Spatial filtering/ Filtragem

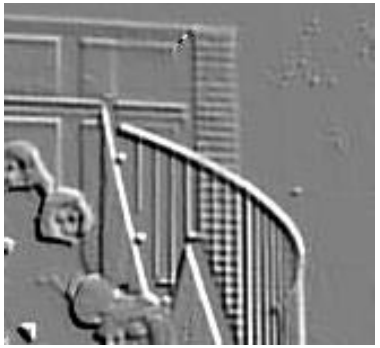
Filtros lineares

- Convolução
- Filtro passa-baixas (suavização)
- Filtro passa-altas (realce)
- Filtros direcionais
- Filtros não lineares

Processamento de imagens

Prof. Dr. Jorge Centeno

Para que servem?



Filtros lineares

Os filtros lineares resultam da *convolução* de uma janela móvel e a imagem.

A Convolução em uma dimensão:

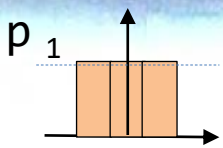
Para uma dimensão a *convolução* de duas funções “p(x)” e “I(x)” é dada por:

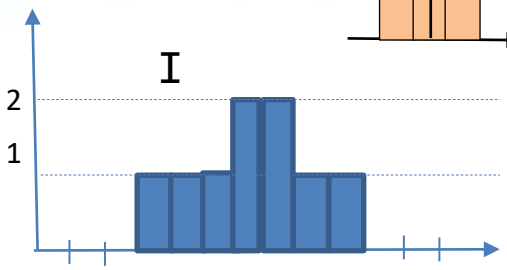
$$G(x) = \Sigma\ p(x) * I(x+i)$$

- Aqui “i” representa a vizinhança do valor “x”,
- por exemplo: i= -1 , 0, +1 pode ser usado para descrever o pixel e seus dois vizinhos na linha.
- p(x) representa o filtro
- G(y,x) é a série resultante, filtrada

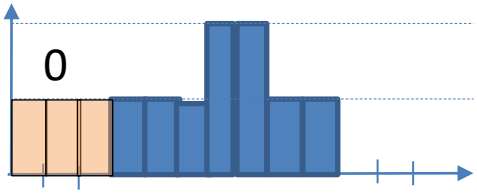
- Resumidamente, a convolução é feita deslocando a função “p” ao longo do domínio de “I” e para cada posição calcular a soma do produto entre as duas funções.

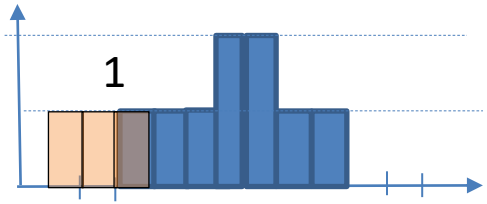
convolução

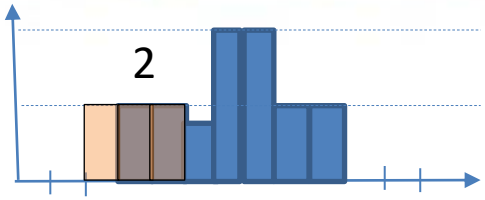


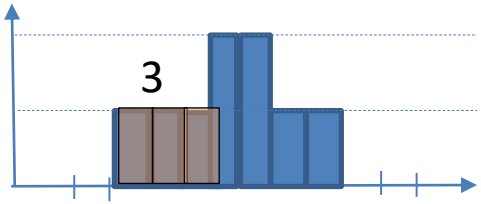


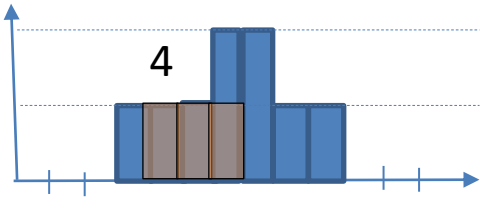
Deslocar “p”

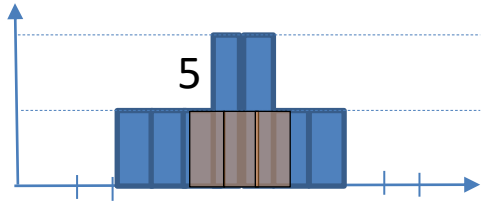


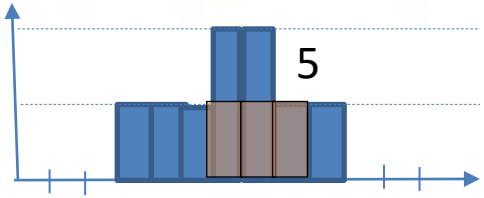


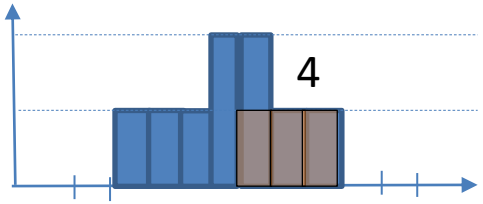


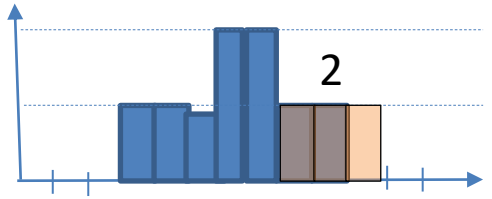


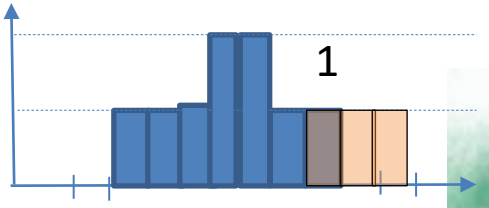




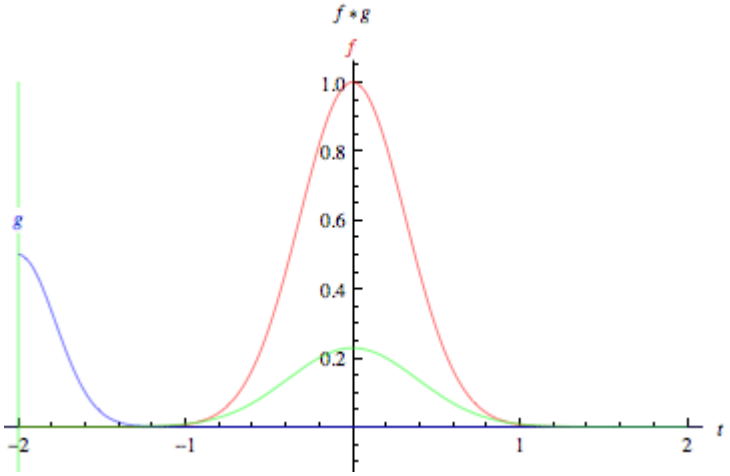








Ao deslocar a segunda função e multiplicar pela primeira, são obtidos valores diferentes em função do deslocamento.



exemplo

Considerando uma função I(x) com os seguintes valores:

I(x)= 5 2 5 5 5 9 5 5 4 4 5 5

e o filtro de tamanho 1x3 com os valores

p(i)=(1,1,1)

Tem-se:

imagem 5 4 5 5 5 9 5 5 4 4 5 5

filtro 1 1 1

Como o filtro é 3x1, as posições relativas são

-1, 0,1, ou seja i=-1, 0,1

exemplo

5 4 5 5 5 9 5 5 4 4 5 5

1 1 1

P=1 

. . . . . .

p=2 

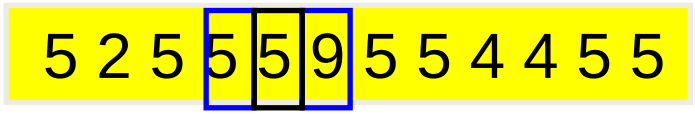
. 4 . . . . . .

p=3 

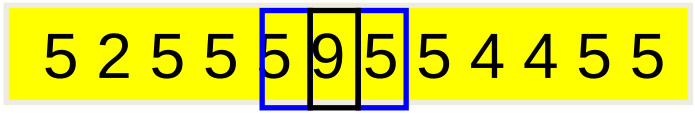
. 4 4 . . . .

p=4 

. 4 4 5 . . . .



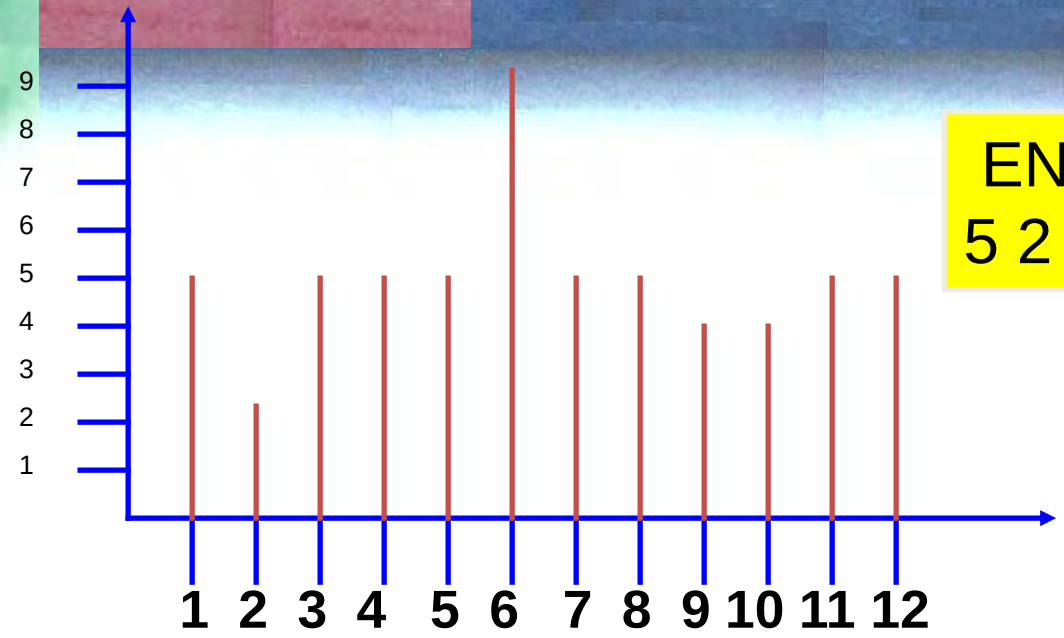
. 4 4 5 6 . . . .



. 4 4 5 6 6 . . . .

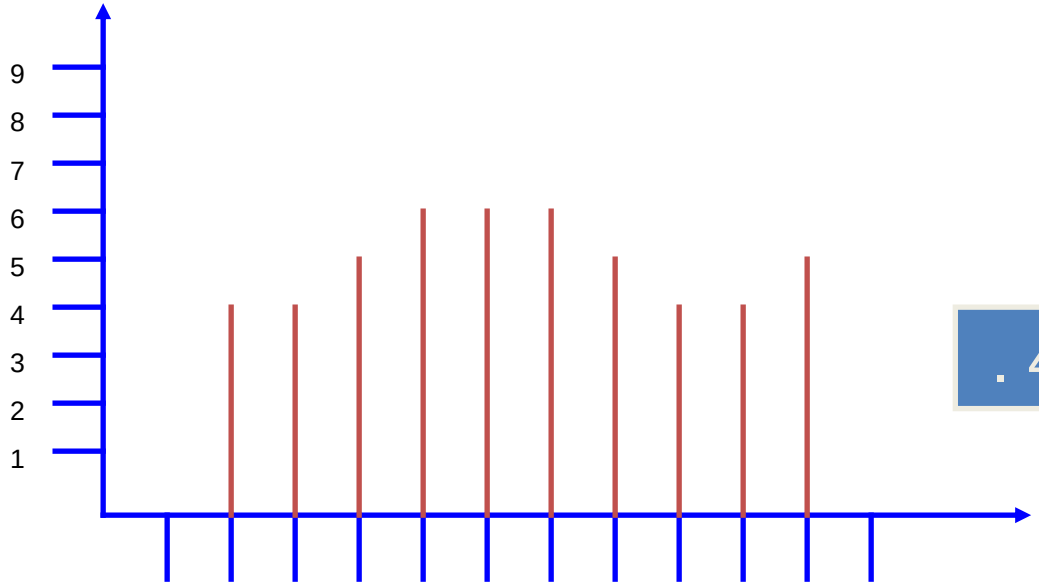
p=12 . 4 4 5 6 6 6 5 4 4 5 .

exemplo



ENTRADA
5 2 5 5 5 9 5 5 4 4 5 5

1 1 1



. 4 4 5 6 6 6 5 4 4 5 .

Filtros lineares

Em processamento de imagens, os filtros lineares resultam da convolução de uma janela móvel e a imagem no espaço 2-D.

O resultado de um filtro linear pode ser escrito na seguinte forma:

$$G(y,x) = \Sigma \Sigma \left(p(i,j) * I(y+i,x+j) \right)$$

onde

- y,x representam as coordenadas do pixel e
- i,j a posição relativa dos vizinhos
- p(i,j) representa o filtro
- G(y,x) é a imagem resultante, filtrada

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

Filtro (3x3)

1 2 1
2 2 2
1 2 1

dividido por 14, a soma dos elementos:

1/14 2/14 1/14
2/14 2/14 1/14
1/14 2/14 1/14

Para a posição linha=5 coluna=3:
Considerando a matriz de ”pesos”.

G = 1/14*(1*1 +2*3 +1*1 +2*7 +2*8 +2*9+1*9+2*8+1*8)
= 89 /14
= 6,4

Ou em valor digital (arredondar, ou truncar)
= 6

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

Da definição de filtro, entende-se que:

- Um resultado pode ser calculado para cada posição da janela móvel dentro da imagem.
- O resultado é sempre atribuído ao pixel central.

Este novo valor é calculado somando o produto do pixel na janela 3x3 e o respectivo peso, definido no filtro.

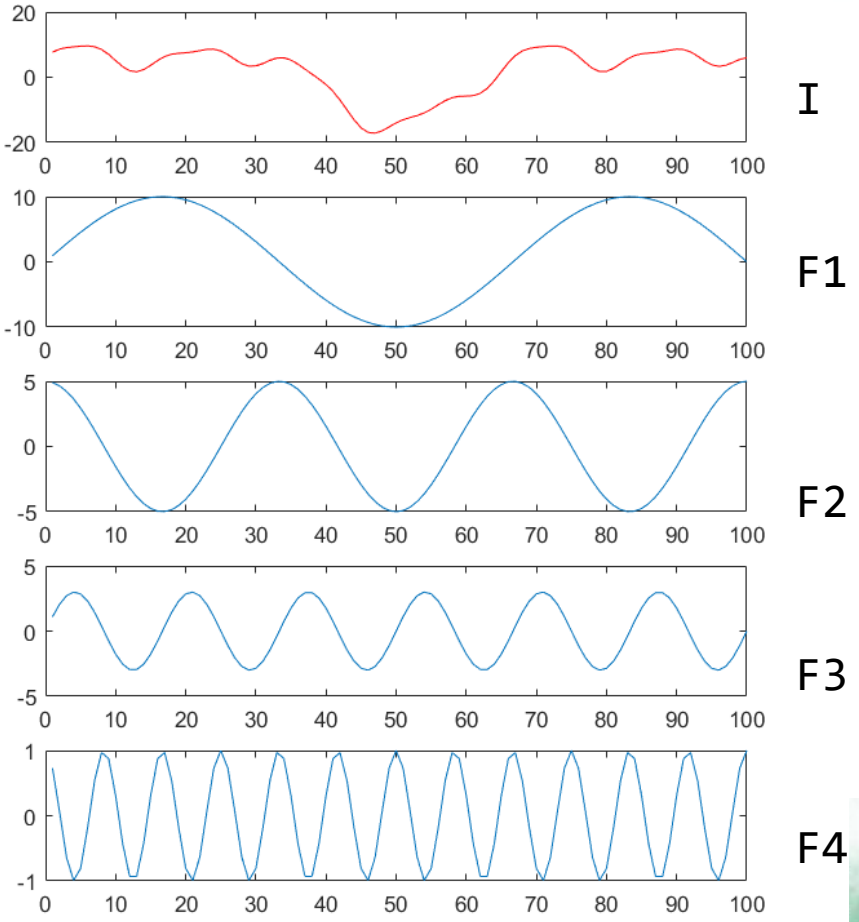
Os filtros lineares são definidos em termos de seu efeito na imagem como *passa baixas e passa altas*.

Conceito de Frequencias

Uma função pode ser descrita como a soma de várias funções senoidais com diferentes frequências.

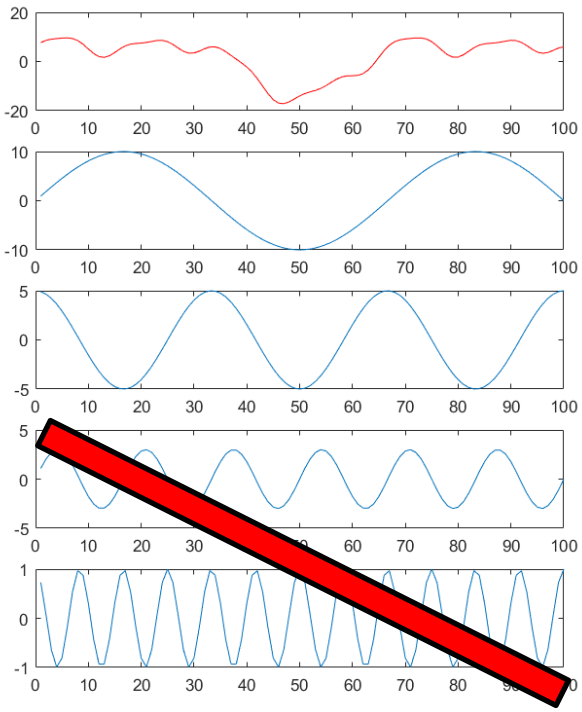
$$I=F1+F2+F3+F4 + \ldots$$

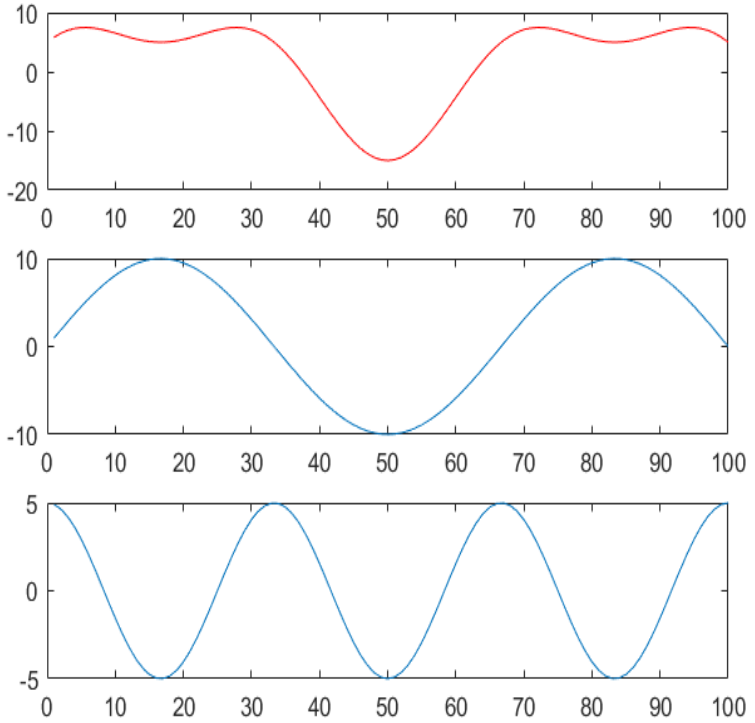
As funções com menores frequencias são responsáveis por dar a forma geral da curva, enquanto as com menores frequencias representam os detalhes



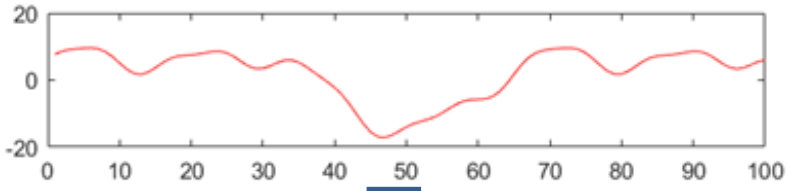
filtragem

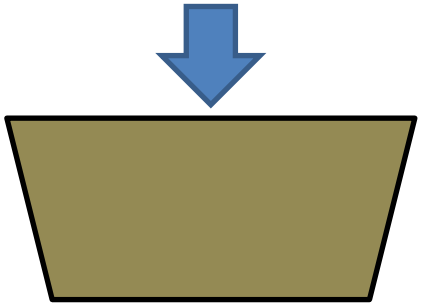
O que ocorre se retiramos as altas frequencias?





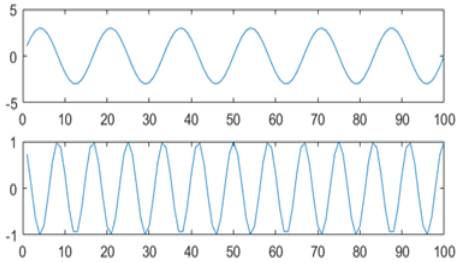
Filtragem seletiva de frequencias



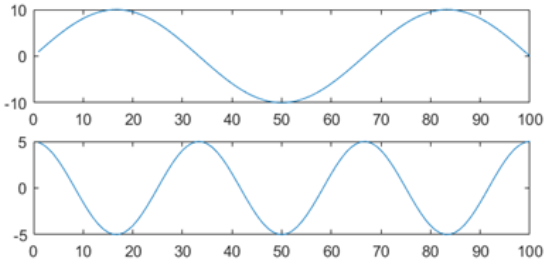


retira

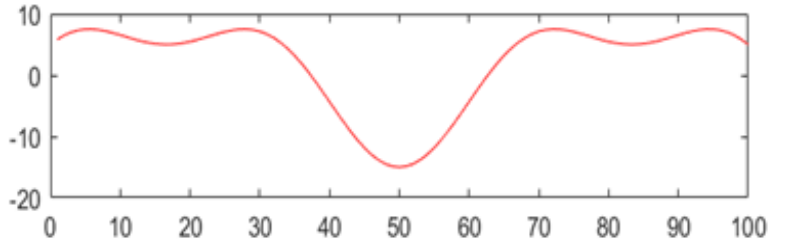
Altas frequencias



passa



baixas frequencias



Filtro passa-baixas (suavização)

- Atenua as altas frequências, aquelas associadas a detalhes na imagem, e deixa apenas as baixas frequências. O efeito deste filtro é a remoção de detalhes da imagem e sua suavização. A imagem filtrada apresenta uma aparência de névoa ou um efeito de ``imagem fora de foco'', e as áreas presentes na imagem tornam-se mais homogêneas.
- O efeito é atingido substituindo o pixel central pela média da janela. A média pode ser uma média simples ou uma média ponderada, onde diferentes pesos são atribuídos aos vizinhos em função de sua proximidade ao pixel central.
- Exemplo:

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

Filtro (3x3)

1	**1**	**1**
1	**1**	**1**
1	**1**	**1**

Exemplos de filtros passa-baixas

1	1	1
1	1	1
1	1	1

1	2	1
2	4	2
1	2	1

1	1	1
1	4	1
1	1	1

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Filtros de média.

Para o cálculo do valor final, o resultado da multiplicação dos pesos e os valores da imagem deve ser dividido pela soma dos pesos.

Passa baixas

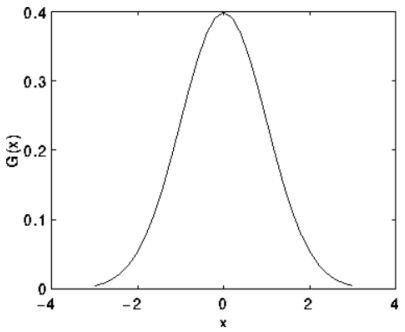
•Original

passa-baixas (suavização)

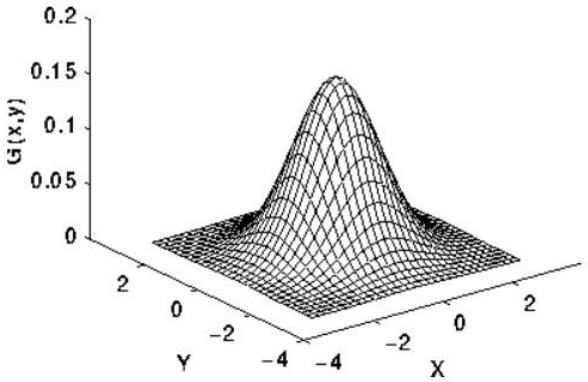
FILTRO GAUSSIANO

O Filtro Gaussiano é um tipo de filtro passa-baixas que usa uma função Gaussiana para calcular os pesos do filtro e, consequentemente, a transformação linear. Assim, maior peso é dado ao central e o peso diminui com a distancia ao pixel central da janela.

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



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



$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

$$
\begin{aligned}
&= \sum_i \sum_j f(x-i, y-j) \exp\left\{\frac{-(i^2+j^2)}{2\sigma^2}\right\} \\
&= \sum_i \left[\sum_j f(x-i, y-j) \exp\left\{\frac{-j^2}{2\sigma^2}\right\}\right] \exp\left\{\frac{-i^2}{2\sigma^2}\right\} \\
&= [f(x,y) * G(y)] * G^T(x).
\end{aligned}
$$

Porém, um filtro Gaussiano 2D pode ser substituído por dois filtros Gaussianos 1D, que são mais rápidos.

- Exemplo

$$f * \left(\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \right) = \left(f * \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} \right) * \left(\frac{1}{4} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \right)$$

$$\frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} * \frac{1}{4} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}.$$

Filtro passa-altas (realce)

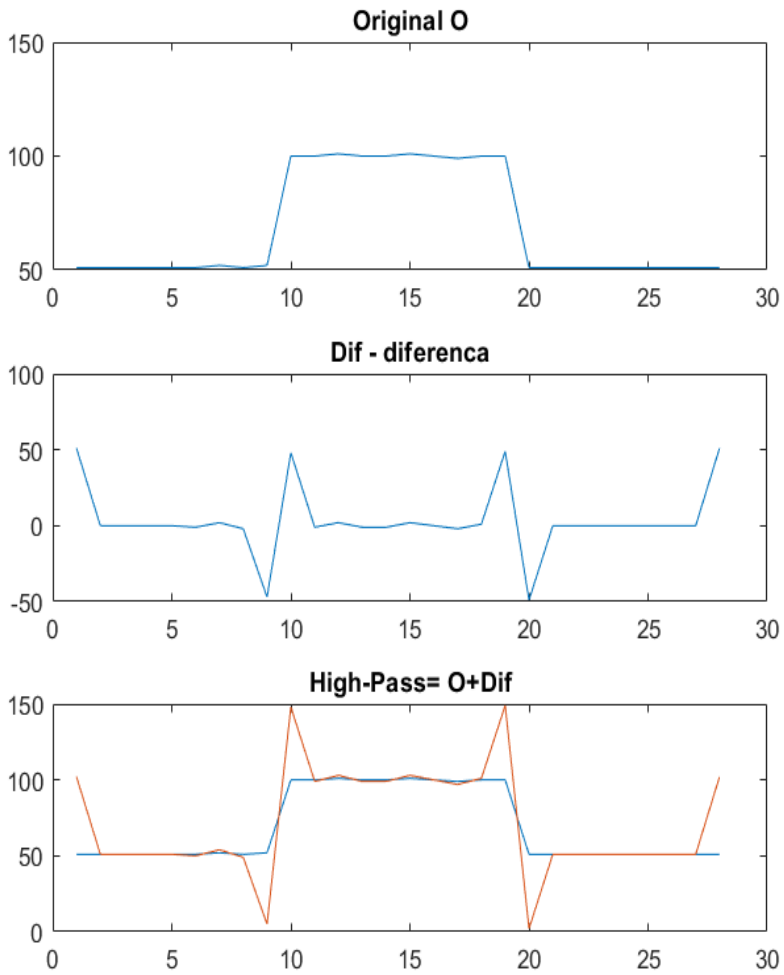
- Enfatiza os contrastes, realçando os detalhes da imagem.
- O nome do filtro explica seu funcionamento, pois nesta transformação as baixas frequências são eliminadas, sendo as altas frequências as únicas remanescentes.
- Este efeito pode ser atingido adicionando à imagem original a diferença entre a imagem original e o resultado de um filtro-passa baixas.
- O resultado da operação é nulo em regiões homogêneas. Em regiões com detalhes, o valor resultante é alto, em função do contraste entre o pixel central e a vizinhança.

Para entender...

Se aplicarmos a uma série de dados (1D)

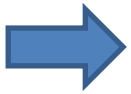
A diferença retira o valor original, pode ser igual em regiões claras e escuras.

Somando esta diferença ao valor original se salienta o contraste nas regiões de bordas.
Em regiões uniformes, não ocorre alteração.



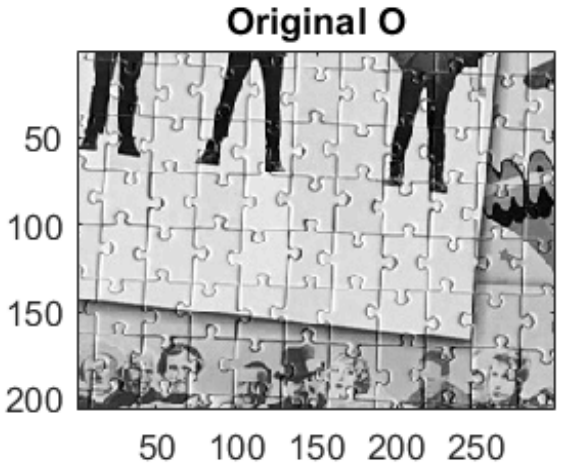
Diferença entre o central e seus vizinhos

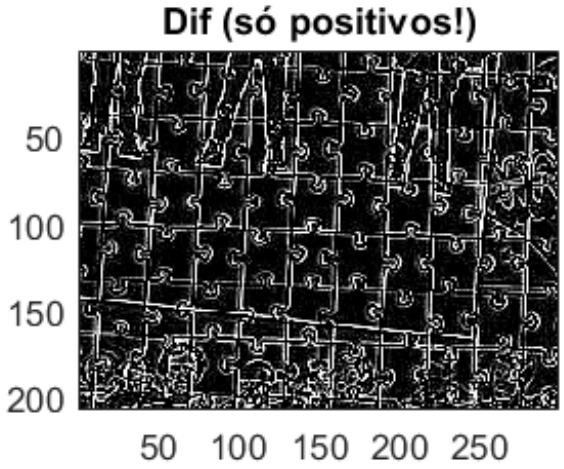
```
-1 0 0   0 -1 0    0  0 -1
 0 1 0   0  1 0    0  1  0
 0 0 0   0  0 0    0  0  0

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

 0 0 0   0  0 0    0  0  0
 0 1 0   0  1 0    0  1  0
-1 0 0   0 -1 0    0  0 -1
```



```
-1 -1 -1
-1  8 -1
-1 -1 -1
```

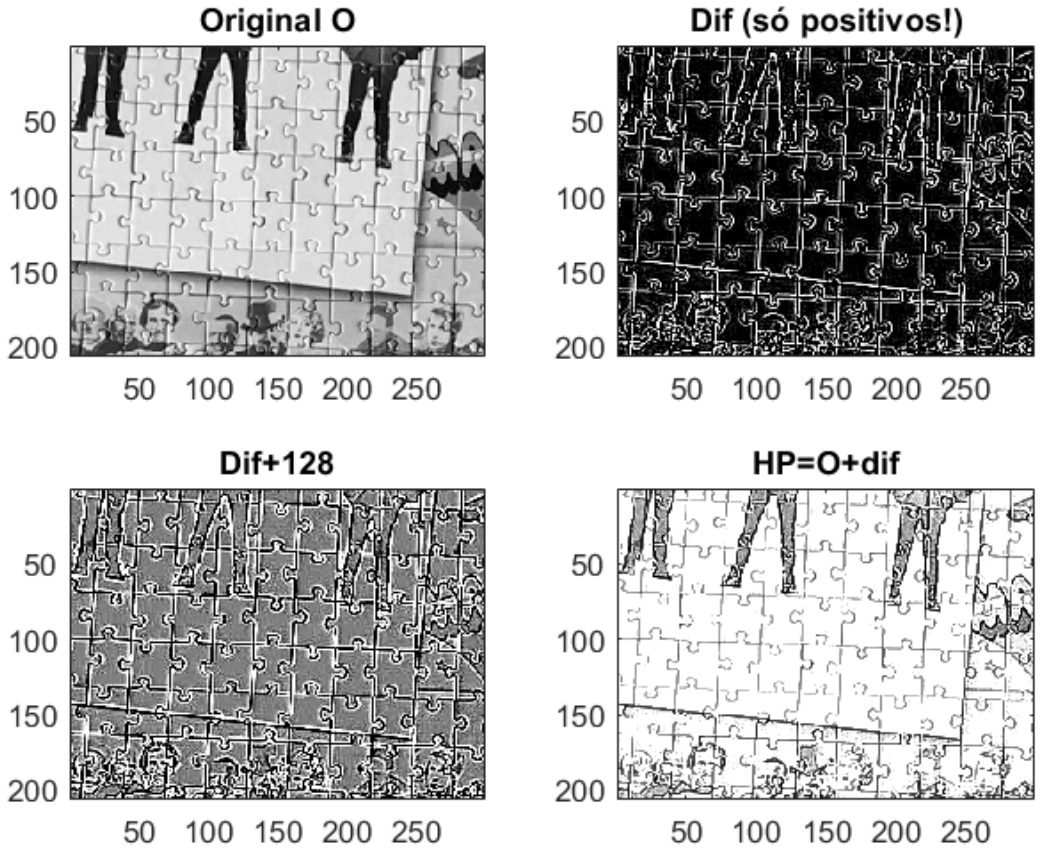
A diferença entre o central e seus oito vizinhos





Central + Diferença

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



Somando a diferença ao valor central acentua o contraste

Exemplos de passa-altas

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \qquad \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 25 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix}$$

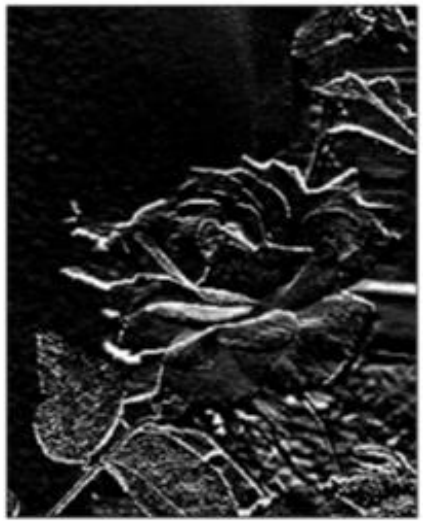
Passa altas

•Original

passa-altas

Filtros direcionais

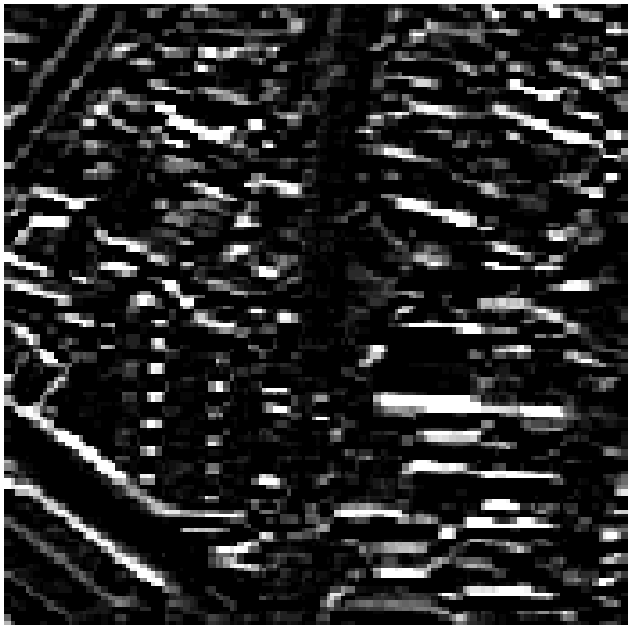
- A convolução de uma janela e a imagem também é útil para salientar determinadas linhas ou bordas. Por exemplo, as técnicas de filtragem permitem salientar as bordas ou linhas que ocorrem numa determinada direção, fazendo a diferença dos valores na janela considerando sua posição em relação ao pixel central da janela. A seguir são mostrados alguns exemplos destes filtros.



Exemplo: bordas horizontais

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

Norte



- Os contrates na direção norte são salientados.
- Algumas linhas diagonais também são salientadas, pois possuem uma componente norte forte.

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

Norte

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

Nordeste

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

Sul

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

Noroeste

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

Leste

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

Sudeste

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

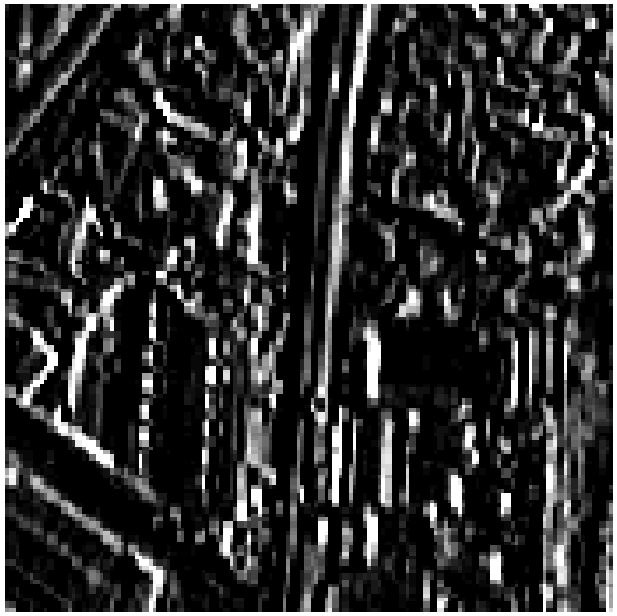
Oeste

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

Sudoeste

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

leste



Filtros não lineares

- Resultam da análise da vizinhança em torno do pixel, mas neste caso seu funcionamento não pode ser representado usando a forma geral da convolução.
- **O filtro de moda** (valor mais frequente) : Usado para suavizar imagens, especialmente temáticas, pois o novo valor atribuído ao pixel central corresponde ao valor mais frequente da vizinhança e por este motivo é igual a pelo menos um dos pixels vizinhos.
- **O filtro de mediana** (valor central) : o novo valor corresponde ao valor central após ordenar os valores de forma crescente.

```
1 1 1 1 2 1 1 0
1 1 2 1 3 1 1 1
1 1 3 1 2 1 1 0
1 1 3 1 4 1 1 1
9 7 8 8 8 9 9 8
9 9 8 8 9 9 9 8
9 9 9 8 9 8 9 9
9 8 9 8 8 9 9 9
```

Serie=[3 1 2 3 1 4 8 8 8]
Moda: 8 (mais frequente)

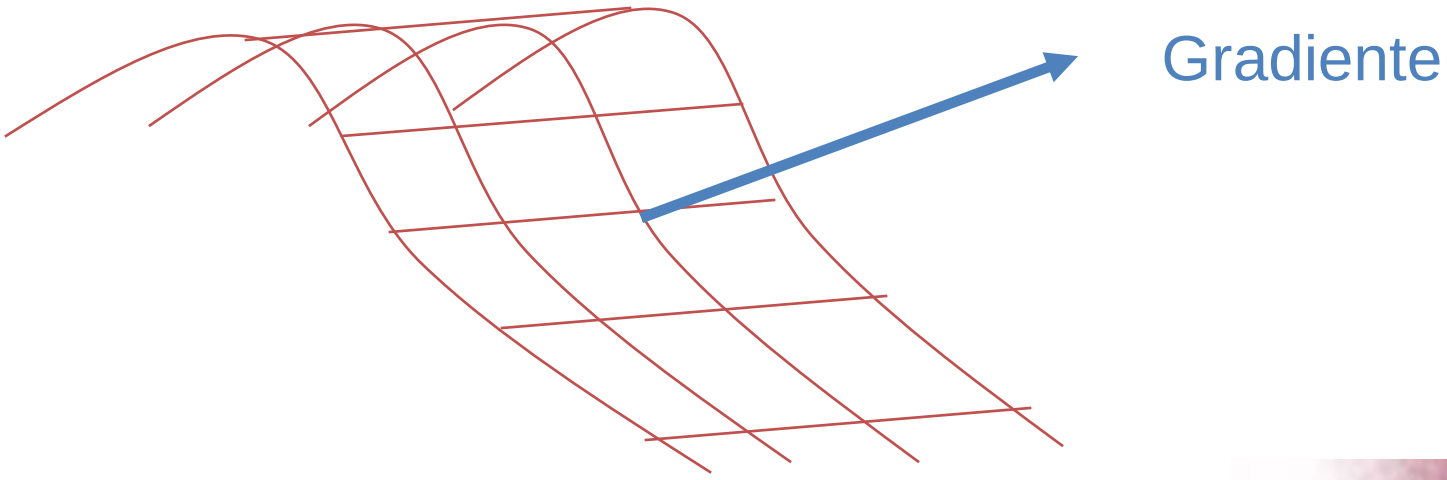
Mediana:
[1 1 2 3 3 4 8 8 8] = 3

Mediana

Filtro da mediana: O valor resultante é a mediana da vizinhança. Este filtro introduz um certo grau de suavização na imagem resultante, do que decorre perda de detalhe. A diferença em relação ao filtro passa baixas é que as bordas não são degradadas em extremo, pois os valores originais são preservados.

Filtros de Gradiente

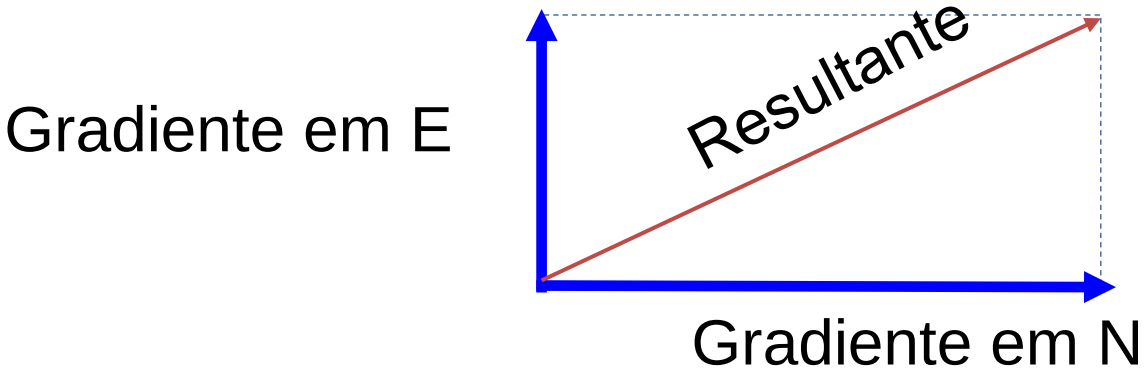
- O gradiente de uma superfície descreve sua inclinação em relação ao sistema adotado.
- O gradiente é tradicionalmente representado perpendicular à superfície.



Filtros de Gradiente

O gradiente pode ser calculado a partir de suas duas componentes (Norte e Leste), ou seja, a derivada parcial da função da superfície em relacao a cada uma das componentes.

$$G(x,y)=\{\ \delta F(x,y)/\delta(x)\ ,\ \delta F(x,y)/\delta(y)\ \}'$$



No processamento de imagens, pode-se assumir que a variação dos valores digitais se assemelha a uma superfície, similar a um Modelo Digital do Terreno (MDT).

Logo, torna-se possível calcular o gradiente para qualquer pixel, analisando a variação dos valores em sua vizinhança.

Para isto:

- Estima-se o gradiente em X
- Estima-se o gradiente em Y (Y perpendicular a X)
- Calcula-se a resultante da soma destes dois vetores.
- O pixel recebe um valor proporcional à magnitude do gradiente.

A diferença entre os filtros de gradiente radica na maneira de estimar as duas derivadas parciais.

Ex: filtros de Roberts:

1	0
0	-1

0	1
-1	0

$\delta F(x,y)/\delta(x)$ $\delta F(x,y)/\delta(y)$

Magnitude:

$$G = (\delta F(x,y)/\delta(x)^2 + \delta F(x,y)/\delta(y)^2)^{1/2}$$

$$\delta F(x,y)/\delta(x) = F(x-1, y-1) - F(x,y)$$
$$\delta F(x,y)/\delta(y) = F(x-1, y+1) - F(x,y)$$

- Ex: filtros de Prewitt

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

$\delta F(x,y)/\delta(x)$

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

$\delta F(x,y)/\delta(y)$

Magnitude:

$$G = \left(\delta F(x,y)/\delta(x)^2 + \delta F(x,y)/\delta(y)^2 \right)^{1/2}$$

- Ex: filtros de Sobel

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

$\delta F(x,y)/\delta(x)$

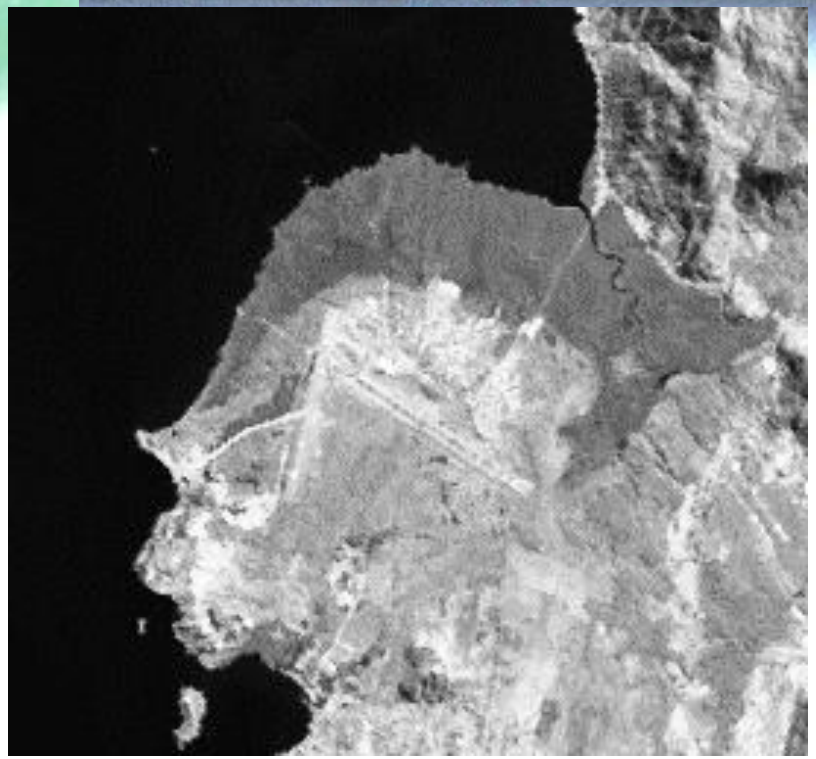
-1	-2	-1
0	0	0
1	2	1

$\delta F(x,y)/\delta(y)$

Magnitude:

$$G = \left(\delta F(x,y)/\delta(x)^2 + \delta F(x,y)/\delta(y)^2 \right)^{1/2}$$

Ex: filtro de SOBEL



Áreas uniformes: baixo gradiente
áreas de fronteira são salientadas, bem como feições lineares.

Laplaciano

O filtro Laplaciano é um operador que calcula a derivada isotrópica (não depende da direção, em tidas as direções)

Gradiente local em todas as direções

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

Exemplo:

0	1	0
1	−4	1
0	1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right]e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Na prática, o Laplaciano pode ser muito demorado para calcular e é sensível à presença de ruído. Por isso, não se usa diretamente sua formulação original.

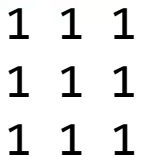
Usa-se a diferença entre a imagen original e a imagen suavizada com um filtro Gaussiano.

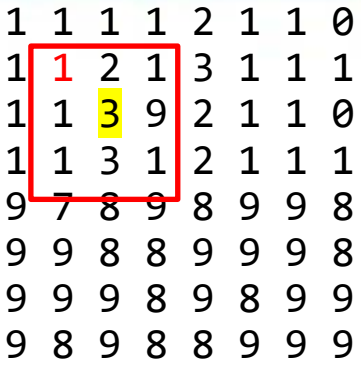
Isto é conhecido como o

Laplaciano do Gaussiano

LoG

Programa (Passa-baixas)





Tamanho do filtro = dim dim=3,5,7,..?

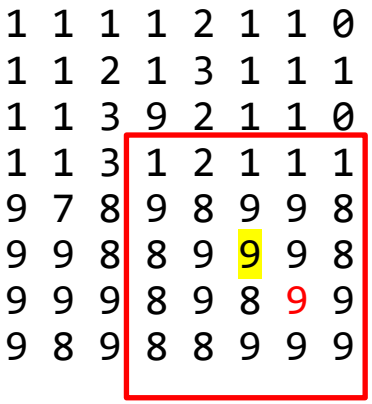
Vizinhos antes e depois

lado=(dim-1)/2

Verifique para dim=3, ou 5

A varredura não pode ser feita para o primeiro pixel, devemos começar no elemento (lado+1)

e não podemos terminar na ultima linha (n) mas devemos terminar em (n-[lado-1])



inicio

```
## ################## cria janela de filtro tamanho DIM ##################
dim=5                   # dimensao do filtro exe 3x3 dim=3, 5x5 dim=5...
lado=(dim-1)/2          # numero de vizinhos antes ou depois do central
lado=np.uint8(lado)
#__________________ escolha o filtro aqui
f=low_pass(dim)     # escolhe passa baixas (3x3)
# f=high_pass(dim)
print('dim=',dim, 'vizinho=',lado)
print('F=',f)

# ler imagem de entrada e recuperar dimensoes
X1= plt.imread('small.tif')
nl,nc = X1.shape
X=np.array(X1, dtype=float) #transforma em float para facilitar multiplicacoes
Y=np.zeros((nl,nc),dtype = float) # replica imagem para gerar uma saida
Y=np.uint8(Y) # em uint 8 para armazenar em byte
```

Filtragem

```
for L in range(lado,nl-lado): # varrer em linhas lado+1, para vizinhos+central
    for C in range (lado, nc-lado): # varrer em colunas
        s=0. # zerar soma para convolucao
        for i in range(dim): # varrer filtro em linhas e colunas
            for j in range(dim):
                # determina o pixel na imagem
                p=(L-lado)+i
                q=(C-lado)+j
                s=s+X[p,q]*f[i,j] #pixel*peso
            if s>255: #truncar se ficar fora da faixa
                s=255
            if s<0:
                s=0
            v=np.uint8( np.round( s ) ) # muda a uint8
            Y[L,C]=v # salva na posicao do central
plt.imsave('saida.png',Y,cmap='gray')
```

Dois filtros em python

```
## ################## LOW_pass ##################
def low_pass(dim):
    f= np.zeros((dim,dim),dtype = float) # matriz vazia
    f=f+1 # cria matriz com tudo igual a 1, zeros+1=1
    f=f/(dim*dim ) # dividir para calcular a media
    return(f)


## ################## HIGH_pass ##################
def high_pass(dim):
    f= np.zeros((dim,dim),dtype = float) # matriz vazia
    f=f-1 # cria matriz com tudo igual a -1, zeros-1=-1
    f[lado,lado]=dim*dim # e muda o central
    return(f)
```

python

```
0 0 0
0 0 0
0 0 0
```

```
1 1 1
1 1 1
1 1 1
```

```
0.11 0.11 0.11
0.11 0.11 0.11
0.11 0.11 0.11
```

```
def low_pass(dim):
    f= np.zeros((dim,dim),dtype = float)

    f=f+1

    f=f/(dim*dim )
```

python

```
def high_pass(dim):
```

```
0 0 0
0 0 0
0 0 0
```

```
    f= np.zeros((dim,dim),dtype = float)
```

```
-1 -1 -1
-1 -1 -1
-1 -1 -1
```

```
    f= f-1
```

```
    f[lado,lado]=dim*dim
```

```
-1 -1 -1
-1  9 -1
-1 -1 -1
```