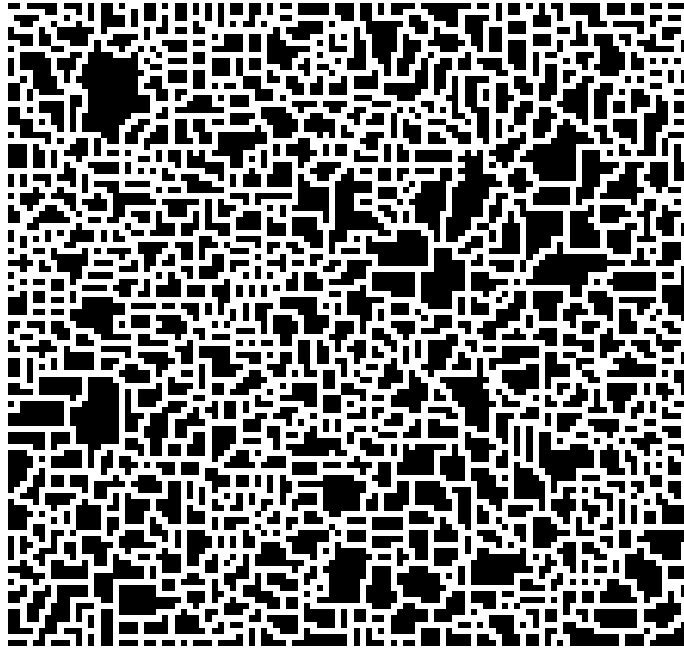
# Transformação watershed

Fernando Soares

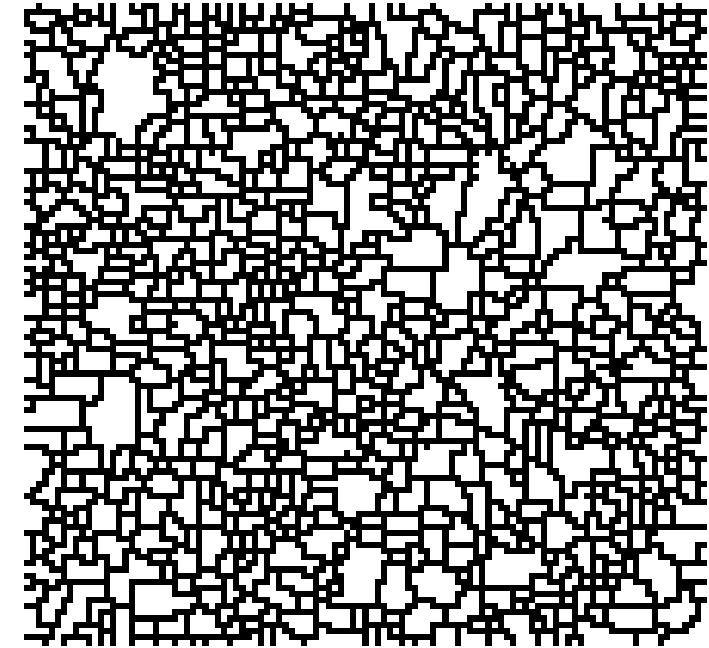
Exemplo da transformação *watershed*, a partir dos mínimos regionais.



Imagen inicial



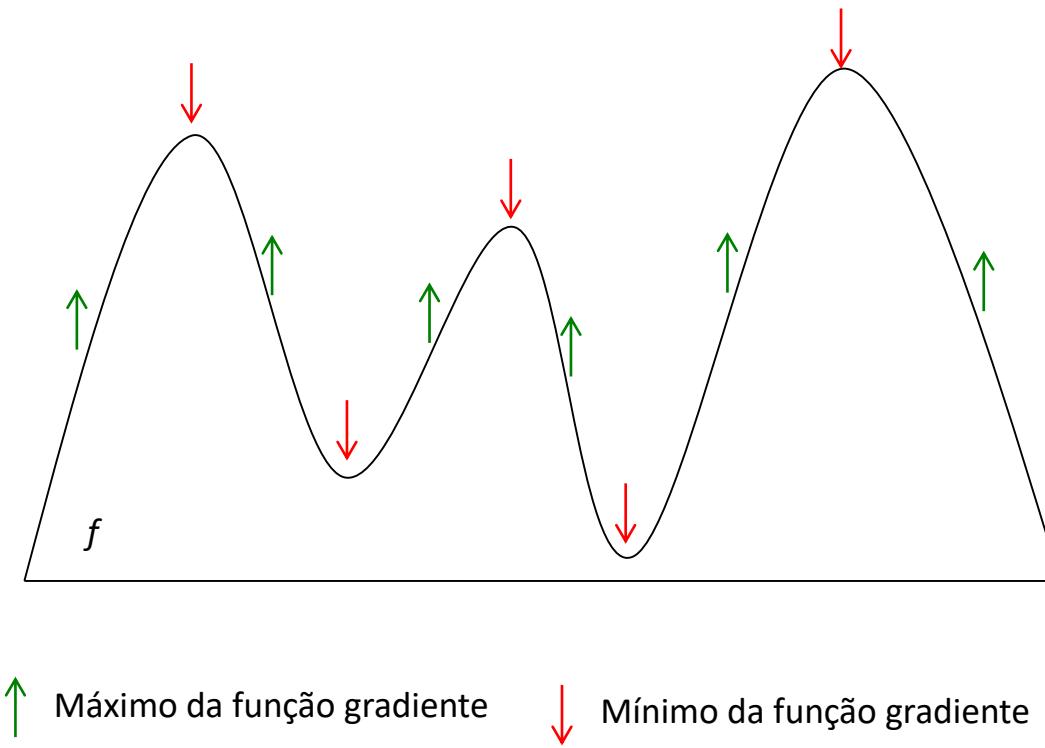
Linhos de *watershed*



Bacias de escoamento

# Transformação watershed

**Watershed da imagem do gradiente:** em vez da imagem original  $f$ , aplica-se sobre a imagem de cinzentos do gradiente morfológico  $\nabla$  (ou outro), a partir dos respectivos mínimos regionais.

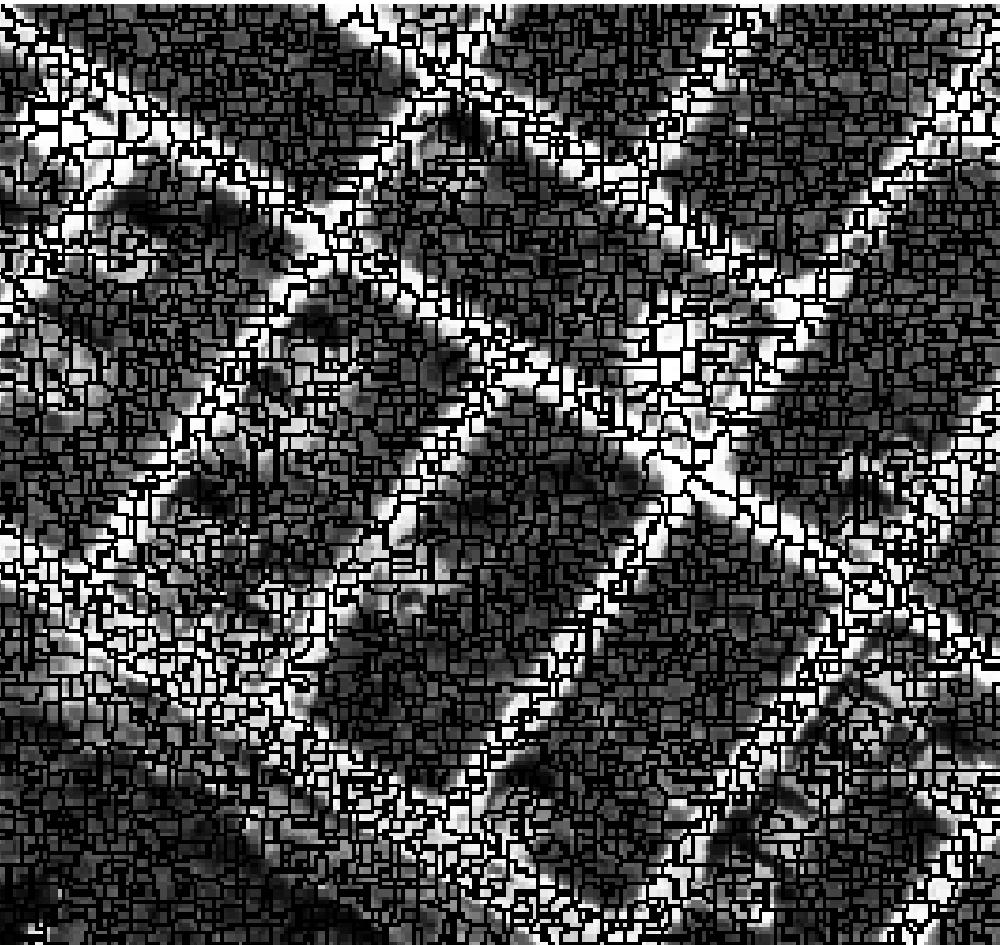


# Transformação watershed

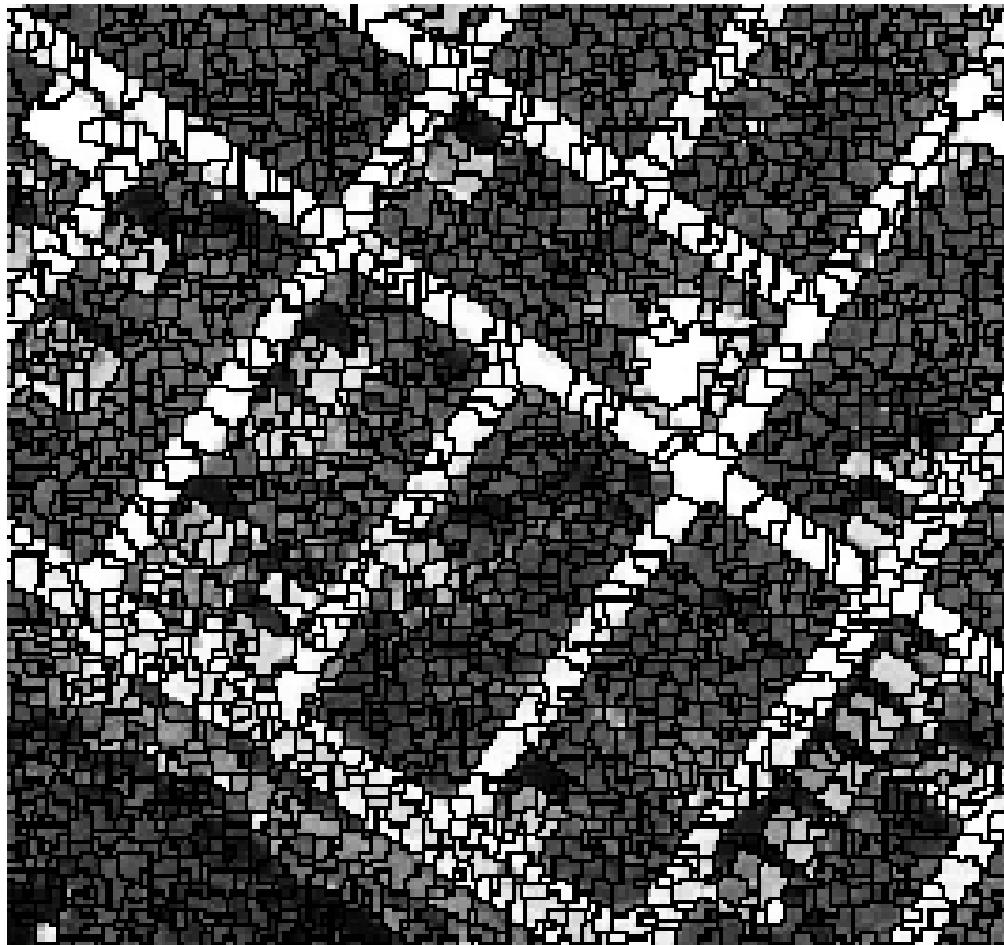
Fernando Soares

Watershed da imagem do gradiente G.

Watershed de F



Watershed de G



# 6. Segmentação de imagem

Técnicas de limiarização do histograma (multi-threshold, média, p-tile, histerese, moda, máxima distância, Otsu e adaptada). Clustering. Técnicas baseadas em regiões (crescimento, separação e split-and-merge). Técnicas baseadas em transições (Canny Edge Detector). Transformação de Hough. Template Matching. Algoritmos para segmentação.

# Introdução

A **segmentação de imagem (SEG)** refere-se à decomposição de uma imagem digital em uma série de regiões distintas (conjuntos de pixels).

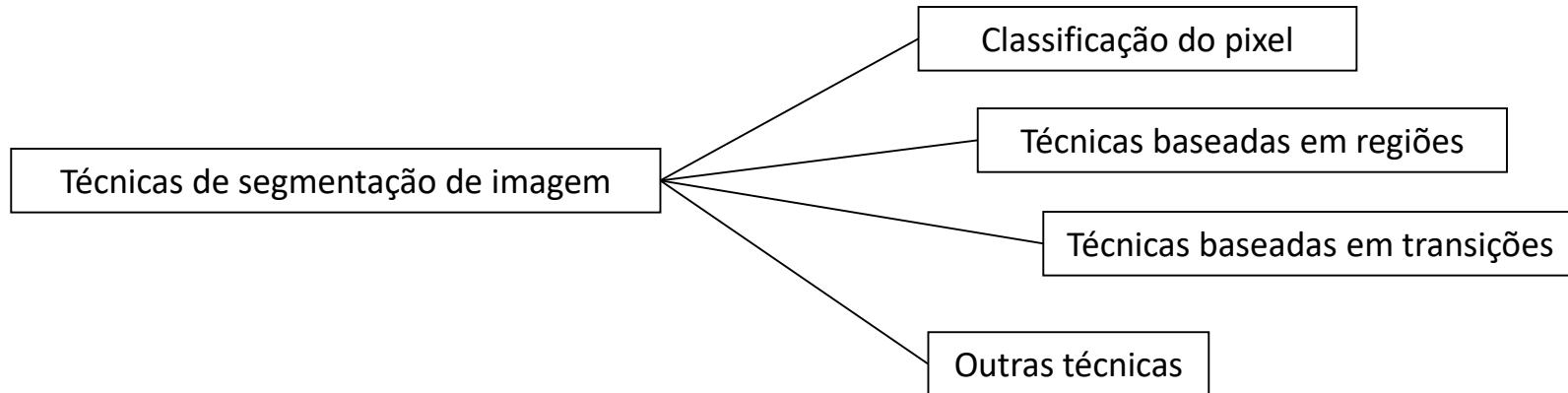
O objectivo da SEG é o de simplificar, ou alterar, a representação inicial de uma imagem, para outra representação que tenha maior significado e, consequentemente, que seja mais simples de analisar.

A SEG é tipicamente usada para localizar objectos e limites de objectos nas imagens.

Todos os pixels de uma região segmentada são semelhantes no que respeita a determinadas características, ou propriedades computacionais, tais como a cor, a intensidade, ou a textura.

Regiões adjacentes são significativamente distintas quanto às mesmas características.

A SEG reduz a informação dos pixels a informação contextualizada a regiões.



# Classificação do pixel

Seja  $f(x,y)$  uma imagem digital: as técnicas de segmentação do pixel assentam na atribuição de uma certa classe a cada pixel  $(x,y)$  da imagem  $f$ , num conjunto de classes possíveis  $\Omega = \{\omega_k: k = 1, \dots\}$ .

Pixels de uma mesma classe  $\omega_k$  formam todos juntos um segmento da função  $f$ .

A atribuição de uma classe a um pixel é guiada pela definição de um certo vector  $v(x,y)$  que pode ser estendido a diversos domínios (espacial, espectral, temporal), melhorando-se assim a segmentação.

O volume de cálculo é proporcional ao número de pixels que contém a imagem (elevadas resoluções  $\Rightarrow$  métodos que visem a extracção de objectos por inteiro).

# Classificação do pixel

**Limiarização (*threshold*) simples do histograma:** é uma das formas de segmentação mais usuais, associada à técnica de classificação do pixel.

- É útil na discriminação entre objectos de primeiro plano (*foreground*) e regiões de contexto (*background*).
- Mediante a selecção de um certo valor de *threshold*  $t$ , a imagem de intensidade é convertida em imagem binária.
- Estimam-se valores para  $t$  por observação e segmentação directa do histograma, ou por metodologias de segmentação do histograma com diferentes graus de automatização.

# Classificação do pixel

Exemplos de *threshold* simples do histograma.



F



$150 < F \leq 220$

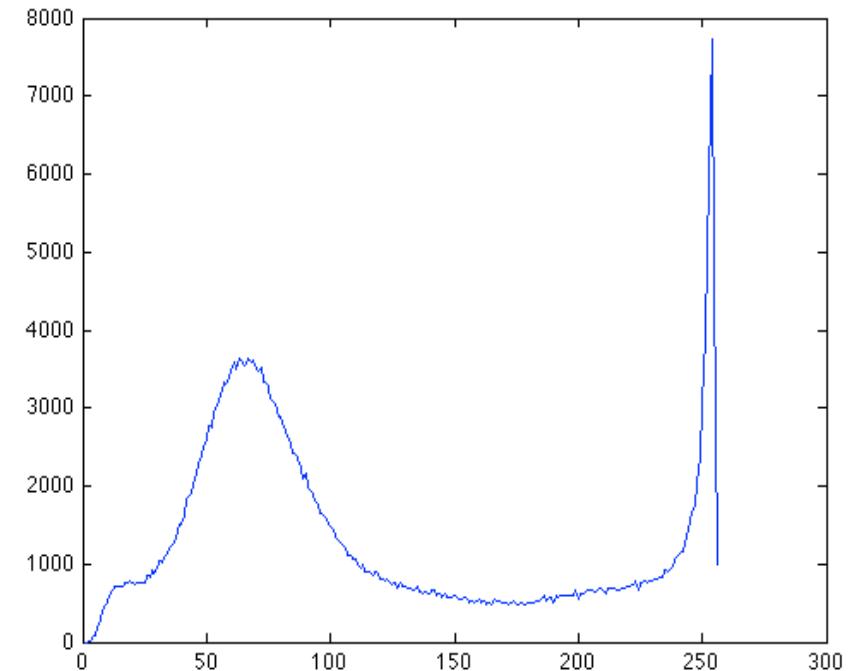


$F \leq 150$



$220 < F \leq 255$

Histograma de F

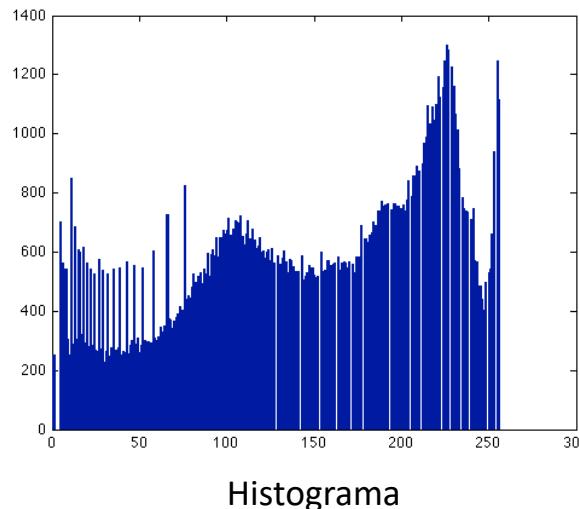


# Classificação do pixel

**Threshold pela média:** calcula-se o valor médio da imagem e usa-se como valor para  $t$ . Aplica-se este método quando a imagem tem aproximadamente a mesma quantidade de pixels nos motivos de *foreground* e *background*.



F



F &lt; 143 (média)

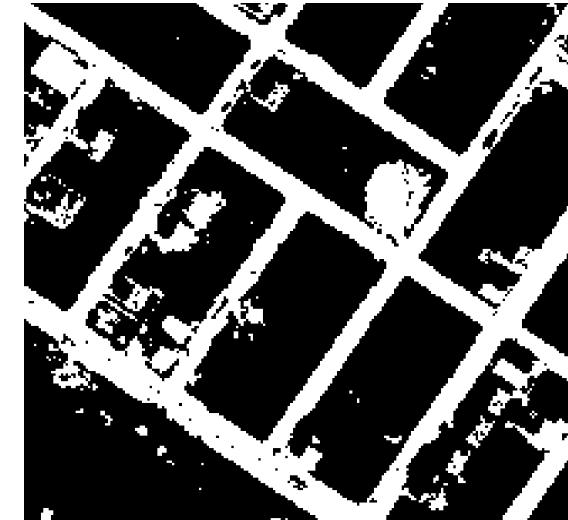
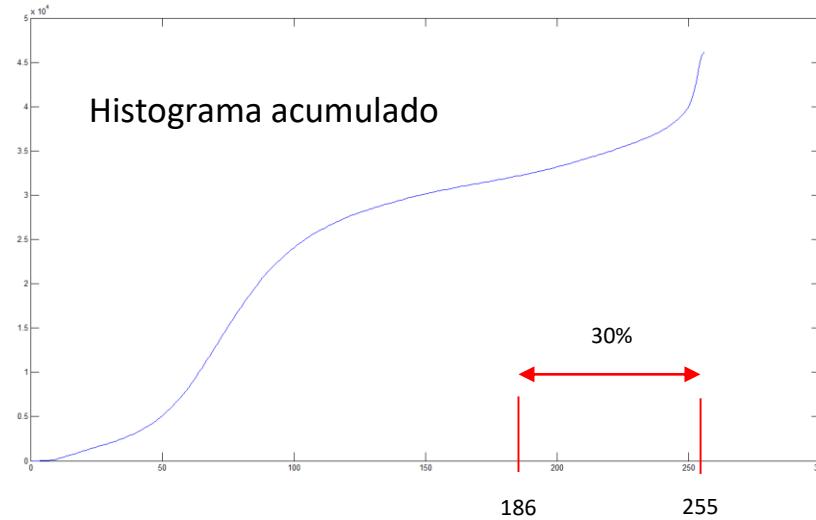
# Classificação do pixel

**Threshold pelo método “*p-tile*”:** Aplica-se quando se conhece a fracção aproximada que um certo objecto ocupa na imagem inteira.

Determina-se o valor de  $t$ , encontrando o valor de intensidade que corresponda à fracção do número de pixels pretendida. O valor de  $t$  é facilmente extraído a partir do histograma acumulado  $A$ , estimando  $A(t) = 1/p$ .



F

P-tile (30%):  $F > 186$

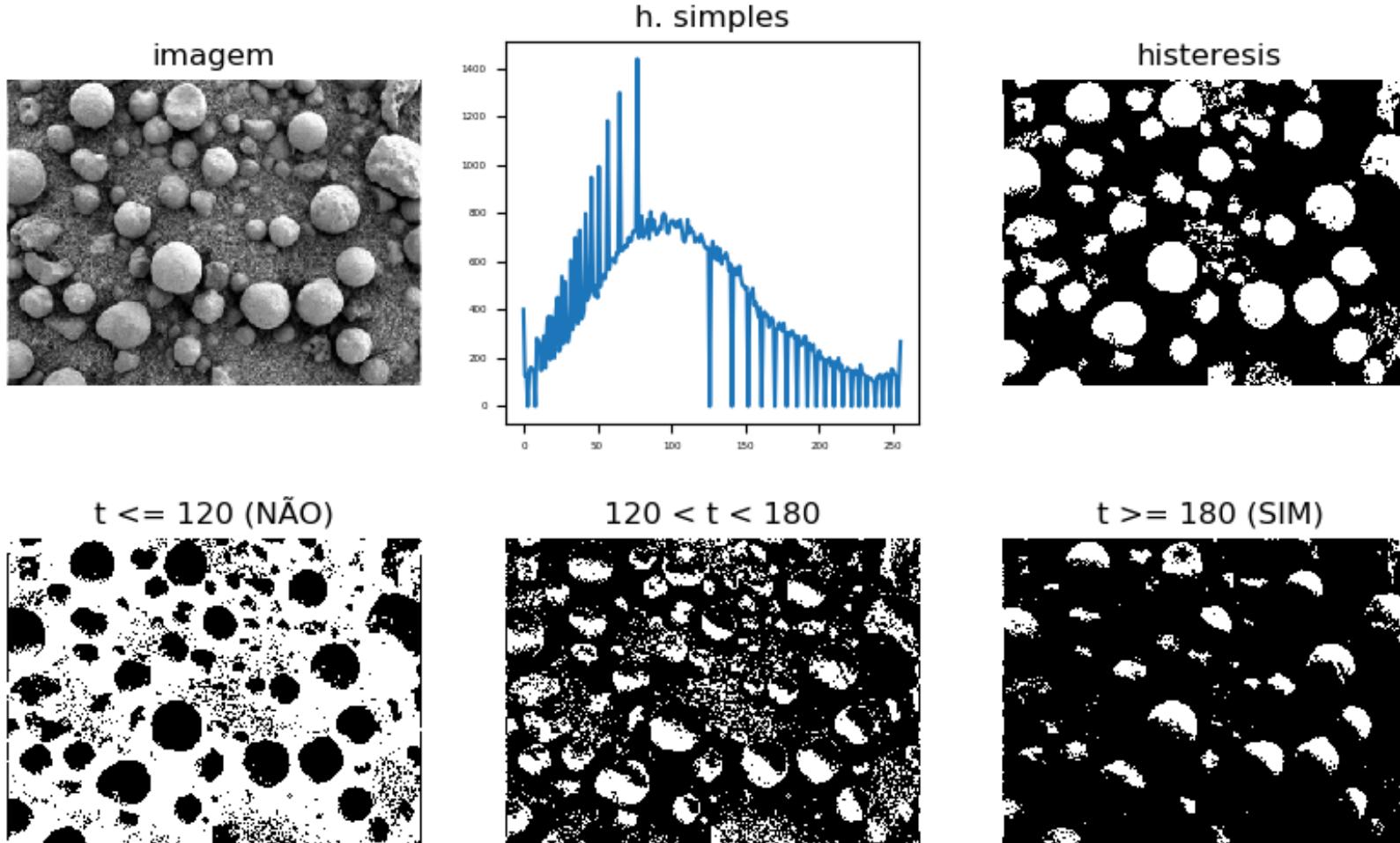
# Classificação do pixel

**Threshold por histerese:** se não existe um “vale” bem definido no histograma, significa que há pixels do *background* que têm valores de cinzento semelhantes aos pixels do *foreground* e vice-versa. Como tal efectuam-se duas limiarizações  $t_1$  e  $t_2$  ( $t_2 > t_1$ ) seguindo os seguintes princípios:

- i) Pixels com valores  $< t_1$  pertencem ao *background*;
- ii) Pixels com valores  $> t_2$  pertencem ao objecto;
- iii) Pixels com valores dentro do intervalo  $[t_1, t_2]$  pertencem ao objecto se forem adjacentes aos pixels do objecto.

# Classificação do pixel

Exemplo do método de *threshold* por histerese.

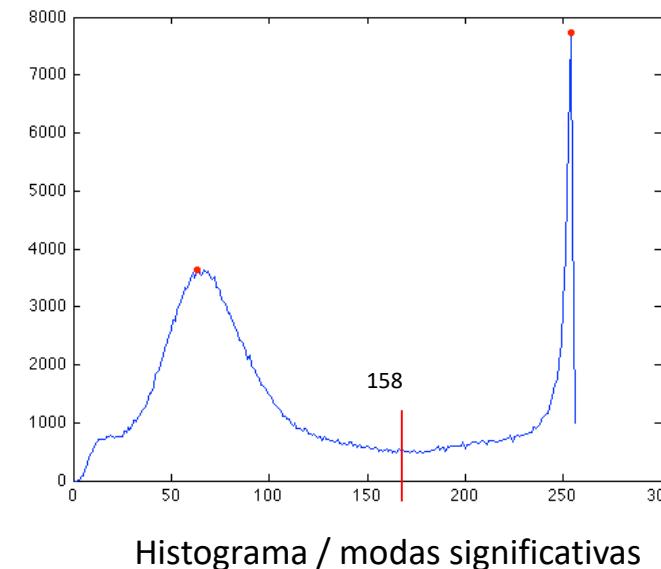


# Classificação do pixel

**Threshold por pesquisa das modas:** determinação dos picos de frequência e dos mínimos locais entre os picos (histogramas multimodais).



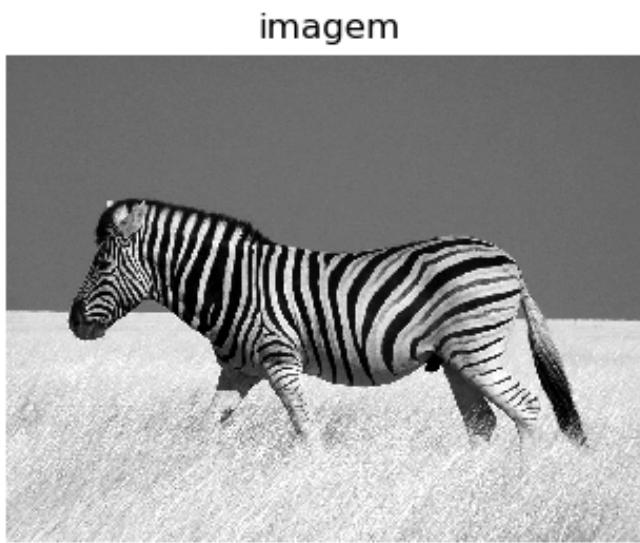
F



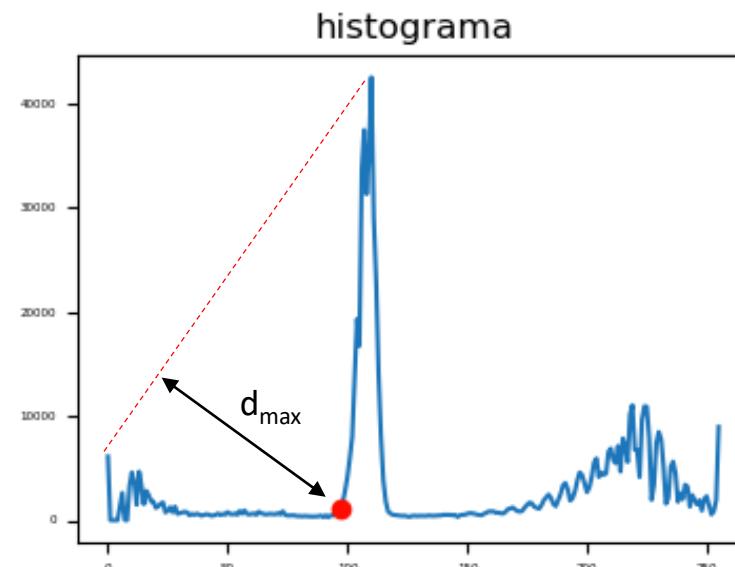
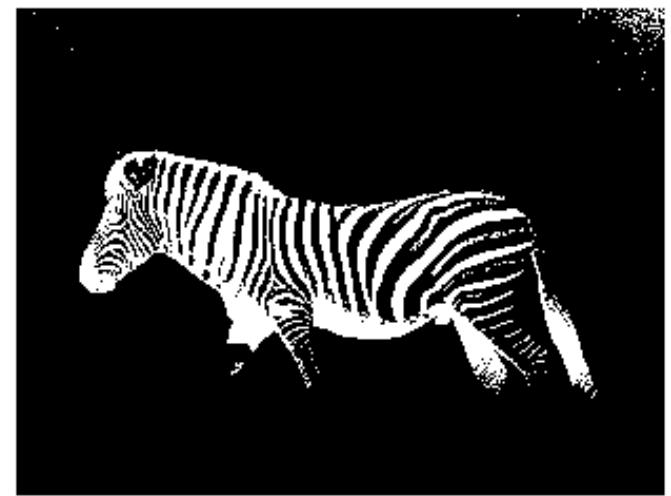
F &lt; 158

# Classificação do pixel

**Threshold pela máxima distância ( $d$ ):** determinação do nível de cinzento que se encontra à maior distância da recta que une o primeiro índice de frequência diferente de zero e o pico significativo do histograma.



F

Máx. Dist. ( $t = 97$ ) $F < 97$

# Classificação do pixel

**Threshold pelo método de Otsu:** Este método é usado para executar a limiarização automática de uma imagem de cinzentos em duas classes, por intermédio de uma análise estatística optimizada.

O valor óptimo de limiarização que separa as duas classes resulta da minimização de ambas as variâncias ponderadas de entre todos os *thresholds* do histograma possíveis.

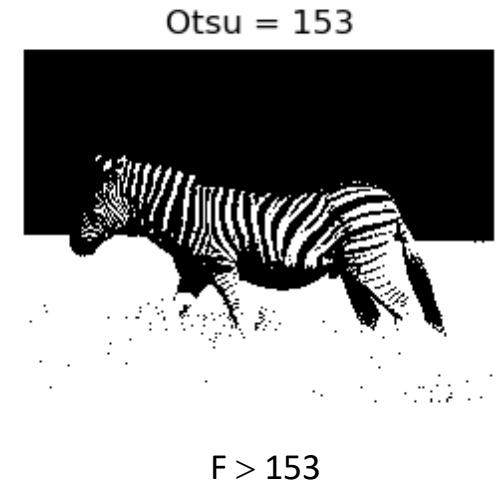
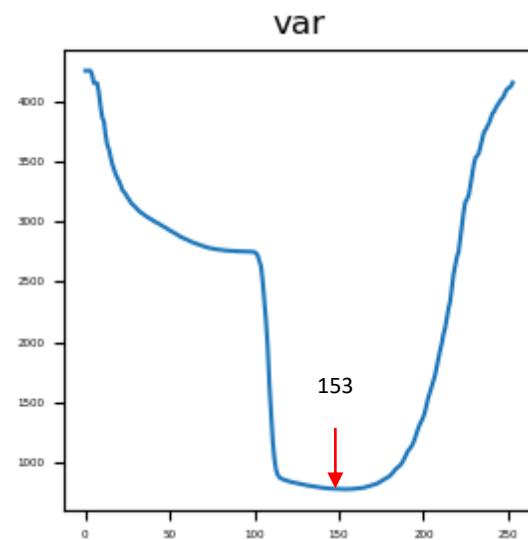
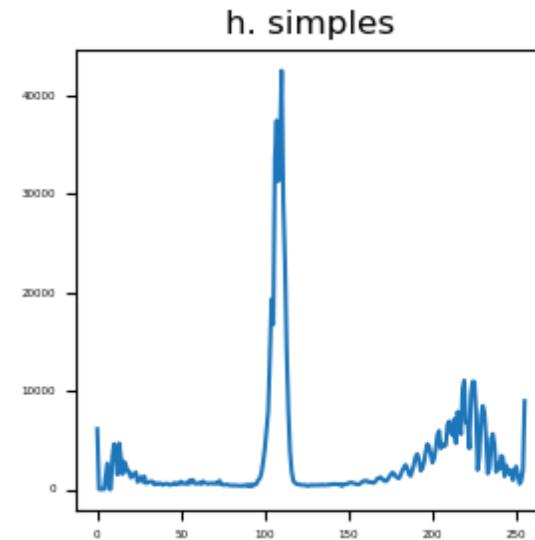
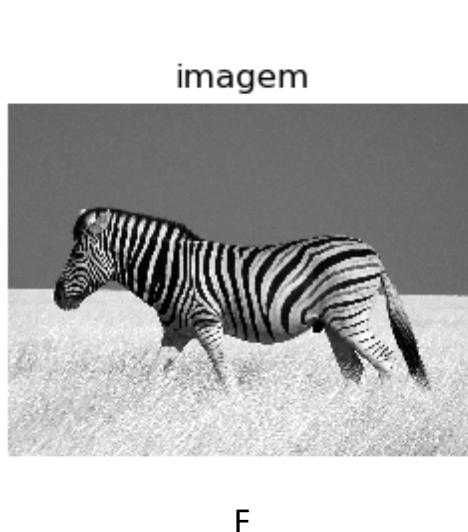
$$\sigma_i^2 = w_e(t_i) \times \sigma_e^2(t_i) + w_d(t_i) \times \sigma_d^2(t_i)$$

$$\sigma^2 = \min(\sigma_i^2)$$

$$w_e(t) = \sum_{i=0}^{t-1} p_i \quad w_d(t) = \sum_{i=t}^{255} p_i$$

# Classificação do pixel

Exemplo do método de *threshold* de Otsu.



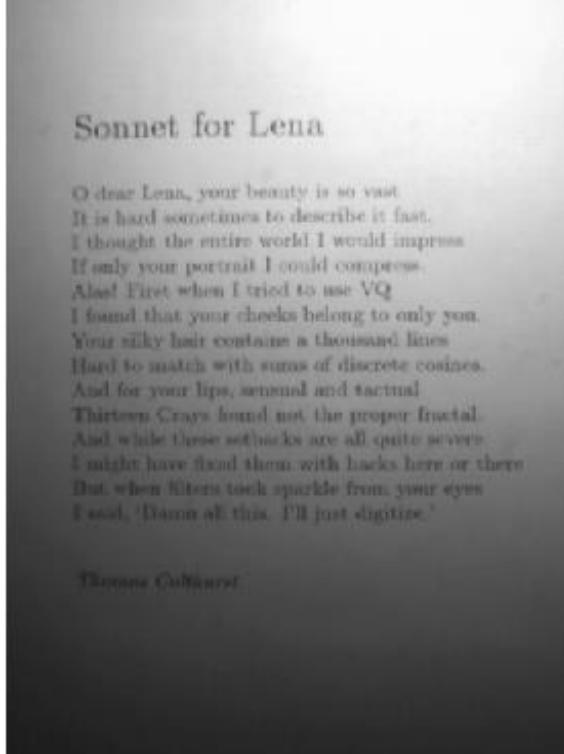
# Classificação do pixel

**Threshold adaptado (adaptive threshold):** É uma técnica de segmentação do pixel um pouco mais refinada, aplicada em circunstâncias de variação de luminosidade ao longo da imagem. Assim, procede-se da seguinte forma:

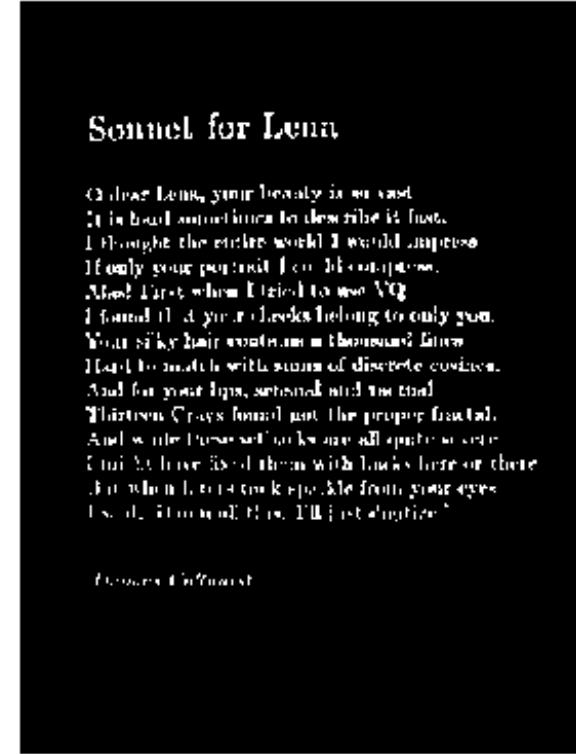
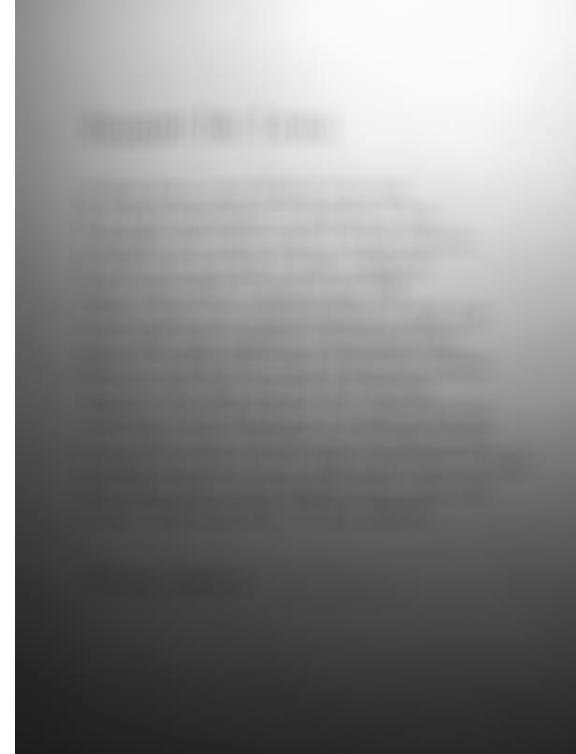
- i) Aplicar filtro passa-baixa (média) com filtro de dimensões suficientemente grandes (por exemplo 11x11) para “esbater” a informação a segmentar;
- ii) Determinar o valor absoluto da subtração da imagem filtrada à imagem original, seguido da subtração de uma constante K (i.e. 10) ao resultado;
- iii) Limiarização simples do resultado anterior.

# Classificação do pixel

Exemplo do método de *threshold* adaptado.



F



# Classificação do pixel

**Threshold multi-níveis (quantização):** Algumas imagens podem conter objectos de diferentes intensidades. Para tal podem-se usar limiarizações em vários níveis de intensidade para separar esses objectos.



F

 $F \leq 75$  $75 < F \leq 125$  $125 < F \leq 225$  $F > 225$ 

Multi-level threshold

Em resumo, as operações de limiarização apresentam geralmente as seguintes:

## Vantagens:

- Facilidade de implementação.
- Rapidez (especialmente se repetido em imagens similares).
- Adequado para certos tipos de imagens (por exemplo, documentos de texto).

## Desvantagens:

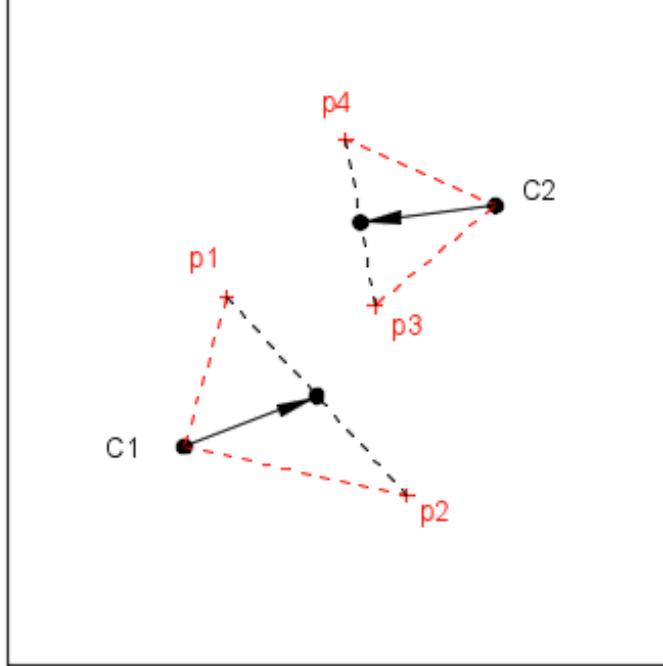
- A segmentação é por vezes pouco eficaz, pois não garante a coerência dos objectos (por exemplo, a existência de vazios, pixels “estranhos”, etc.).

**Clustering:** Consiste numa técnica de segmentação que visa organizar/agrupar os pixels em grupos (*clusters*) com base nos seus atributos de intensidade radiométrica.

- Definir centroides iniciais para os clusters.
- Determinar os pixels que estão mais próximos de cada centroide dos *clusters*.
- Cada centroide é re-determinado a partir de uma medida do respectivo *cluster* (por exemplo, a distância euclidiana).
- O processo é iterativo e termina quando os centroides não sofrerem alterações.

# Pixel clustering

**K-Means clustering:** técnica que visa agrupar os pixels em *clusters*, pela sua média.

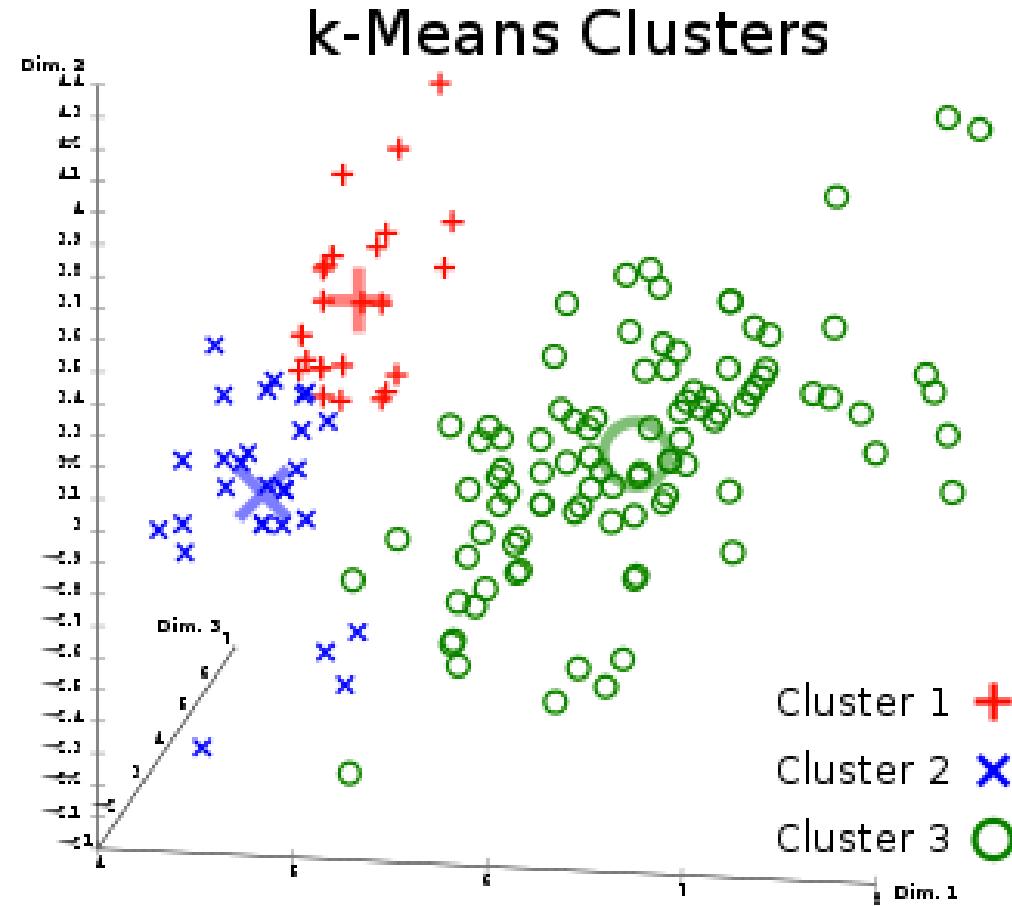


$$W(C) = \sum_{k=1}^K N_k \sum_{C(i)=k} \|x_i - m_k\|^2$$

$x_i$ : observações

$m_k$ : valor médio do *cluster* k

$N_k$ : número de observações do *cluster* k



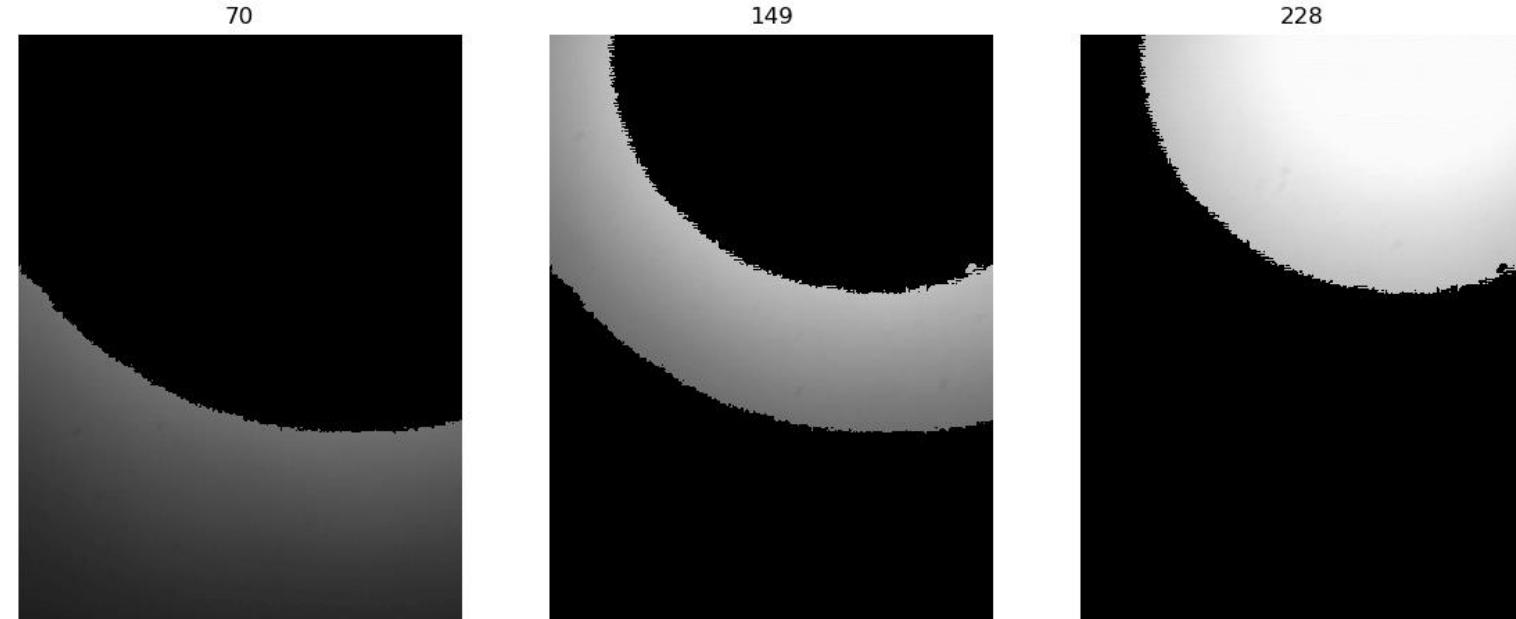
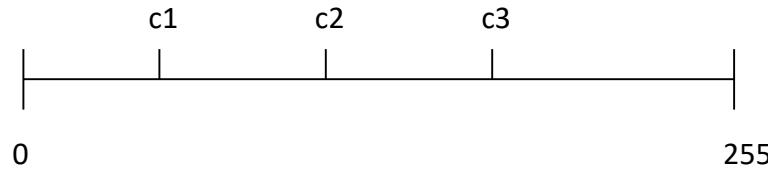
Cluster 1 +

Cluster 2 x

Cluster 3 o

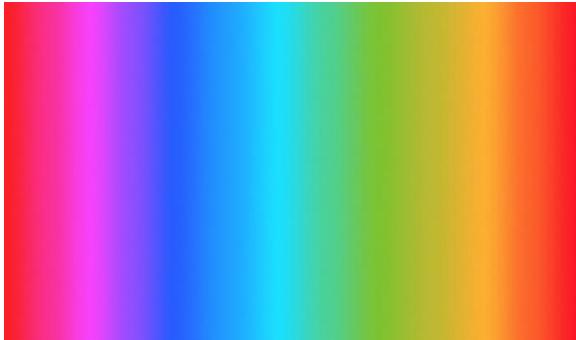
# Pixel clustering

**Gray K-Means clustering:** agregação dos pixels de uma imagem de cinzentos em clusters.



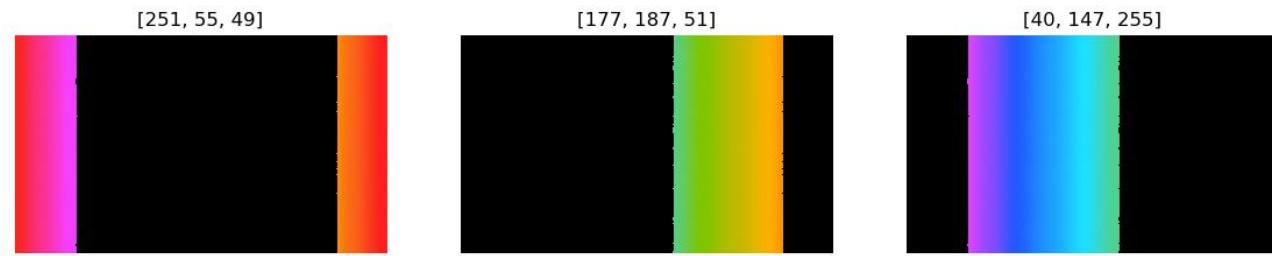
# Pixel clustering

**Color K-Means clustering:** agregação dos pixels de uma imagem de cor.

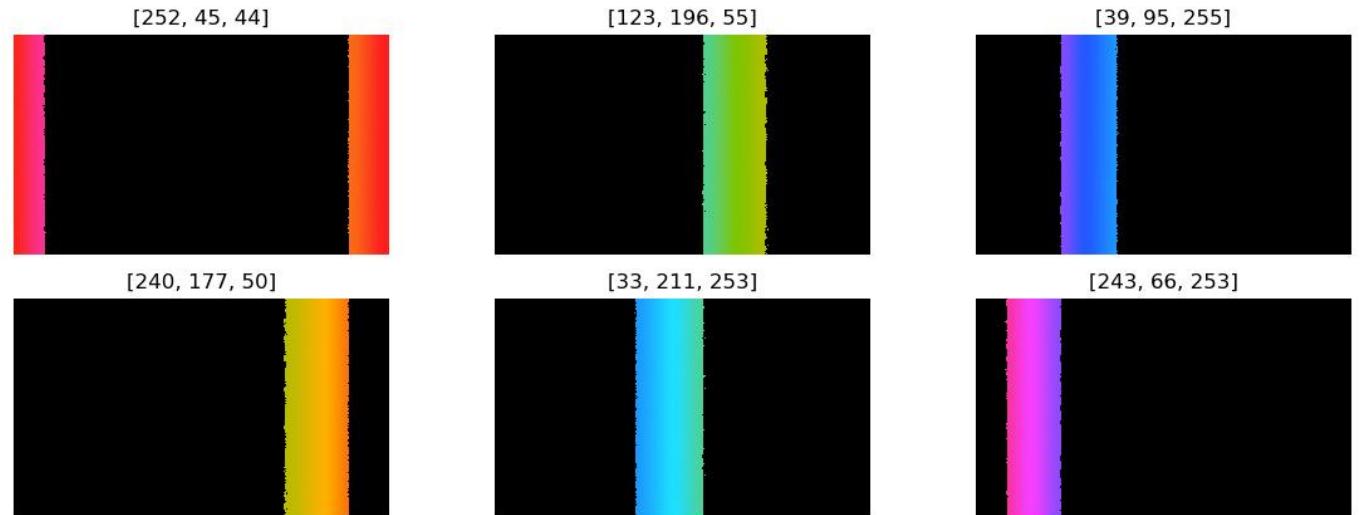


Imagen

3 clusters



6 clusters



# Pixel clustering

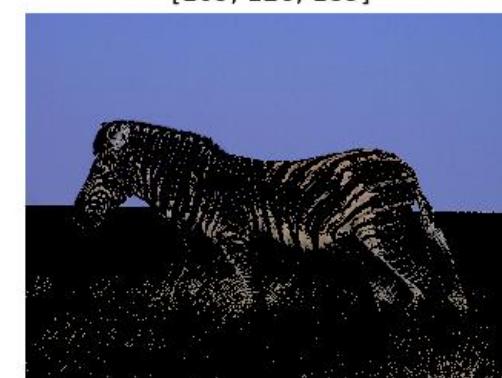
Exemplo do método de *clustering* em imagem de 24-bits, com três centroides.



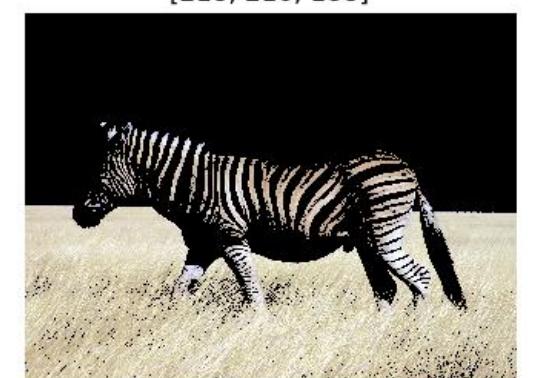
Imagen



[24, 21, 25]



[109, 126, 189]



[218, 210, 188]

# Histograma multivariado (HMV)

O objetivo da análise multivariada é estabelecer e utilizar as correlações espaciais das intensidades dos pixels para reduzir a informação dos dados a um nível mais refinado, com vista a uma interpretação mais fácil.

Como a análise multivariada investiga a correlação espacial de intensidades entre imagens, se houver poucas ou nenhuma informações complementares entre elas, é improvável que a técnica produza informação relevante adicional.

Neste contexto, a análise multivariada do histograma deve ser vista como uma ferramenta complementar dos meios convencionais de análise de imagem.

# Histograma multivariado (HMV)

As dimensões do HMV de uma imagem RGB ( $2^n \times 2^n \times 2^n$  bits) são de  $(n \times n \times 3)$ . O cálculo do HMV de uma imagem RGB faz-se da seguinte forma ( $p, q = 1 \dots n$ ):

$$HMV[p, q, 0] = HMV[p, q, 0] + 1 \quad \begin{cases} p = rgb[i, j, 0] \\ q = rgb[i, j, 1] \end{cases} \quad \text{Bandas 1 e 2}$$

$$HMV[p, q, 1] = HMV[p, q, 1] + 1 \quad \begin{cases} p = rgb[i, j, 0] \\ q = rgb[i, j, 2] \end{cases} \quad \text{Bandas 1 e 3}$$

$$HMV[p, q, 2] = HMV[p, q, 2] + 1 \quad \begin{cases} p = rgb[i, j, 1] \\ q = rgb[i, j, 2] \end{cases} \quad \text{Bandas 2 e 3}$$

# Histograma multivariado (HMV)

Fernando Soares

Exemplo.

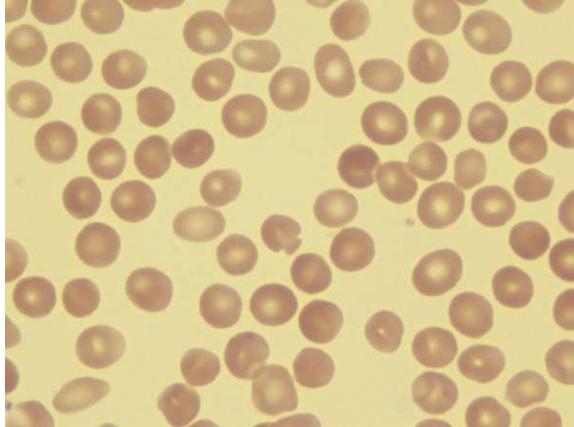
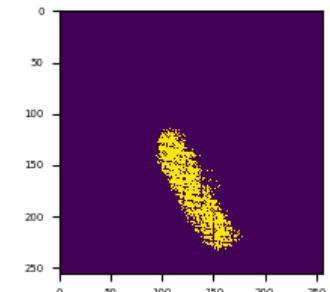
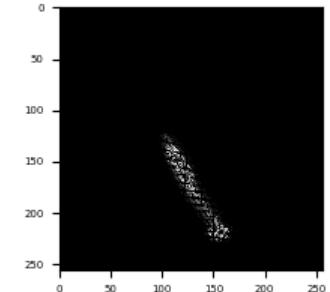
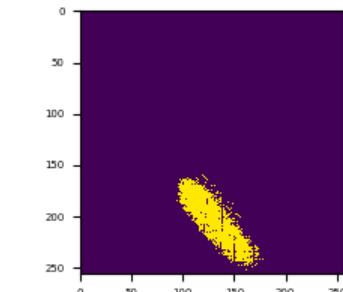
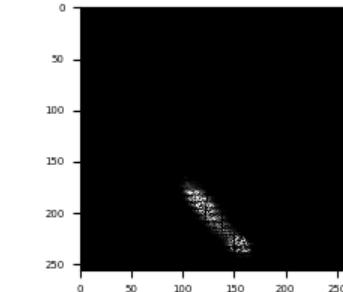
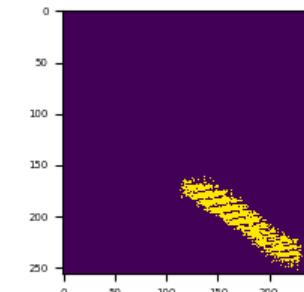
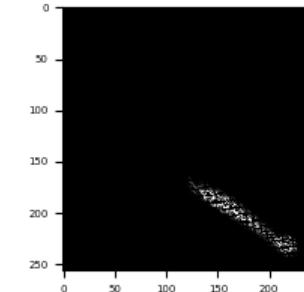
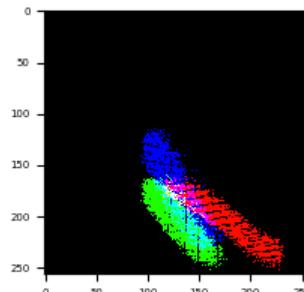
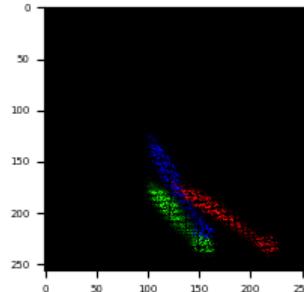


Imagen inicial

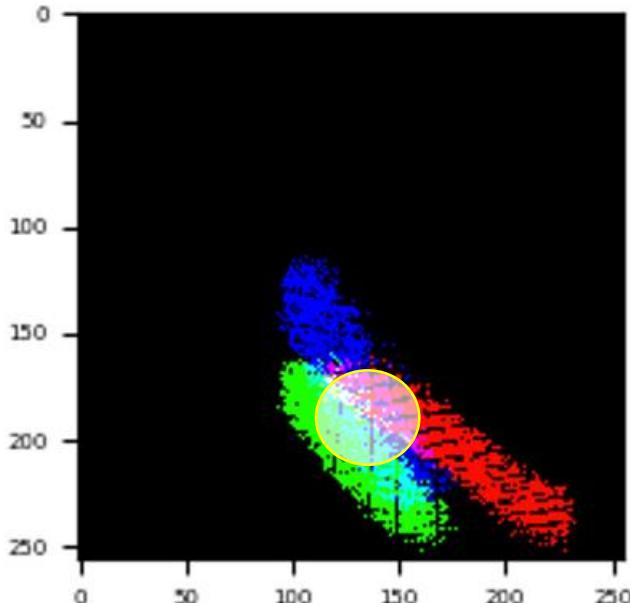


Histograma multivariado

Componentes

# Histograma multivariado (HMV)

Segmentação do histograma multivariado.



Região de interesse do HMV



Seleção dos pixels situados dentro da região de interesse

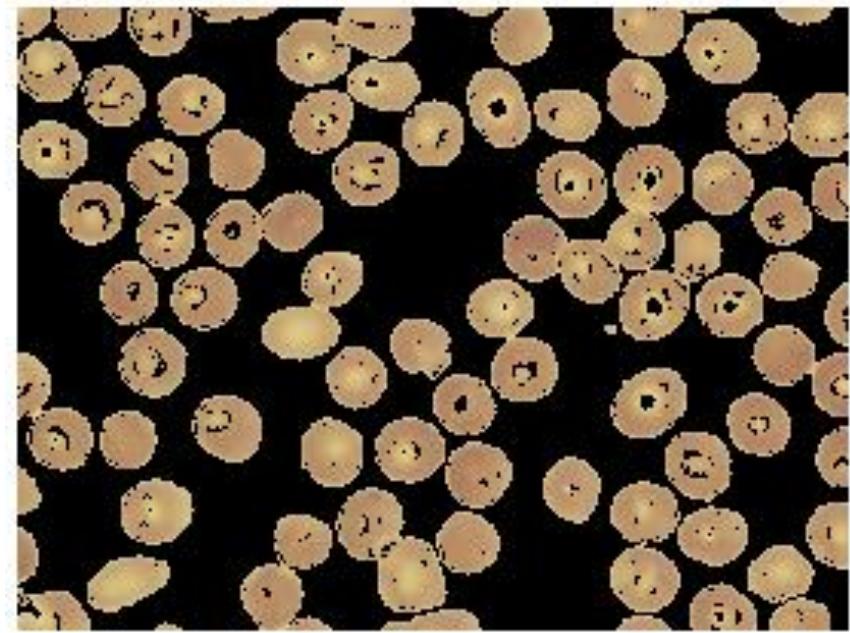


Imagen resultante

# Técnicas baseadas em regiões

Fernando Soares

Estas técnicas de SEG visam a partição, ou a agregação de regiões, de acordo com características associadas aos pixels da imagem (intensidade; textura, ou padrão dos objectos; perfis espectrais que fornecem dados multidimensionais).

Com base nessas características, e de acordo com critérios de similitude entre os pixels da imagem, constrói-se progressivamente, com algoritmos iterativos, as regiões homogéneas e termina-se o processo iterativo assim que todos os pixels da imagem forem classificados segundo aquelas as características.

A fronteira entre cada duas diferentes regiões define-se onde ambas se encontram.

Na caracterização final das regiões  $R_i$  de uma imagem  $I$ , dever-se-á respeitar a condição de independência espacial para todas elas.

$$I = \bigcup_i^s R_i \quad \wedge \quad R_i \cap R_j = \emptyset \quad , \quad i \neq j$$

# Técnicas baseadas em regiões

Fernando Soares

**Crescimento de regiões (*region growing*):** inicia-se a partir de um conjunto de pequenas regiões previamente segmentadas e com propriedades consideradas homogéneas, no conjunto dos pixels que as constituem.

- Nos métodos mais simples começa-se a anexação de regiões a partir de pequenas regiões de  $2\times 2$ ,  $3\times 3$ , ou  $4\times 4$  pixels (regiões “semente”).
- Cada região é geralmente descrita segundo propriedades estatísticas dos seus níveis de cinzento. Quando regiões adjacentes se assemelham, estas anexam-se, originando assim uma nova região de maior tamanho. Caso contrário serão classificadas como não semelhantes e mantêm-se separadas.

- O processo de anexação continua assim entre outras regiões vizinhas, incluindo recém- anexadas.
- Quando uma certa região não pode ser anexada a nenhuma outra, termina o processo local de anexação, ou é reformulado o critério de anexação.
- O método termina quando cada região é classificada de forma única relativamente às restantes sem se verificar qualquer outra anexação. Os critérios de homogeneidade que determinam as várias anexações das regiões são diversos: níveis de cinzento, cor, textura, forma, segundo um certo modelo, etc.

# Técnicas baseadas em regiões

Fernando Soares

**Separação de regiões (*region splitting*):** separação da imagem, ou de uma ou mais áreas de interesse de uma imagem, num conjunto de regiões disjuntas, por forma a que haja coerência entre os pixels de cada região, quanto a certas propriedades.

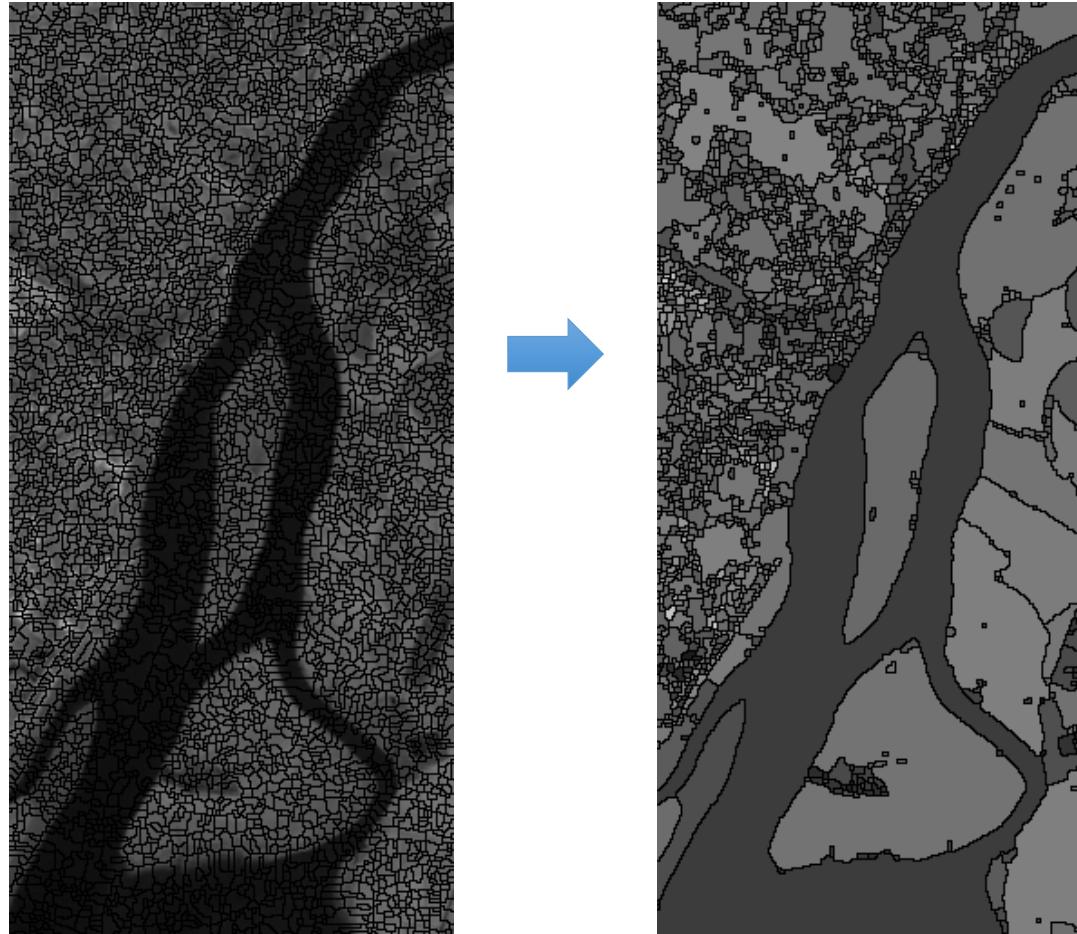
Exemplos de técnicas:

- I. A transformação morfológica *watershed* é frequentemente referenciada como técnica usada na definição preliminar das regiões.
- II. Área total dividida em regiões disjuntas se os correspondentes pixels não respeitarem um certo constrangimento de similitude. Por norma, a partição inicial faz-se em quatro quadrantes. O processo continua em cada nova área de interesse, até que não aconteça mais nenhuma partição.

# Técnicas baseadas em regiões

Fernando Soares

I. Exemplo da SEG resultante do processo de crescimento de regiões a partir da anexação de regiões mais pequenas (obtidas por watershed), com base em critérios radiométricos.



# Técnicas baseadas em regiões

II. Sucessiva divisão em quatro quadrados iguais; uma região mantém-se se cumpre o critério de homogeneidade (diferença  $\leq 1$ ); termina quando não houver mais divisões.

1	1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0	
1	1	8	8	8	4	1	0	
1	1	6	6	6	3	1	0	
1	1	5	6	6	3	1	0	
1	1	5	6	6	2	1	0	
1	1	1	1	1	1	0	0	

F

1	1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0	
1	1	8	8	8	4	1	0	
1	1	6	6	6	3	1	0	
1	1	5	6	6	3	1	0	
1	1	5	6	6	2	1	0	
1	1	1	1	1	1	0	0	

1º passo

1	1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0	
1	1	8	8	8	4	1	0	
1	1	6	6	6	3	1	0	
1	1	5	6	6	3	1	0	
1	1	5	6	6	2	1	0	
1	1	1	1	1	1	0	0	

2º passo

1	1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0	
1	1	8	8	8	4	1	0	
1	1	6	6	6	3	1	0	
1	1	5	6	6	3	1	0	
1	1	5	6	6	2	1	0	
1	1	1	1	1	1	0	0	

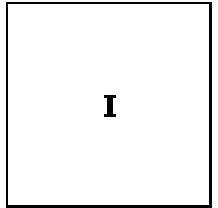
3º passo

***Split and merge***: Combinação entre as técnicas de *region splitting* e *region growing*.

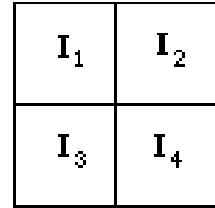
- Após a partição por *region splitting*, executa-se um processo de fusão das sub-regiões que cumpram os critérios de homogeneidade estabelecidos.
- O método de SEG, conhecido por *quadtree decomposition*, é um exemplo da técnica *split-and-merge*, que consiste em partitionar a imagem em quadrados idênticos (imagens com área de representação quadrada e com dimensão  $2^L$ ) e depois anexar as regiões que tenham propriedades radiométricas semelhantes.

# Técnicas baseadas em regiões

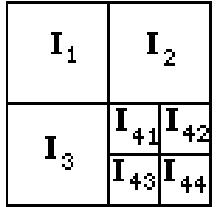
Ilustração do método de *split and merge*.



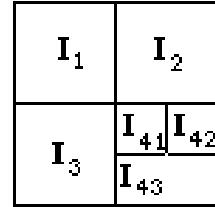
(a) Imagem inicial



(b) 1<sup>a</sup> partição



(c) 2<sup>a</sup> partição



(d) Fusão

1	1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0	
1	1	8	8	8	4	1	0	
1	1	6	6	6	3	1	0	
1	1	5	6	6	3	1	0	
1	1	5	6	6	2	1	0	
1	1	1	1	1	1	0	0	

Splitting

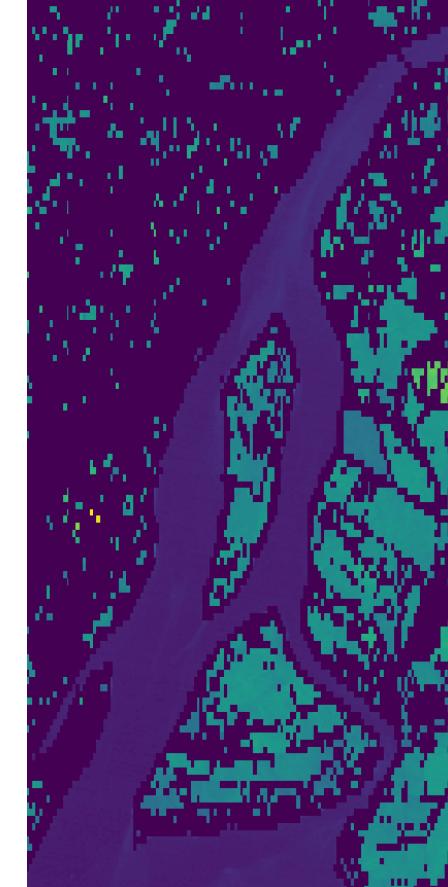
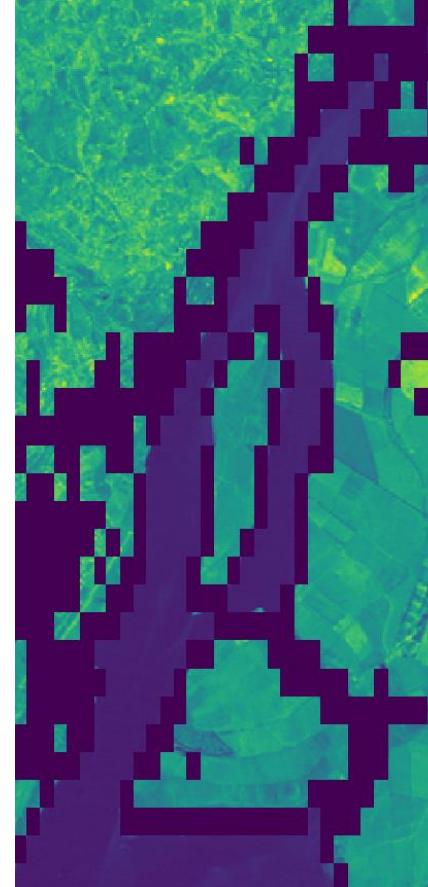
1	1	1	1	1	1	1	1	2
1	1	1	1	1	1	1	1	0
3	1	4	9	9	8	1	0	
1	1	8	8	8	4	1	0	
1	1	6	6	6	3	1	0	
1	1	5	6	6	3	1	0	
1	1	5	6	6	2	1	0	
1	1	1	1	1	1	0	0	

Merging

# Técnicas baseadas em regiões

Fernando Soares

Exemplo com os critérios de  $\sigma \leq 10$  (até 32 divisões) e  $\sigma \leq 2$  (até 128 divisões).



# Técnicas baseadas em transições

Fernando Soares

Transições são descontinuidades na intensidade de uma dada imagem, e podem ser dos tipos:

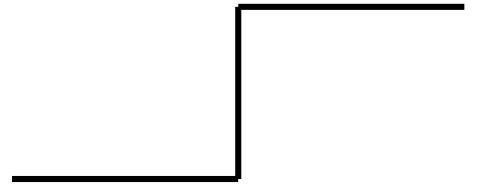
- i) “Degrau”: onde a mudança se faz de forma abrupta entre dois valores distintos.
- ii) “Linha”: onde a mudança acontece, regressando no entanto ao valor anterior após uma reduzida distância.

Contudo, estas mudanças são efectivamente raras de acontecer, verificando-se quase sempre os modelos de “rampa” e “telhado”.

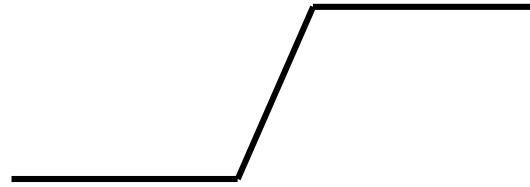
# Técnicas baseadas em transições

Fernando Soares

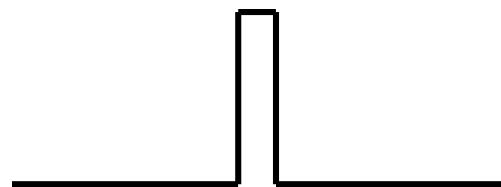
Ilustração dos modelos de transição.



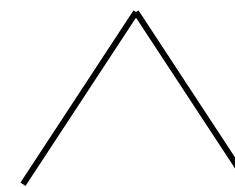
i) Degrau



ii) Rampa



iii) Linha



iv) Telhado

# Técnicas baseadas em transições

Fernando Soares

Estas técnicas assentam na determinação de linhas de transição numa imagem, por uso de operadores de detecção de zonas de transição (as transições correspondem a zonas de descontinuidade radiométrica, textural, etc.).

- A variação radiométrica nos pixels da fronteira entre duas regiões distintas adjacentes é bem mais significativa que a variação entre os pixels internos a cada uma dessas regiões.
- No entanto os resultados de segmentação por detecção de transições devem passar por fases suplementares de processamento, no sentido de adequar as linhas encontradas aos objectos específicos que se pretendem ver delimitados.

# Técnicas baseadas em transições

Fernando Soares

- Executada uma das técnicas de segmentação, crescimento de regiões, ou detecção de transições, torna-se intuitivo definir a outra, já que a construção de regiões torna-se acessível a partir do conhecimento das respectivas transições, assim como também é acessível definir as transições a partir do conhecimento das regiões.
- Pode acontecer que as regiões de uma imagem, obtidas com metodologias de crescimento de regiões, não coincidam com as que se obtêm por metodologias de detecção de transições, ou vice-versa.

# Canny-edge detector

**Canny-edge detector:** é um algoritmo desenhado para ser um detector de fronteiras.

Parte-se de um imagem de cinzentos e obtém-se uma imagem binária com as posições das descontinuidades radiométricas marcantes da imagem inicial.

É executado em diversas etapas que focam conceitos diversos de processamento de imagem, tais como filtragem, quantização, e gradiente.

Original

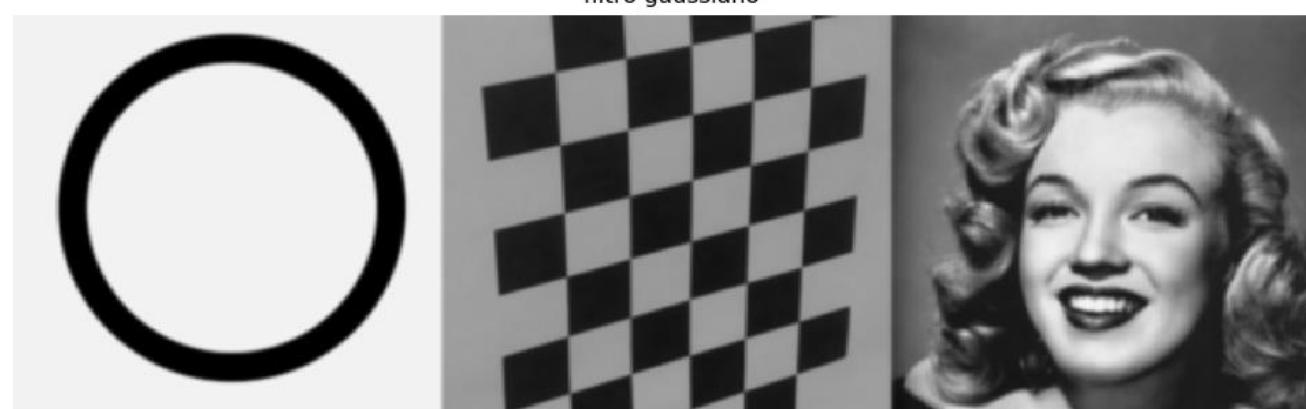
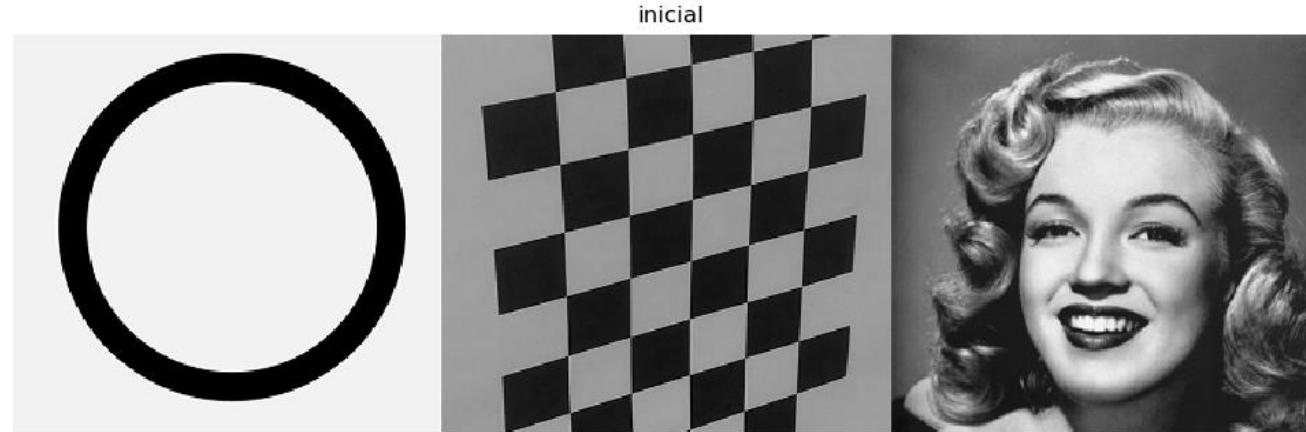


Canny



## Síntese do algoritmo.

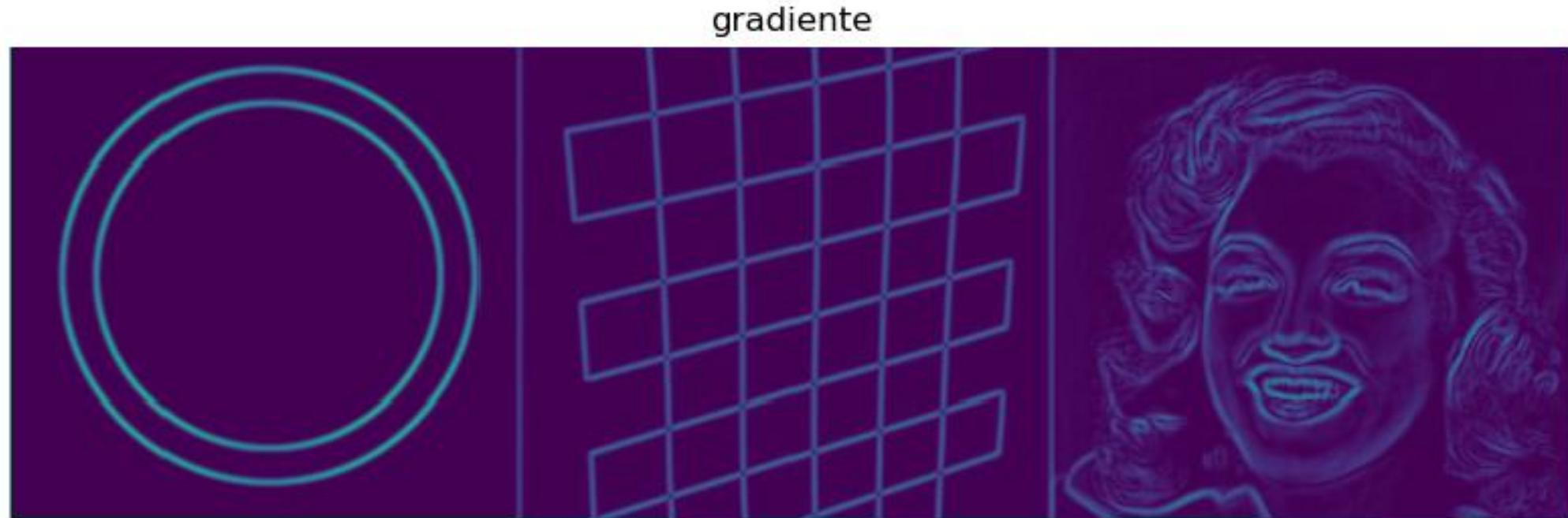
1. A imagem inicial é suavizada com um filtro gaussiano.



1

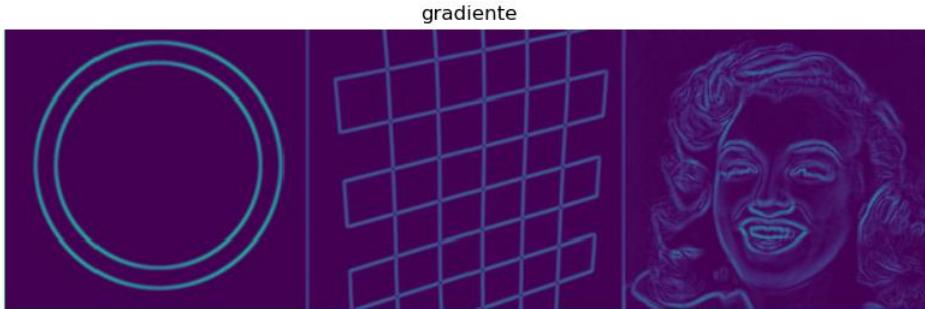
# Canny-edge detector

2. Executa-se um filtro derivativo simples (por exemplo o de Sobel) sobre a imagem suavizada, para realçar as primeiras derivadas espaciais mais elevadas nas direcções horizontal e vertical.

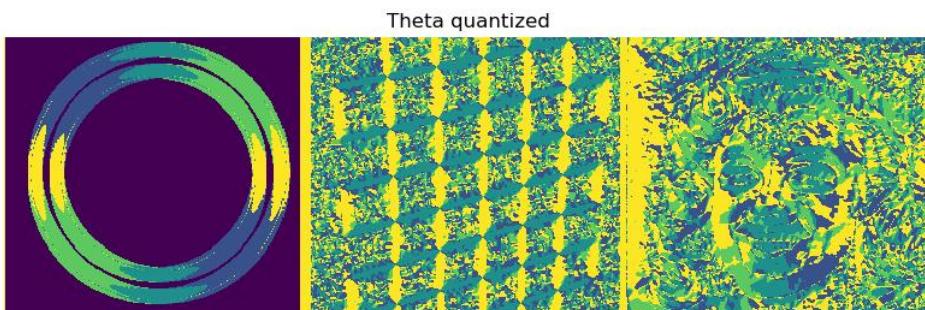
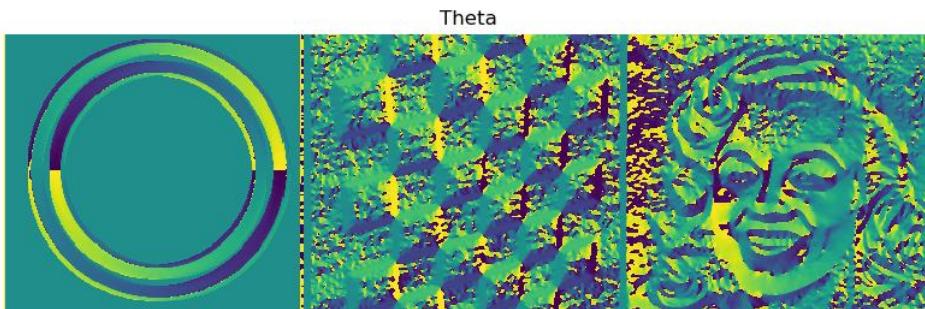


# Canny-edge detector

3. Determina-se, em cada pixel, o ângulo de inclinação entre as direcções horizontal e vertical do gradiente, e quantizam-se os valores para as direcções de  $k \times 45^\circ$  ( $k=0\dots3$ ).

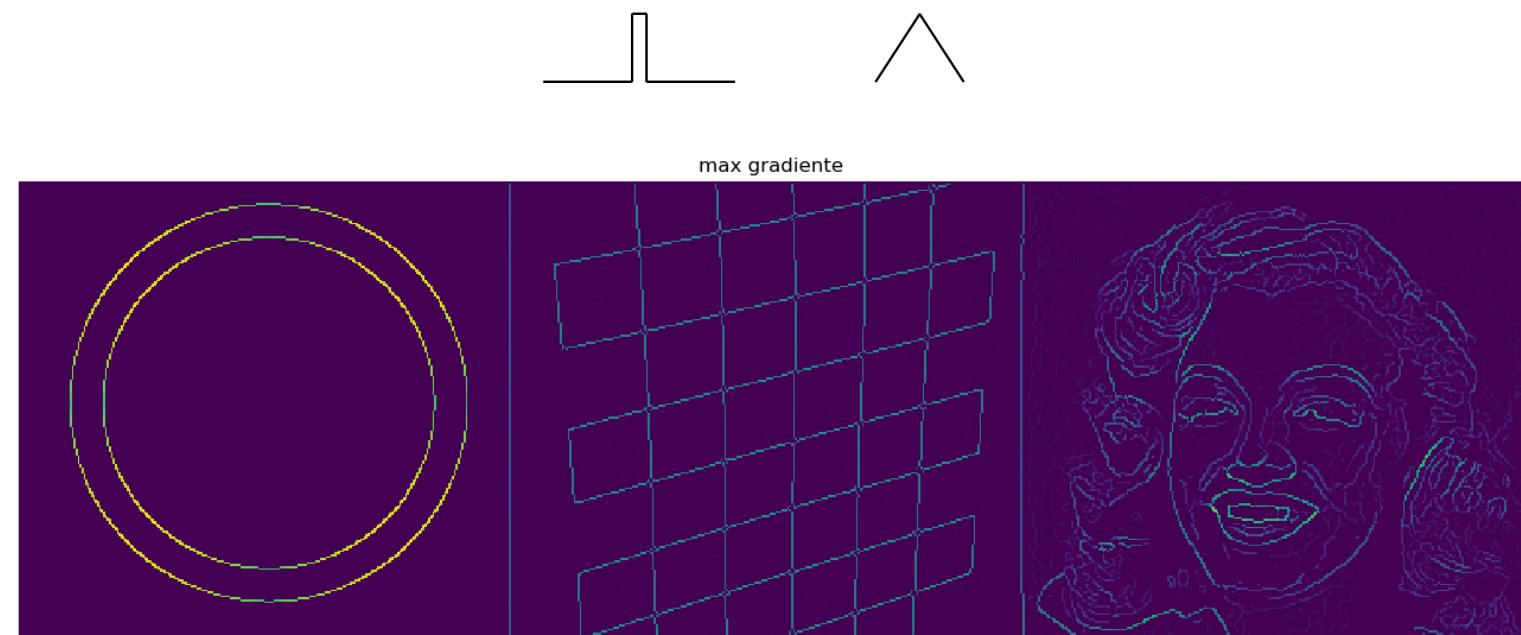
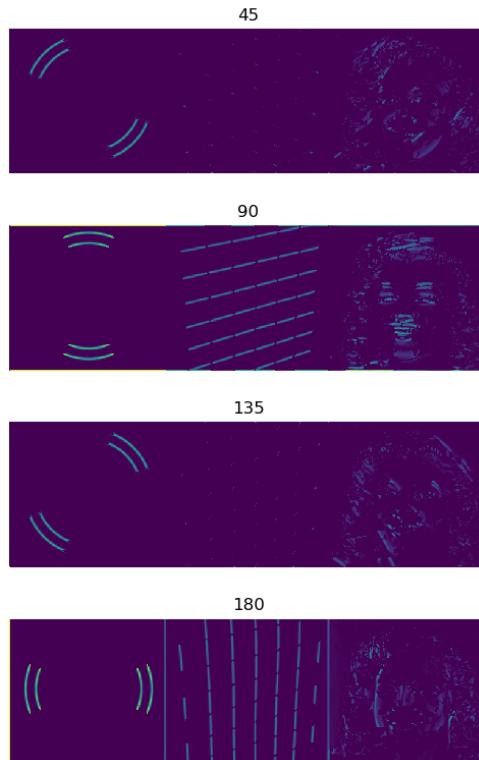


$$\text{Theta} = \text{atan}\left(\frac{S_y}{S_x}\right)$$



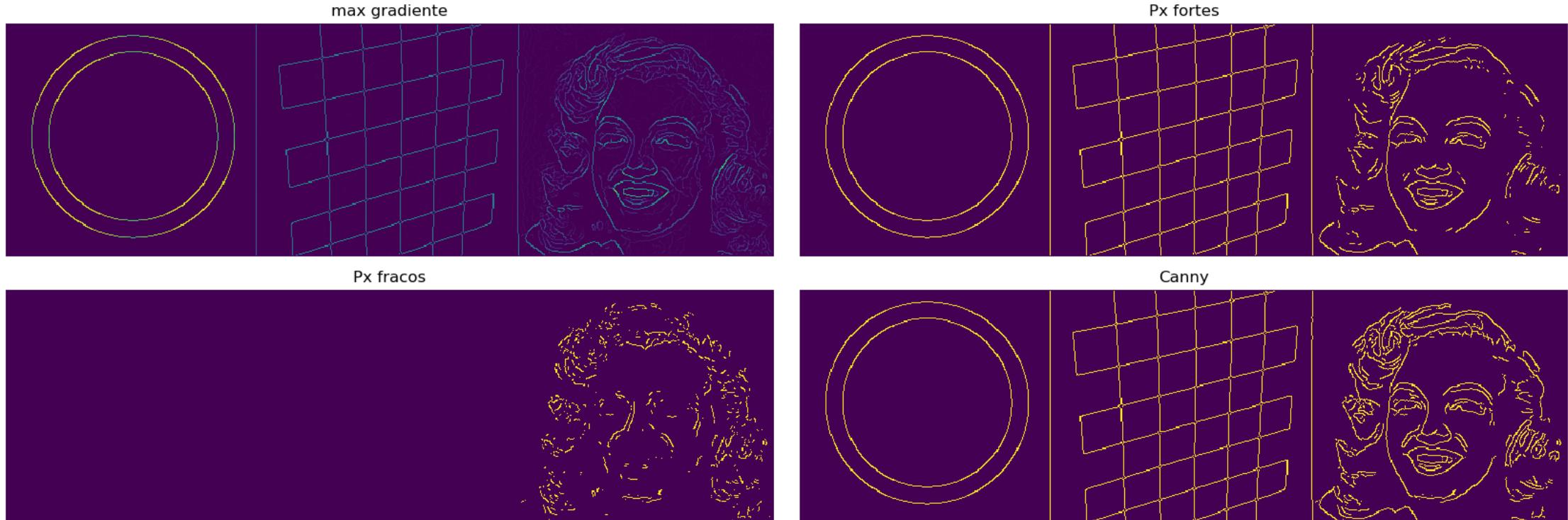
# Canny-edge detector

4. Executa-se, ao longo das linhas de junção do gradiente, uma busca dos pixels máximos nas quatro direcções de (3) e coloca-se a zero todos os pixels que não pertencem à linha de topo, resultando apenas uma linha (processo conhecido por supressão de pixels não-máximos (*non-maximal suppression*)).



# Canny-edge detector

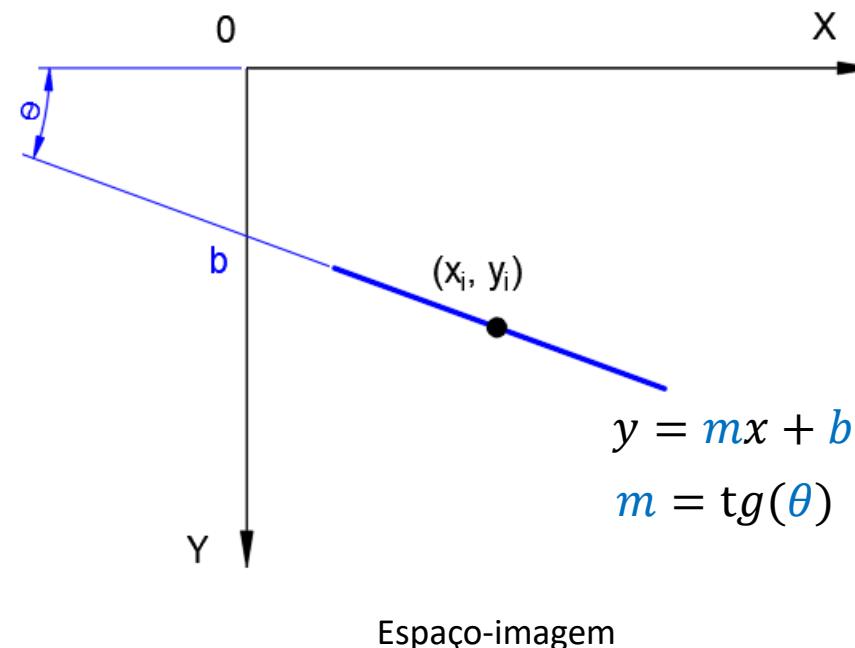
5. A segmentação das linhas é controlada por dois *thresholds*  $t_1$  e  $t_2$ , com  $t_1 > t_2$ . Todos os pixels acima de  $t_2$  (pixels fortes) pertencem às linhas de fronteira; todos os que estão abaixo de  $t_1$  (pixels fracos) não pertencem. Entre  $t_1$  e  $t_2$ , a escolha dos pixels é executada por histerese (*hysteresis*).



# Transformação de Hough

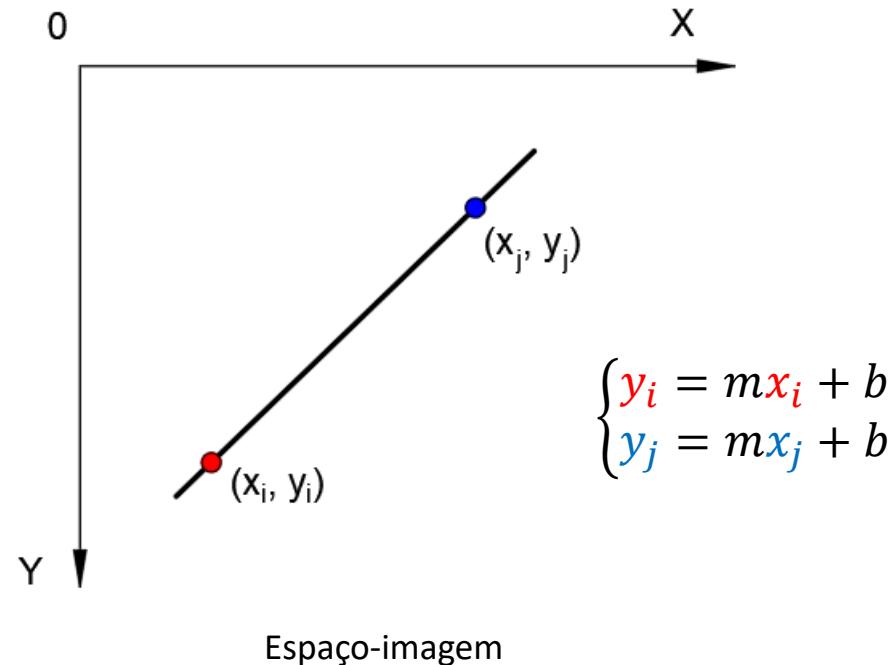
**Transformação de Hough:** é um algoritmo originalmente desenhado para detectar linhas rectas (tendo sido posteriormente alargado para linhas curvas).

A equação reduzida de uma linha, representada no espaço-imagem, escreve-se com base no conhecimento do seu declive  $m$  e do valor da ordenada na origem  $b$ . O ponto de coordenadas  $(x_i, y_i)$  pertence à linha.



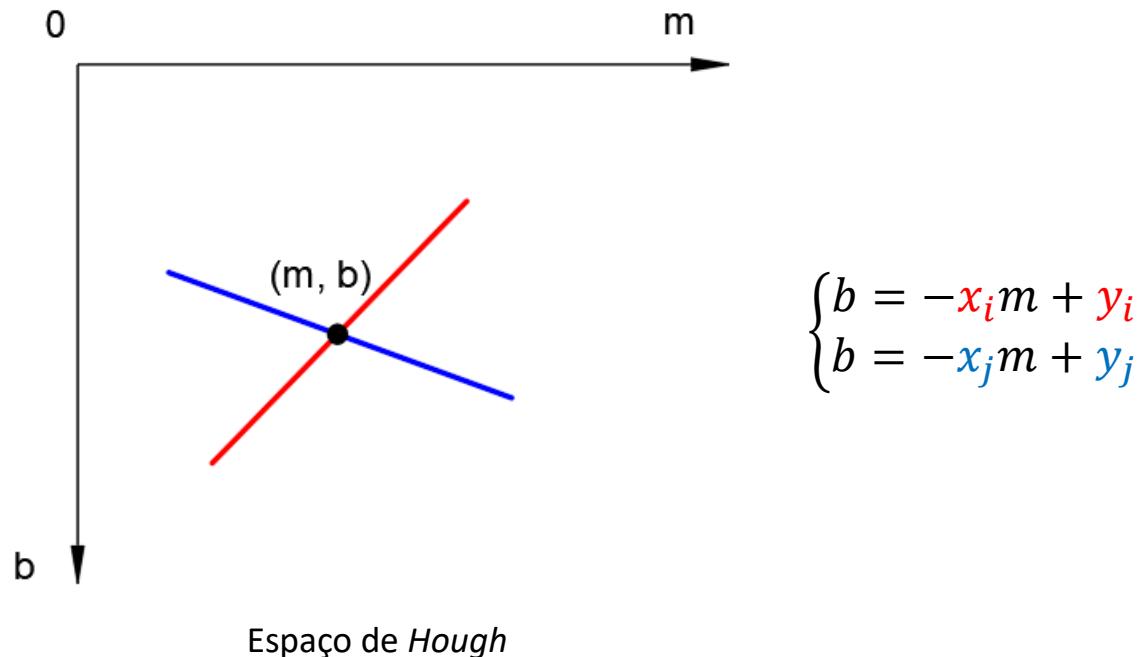
# Transformação de Hough

No espaço-imagem, dois pontos de uma mesma linha dão origem a duas equações reduzidas, em que se mantêm iguais os parâmetros m e b.



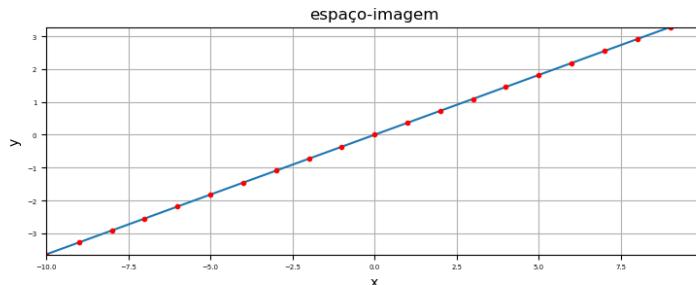
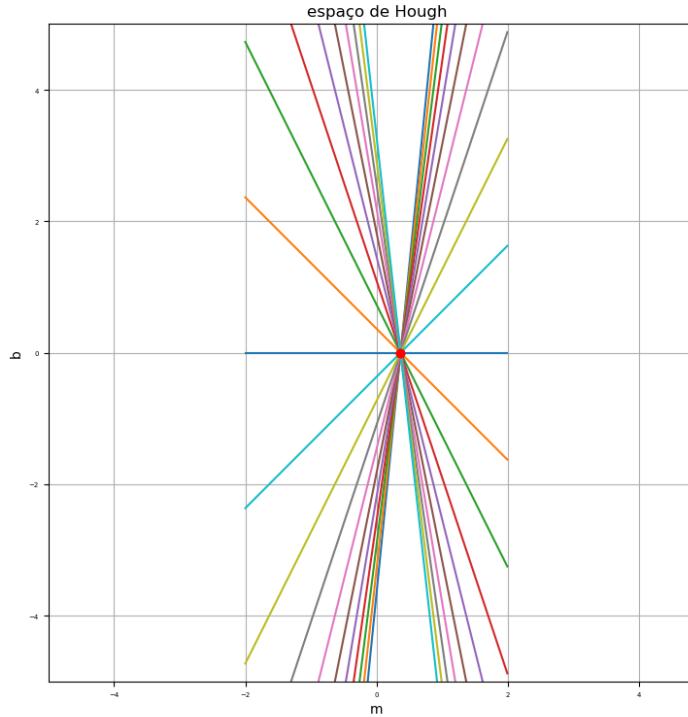
# Transformação de Hough

Reescrevendo as equações com base nos parâmetros ( $m$ ,  $b$ ), define-se o **espaço de Hough**. Neste espaço, para um dado conjunto de pontos  $\{(x_k, y_k): k>1\}$ , sobre a linha do espaço-imagem, obtém-se linhas com diferentes inclinações, intersectando-se todas num só ponto de coordenadas ( $m$ ,  $b$ ).



# Transformação de Hough

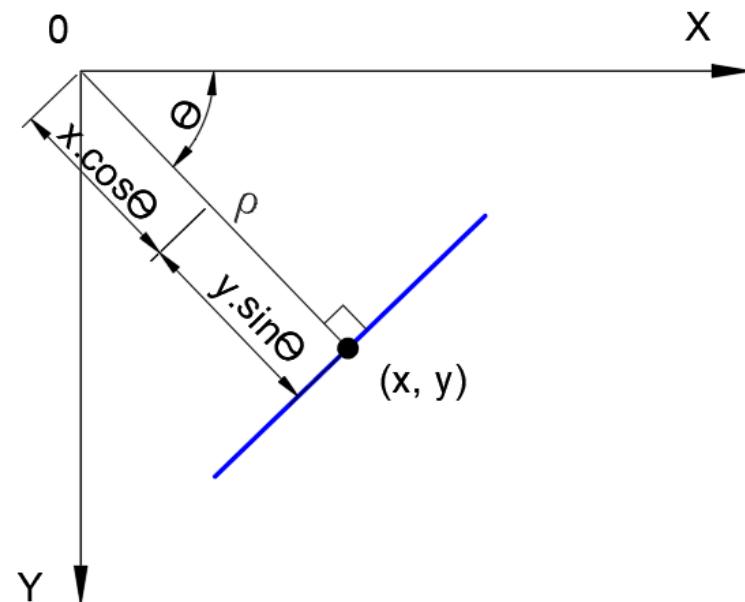
O espaço de *Hough* é, na verdade, uma “função acumuladora”, ou seja, em cada ponto desse espaço é adicionada uma unidade sempre que por ele passe uma linha. No caso da figura, o pico da soma está na coordenada  $(m, b) = (\tan(20^\circ), 0) = (0.36, 0)$ , ou seja os pontos pertencem todos à mesma recta de declive 0.36, que passa em  $(0, 0)$ .

20 pontos;  $m = \tan(20^\circ)$ ;  $b = 0$ 

# Transformação de Hough

Este processo tem o inconveniente de, quando os pontos estão alinhados numa linha vertical, o valor do declive é infinito, comprometendo os cálculos e, consequentemente, a detecção dessa linha.

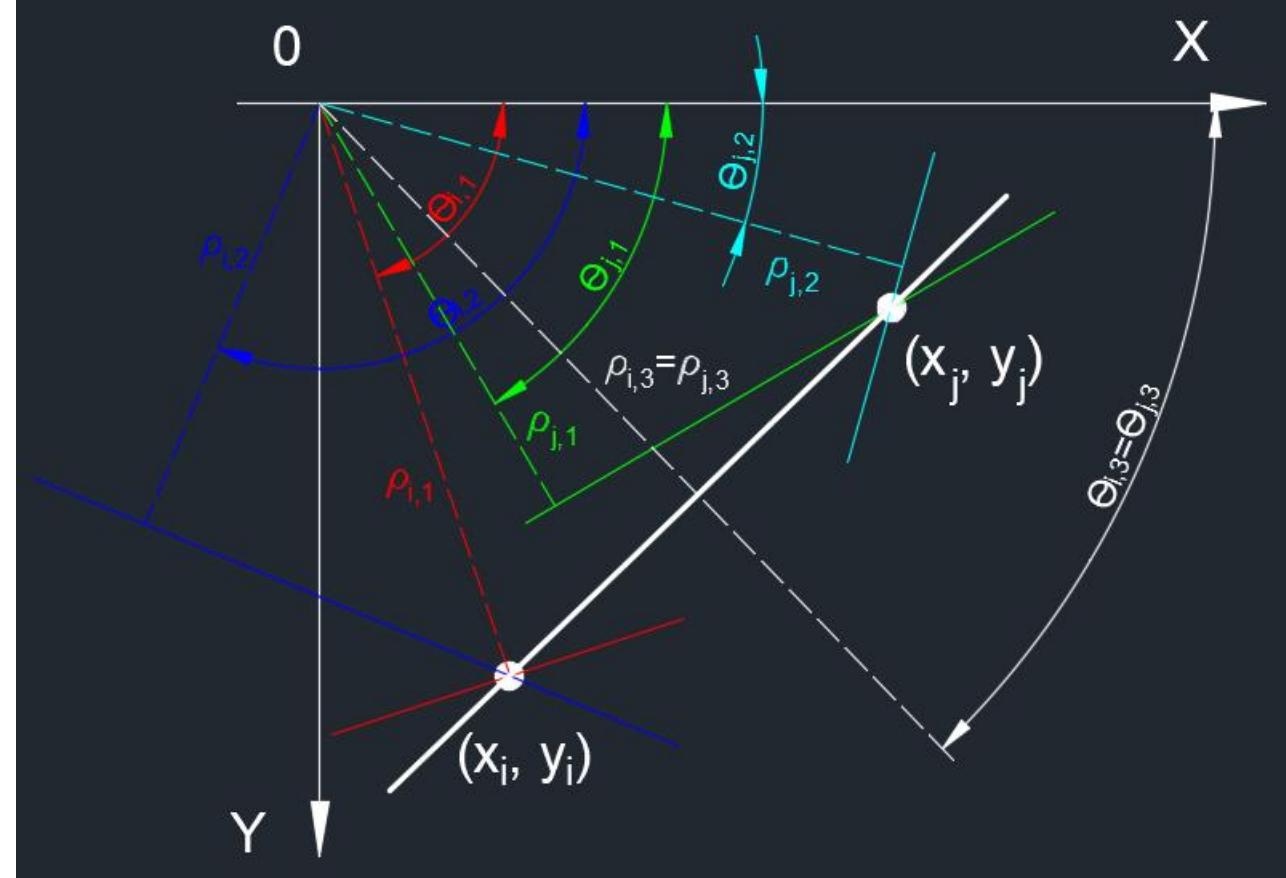
Para resolver esta questão, em vez da equação reduzida da recta, utiliza-se a equação de uma recta definida em coordenadas polares  $(\rho, \theta)$ .



$$\rho = x \cos \theta + y \sin \theta$$

# Transformação de Hough

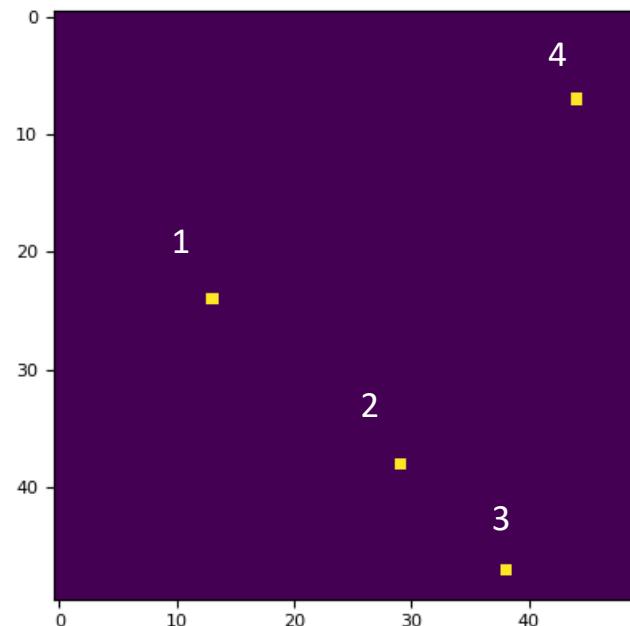
Ilustração de como construir a função acumuladora de Hough.



## Síntese do algoritmo.

1. Para cada pixel da imagem  $(x_i, y_i)$ , determinar os valores de  $\rho_{i,k}$  correspondentes aos valores de  $\theta_k$  de um dado conjunto pré-estabelecido (por exemplo  $\theta = \{\theta_k\}$ ,  $k = 0^\circ, 1^\circ, \dots, 359^\circ$ ).

$$\rho_{i,k} = x_i \cos \theta_k + y_i \cos \theta_k$$

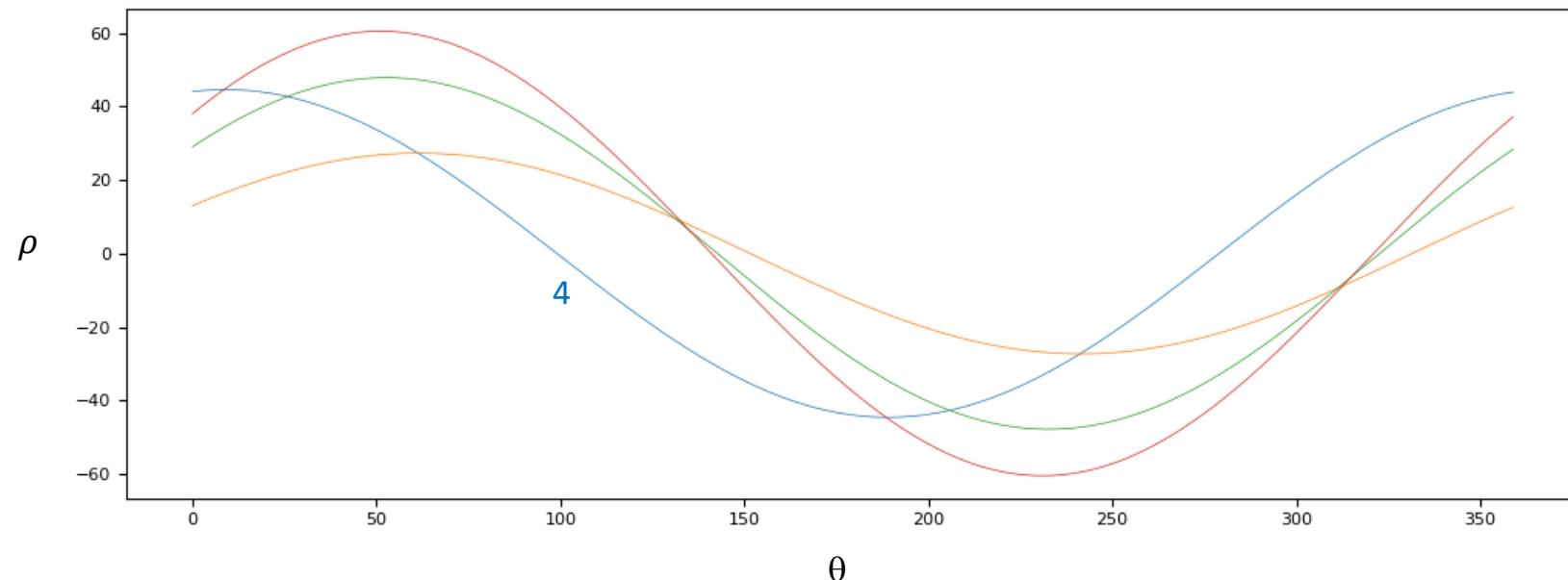


Exemplo para 4 pixels, dos quais 3 são aparentemente colineares.

# Transformação de Hough

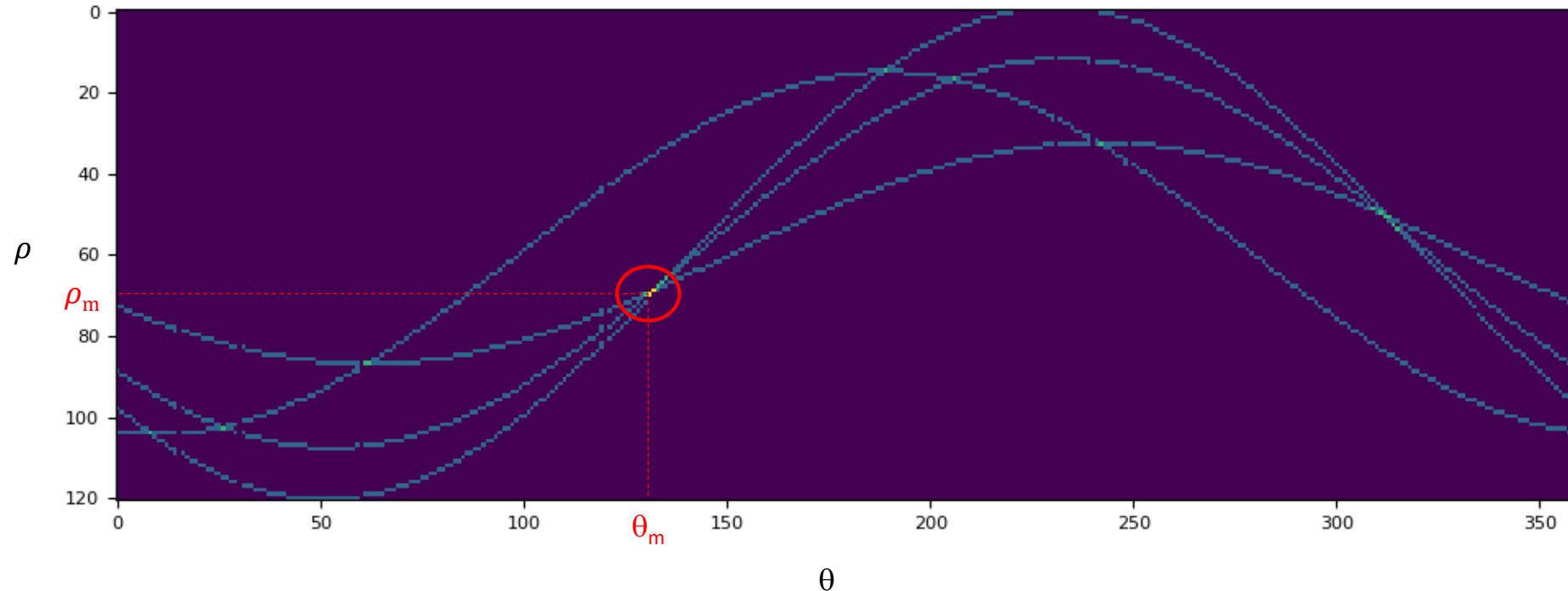
2. Construir a matriz (função acumuladora) do espaço de *Hough*:  $h = (\rho_{i,k}, \theta_k)$ . Em vez de linhas rectas, tem-se curvas senoidais.

$$h(\rho_{i,k}, \theta_k) = h(\rho_{i,k}, \theta_k) + 1$$



# Transformação de Hough

3. Extrair os máximos da matriz  $h$ :  $(\rho_m, \theta_m)$ .

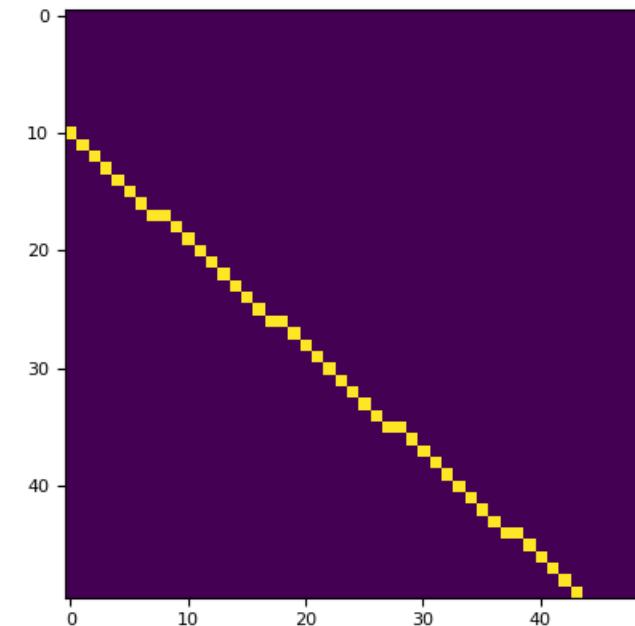


# Transformação de Hough

4. Determinar a equação reduzida da recta, por inversão da sua representação em coordenadas polares.

$$\rho = x\cos\theta + y\sin\theta \Leftrightarrow y = -\frac{\cos\theta}{\sin\theta}x + \frac{\rho}{\sin\theta}$$

$$declive = -\frac{1}{\tan\theta}; ordenada na origem = \frac{\rho}{\sin\theta}$$



A aplicação da transformação de Hough parte de uma imagem de fronteiras previamente obtida (por exemplo, a partir do Canny-Edge)

# Técnicas baseadas em morfologia matemática

Fernando Soares

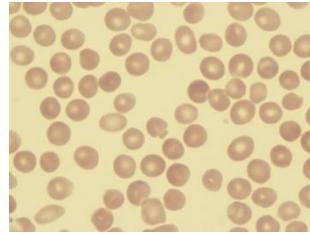
A abordagem morfológica de segmentação de imagem combina técnicas de crescimento de regiões e de detecção de transições. Esta abordagem agrupa os pixels em torno dos mínimos, ou máximos regionais da imagem. As fronteiras de grupos de pixels adjacentes estão precisamente posicionadas nas linhas de “crista” das bacias de watershed.



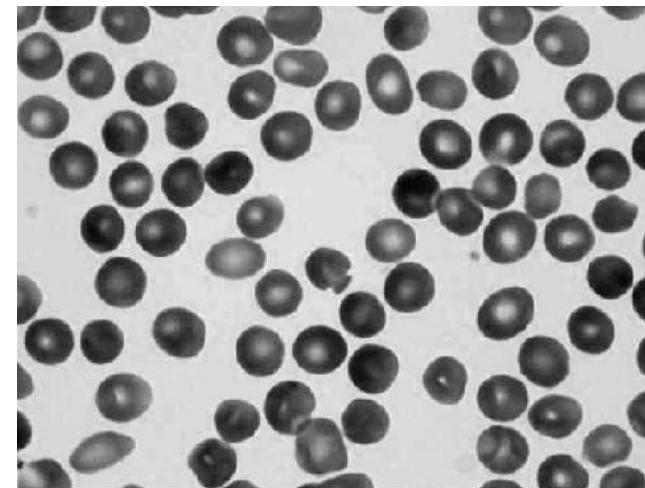
**Separação de objectos:** pode ser executada combinando várias técnicas morfológicas.

## Síntese do algoritmo.

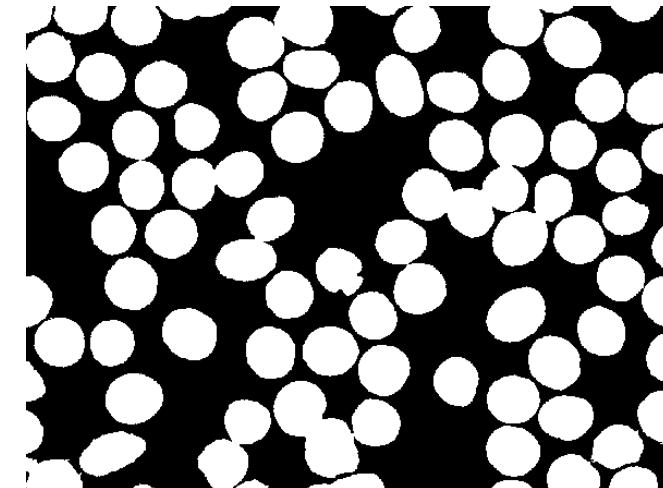
1. Threshold e da imagem de cinzentos, para se obter uma primeira aproximação dos objectos. Há alguns objectos que estão conectados.



RGB



8 bits

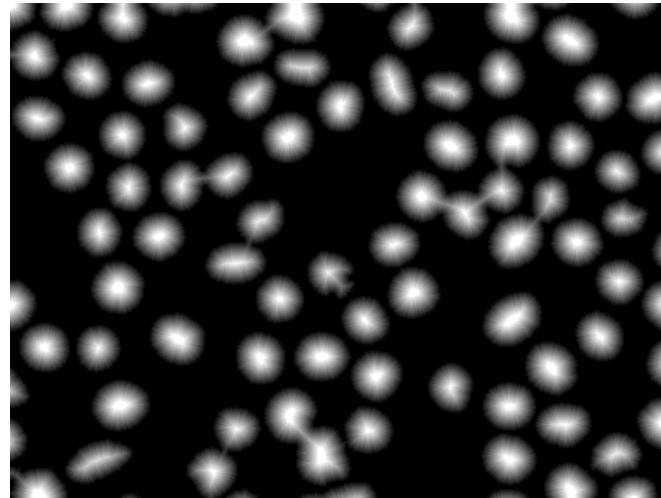


Threshold + Filtragem (TH)

# Técnicas baseadas em morfologia matemática

Fernando Soares

2. Determinação da função distância no interior dos objectos. Determinação dos máximos regionais, usando a reconstrução geodésica numérica por dilatações geodésicas sucessivas.



Função distância (D)

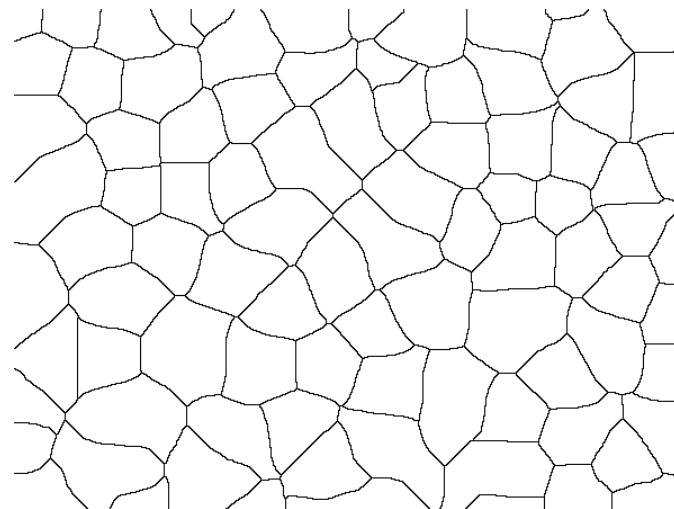


Máximos de D

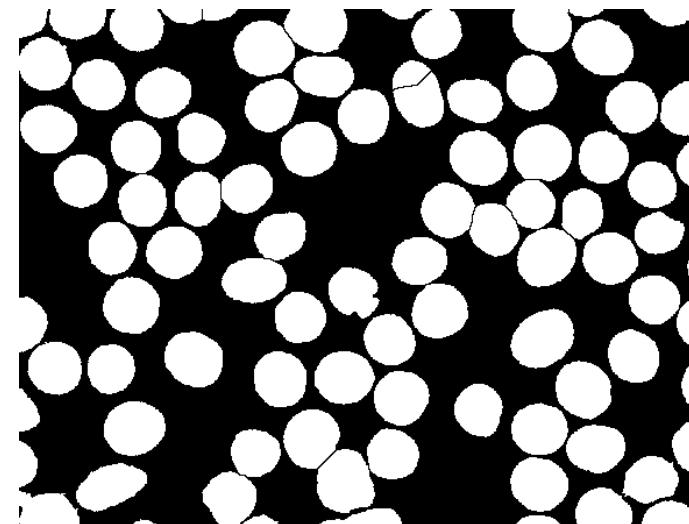
# Técnicas baseadas em morfologia matemática

Fernando Soares

3. Determinação do watershed do complementar numérico da função D, usando os máximos regionais como marcadores binários. Intersecção do watershed com a imagem de threshold: os objectos ficam separados.



Watershed de  $[\max(D)-D]$  (W)

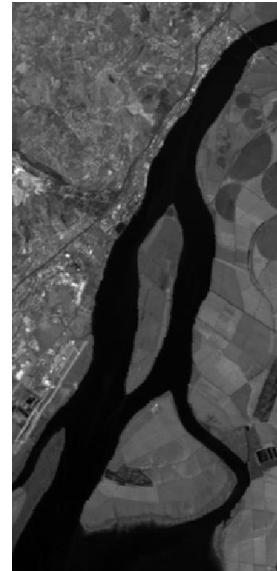


W AND TH

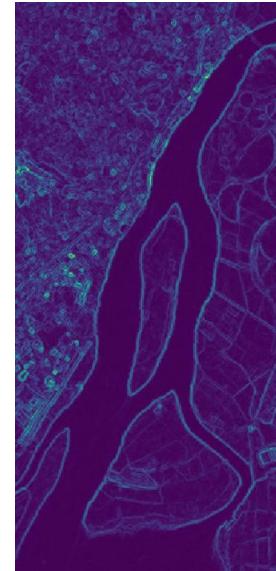
**Imagen-mosaico:** reconfiguração da imagem por unificação dos valores de cinzento das bacias de watershed.

## Síntese do algoritmo.

1. Determinação da imagem do gradiente morfológico.



Inicial

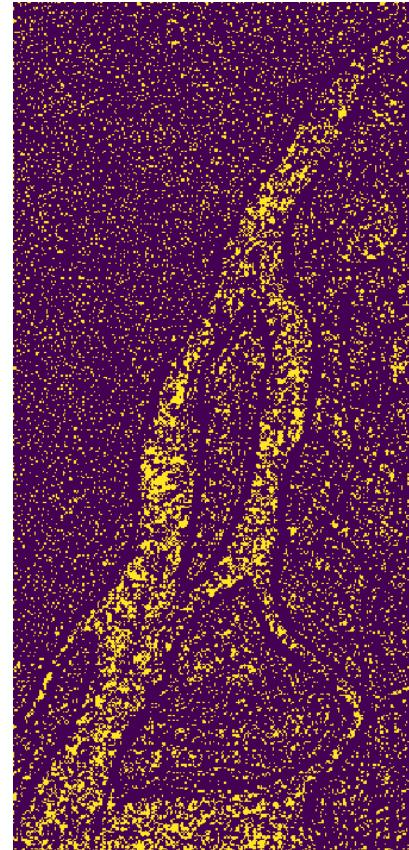


Gradiente morfológico

# Técnicas baseadas em morfologia matemática

Fernando Soares

2. Determinação da imagem de watershed do gradiente, a partir dos respectivos mínimos regionais.



Mínimos regionais

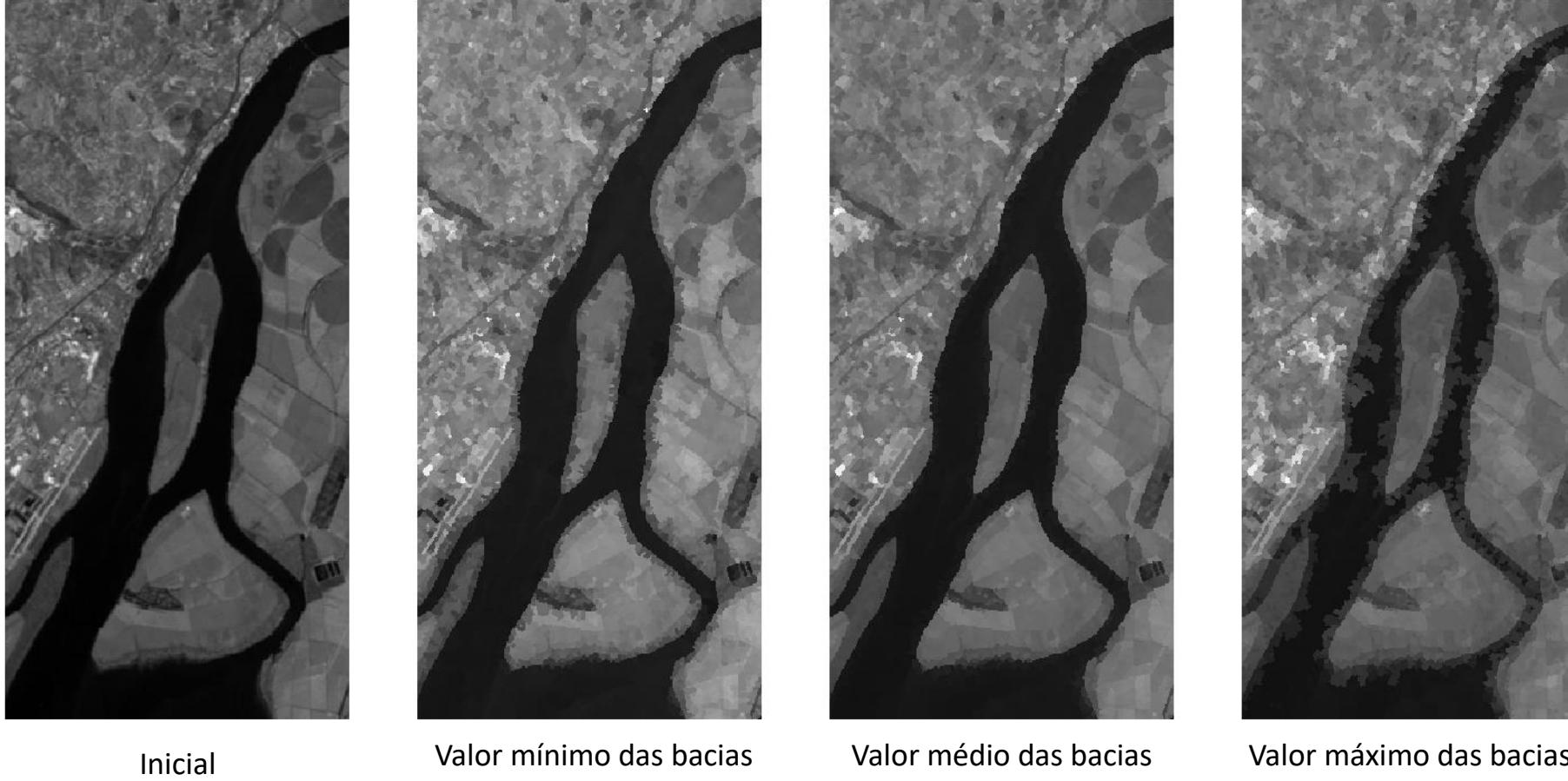


Watershed

# Técnicas baseadas em morfologia matemática

Fernando Soares

3. Atribuição de um valor de cinzento único a cada bacia de watershed.



# Medidas de similitude

**Erro Médio Quadrático (EMQ):** mede a quantidade de alteração ocorrida no processamento de uma imagem. Os valores do EMQ não são negativos e um valor de 0 significa que a imagem ou os vídeos comparados são idênticos. O EMQ entre uma imagem processada ( $t$ ) e uma imagem de referência ( $f$ ) é dado por:

$$EMQ(x, y) = \sqrt{\frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f(x, y) - t(x, y)]^2}$$

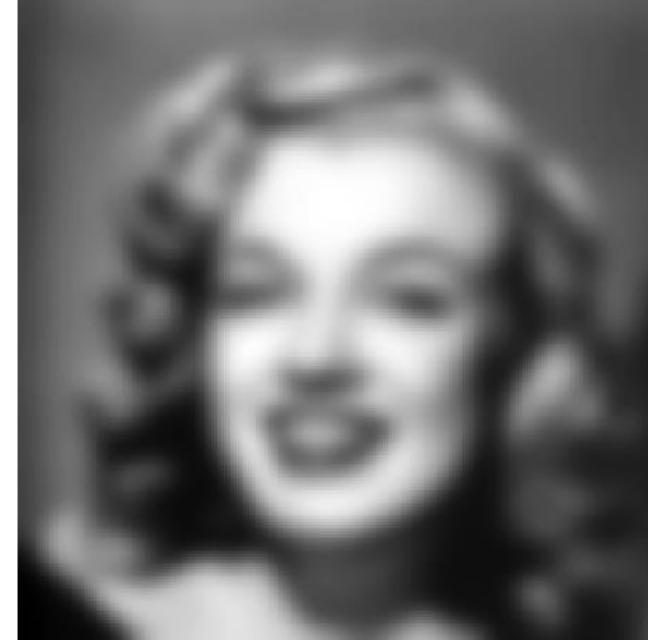
$(M, N) = (\text{nº de colunas}, \text{nº de linhas})$

## Erro Médio Quadrático (EMQ).

F



T (emq = 0.099)



# Medidas de similitude

Fernando Soares

Sejam  $f$  e  $h$  duas imagens matriciais com dimensões  $M \times N$  e  $m \times n$ , respectivamente. A medida de similitude, em cada pixel  $(x, y)$  é dada pela expressão seguinte:

$$S(x, y) = \sum_{i=-a}^a \sum_{j=-b}^b (f(x + i, y + j) - t(i, j))^2 \quad a = \text{int}\left(\frac{m}{2}\right); b = \left(\frac{n}{2}\right)$$

Com esta medida pode-se verificar as possíveis correspondências de um certo padrão em vários locais de uma imagem.

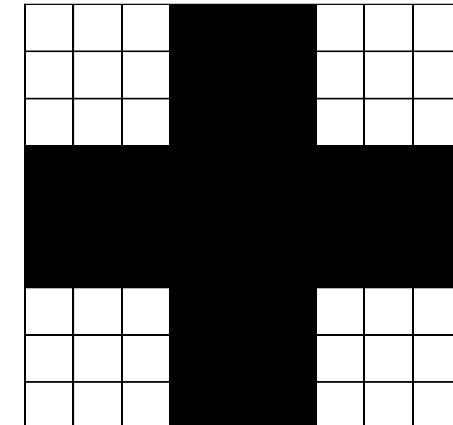
# Medidas de similitude

Exemplo: identificar o movimento do objecto do instante  $t_1$  para o instante  $t_2$ .

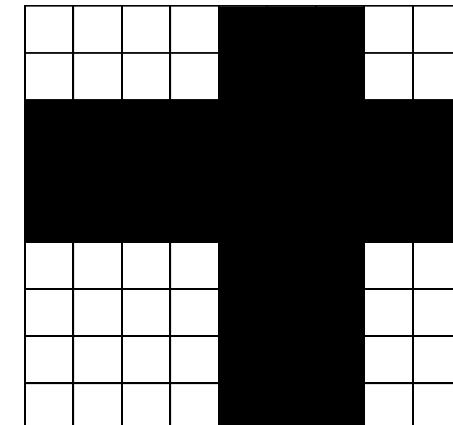
255	255	255	0	0	0	255	255	255
255	255	255	0	0	0	255	255	255
255	255	255	0	0	0	255	255	255
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
255	255	255	0	0	0	255	255	255
255	255	255	0	0	0	255	255	255
255	255	255	0	0	0	255	255	255

255	255	255	255	0	0	0	255	255
255	255	255	255	0	0	0	255	255
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
255	255	255	255	0	0	0	255	255
255	255	255	255	0	0	0	255	255
255	255	255	255	0	0	0	255	255
255	255	255	255	0	0	0	255	255

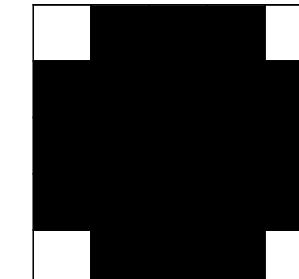
Na memória



No ecrã



255	0	0	0	255
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
255	0	0	0	255



$h$  ( $5 \times 5$ ) retirado do instante  $t_1$

# Medidas de similitude

No exemplo, o valor de S é muito elevado. Como tal não há correspondência.

								2	
255	255	255	255	0	0	0	255	255	
255	255	255	255	0	0	0	255	255	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
255	255	255	255	0	0	0	255	255	
6	255	255	255	255	0	0	0	255	255
255	255	255	255	0	0	0	255	255	
255	255	255	255	0	0	0	255	255	
255	255	255	255	0	0	0	255	255	
255	0	0	0	0	255				
0	0	0	0	0	0				
0	0	0	0	0	0				
0	0	0	0	0	0				
255	0	0	0	0	255				

$$\begin{aligned} S(2,6) &= \sum_{i=-2}^2 \sum_{j=-2}^2 (f(2+i, 6+j) - h(i, j))^2 = \\ &= (0 - 255)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 255)^2 + \\ &\quad + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (0 - 0)^2 + \\ &\quad + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (0 - 0)^2 + \\ &\quad + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (0 - 0)^2 + \\ &\quad + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (0 - 0)^2 + \\ &\quad + (255 - 255)^2 + (255 - 0)^2 + (255 - 0)^2 + (255 - 0)^2 + (0 - 255)^2 = 1\,170\,450 \end{aligned}$$

# Medidas de similitude

O valor de S é zero, logo há correspondência total. Está encontrada a posição pretendida.

				5					
255	255	255	255	0	0	0	255	255	
255	255	255	255	0	0	0	255	255	
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
t <sub>2</sub>	3								
0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	
255	255	255	255	0	0	0	255	255	
255	255	255	255	0	0	0	255	255	
255	255	255	255	0	0	0	255	255	
255	255	255	255	0	0	0	255	255	
255	0	0	0	0	255				
0	0	0	0	0	0				
0	0	0	0	0	0				
0	0	0	0	0	0				
255	0	0	0	0	255				

$$\begin{aligned}S(5,3) &= \sum_{i=-2}^2 \sum_{j=-2}^2 (f(5+i, 3+j) - h(i,j))^2 = \\&= (255 - 255)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (255 - 255)^2 + \\&\quad + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + \\&\quad + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + \\&\quad + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + \\&= (255 - 255)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (255 - 255)^2 = 0\end{aligned}$$

# Medidas de similitude

Na realidade a imagem tem sempre um certo ruído. Daí que o objectivo é encontrar a correspondência onde o valor de  $S$  é mínimo e não especificamente o valor zero.

							5		
254	253	252	253	3	0	2	249	253	
250	255	251	252	1	2	0	253	254	
0	1	3	3	2	0	1	2	0	
t <sub>2</sub>	1	0	3	0	2	1	1	0	3
	1	3	2	0	1	1	2	2	0
253	254	250	252	0	2	1	255	254	
252	254	253	251	3	2	0	250	253	
255	249	253	250	0	1	1	253	251	
251	253	254	251	2	3	3	254	252	
	252	0	0	0	250				
	0	1	1	2	3				
	3	0	2	1	0				
	2	0	3	0	2				
	250	0	1	0	251				

$$\begin{aligned} S(5,3) &= \sum_{i=-2}^2 \sum_{j=-2}^2 (f(5+i, 3+j) - h(i,j))^2 = \\ &= (252 - 252)^2 + (1 - 0)^2 + (2 - 0)^2 + (0 - 0)^2 + (253 - 250)^2 + \\ &\quad + (3 - 0)^2 + (2 - 1)^2 + (0 - 1)^2 + (1 - 2)^2 + (2 - 3)^2 + \\ &\quad + (0 - 3)^2 + (2 - 0)^2 + (1 - 2)^2 + (1 - 1)^2 + (0 - 0)^2 + \\ &\quad + (0 - 2)^2 + (1 - 0)^2 + (1 - 3)^2 + (2 - 0)^2 + (2 - 2)^2 + \\ &= (252 - 250)^2 + (2 - 0)^2 + (3 - 1)^2 + (3 - 0)^2 + (255 - 251)^2 = 89 \end{aligned}$$

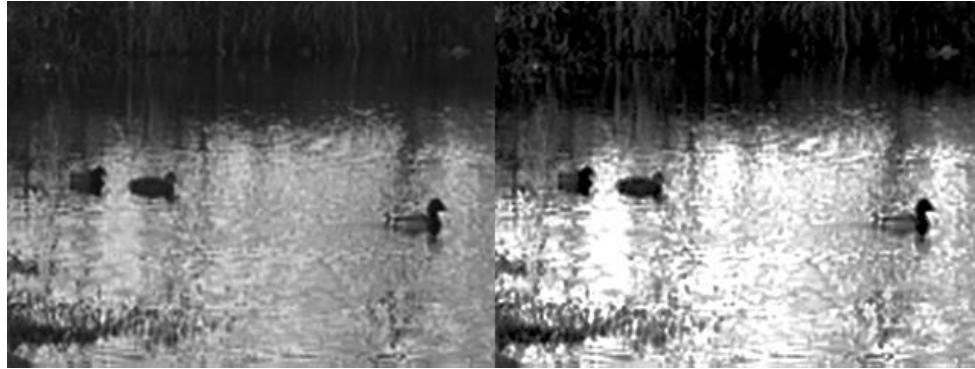
**Correlação cruzada:** é o processo de fazer correr uma matriz  $t$  de pequenas dimensões ( $m \times n$ ) por todas as posições de uma matriz  $f$  de maiores dimensões ( $M \times N$ ), calculando o valor da correlação em cada posição (multiplicando os pixels de  $t$  pelos pixels de  $f$  e somando todas essas parcelas).

$$CC(x, y) = \sum_{i=-a}^a \sum_{j=-b}^b f(x + i, y + j) \times t(i, j) \quad a = \text{int}\left(\frac{m}{2}\right); b = \left(\frac{n}{2}\right)$$

- Os valores de CC estão compreendidos entre 0 e  $\infty$ .
- A correlação é sensível às variações de luminosidade.

# Medidas de similitude

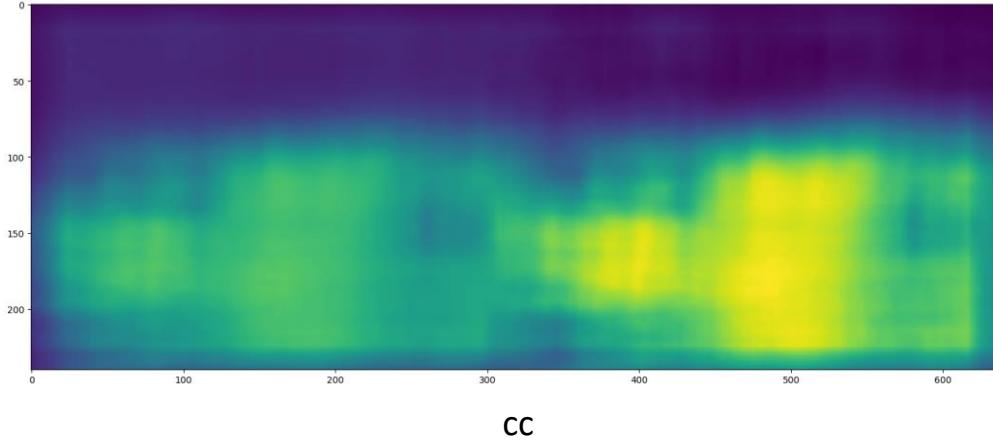
Correlação cruzada:



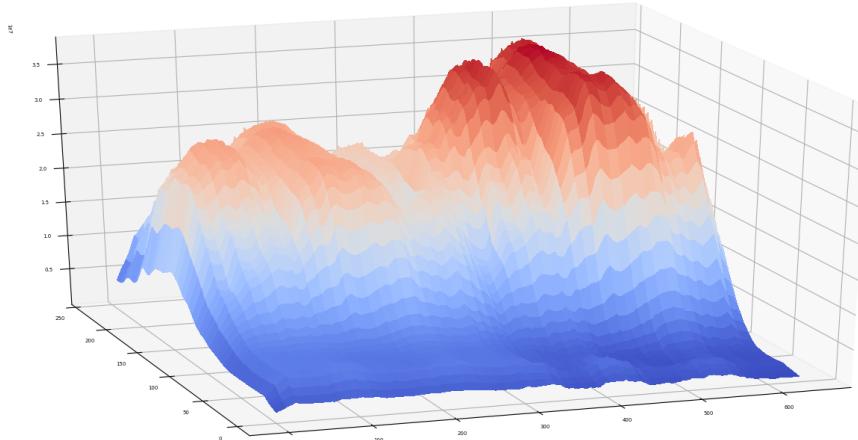
f



t



CC



**Correlação cruzada normalizada:** Calcula-se pela seguinte expressão:

$$CCN(x, y) = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [(f_{u,v} - \bar{f}_{u,v}) \times (t - \bar{t})]}{\sqrt{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f_{u,v} - \bar{f}_{u,v}]^2} \times \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [t - \bar{t}]^2}}$$

- A subtracção das médias torna o processo insensível às diferenças de brilho.
- A divisão pelos quadrados das diferenças torna o processo insensível às diferenças de contraste.
- Os valores de CCN estão compreendidos entre -1 e 1.

# Medidas de similitude

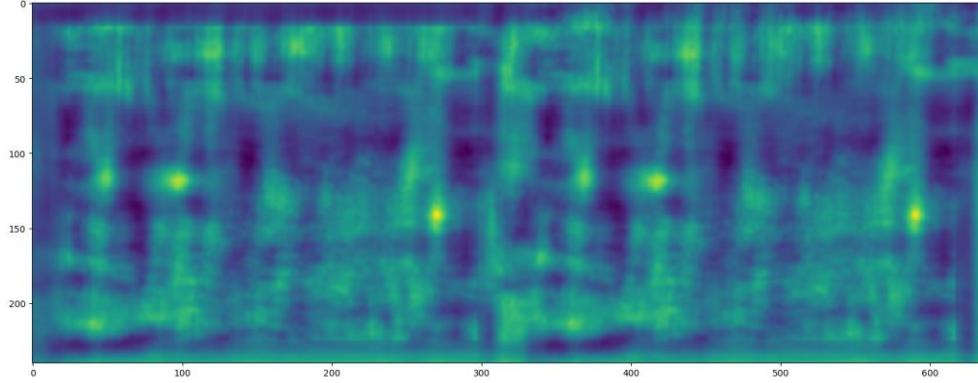
## Correlação cruzada normalizada.



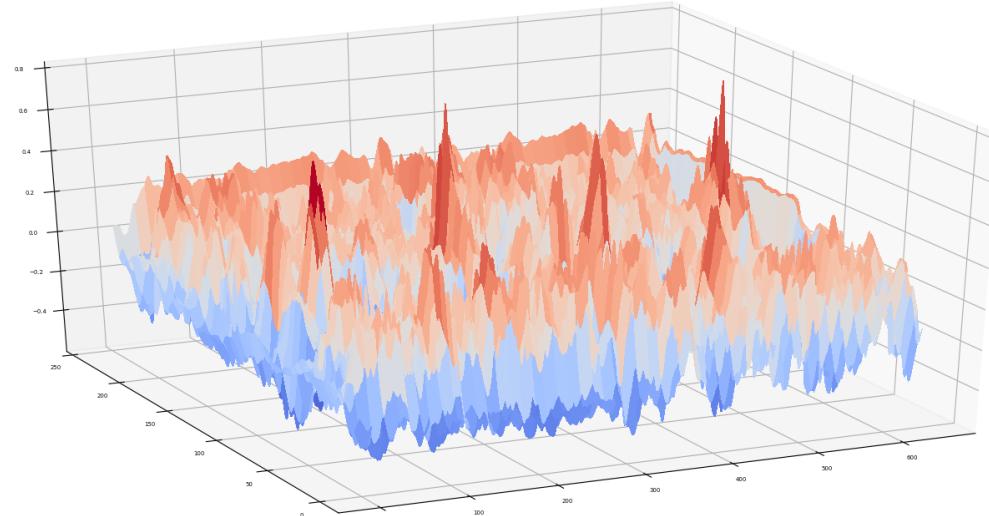
f



t

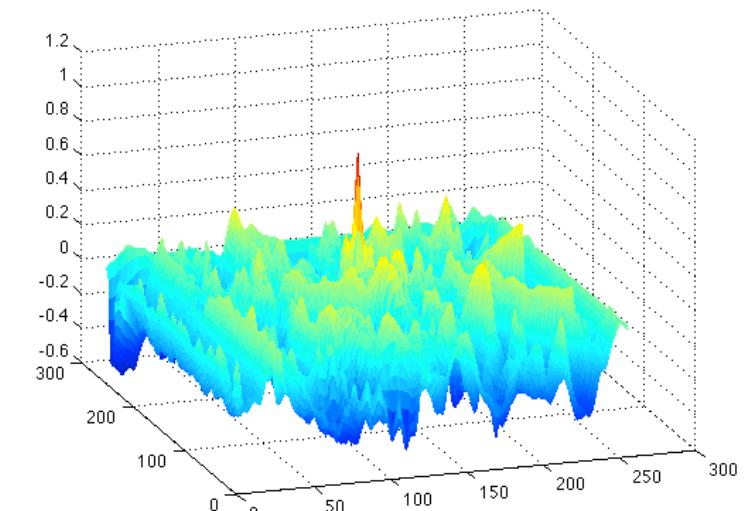
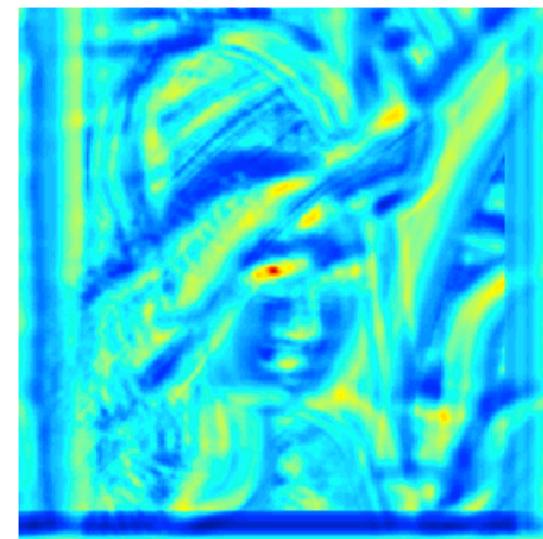
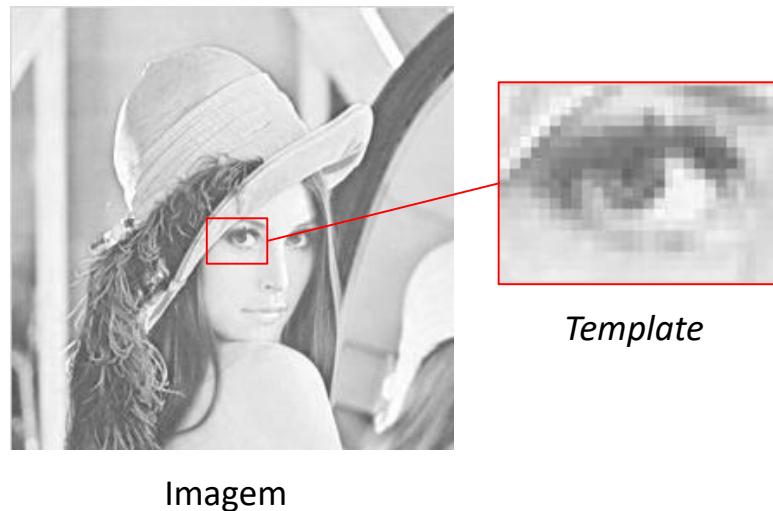


CC



# Template Matching

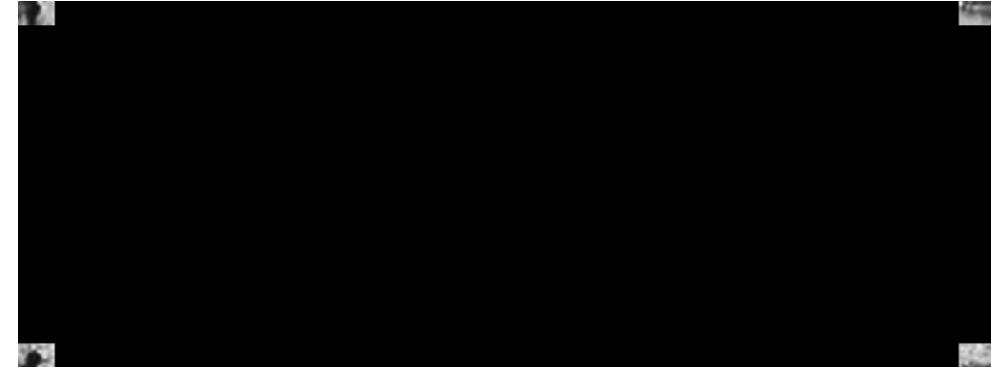
Técnica de segmentação que visa encontrar objectos semelhantes a um dado objecto que se designa por *template*. É geralmente realizada com base no cálculo da correlação cruzada normalizada.



Correlação cruzada normalizada

# Medidas de similitude

**Correlação cruzada (com a transformada de Fourier):** aplica-se o teorema da convolução (como antes enunciado).



# Correlação de fase

Técnica que permite determinar uma translação de imagem, através da determinação do espectro de potência cruzado (*cross power spectrum*). Sejam  $f_1$  e  $f_2$  duas imagens deslocadas uma da outra:

O desfasamento entre as duas imagens pode ser calculado da seguinte forma:

- a)  $g_1$  = transformada de Fourier da imagem  $f_1$ ;
- b)  $g_2$  = transformada de Fourier da imagem  $f_2$ ;
- c) Cálculo do espectro de potência cruzado:

$$CF = \text{real} \left[ \text{IFT} \left( \frac{g_1 \times \overline{g_2}}{|g_1 \times \overline{g_2}|} \right) \right]$$

IFT: inversa da transformada de Fourier

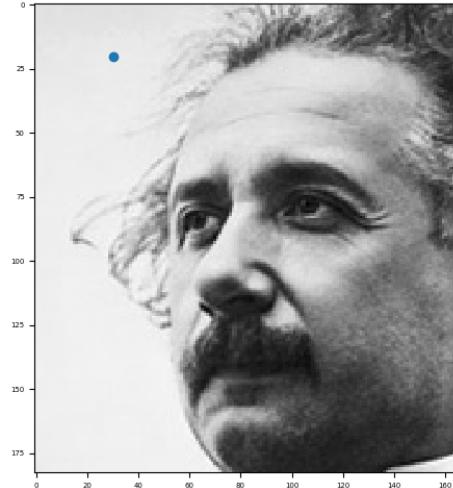
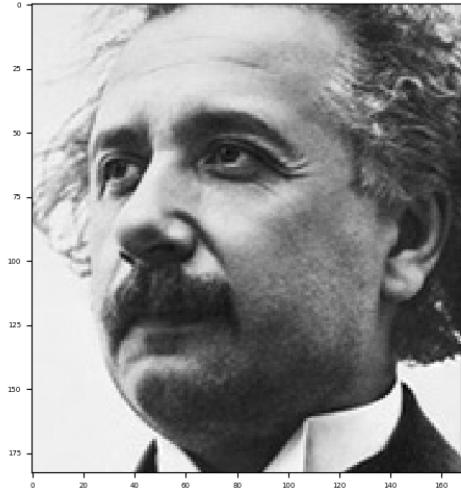
real(z): parte real do número complexo z

$\overline{g_2}$  : conjugado do número complexo  $g_2$

- d) Determinação do pico máximo de CF.

# Correlação de fase

Exemplo: translação de imagem.

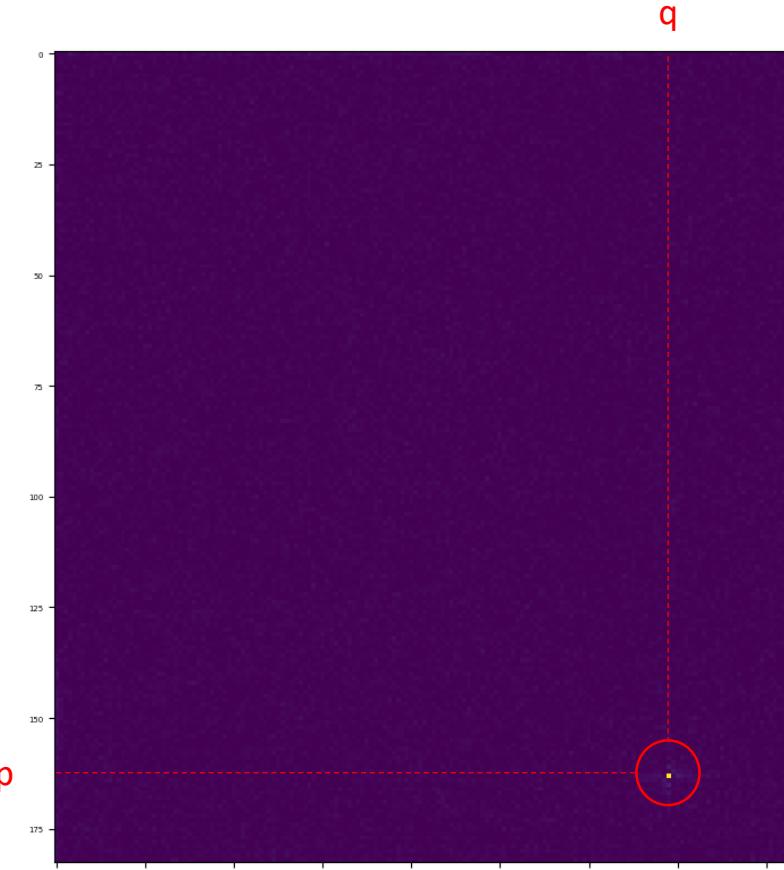


$f_1$

$f_2$

Pico = máximo de CF

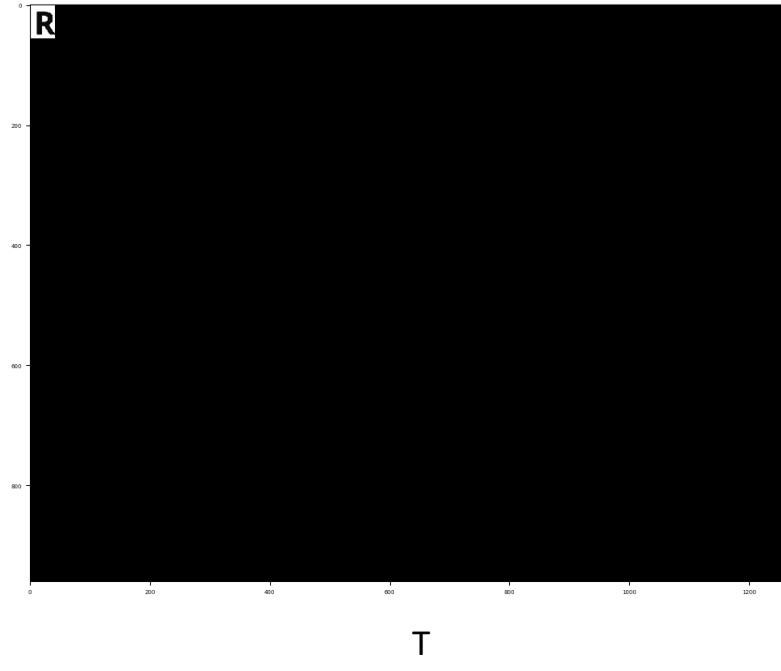
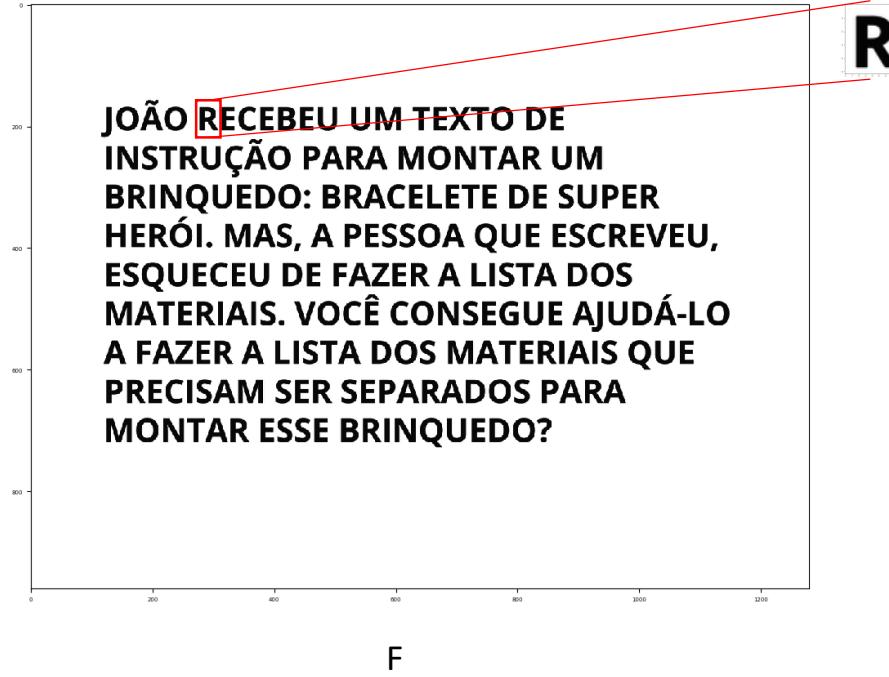
Deslocamento =  $(M - p, N - q)$



CF

# Correlação de fase

Exemplo: *template matching*.



# Correlação de fase

Exemplo: *template matching*:

