

# **PMR2560 – Visão Computacional**

## **Filtragem e Suavização**

Prof. Eduardo L. L. Cabral



# Objetivos

- Processamento de imagens:
  - Convolução;
  - Filtragem 2D;
  - Suavização de imagens.

# Convolução

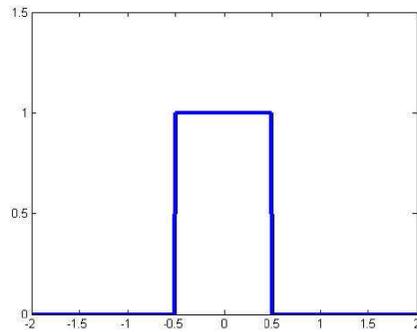
- Definição de convolução no tempo em 1D:

$$y(t) = g(t) \otimes u(t) = \int_{-\infty}^{+\infty} g(\tau)u(t-\tau)d\tau$$

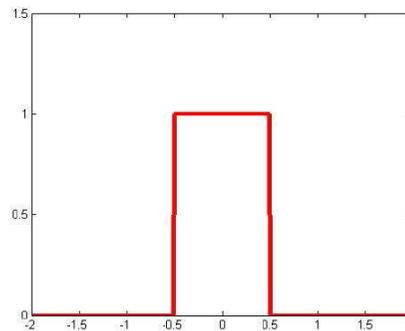
- O que faz a convolução?
  - Para um dado instante de tempo “ $t$ ”:
    - Pegar a imagem de  $u \Rightarrow u(-\tau)$ ;
    - Deslocar  $u$  por um dado tempo  $\Rightarrow u(t-\tau)$  ;
    - Multiplicar por  $g(t) \Rightarrow g(\tau)u(t-\tau)$ ;
    - Integrar.
  - Repetir para todo instante de tempo de  $[-\infty, +\infty]$ .
  - Como as imagens são finitas e no espaço  $\Rightarrow$  convolução no espaço sobre uma região finita.

# Convolução

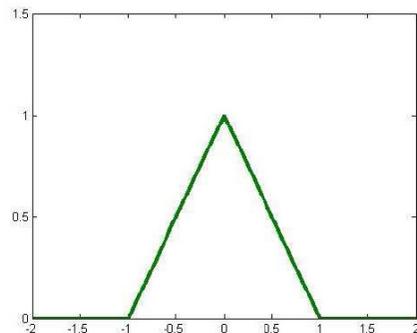
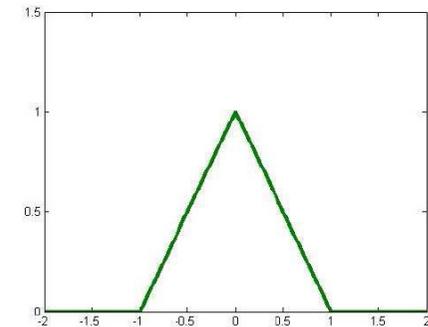
- Exemplos de convolução em 1D:



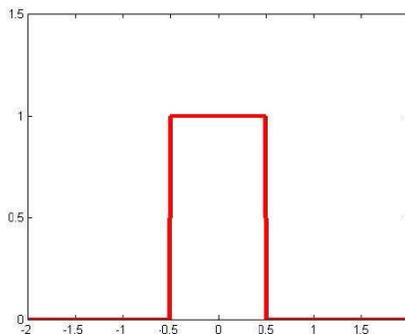
$\otimes$



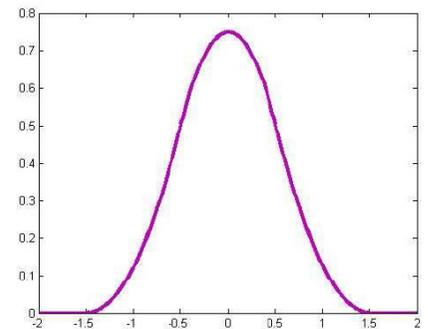
=



$\otimes$



=



⇒ Mais exemplos: <http://mathworld.wolfram.com/Convolution.html>

# Convolução discreta

- Para funções discretas a integral é trocada por uma somatória:
- A convolução em tempo discreto é definida por:

$$h(nT) = g(nT) \otimes u(nT) = \sum_{k=0}^N g(kT)u(nT - kT)$$

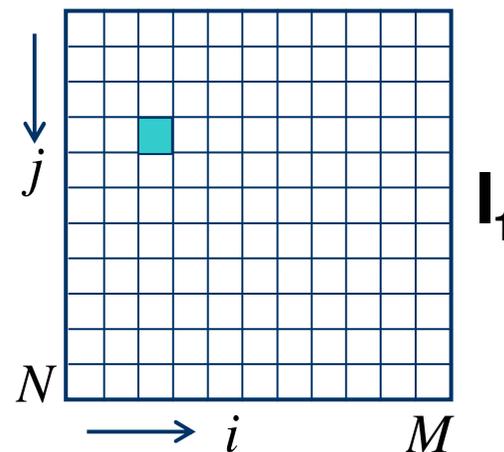
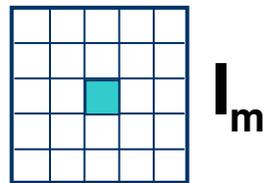
- No caso de imagens:
  - $g(nT)$  é uma imagem que é função do espaço;
  - $u(nT)$  é um filtro (máscara) que também é função do espaço.

# Convolução em 2D

- Convolução em 2D de duas imagens digitais é definida por:

$$I_2(i, j) = I_m \otimes I_1 = \sum_{k=1}^N \sum_{l=1}^M I_1(k, l) I_m(i-k, j-l)$$

onde  $I_1$  e  $I_2$  são imagens de dimensão  $N \times M$  e  $I_m$  é a máscara de dimensão  $n \times m$ .



# Convolução em 2D

- Exemplos de máscaras com dimensão 3x3:

$$\mathbf{I}_m = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \mathbf{I}_m = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

- Máscaras de 2D podem ser vistas como sendo a convolução de duas máscaras de 1D.

(a) usar 

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

 ou 

1
2
1

 ou (b) usar 

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

 \* 

1
2
1

 = 

1	2	1
2	4	2
1	2	1

- Eficiência  $\Rightarrow$  convolver uma imagem com uma máscara de dimensão  $n \times m$  requer  $n+m$  multiplicações versus  $n \times m$  multiplicações exigidas para convolver a imagem com duas máscaras com dimensões  $n \times 1$  e  $1 \times m$ .

# Convolução em 2D

- Exemplo  $\Rightarrow I_2 = I_1 \otimes I_m$

$I_m$

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

Girar ↓

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

$I_1$

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

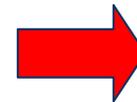
# Convolução em 2D

Passo 1

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

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

1	1	1		
-1	4	2	2	3
-1	-2	1	3	3
$I_1$	2	2	1	2
	1	3	2	2



5			

$I_2$

# Convolução em 2D

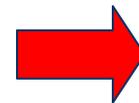
Passo 2

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

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

$I_1$

1	1	1	
-2	4	2	3
-2	-1	3	3
2	2	1	2
1	3	2	2



$I_2$

5	4		

# Convolução em 2D

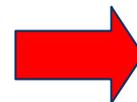
Passo 3

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

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

$I_1$

		1	1	1	
2	-2	4	3		
2	-1	-3	3		
2	2	1	2		
1	3	2	2		



$I_2$

5	4	4	

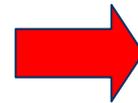
# Convolução em 2D

Passo 4

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

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

		1	1	1
2	2	-2	6	1
2	1	-3	-3	1
2	2	1	2	
1	3	2	2	



5	4	4	-2

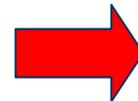
# Convolução em 2D

Passo 5

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

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

1	2	2	2	3
-1	4	1	3	3
-1	-2	2	1	2
$I_1$	1	3	2	2



5	4	4	-2
9			

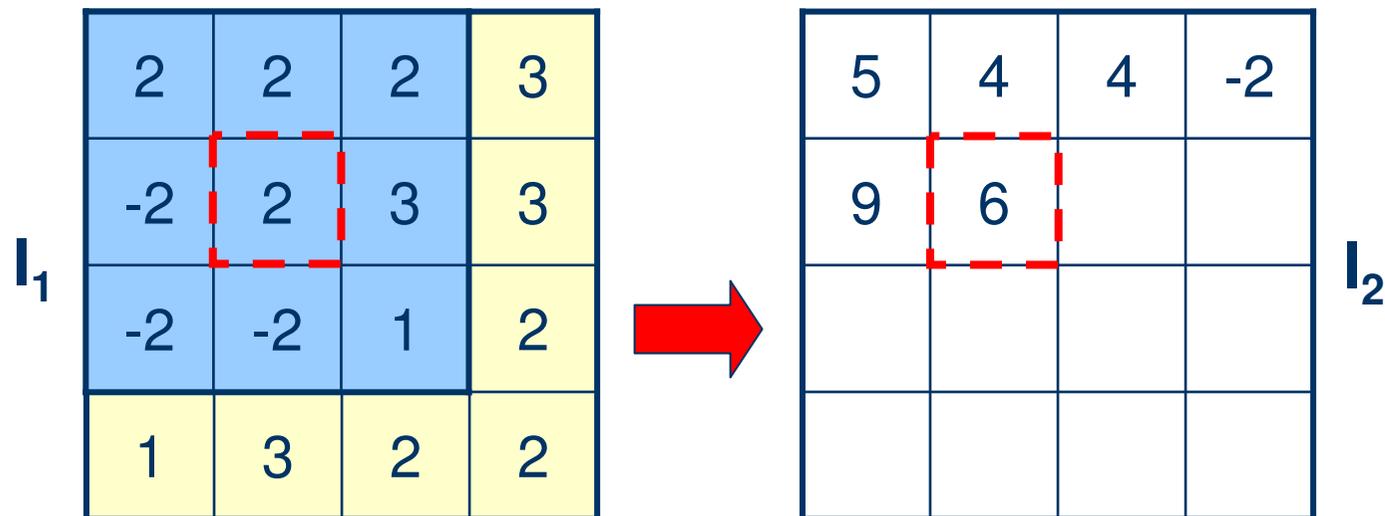
$I_2$

# Convolução em 2D

Passo 6

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

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2



# Convolução em 2D

## Resultado Final

5	4	4	-2
9	6	14	5
11	7	6	5
9	12	8	5

$I_2$

# Filtragem em 2D

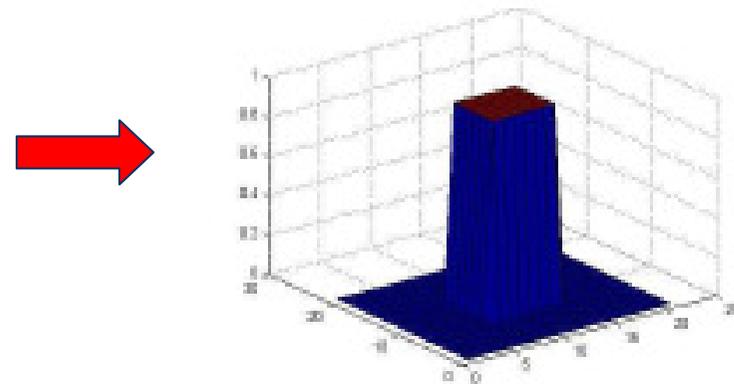
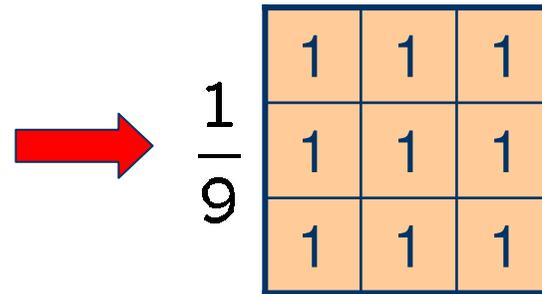
- Filtragem = processo de convolução sem inversão da máscara.
- Usada para obter transformar a imagem:
  - Suavização;
  - Ressaltamento;
  - Cálculo de gradiente;
  - Determinação de bordas;
  - Identificação de texturas.

# Suavização de imagens

- Suavização  $\Rightarrow$  **filtragem passa baixa**.
- Troca cada pixel da imagem por uma média ponderada de seus vizinhos.
- Benefícios  $\Rightarrow$  elimina ruídos e efeitos de “aliasing”.
- Problemas  $\Rightarrow$  a imagem se torna embaçada.
- Tipos de filtros:
  - Média;
  - Mediana (não-linear)  $\Rightarrow$  mediana é o valor (pertencente ou não à amostra) que a divide ao meio, isto é, 50% dos elementos da amostra são menores ou iguais à mediana e os outros 50% são maiores ou iguais à mediana;
  - Gaussiana.

# Suavização de imagens - média

- Substitui o valor do pixel pela média aritmética simples dos valores dos pixels vizinhos.
- Exemplo de máscara de filtro tipo média de dimensão 3x3.
- Também conhecido como filtro “caixa”.



# Suavização de imagens - média

- Exemplo de imagem suavizada.



Imagem original

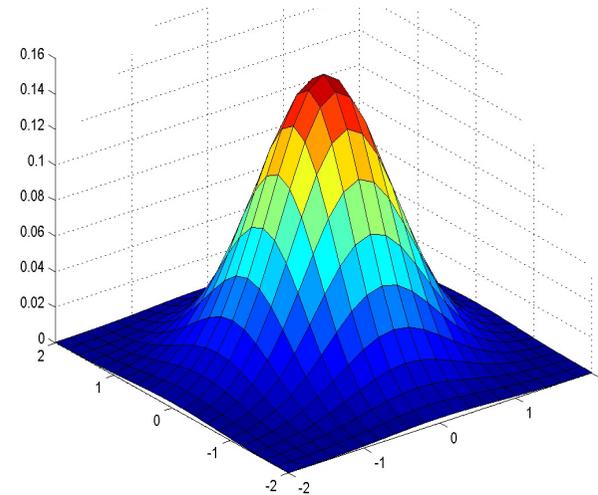


Imagem suavizada com filtro 7x7

# Suavização de imagens - Gaussiana

- Baseado na distribuição Gaussiana:

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



- Simetria rotacional trata todas as direções igualmente (isotropia).
- Regra geral é usar filtro com dimensão  $\geq 5\sigma$ .
- Tipicamente se usa filtro de dimensão 5x5:
  - Engloba 98,8% da distribuição gaussiana quando  $\sigma = 1$  pixel.

# Suavização de imagens - Gaussiana

- Geração de filtros Gaussianos:
  - Filtros gaussianos discretos de 1D, com  $\sigma = 1$ , podem ser gerados pelos coeficientes do triângulo de Pascal;

```
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
 1 5 10 10 5 1
1 6 15 20 15 6 1
```

- Coeficientes do filtro em 2D podem ser obtidos pela convolução de dois filtros em 1D com componentes horizontal e vertical (Propriedade da Separabilidade).

# Suavização de imagens - Gaussiana

- Distribuição Gaussiana com média zero em 1D:

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

- Área da distribuição Gaussiana:

Coordenada $x$ (pixel)	Meia área (%)
$1\sigma$	68,26
$2\sigma$	95,0
$2,5\sigma$	98,8
$3\sigma$	99,7
$4\sigma$	99,99

# Suavização de imagens - Gaussiana

- Filtro 1D com  $\sigma = 1$  pixel:
  - Número de pixels para se ter 98,8% da área =  $2,5\sigma = 2,5$  pixels.
  - Perfil da distribuição Gaussiana:

$x$ (pixel)	$G(x)$	<i>Filtro</i>
0	0,3989	6
1	0,2420	4
2	0,0540	1

onde  $G(x)$  é a Gaussiana original e *Filtro* é a máscara utilizada obtida pela multiplicação da Gaussiana por uma constante e arredondada para inteiro.

# Suavização de imagens - Gaussiana

- Filtro 1D com  $\sigma = 2$  pixels:
  - Número de pixels para se ter 98,8% da área =  $2,5\sigma = 5$  pixels.
  - Perfil da distribuição Gaussiana:

$x$ (pixel)	$G(x)$	<i>Filtro</i>
0	0,2821	7
1	0,2489	6
2	0,1711	4
3	0,0916	2
4	0,0382	1

onde  $G(x)$  é a Gaussiana original e *Filtro* é a máscara utilizada obtida pela multiplicação da Gaussiana por uma constante e arredondada para inteiro.

# Suavização de imagens - Gaussiana

- Filtro Gaussiano 2D  
com  $\sigma = 1$  pixel:

$$\frac{1}{16} [1 \ 4 \ 6 \ 4 \ 1] \otimes \frac{1}{16} \begin{bmatrix} 1 \\ 4 \\ 6 \\ 4 \\ 1 \end{bmatrix} = \frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

- Filtro Gaussiano 2D  
com  $\sigma = 2$  pixels:

$$\frac{1}{33} [1 \ 2 \ 4 \ 6 \ 7 \ 6 \ 4 \ 2 \ 1] \otimes \frac{1}{33} \begin{bmatrix} 1 \\ 2 \\ 4 \\ 6 \\ 7 \\ 6 \\ 4 \\ 2 \\ 1 \end{bmatrix} = \frac{1}{1089} \begin{bmatrix} 1 & 2 & 4 & 6 & 7 & 6 & 4 & 2 & 1 \\ 2 & 4 & 8 & 12 & 14 & 12 & 8 & 4 & 2 \\ 4 & 8 & 16 & 24 & 28 & 24 & 16 & 8 & 4 \\ 6 & 12 & 24 & 36 & 42 & 36 & 24 & 12 & 6 \\ 7 & 14 & 28 & 42 & 49 & 42 & 28 & 14 & 7 \\ 6 & 12 & 24 & 36 & 42 & 36 & 24 & 12 & 6 \\ 4 & 8 & 16 & 24 & 28 & 24 & 16 & 8 & 4 \\ 2 & 4 & 8 & 12 & 14 & 12 & 8 & 4 & 2 \\ 1 & 2 & 4 & 6 & 7 & 6 & 4 & 2 & 1 \end{bmatrix}$$

# Suavização de imagens - Gaussiana

- Exemplo de imagem suavizada:



Imagem original



$\sigma = 1$

Máscara  
7x7



$\sigma = 3$

# Suavização de imagens - Gaussiana

Imagem original



$\sigma = 2$



$\sigma = 2,8$



$\sigma = 4$



# Efeito de bordas

- Como lidar com as bordas da imagem?
  - Repetir o valor da borda da imagem original:
    - Pode introduzir altas derivadas de 1ª ordem.
  - Ignorar:
    - Imagem resultante será menor do que a imagem original.
  - Colocar valor constante nas bordas:
    - Pode introduzir altas derivadas de 1ª ordem.
  - Repetir valor do pixel interno na borda:
    - Pode introduzir altas derivadas de 2ª ordem.

10	11	11	10	10
12	150	15	14	14
13	13	255	14	14
10	11	11	10	10
10	11	11	10	10

0	0	0	0	0	0
0	10	11	11	10	10
0	12	150	15	14	14
0	13	13	255	14	14
0	10	11	11	10	10
0	10	11	11	10	10

# Exercícios

1. Faça a filtragem da imagem abaixo com a máscara dada e gere uma nova imagem. Ignore as bordas da imagem.
2. Discuta os efeitos dessa filtragem na imagem original.

$$\begin{bmatrix} 12 & 13 & 13 & 14 & 13 \\ 13 & 15 & 16 & 15 & 16 \\ 200 & 198 & 199 & 199 & 199 \\ 201 & 200 & 200 & 200 & 200 \\ 202 & 201 & 200 & 200 & 201 \end{bmatrix} \otimes \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & - & - & - & 0 \\ 0 & - & - & - & 0 \\ 0 & - & - & - & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

# Exercícios

3. Forme um filtro gaussiano de dimensão 3x3 com desvio padrão de 1 pixel.
4. Faça a filtragem da imagem do exercício 1 com o filtro do exercício 3.
5. Forme um filtro gaussiano com desvio padrão de 1,5 pixels. Qual a dimensão ideal para esse filtro?
6. Faça a filtragem da imagem do exercício 1 com o filtro do exercício 5.
7. Você observa alguma diferença entre as imagens resultantes dos exercícios 4 e 6?