

Filtragem Espacial

Prof. Jefferson T. Oliva
jeffersonoliva@utfpr.edu.br

Processamento de Imagens
Engenharia de Computação
Departamento Acadêmico de Informática (Dainf)
Universidade Tecnológica Federal do Paraná (UTFPR)
Campus Pato Branco



This work is licensed under a Creative Commons "Attribution-ShareAlike 4.0 International" license.

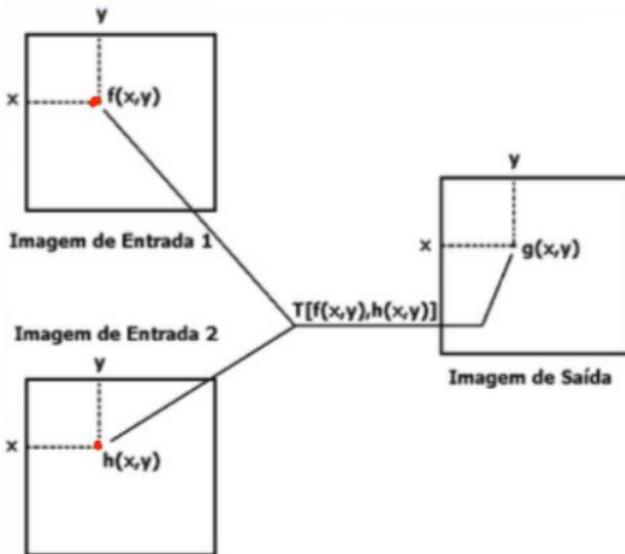


- Filtragem no Domínio Espacial

Filtragem no Domínio Espacial

Filtragem no Domínio Espacial

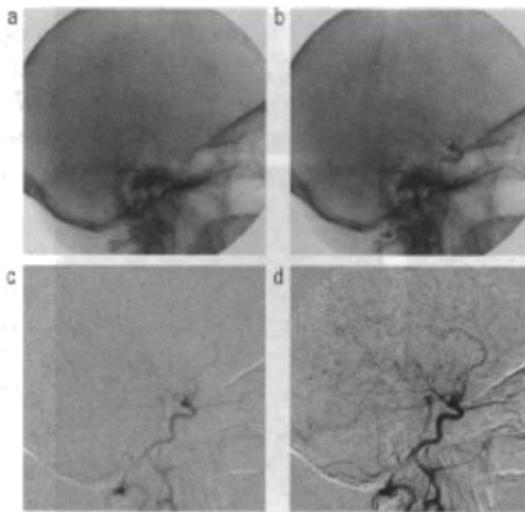
- $T[f(x,y), h(x,y)]$: Operação envolvendo dois pixeis nas imagens de entrada, na mesma coordenada (x, y) gerando o pixel correspondente na Imagem de saída



Filtragem no Domínio Espacial

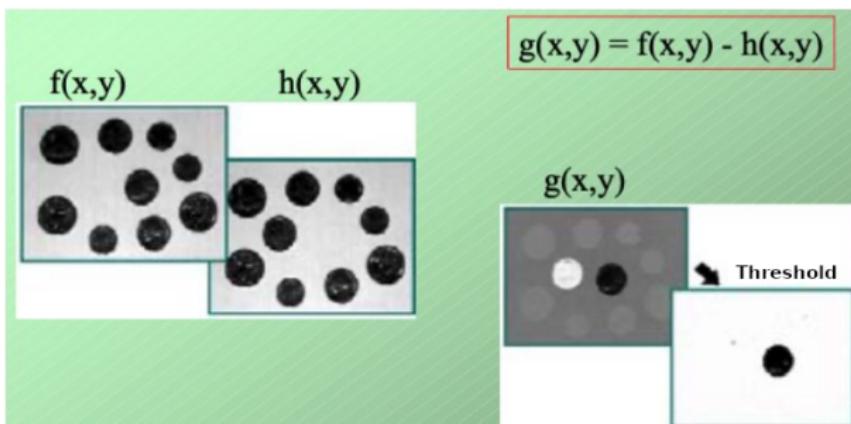
- Exemplo na área médica

- $h(x,y)$ é a máscara (a): imagem de raio-X obtida de uma região do corpo do paciente
- $f(x)$ é a imagem (b): obtida após a injeção de contraste, na mesma região do corpo do paciente
- $b - a$ (c)
- Imagen realçada (d)



Filtragem no Domínio Espacial

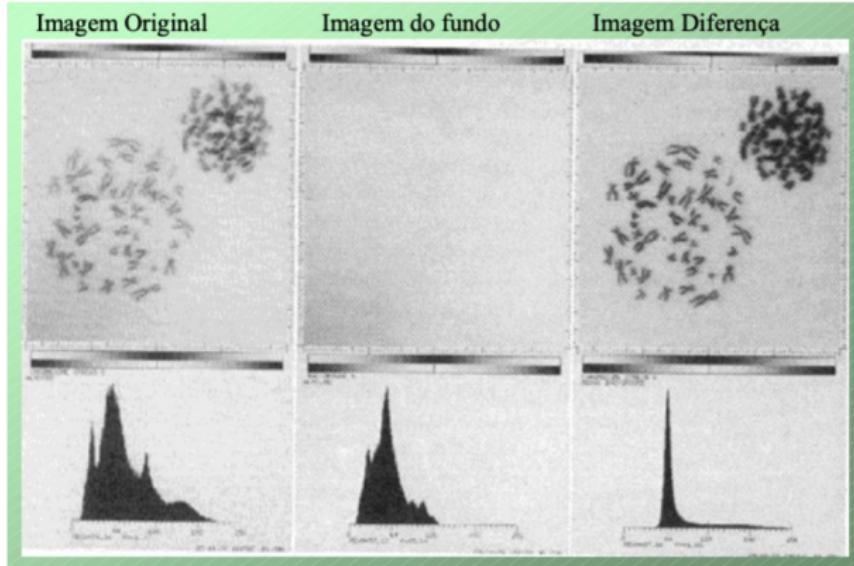
- Exemplo na detecção de movimento: a subtração das duas imagens produz uma Imagem Diferença que pode detectar se houve movimento de algum dos componentes da $f(x,y)$ para a $h(x,y)$



Filtragem no Domínio Espacial

- Subtração do fundo (*background*): remove sombreamentos ou descobre movimentos entre duas imagens
 - O nível de cinza em cada pixel de uma imagem é subtraído do nível de cinza do pixel correspondente na outra imagem

$$f_m(x) = f(x) - fb(x)$$



- Adição de imagens: a soma de duas ou mais imagens é geralmente aplicada com o objetivo de obter a média entre as mesmas para atenuação de ruídos
 - Seja $g(x, y)$ a imagem $f(x, y)$ com ruído $\eta(x, y)$ não correlacionado e de média zero em cada coordenada (x, y) :

$$g(x, y) = f(x, y) + \eta(x, y)$$

- A média entre M imagens $g(x, y)$ é dada por:

$$\bar{g}(x, y) = \frac{1}{M} \sum_{i=1}^M g_i(x, y)$$

- Adição de imagens
 - As variâncias da média e do ruído são:

$$\sigma_{\bar{g}(x,y)}^2 = \frac{1}{M} \sigma_{\eta(x,y)}^2$$

- O desvio padrão em qualquer ponto da imagem média é dado por

$$\sigma_{\bar{g}(x,y)} = \frac{1}{M} \sigma_{\eta(x,y)}$$

Filtragem no Domínio Espacial

- Adição de imagens
 - Quanto maior for o número de imagens utilizadas na soma, menor será o desvio padrão e a variância

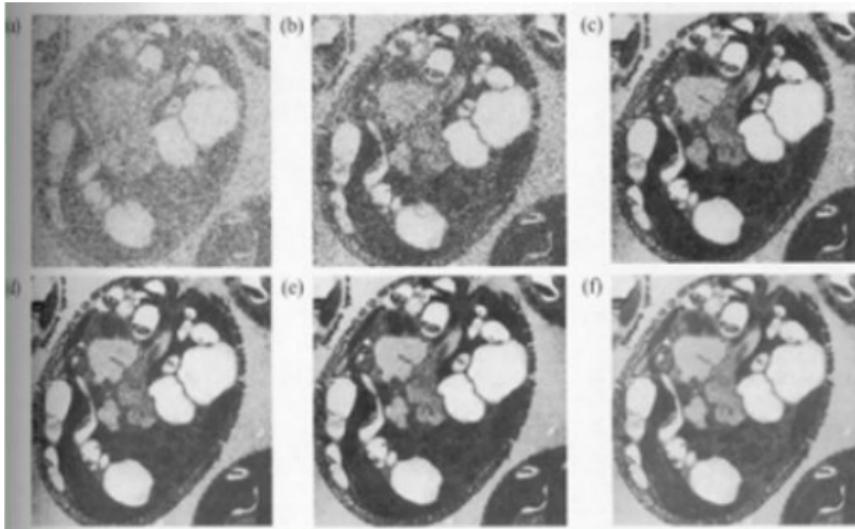


Figura: a) imagem original b) média de 2, c) média de 8 d) média de 16 e) média de 21 e f) média de 128 imagens ruidosas

- Multiplicação e divisão de imagens
 - Um dos principais usos da multiplicação ou divisão de imagens é para corrigir sombras de níveis de cinza produzidas por não uniformidades da iluminação ou no sensor utilizado para a aquisição da imagem

$$g(x, y) = f(x, y) \cdot h(x, y)$$

$$g(x, y) = f(x, y) / h(x, y)$$

Filtragem no Domínio Espacial

- Multiplicação e divisão de imagens

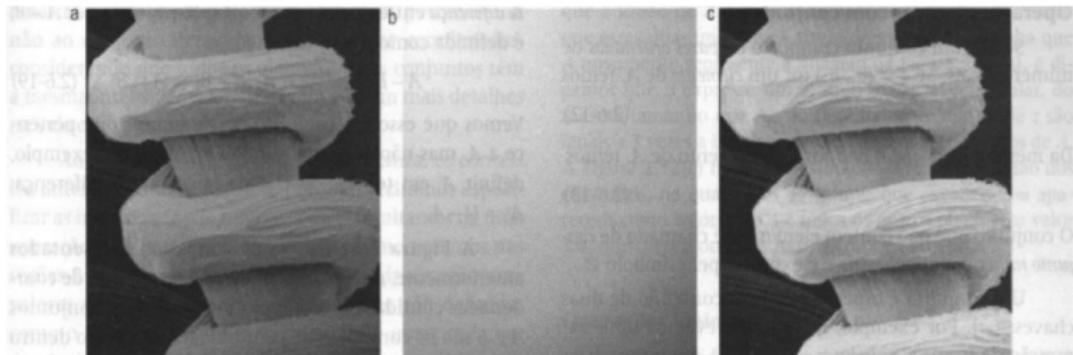
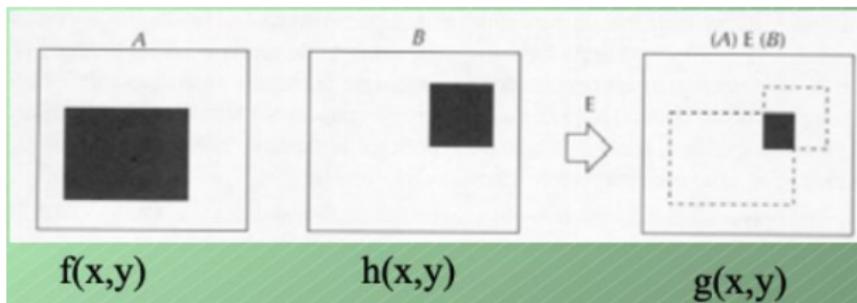


Figura 2.29 Correção de sombreamento. (a) Imagem sombreada de um filamento de tungstênio e suporte gerada por um microscópio eletrônico por varredura, ampliada aproximadamente 130 vezes. (b) O padrão de sombreamento. (c) Produto de (a) pelo inverso de (b). (Imagen original: cortesia de Michael Shaffer, Departamento de Ciências Geológicas, Universidade de Oregon, Eugene.)

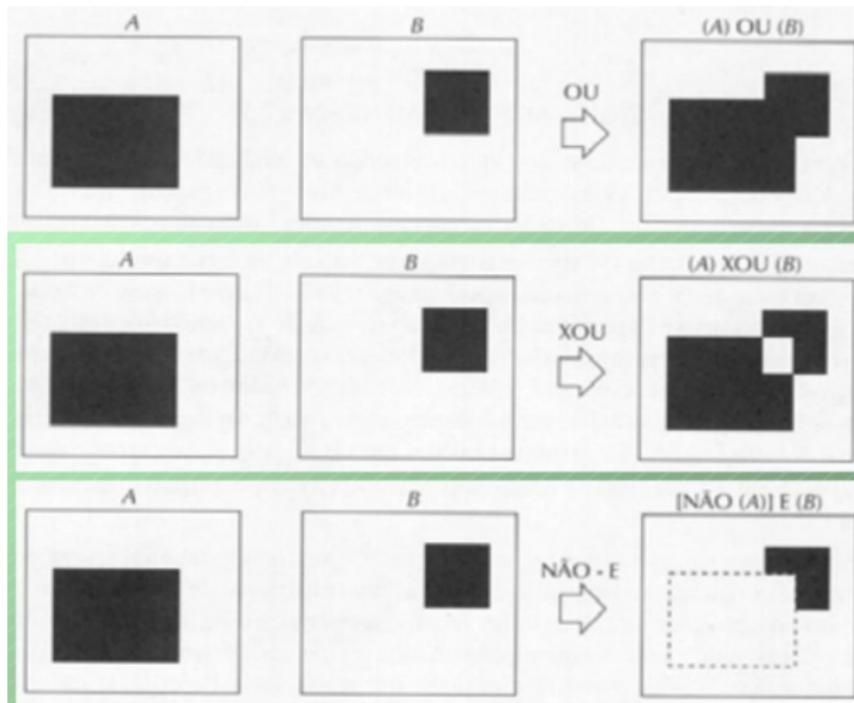
Filtragem no Domínio Espacial

- Operações lógicas: aplicadas apenas em imagens binárias, onde, nos exemplos abaixo, 1 = preto e 0 = branco
 - Operação AND: $f(x, y) \text{ AND } h(x, y)$



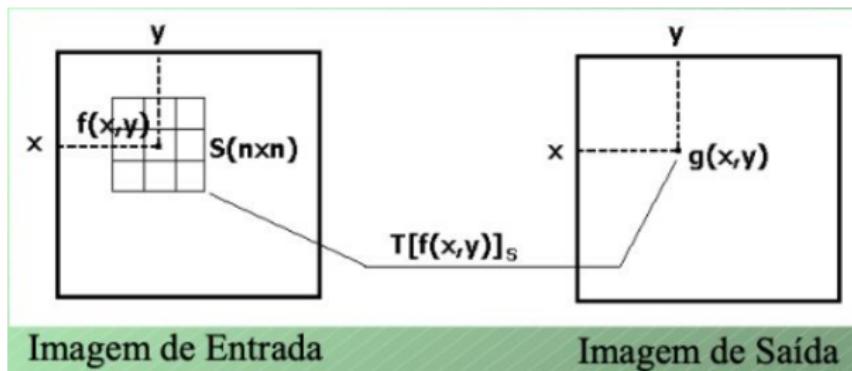
Filtragem no Domínio Espacial

- Operações lógicas



Filtragem no Domínio Espacial

- Operadores locais: combina a Intensidade de um certo número de pixels (janela), para computar o valor da nova intensidade na Imagem de Saída
 - $T[f(x,y)]_S$: operação sobre todos os pixels dentro da janela S centrada em $f(x,y)$

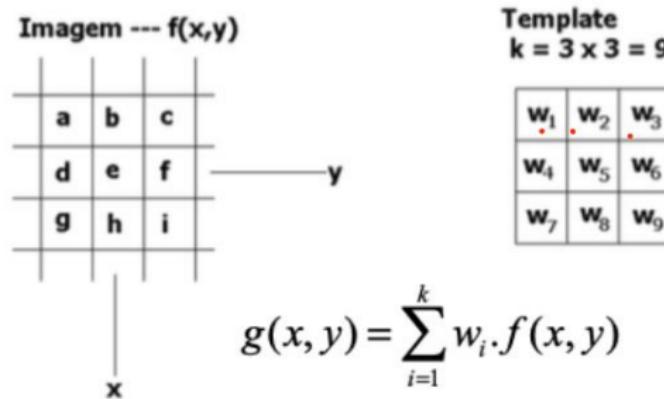


- Máscaras, *templates* e janelamento são os operadores locais mais empregados em Processamento de Imagens
 - Operadores de Vizinhança
 - Tipicamente, cada elemento do *template* é multiplicado pelo valor do pixel correspondente e a soma desses resultados é armazenada como o novo valor do nível de cinza na Nova Imagem
 - Seja w uma janela de $n \times n = k$ pixels. A Função de Transformação para cada pixel na imagem $g(x, y)$ será dada por:

$$g(x, y) = \sum_{i=1}^k w_i \cdot f(x, y)$$

Filtragem no Domínio Espacial

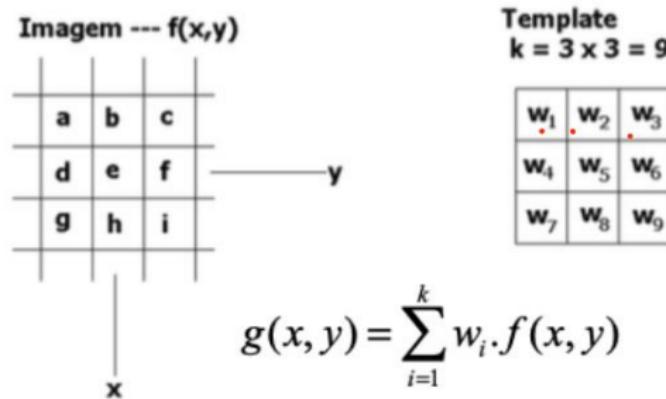
- Exemplo de aplicação de uma janela 3×3



- $(a, b, c, d, e, f, g, h, i)$ são os valores dos níveis de cinza na mesma vizinhança de $f(x,y)$, comparativamente ao *template*
- $(w_1 \text{ a } w_9)$ são os pesos, ou seja, os valores dos níveis de cinza em cada posição do *template*

Filtragem no Domínio Espacial

- Exemplo de aplicação de uma janela 3×3



- O valor do pixel $g(x,y)$ na nova imagem, na posição (x,y) será dado por:

$$g(x,y) = w_1.a + w_2.b + \dots + w_9.i$$

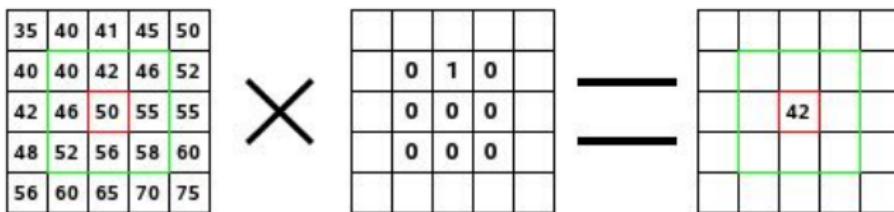
- Convolução: é o processo de aplicar uma matriz (chamada de núcleo ou kernel), sobre outra matriz que contém a imagem (chamada de matriz primária)
 - A convolução Discreta 2-D de $f(x, y)$ e $g(x, y)$ é definida por:

$$f(x) * g(x) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(i, j) \cdot g(x - i, y - j)$$

- Onde a matriz $M \times N$ é um período da convolução discreta bi-dimensional
- Um grande número de filtros usa a matriz de convolução internamente, como os de desfocagem, nitidez, mapa de relevo, dentre outros

Filtragem no Domínio Espacial

- Convolução



- **Isso é o que acontece:** o filtro lê sucessivamente, da esquerda para a direita e de cima para baixo, todos os pixels da área de ação do núcleo. O valor de cada pixel é multiplicado pelo valor correspondente do kernel e, em seguida, soma os resultados. O pixel inicial se tornou 42: $(40*0)+(42*1)+(46*0) + (46*0)+(50*0)+(55*0) + (52*0)+(56*0)+(58*0) = 42$.

Filtragem no Domínio Espacial

• Convolução

| | | | | |
|-----|-----|-----|-----|-----|
| 105 | 102 | 100 | 97 | 96 |
| 103 | 99 | 103 | 101 | 102 |
| 101 | 98 | 104 | 102 | 100 |
| 99 | 101 | 106 | 104 | 99 |
| 104 | 104 | 104 | 100 | 98 |

Matriz Imagem

| | | |
|----|----|----|
| 0 | -1 | 0 |
| -1 | 5 | -1 |
| 0 | -1 | 0 |

Núcleo

| | | | | |
|--|----|--|--|--|
| | | | | |
| | 89 | | | |
| | | | | |
| | | | | |
| | | | | |

Matriz Filtrada

| | | | | |
|-----|-----|-----|-----|-----|
| 105 | 102 | 100 | 97 | 96 |
| 103 | 99 | 103 | 101 | 102 |
| 101 | 98 | 104 | 102 | 100 |
| 99 | 101 | 106 | 104 | 99 |
| 104 | 104 | 104 | 100 | 98 |

Matriz Imagem

| | | |
|----|----|----|
| 0 | -1 | 0 |
| -1 | 5 | -1 |
| 0 | -1 | 0 |

Núcleo

| | | | | |
|--|----|-----|--|--|
| | | | | |
| | 89 | 111 | | |
| | | | | |
| | | | | |
| | | | | |

Matriz filtrada

$$\begin{aligned}102 * 0 + 100 * -1 + 97 * 0 \\+ 99 * -1 + 103 * 5 + 101 * -1 \\+ 98 * 0 + 104 * -1 + 102 * 0 = 111\end{aligned}$$

- Correlação
 - Se $f(x)$ e $g(x)$ forem iguais a relação é usualmente chamada de autocorrelação
 - Se $f(x)$ e $g(x)$ forem diferentes, o termo normalmente utilizado é Correlação Cruzada
 - Para o caso discreto bi-dimensional, a correlação entre $f(x, y)$ e $g(x, y)$ é dada por:

$$f(x) \circ g(x) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(i, j) \cdot g(x + i, y + j)$$

Filtragem no Domínio Espacial

- Convolução e correlação: a diferença entre esses operadores reside no espelhamento do *template* a ser utilizado
 - No geral, os *template* são simétricos
 - A correlação cruzada tem sido empregada com o nome de convolução de um *template* com uma imagem na área de processamento de imagens

| Template | Imagen | Resultado |
|--|--|--|
| $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ | $f(x,y)$ | $T(i,j)*f(x,y)$ |
| $T(i,j)$ | $\begin{bmatrix} 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 4 & 4 & 3 \\ 2 & 1 & 3 & 3 & 3 \\ 1 & 1 & 1 & 4 & 4 \end{bmatrix}$ | $\begin{bmatrix} 2 & 5 & 7 & 6 & * \\ 2 & 4 & 7 & 7 & * \\ 3 & 2 & 7 & 7 & * \\ * & * & * & * & * \end{bmatrix}$ |
| | $\begin{bmatrix} 1 & 0 & 3 & 3 & 4 \\ 0 & 1 & 4 & 4 & 3 \\ 2 & 1 & 3 & 3 & 3 \\ 1 & 1 & 1 & 4 & 4 \end{bmatrix}$ | |
| | | |

$$T(x,y) \circ f(x,y) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} T(i,j).f(x+i, y+j)$$

- Convenções para convolução
 - Máscaras de organização par (2×2 , 4×4 , ...) o resultado é colocado sobre o primeiro pixel
 - Máscaras de organização ímpar (3×3 , 5×5 , ...) o resultado é colocado sobre o pixel de centro
 - A imagem resultante da convolução não necessita, obrigatoriamente, ser menor que a imagem original
- tipos de convolução: aperiódica e periódica

Filtragem no Domínio Espacial

- Convolução aperiódica
 - Opção 1: É atribuído o valor 0 aos resultados não calculáveis
 - No exemplo abaixo, o primeiro ponto calculável é resultante da seguinte operação: $(1 * 1) + (1 * 2) + (1 * 3) + (0 * 0) + (0 * 1) + (0 * 3) + (1 * 1) + (1 * 1) + (1 \times 3) = 11$

Exemplo:

Template

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 0 | 1 | 3 |
| 1 | 1 | 3 |
| 0 | 0 | 4 |

Imagen

| | | | | |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| 0 | 1 | 3 | 4 | 0 |
| 1 | 1 | 3 | 2 | 0 |
| 0 | 0 | 4 | 5 | 6 |
| 1 | 0 | 7 | 8 | 0 |

Resultado

| | | | | |
|---|----|----|----|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 11 | 15 | 17 | 0 |
| 0 | 8 | 17 | 22 | 0 |
| 0 | 13 | 21 | 20 | 0 |
| 0 | 0 | 0 | 0 | 0 |

- Convolução aperiódica
 - Opção 2: Centra-se o *template* com o primeiro pixel da Imagem atribuindo o valor 0 aos valores inexistentes na Imagem
 - No exemplo abaixo, o primeiro ponto calculável é resultante da seguinte operação: $(1 * 0) + (1 * 0) + (1 * 0) + (0 * 0) + (0 * 1) + (0 * 2) + (1 * 0) + (1 * 0) + (1 * 1) = 1$

Exemplo:

| Template | | |
|----------|---|---|
| 1 | 1 | 1 |
| 0 | 0 | 0 |
| 1 | 1 | 1 |

Imagen

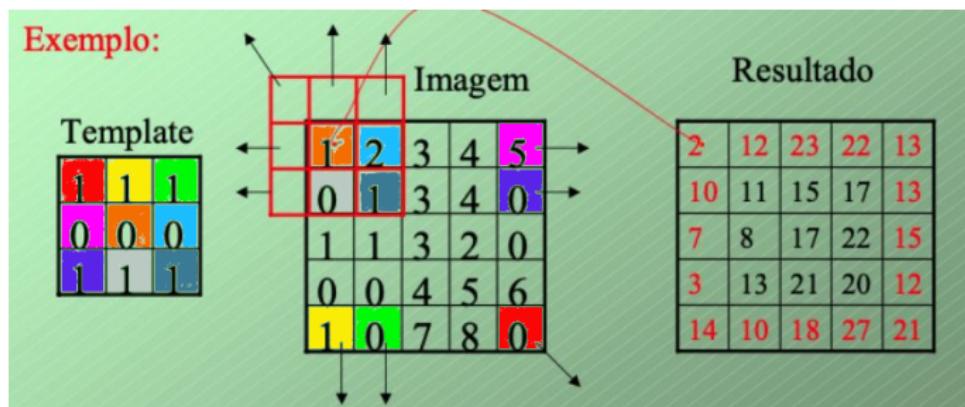
| | | | | |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| 0 | 1 | 3 | 4 | 0 |
| 1 | 1 | 3 | 2 | 0 |
| 0 | 0 | 4 | 5 | 6 |
| 1 | 0 | 7 | 8 | 0 |

Resultado

| | | | | |
|---|----|----|----|----|
| 1 | 4 | 8 | 7 | 4 |
| 4 | 11 | 15 | 17 | 11 |
| 1 | 8 | 17 | 22 | 15 |
| 3 | 13 | 21 | 20 | 10 |
| 0 | 4 | 9 | 15 | 11 |

Filtragem no Domínio Espacial

- Convolução periódica (ou circular)
 - Preenche os pixels inexistentes fora das bordas da imagem considerando que a mesma é circular
 - No exemplo abaixo, o primeiro ponto calculável é resultante da seguinte operação: $(1 * 0) + (1 * 1) + (1 * 0) + (0 * 5) + (0 * 1) + (0 * 2) + (1 * 0) + (1 * 0) + (1 * 1) = 2$
 - Por exemplo, o elemento na posição (0, 0) da máscara é multiplicado com o valor em (4, 4) da imagem



- Convolução
 - O custo computacional da convolução espacial é alto
 - Se a imagem é de tamanho $M \times M$ e o *template* tem dimensão $N \times N$, o número de multiplicações é na ordem de $M^2 * N^2$
 - Por exemplo, para uma imagem 1024×1024 e um *template* 12×12 são necessárias 150.994.944 multiplicações
 - A alternativa é converter a imagem e o *template* do domínio espacial para o domínio da frequência (e.g. transformada de Fourier) e multiplicar elemento a elemento
 - A transformada é justificável caso o *template* seja maior que 32×32 , devido ao custo da transformada de Fourier

Filtragem no Domínio Espacial

- Uso de convoluções para filtros
 - Realce

| | | | | |
|---|----|----|----|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | -1 | 0 | 0 |
| 0 | -1 | 5 | -1 | 0 |
| 0 | 0 | -1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |



Filtragem no Domínio Espacial

- Uso de convoluções para filtros
 - Desfoque

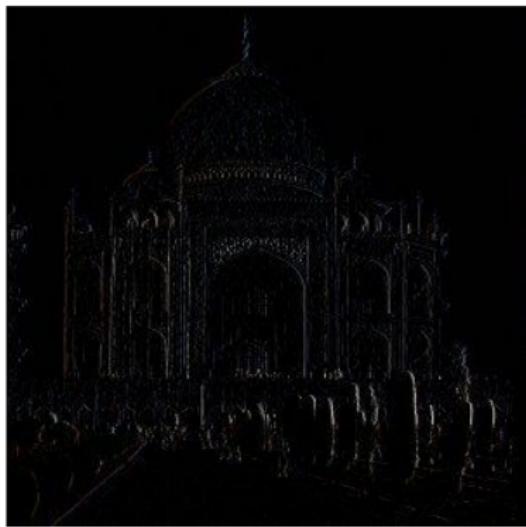
| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |



Filtragem no Domínio Espacial

- Uso de convoluções para filtros
 - Realce de bordas

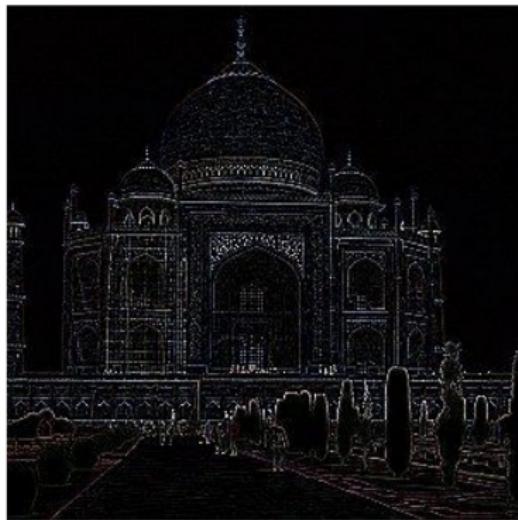
$$\begin{array}{|c|c|c|} \hline & & \\ \hline & 0 & 0 & 0 \\ \hline -1 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline & & \end{array}$$



Filtragem no Domínio Espacial

- Uso de convoluções para filtros
 - Detecção de bordas

$$\begin{array}{|c|c|c|} \hline & & \\ \hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array}$$



Filtragem no Domínio Espacial

- Uso de convoluções para filtros
 - Destaque de relevo

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



- Filtragem passa-baixa
 - Uma das aplicações da convolução espacial de uma imagem com templates é a suavização (*smoothing*) ou filtragem passa-baixa
 - Um filtro espacial passa-baixa é implementado através de uma Máscara que opera a média da vizinhança
 - Uma máscara de média é tal que seus pesos são positivos e a soma é igual a 1
 - Exemplos de algumas máscaras de filtros passa -baixa

$$\frac{1}{5} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \frac{1}{32} \begin{bmatrix} 1 & 3 & 1 \\ 3 & 16 & 3 \\ 1 & 3 & 1 \end{bmatrix} \quad \frac{1}{8} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Filtragem no Domínio Espacial

- Filtragem passa-baixa

- A suavização pode ser usada para:
 - Borramento (*blurring*)



$$\frac{1}{N^2} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}_{N \times N}$$

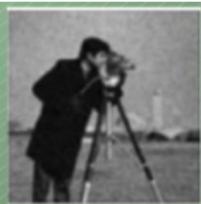


$N = 15$

- Redução de ruídos



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



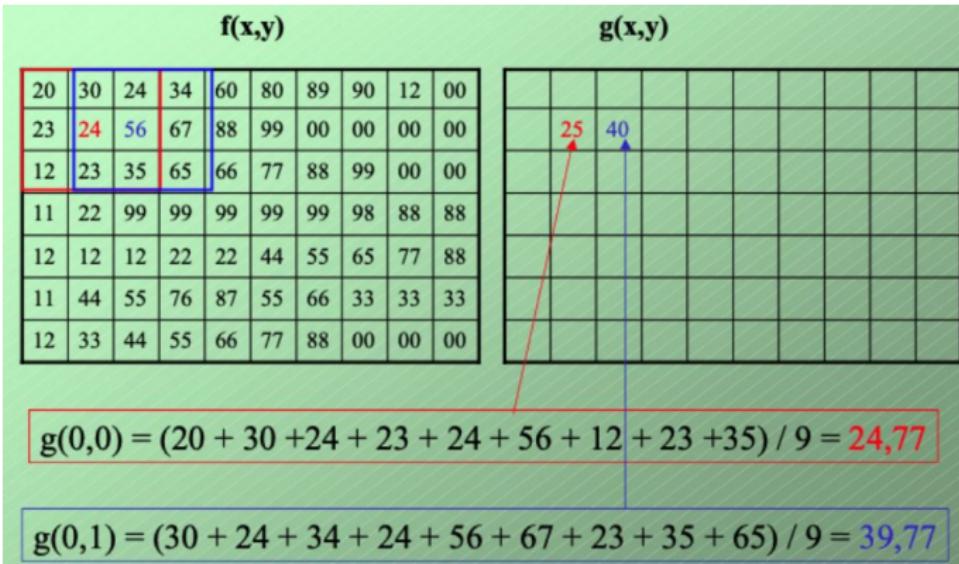
- Suavização: tem o objetivo de diminuir efeitos espúrios em uma imagem
 - Dada uma imagem $f(x,y)$ de $N \times N$ pixeis

$$g(x, y) = \frac{1}{M} \sum_{(n,m) \in S} f(n, m)$$

- S é o conjunto de coordenadas dos pontos na vizinhança de (x, y) , incluindo (x, y)

Filtragem no Domínio Espacial

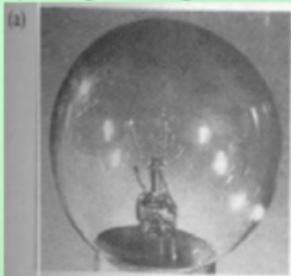
- Suavização



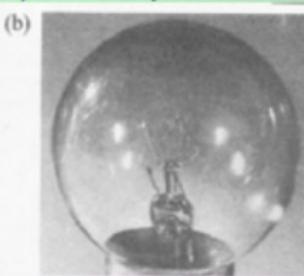
Filtragem no Domínio Espacial

- Suavização

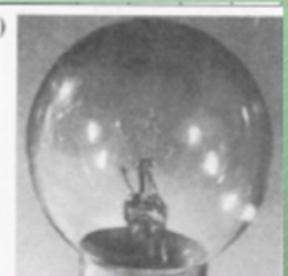
a) Imagem Original



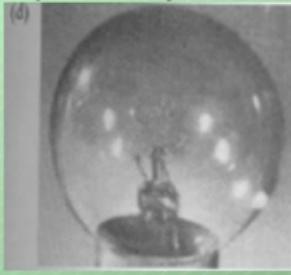
b) Vizinhança 3 x 3



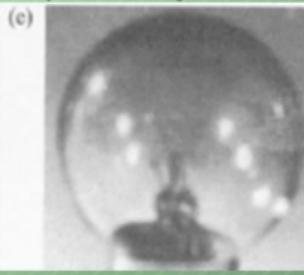
c) Vizinhança 5 x 5



d) Vizinhança 7 x 7



e) Vizinhança 15 x 15



f) Vizinhança 25 x 25



Filtragem no Domínio Espacial

- Média dos k-vizinhos mais próximos: define a média dentro de uma vizinhança de k-vizinhos que mais se aproximem do pixel em questão

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 20 | 30 | 24 | 34 | 60 | 80 | 89 | 90 | 12 | 00 |
| 23 | 24 | 56 | 67 | 88 | 99 | 00 | 00 | 00 | 00 |
| 12 | 23 | 35 | 65 | 66 | 77 | 88 | 99 | 00 | 00 |
| 11 | 22 | 99 | 99 | 99 | 99 | 99 | 98 | 88 | 88 |
| 12 | 12 | 12 | 22 | 22 | 44 | 55 | 65 | 77 | 88 |
| 11 | 44 | 55 | 76 | 87 | 55 | 66 | 33 | 33 | 33 |
| 12 | 33 | 44 | 55 | 66 | 77 | 88 | 00 | 00 | 00 |

Exemplo:

Vizinhança de 5 x 5

$k = 9$

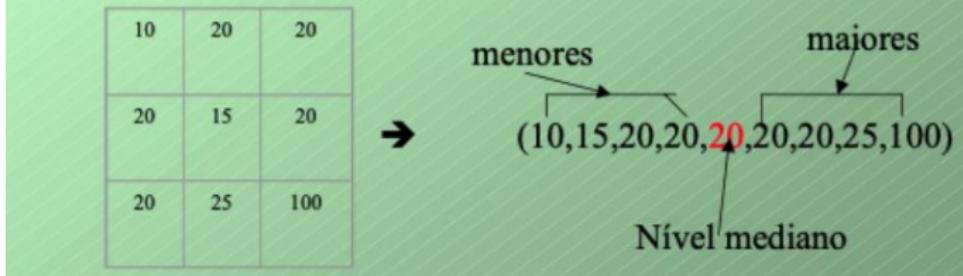
(11,12,12,12,12,20,22,22,22,23,23,24,24,30,34,35,56,60,65,66,67,88,99,99,99)

$$|35-22| = 13 \quad |35-56| = 21$$

$$M = (22+22+22+23+23+24+24+30+34) / 9 = 22,33$$

- Filtragem da mediana
 - Substitui o nível de cinza de cada pixel pelo nível de cinza mediano em uma vizinhança do pixel
 - O nível mediano de um conjunto de valores é tal que exista metade dos valores menores e metade dos valores maiores

Exemplo:



Filtragem no Domínio Espacial

- Filtragem da mediana
 - A) Imagem original
 - B) Imagem com ruído
 - C) Filtro da mediana 3×3
 - D) Filtro da mediana 5×5



- Filtragem passa-alta
 - Outra aplicação essencial da convolução espacial de uma máscara em uma imagem é o realce (*sharpening*) ou filtragem passa-alta
 - É chamada de máscara de realce porque tende a realçar mudanças abruptas de níveis de cinza na imagem
 - A máscara do filtro passa-alta deve ter o peso central positivo e os pesos periféricos negativos tal que a soma seja igual a zero
 - Exemplos de filtros

$$\frac{1}{9} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

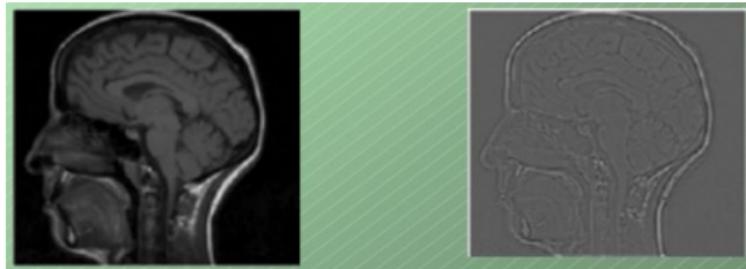
$$\frac{1}{5} \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Filtragem no Domínio Espacial

- Filtragem passa-alta
 - A Imagem filtrada por passa-alta $g(x, y)$ de uma Imagem $f(x, y)$ pode ser expressa também como:

$$g(x, y) = f(x, y) - LowPass(f(x, y))$$

- onde $LowPass(x, y)$ é a imagem resultado da filtragem passa-baixa de $f(x, y)$
- Exemplo de aplicação do filtro passa-alta



Referências I

-  Borges, L. E.
Python para desenvolvedores.
Novatec, 2017.
-  Bradski, G.
The openCV library.
Dr. Dobb's Journal: Software Tools for the Professional Programmer, v. 25, n. 11, p. 120-123, 2000.
-  Foley, J. D., Van, F. D., Van Dam, A., Feiner, S. K., Hughes, J. F., e Hughes, J.
Computer Graphics: Principles and Practice.
Addison-Wesley, 1996.
-  Gonzalez, Rafael C., e Richard E. Woods.
Processamento de imagens digitais.
Editora Blucher, 2000.

Referências II



Oliveira, M. M.

Notas de aula – Fundamentos de Processamento de Imagens.
Porto Alegre, 2010.

Disponível em: http://www.inf.ufrgs.br/~oliveira/Cursos/INF01046/INF01046_descricao_2010_2.html.

Acesso em: 10/12/2021.



Prateek, J., Millan, E. D., e Vinivius, G.

OpenCV by Example.

Packt Publishing, 2016.



Villán, A. F.

Mastering OpenCV 4 with Python.

Packt Publishing, 2019.



Wiggers, K. L.

Notas de aula – Processamento de Imagens: filtragem no domínio espacial. Pato Branco, PR, 2024.