Ciências ULisboa

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.

Filtro digital 2D

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.

$$\begin{array}{|c|c|}\hline w1 & w2 \\ \hline w4 & w5 \\ \hline \end{array} \longrightarrow \begin{array}{|c|c|c|}\hline w1 & w2 & 0 \\ \hline w3 & w4 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} \qquad \begin{array}{|c|c|}\hline w1 & w2 \\ \hline w3 & w4 \\ \hline w5 & w6 \\ \hline \end{array} \longrightarrow \begin{array}{|c|c|c|}\hline w1 & w2 & 0 \\ \hline w3 & w4 & 0 \\ \hline w5 & w6 & 0 \\ \hline \end{array}$$

Correlação / Convolução

Fernando Soares

Sendo f uma imagem matricial, define-se a operação de **correlação** ("⨂") 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 3×3 tem-se:

$$H \otimes f(x,y) = \sum_{i=-1}^{1} \sum_{j=-1}^{1} H(i,j) \times f(x+i, y+j) =$$

$$= H(-1,-1) \times f(x-1, y-1) + H(-1,0) \times f(x-1, y) + H(-1,-1) \times f(x-1, y+1) +$$

$$+ H(0,-1) \times f(x, y-1) + H(0,0) \times f(x,y) + H(0,1) \times f(x, y+1) +$$

$$+ H(1,-1) \times f(x+1, y-1) + H(1,0) \times f(x+1, y) + H(1,1) \times f(x+1, y+1)$$

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

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

Fernando Soares

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).

f

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

Filtro H

h00	h01	h02
h10	h11	h12
h20	h21	h22

```
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

Fernando Soares

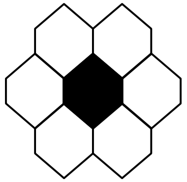
Para os pixels de fronteira da imagem há que fazer uma adaptação para a executar o processo de filtragem. Uma das quatro seguintes opções pode ser seguida:

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.

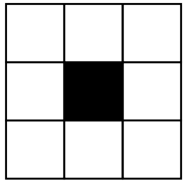
Propriedades dos filtros digitais

Fernando Soares

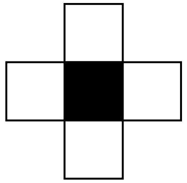
Para além dos valores dos coeficientes associados, a forma e a dimensão são características segundo as quais H também pode variar. Eis alguns exemplos de geometrias de H.



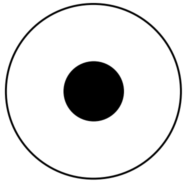
hexágono



quadrado

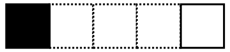


diamante



círculo

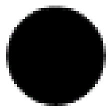
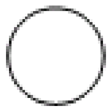
segmento



par de pontos

Propriedades dos filtros digitais

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

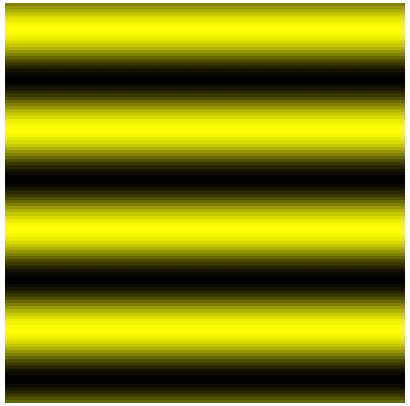
	Isotrópico	**Anisotrópico**
Convexo	Disco 	Segmento 
Não convexo	Contorno do disco 	Par de pontos 

Filtragem espacial

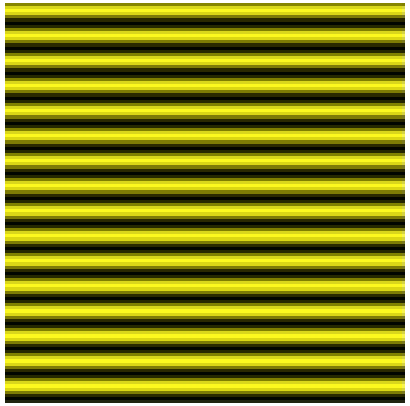
Fernando Soares

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

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



Sinal de baixa frequência



Sinal de alta frequência

Filtragem espacial

Fernando Soares

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.

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

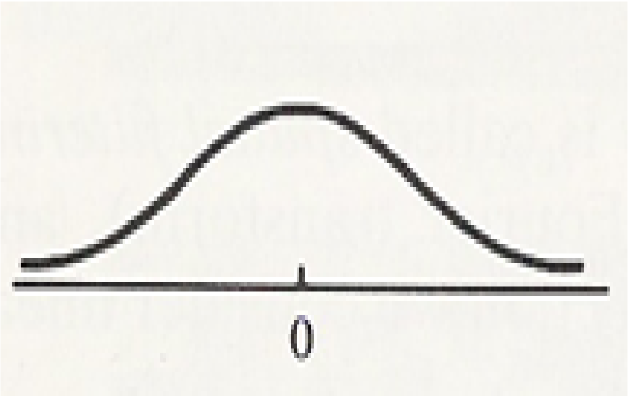
f_1: imagem filtrada
f: imagem inicial
H: filtro

Quaisquer outros filtros são designados por filtros não lineares.

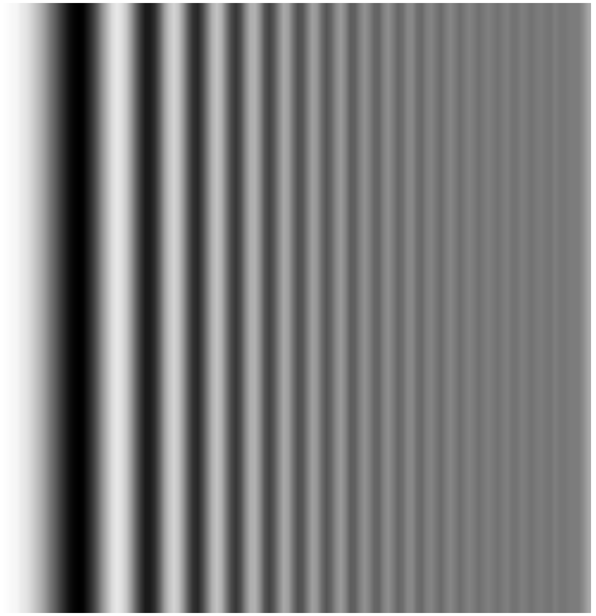
Filtro passa-baixa

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 <u>coeficientes positivos</u>, devendo a respectiva <u>soma ser igual a 1</u>.



Função-resposta do filtro PB



Efeito

Filtro passa-baixa

Fernando Soares

Por exemplo, no filtro passa-baixa da média aritmética 3 x 3, tem-se:

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

Função

1/9 ×

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

Coeficientes

	$PB_{i,j}$	

Valor calculado

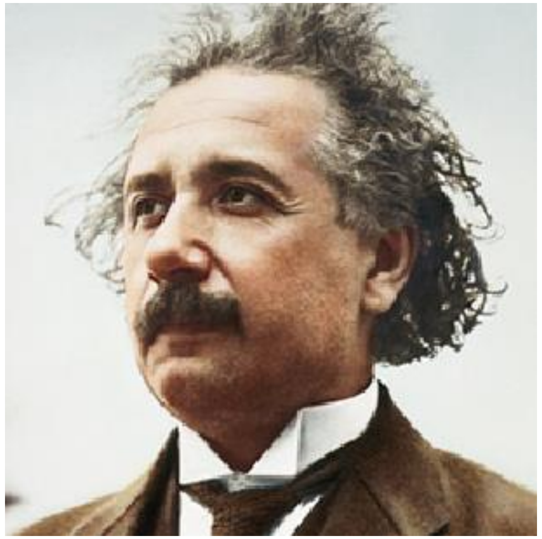
Por convolução, tem-se:

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

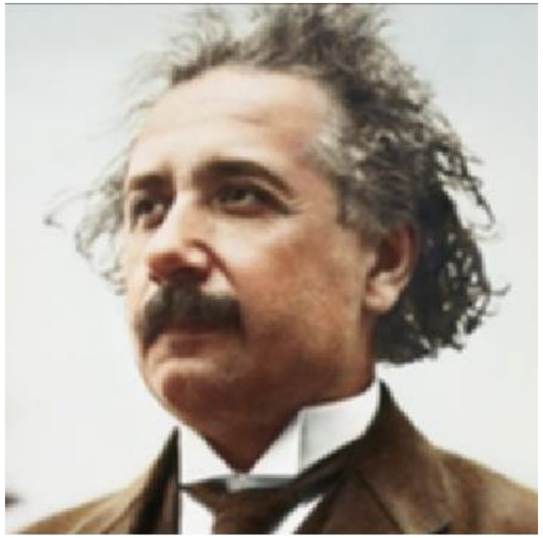
Filtro passa-baixa

Fernando Soares

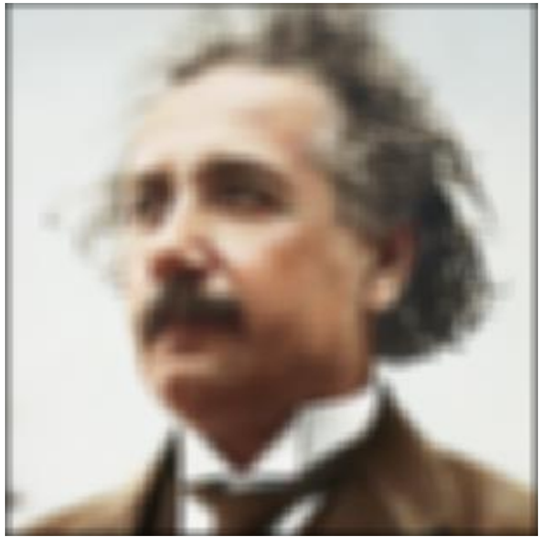
Média: É o mais simples filtro linear passa-baixa. Todos os coeficientes são iguais. Calcula-se a média dos tons de cinzento no interior da janela (H) e substitui-se o pixel central da janela pelo valor resultante (por esta razão as janelas são normalmente quadradas com dimensão ímpar (3 x 3, 5 x 5,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



Inicial



Média 3×3



Média 9×9

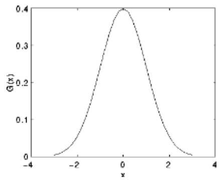
$$H_{3\times3} = \frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$H_{9\times9} = \frac{1}{81} \times \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix}$$

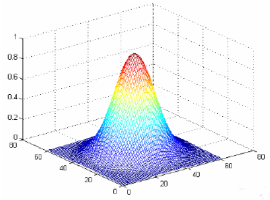
Filtro passa-baixa

Fernando Soares

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 7×7, centrada em (0, 0), com σ^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	1	1	1	0	0
0	1	3	5	3	1	0
0	1	5	8	5	1	0
0	1	3	5	3	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

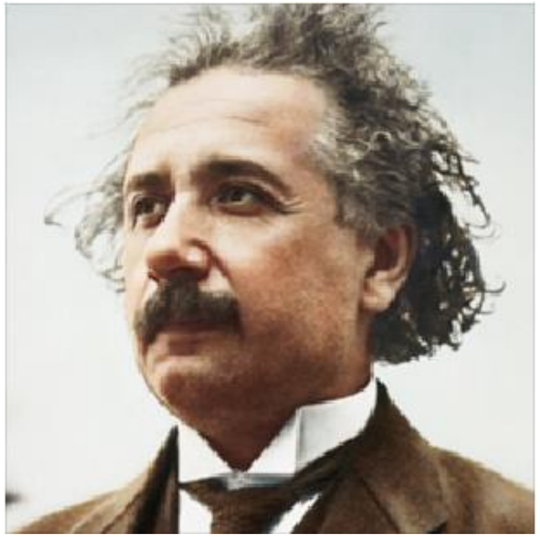
$\times \frac{1}{49} =$

Filtro passa-baixa

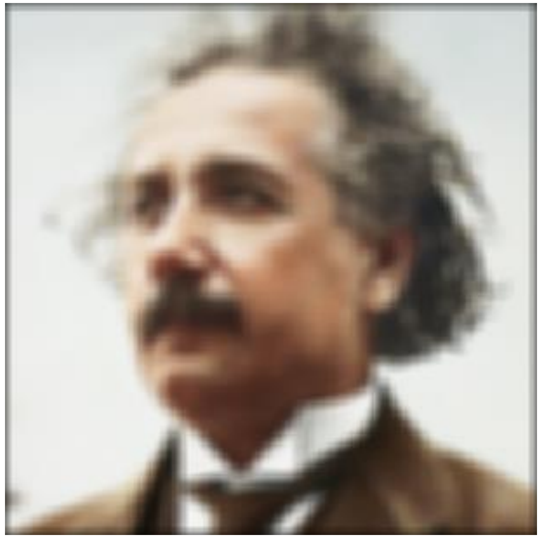
Fernando Soares

O filtro gaussiano é uma média ponderada, com maior peso aplicado ao pixel central, diminuíndo 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 (σ^2 = 0.5)

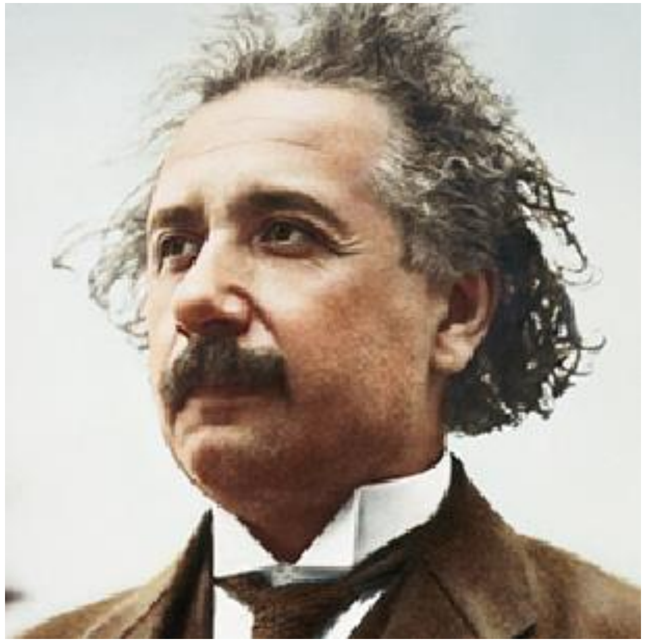


Gaussiana (σ^2 = 5)

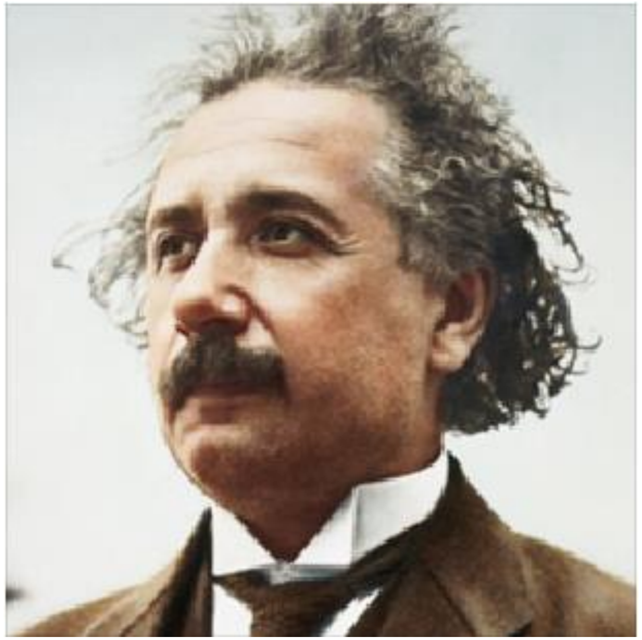
Filtro passa-baixa

Fernando Soares

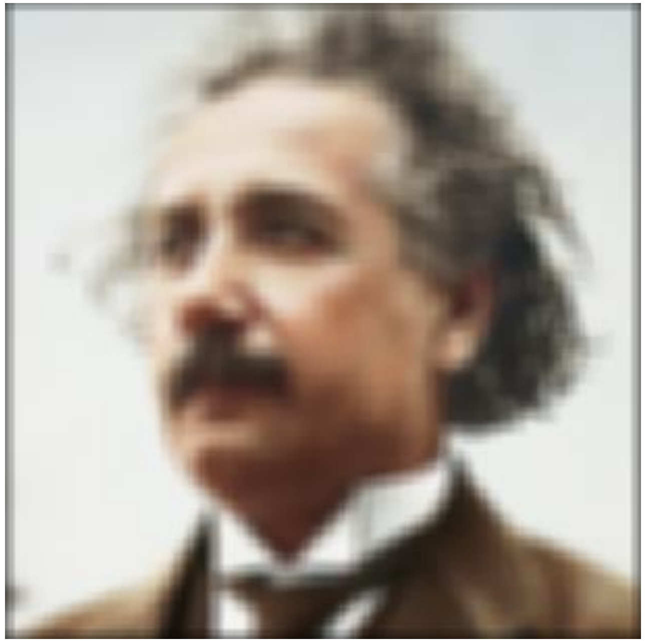
Exemplo: diferença entre os filtros da média e gaussiano.



inicial



Gaussiano 9 x 9

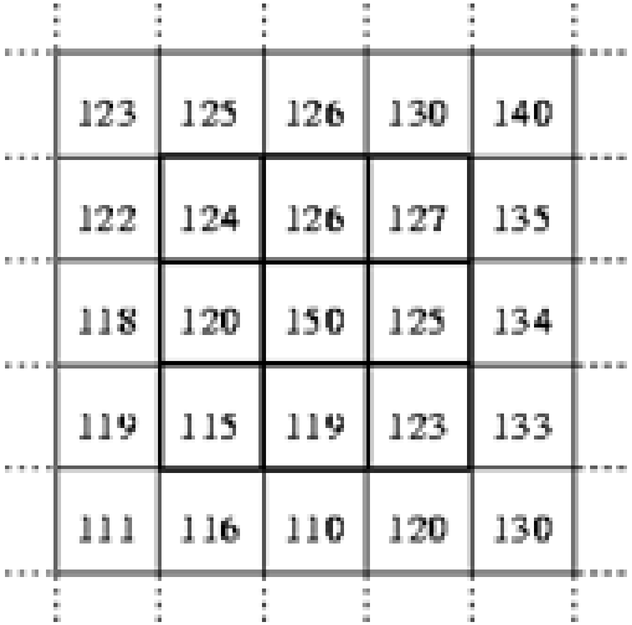


Média 9 x 9

Filtro passa-baixa

Fernando Soares

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.



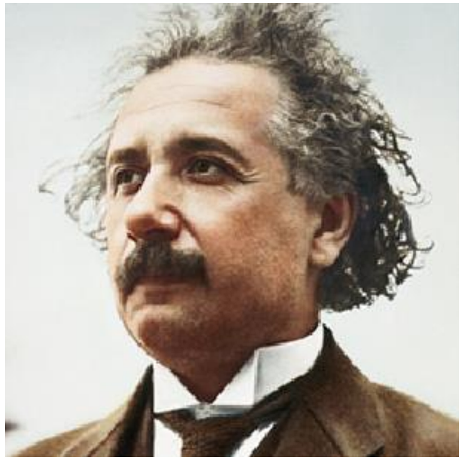
Valores da vizinhança

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

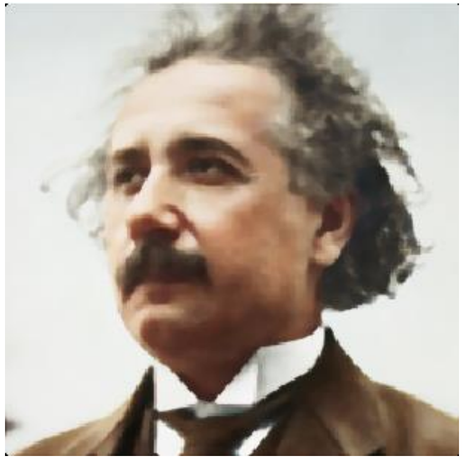
Mediana: 124

Filtro passa-baixa

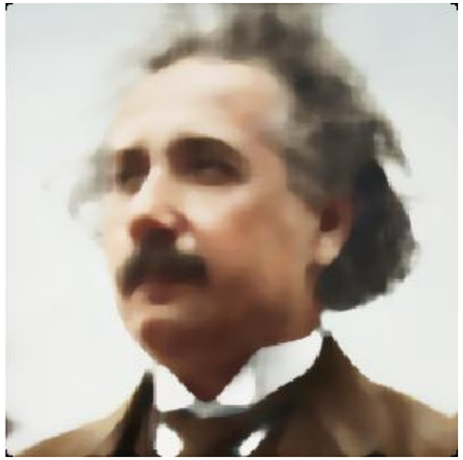
Exemplo: filtro da mediana.



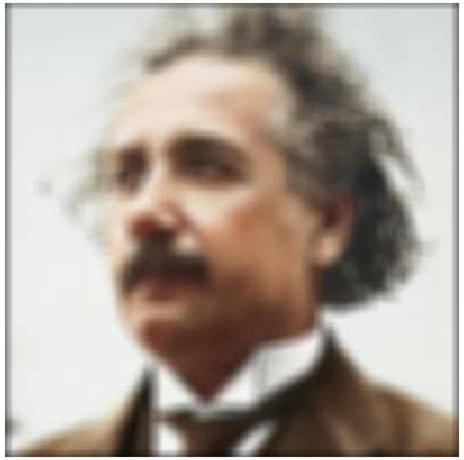
Inicial



Mediana 5 x 5



Mediana 9 x 9



Média 9 x 9

Filtro passa-baixa

Fernando Soares

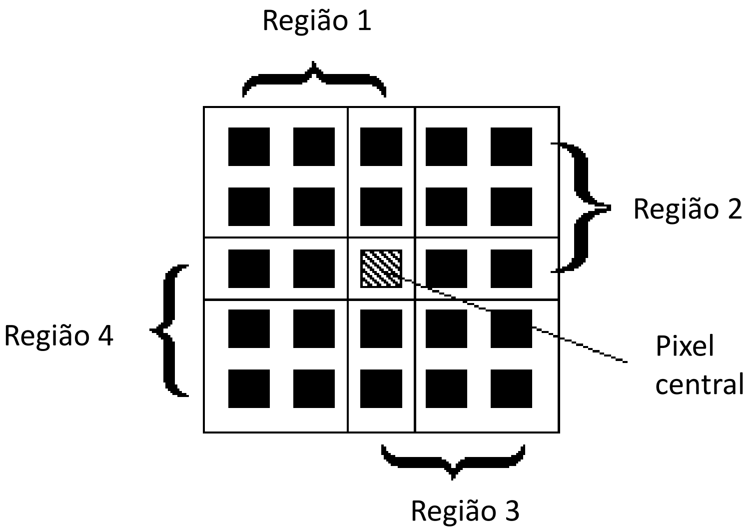
Kuwahara: Suaviza uma imagem sem perturbar a nitidez e a posição das fronteiras.

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).

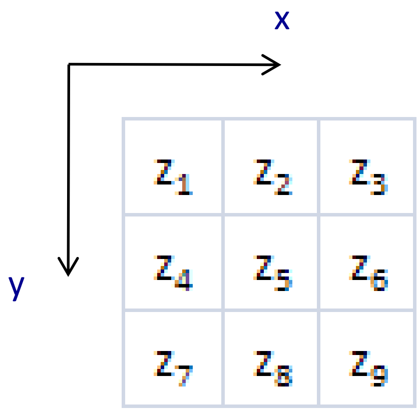
Gradiente: O gradiente de uma função f, no ponto (x,y), define-se por $\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$.

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.



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

ou

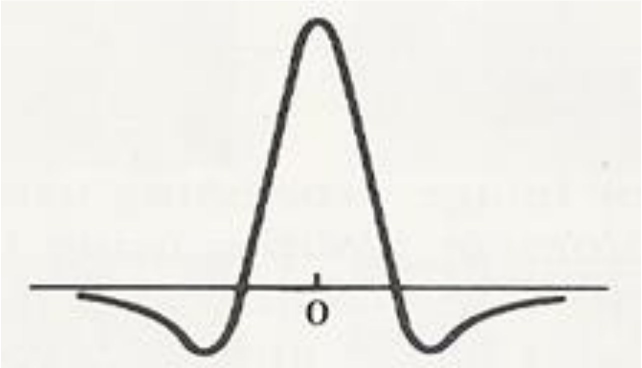
$$f(z_5) = mag(\nabla f) \approx |z_5 - z_6| + |z_5 - z_8|$$

Filtro passa-alta

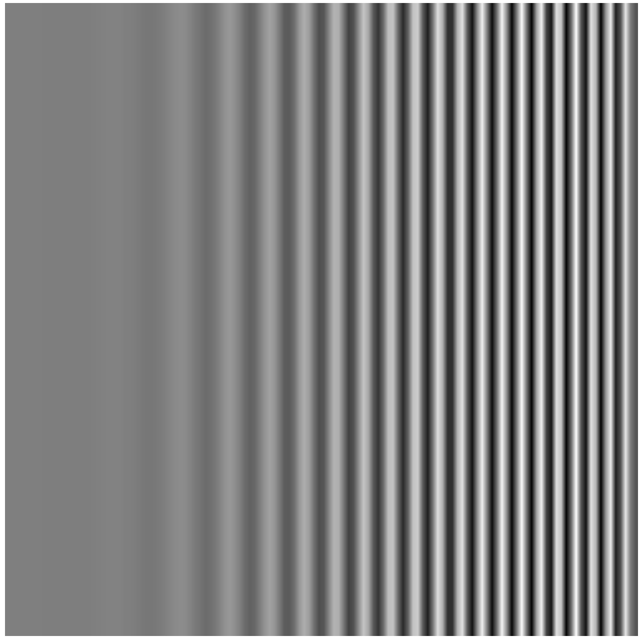
Fernando Soares

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 <u>positivos na vizinhança do centro e negativos na periferia</u>, devendo a respectiva <u>soma ser igual a 0</u>.



Função-resposta



Efeito

Filtro passa-alta

Fernando Soares

Filtro Passa-Alta = Imagem original - Filtro Passa-Baixa

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

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

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

Função

1/9 ×

-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

Coeficientes

	$PA_{i,j}$	

Valor a calcular

Filtro passa-alta

Fernando Soares

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:

1	0
0	-1

0	1
-1	0

x

y

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

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

ou

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

Filtro passa-alta

Roberts (exemplo 1):

0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50

1	0
0	-1

0	0	-50	0	0	...
0	0	-50	0	0	...
0	0	-50	0	0	...
0	0	-50	0	0	...
0	0	-50	0	0	...
...	...	...	...	...	...

$g_1(i,j) = f(i,j) - f(i+1, j+1)$

0	1
-1	0

0	0	50	0	0	...
0	0	50	0	0	...
0	0	50	0	0	...
0	0	50	0	0	...
0	0	50	0	0	...
...	...	...	...	...	...

$g_2(i,j) = f(i,j+1) - f(i+1, j)$

0	0	100	0	0	...
0	0	100	0	0	...
0	0	100	0	0	...
0	0	100	0	0	...
0	0	100	0	0	...
...	...	...	...	...	...

$g(i,j) = |g_1(i,j)| + |g_2(i,j)|$

Filtro passa-alta

Roberts (exemplo 2):

50	50	50	50	50	50
0	50	50	50	50	50
0	0	50	50	50	50
0	0	0	50	50	50
0	0	0	0	50	50
0	0	0	0	0	50

1	0
0	-1

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	...
...	...	...	...	...	...

$g_1(i,j) = f(i,j) - f(i+1, j+1)$

0	1
-1	0

50	0	0	0	0	...
50	50	0	0	0	...
0	50	50	0	0	...
0	0	50	50	0	...
0	0	0	50	50	...
...	...	...	...	...	...

$g_2(i,j) = f(i, j+1) - f(i+1, j)$

50	0	0	0	0	...
50	50	0	0	0	...
0	50	50	0	0	...
0	0	50	50	0	...
0	0	0	50	50	...
...	...	...	...	...	...

$g(i,j) = |g_1(i,j)| + |g_2(i,j)|$

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.

x

-1	0	1
-2	0	2
-1	0	1

y

-1	-2	-1
0	0	0
1	2	1

x

y

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

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

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

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

Filtro passa-alta

Sobel (exemplo):

	-1	*0*	*1*
-1	1	0	-1
0	2	0	-2
1	1	0	-1

0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50

	-1	0	1
-1	1	2	1
0	0	0	0
1	-1	-2	-1

…	…	…	…	…	…
…	0	200	200	0	…
…	0	200	200	0	…
…	0	200	200	0	…
…	0	200	200	0	…
…	…	…	…	…	…

S_x

…	…	…	…	…	…
…	0	0	0	0	…
…	0	0	0	0	…
…	0	0	0	0	…
…	0	0	0	0	…
…	…	…	…	…	…

S_y

…	…	…	…	…	…
…	0	200	200	0	…
…	0	200	200	0	…
…	0	200	200	0	…
…	0	200	200	0	…
…	…	…	…	…	…

$|S_x| + |S_y|$

Filtro passa-alta

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.

x

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

y

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

x

y

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

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

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

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

Filtro passa-alta

Fernando Soares

Prewitt (completar):

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

0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50
0	0	0	50	50	50

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

…	…	…	…	…	…
…					…
…					…
…					…
…					…
…	…	…	…	…	…

S_x

…	…	…	…	…	…
…					…
…					…
…					…
…					…
…	…	…	…	…	…

S_y

…	…	…	…	…	…
…					…
…					…
…					…
…					…
…	…	…	…	…	…

$|S_x| + |S_y|$

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

1	1	1
1	-2	-1
1	-1	-1

N

1	1	1
1	-2	1
-1	-1	-1

NE

1	1	1
-1	-2	1
-1	-1	1

W

1	1	-1
1	-2	-1
1	1	-1

E

-1	1	1
-1	-2	1
-1	1	1

SW

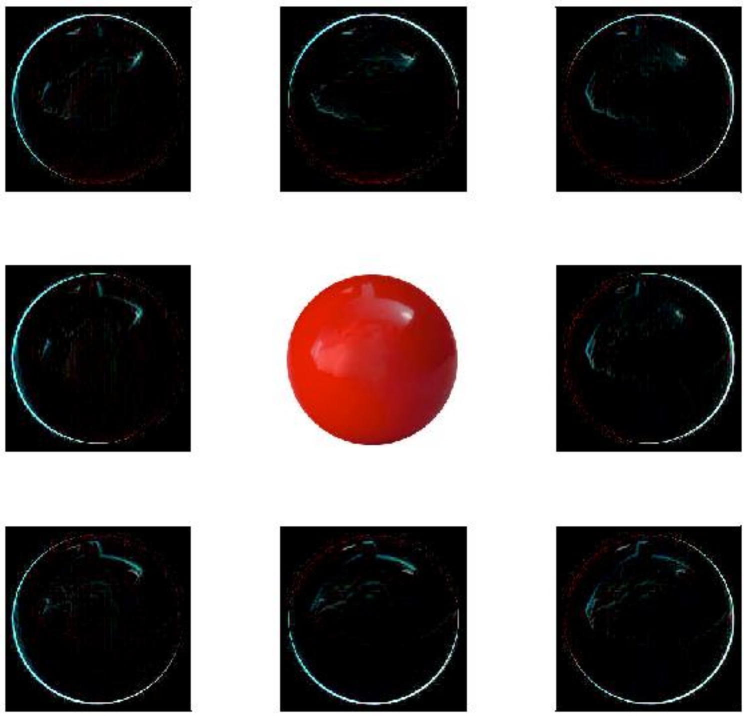
1	-1	-1
1	-2	-1
1	1	1

S

-1	-1	-1
1	-2	1
1	1	1

SE

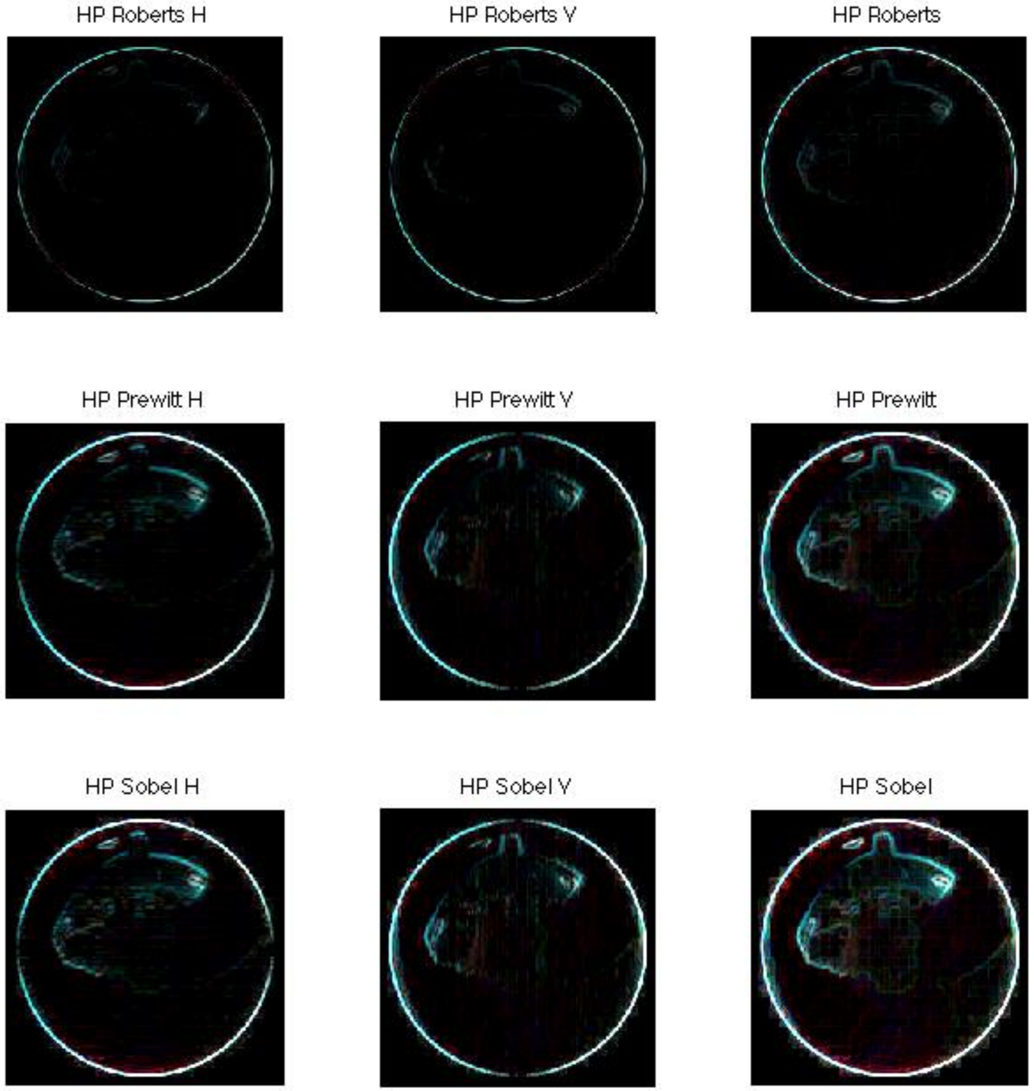
-1	-1	1
-1	-2	1
1	1	1



Filtro passa-alta

Fernando Soares

Operadores de gradiente:

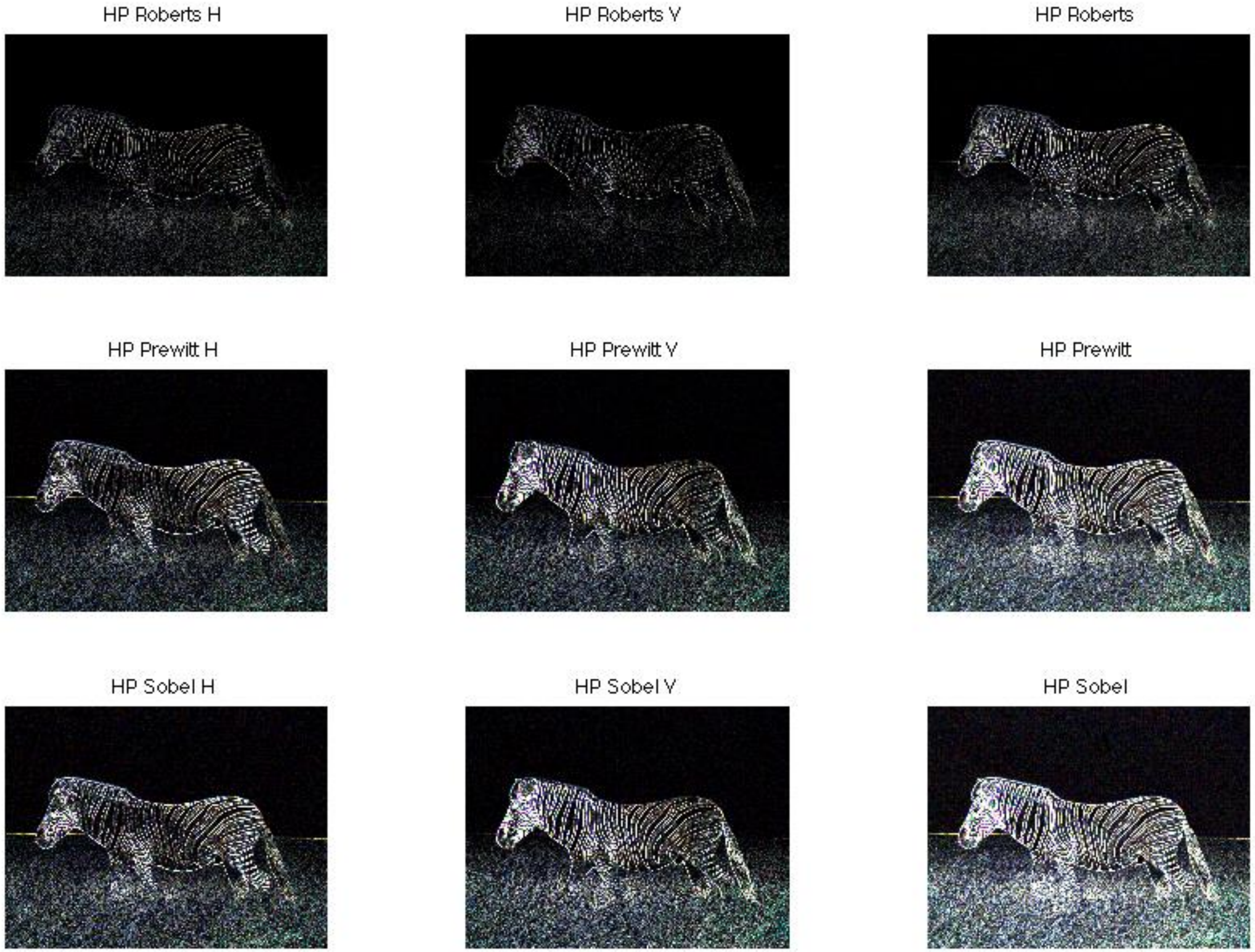


Filtro passa-alta

Ciências ULisboa

Fernando Soares

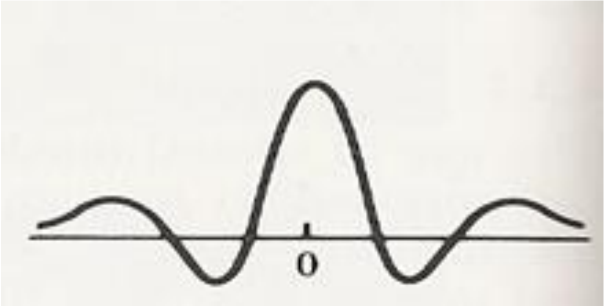
Operadores de gradiente:



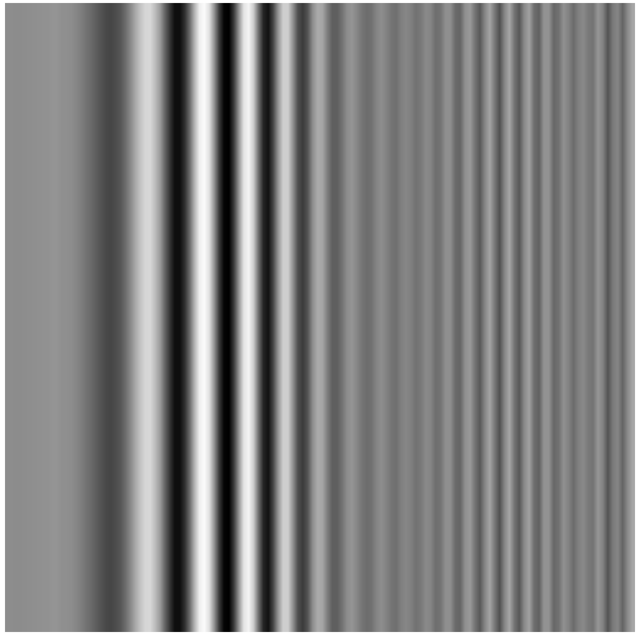
Filtro passa-banda

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

Filtro passa-banda

Fernando Soares

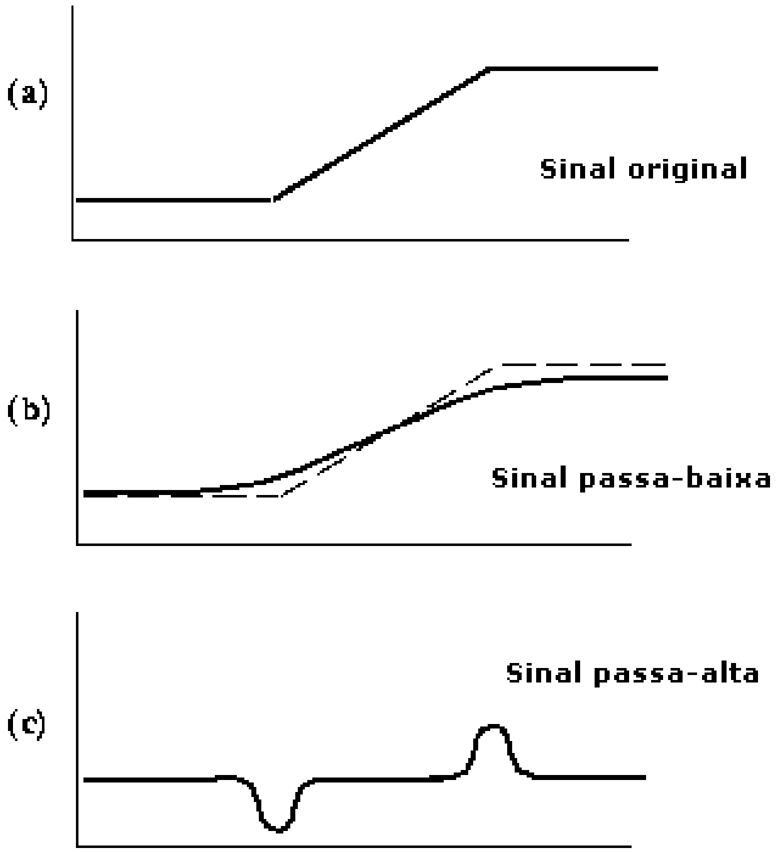
$$\boxed{\textit{Filtro Passa-Banda} = \textit{Filtro Passa-Baixa 1} - \textit{Filtro Passa-Baixa 2}}$$

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

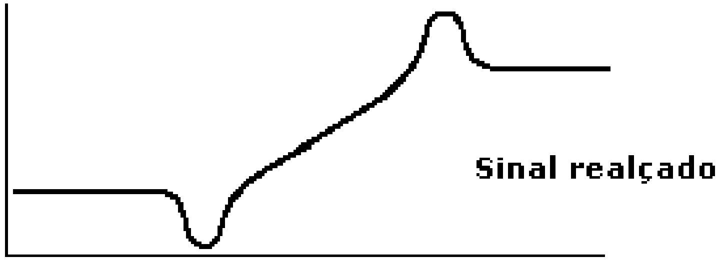
Os filtros Passa-Banda são geralmente usados para realçar as fronteiras e outras características de filtragem passa-alta na presença de ruído.

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.



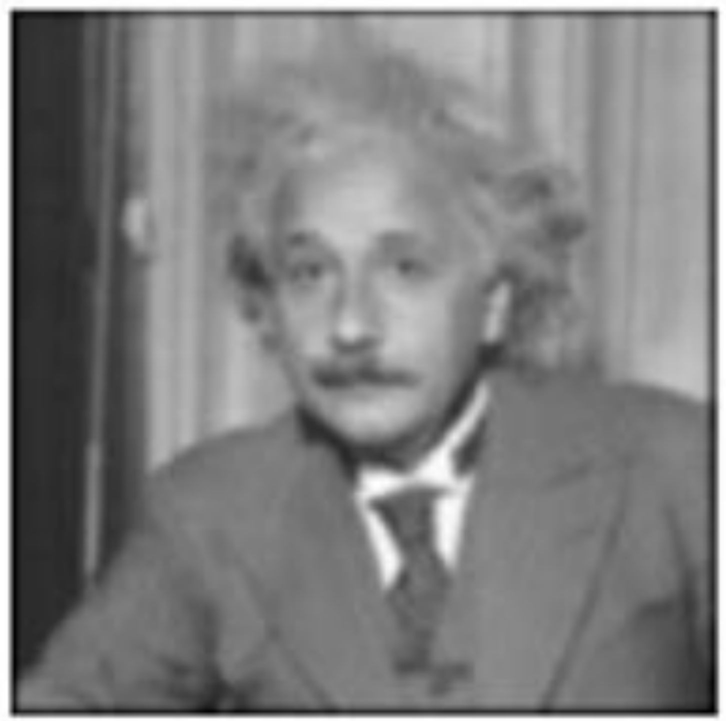
$$u(i,j) = f(i,j) + k \times [f(i,j) - PB(i,j)], \qquad 0 \leq k \leq 1$$



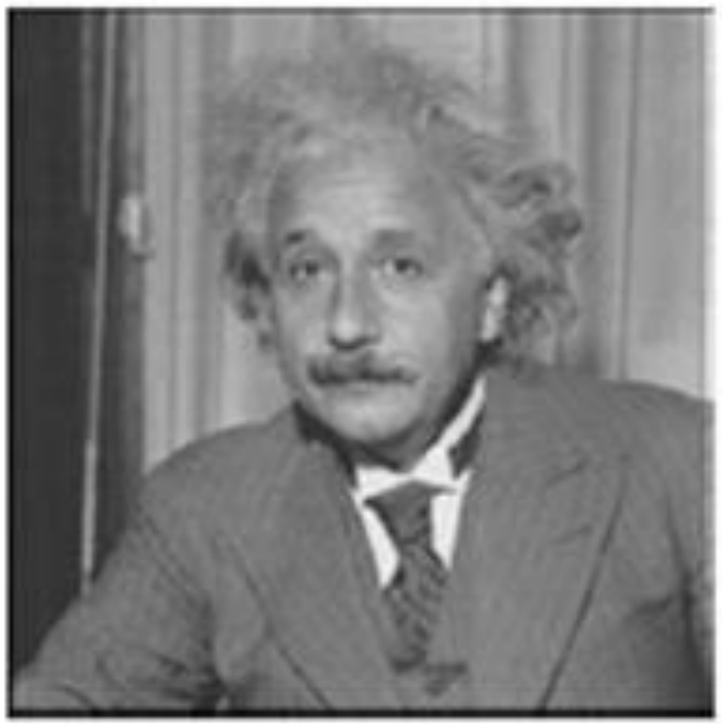
(a) + (c)

Operação de unsharp

Exemplo:



Antes

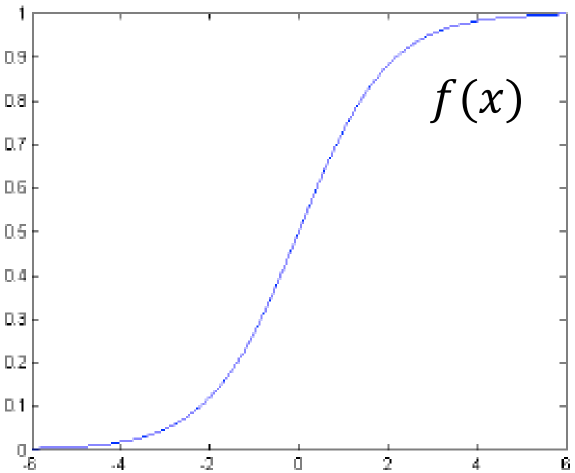


Depois

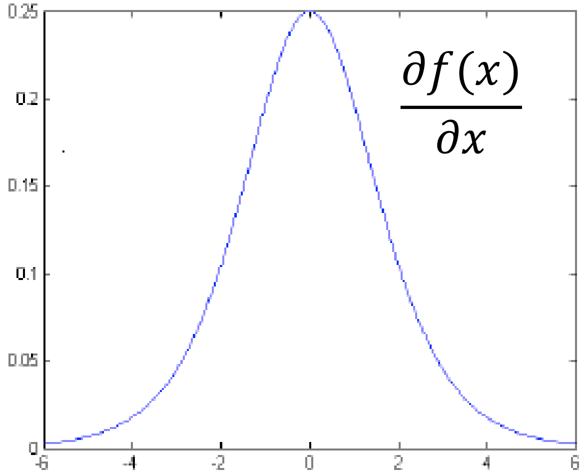
Filtro laplaciano

Fernando Soares

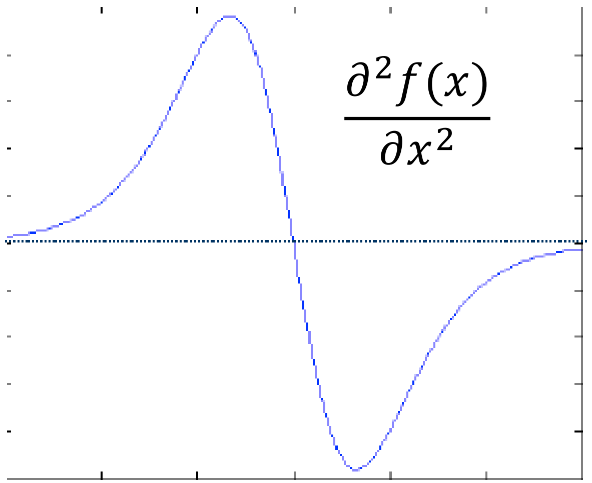
O filtro laplaciano distingue-se dos restantes filtros de realce de fronteiras porque usa a informação de segundas derivadas relativa às variações de intensidade dos pixels.



Função



Gradiente



Laplaciano

Filtro laplaciano

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

Fernando Soares

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

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

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

Filtro laplaciano

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

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

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

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

Filtro laplaciano

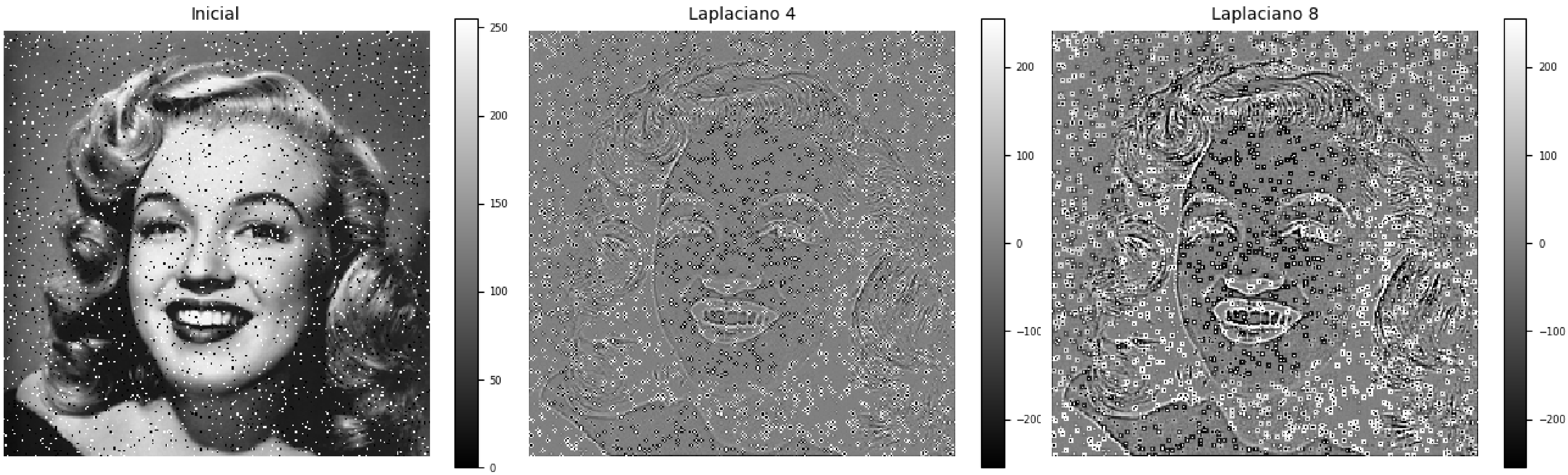
Fernando Soares

Exemplo 1:

Filtro laplaciano

Fernando Soares

Exemplo 2:



Filtro laplaciano do gaussiano

Fernando Soares

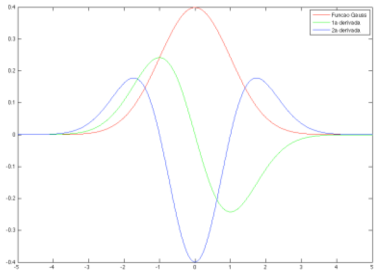
Para atenuar o efeito da presença de ruído, a imagem é geralmente filtrada, primeiro com um filtro passa-baixa gaussiano, antes de aplicar o operador laplaciano. Esta tarefa reduz o ruído de alta frequência antes da diferenciação.

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

Fernando Soares



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

Função gaussiana

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

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

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

Laplaciano do gaussiano

Filtro laplaciano do gaussiano

Fernando Soares

A aproximação discreta do operador LoG pode ser obtida com,

```
LapOfGauss = 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

$=$

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

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

Filtro laplaciano do gaussiano

Fernando Soares

Exemplo 1:

Filtro laplaciano do gaussiano

Exemplo 2: