

UNIP

UNIVERSIDADE PAULISTA

Processamento de Imagem



Relacionamentos entre pixels

e

Operações Aritméticas e Lógicas

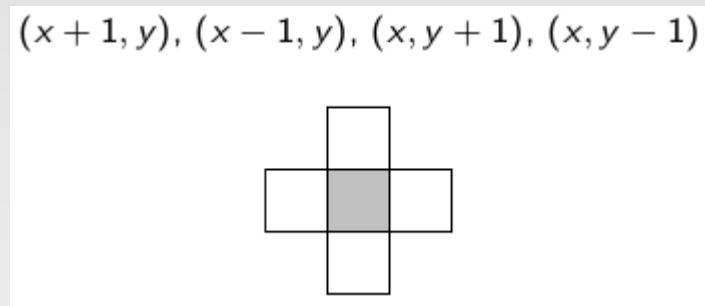
Professora Sheila Cáceres

Relacionamentos básicos entre elementos de uma imagem

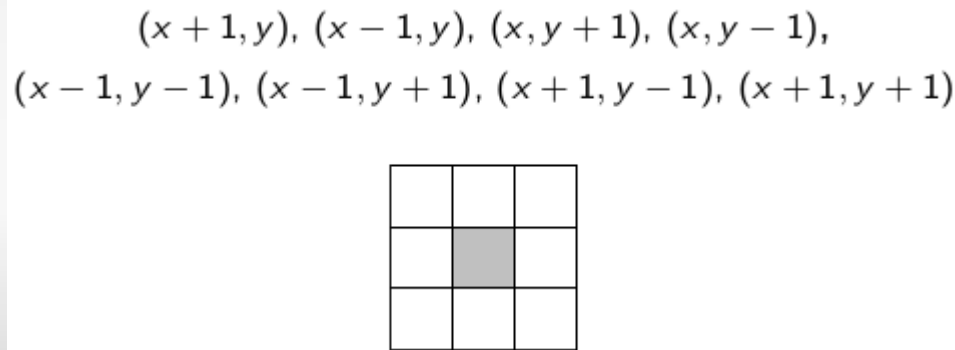
- Vizinhança
- Conectividade
- Adjacência
- Caminho
- Componentes Conexos
- Borda e Interior
- Distancia

Vizinhança

- Vizinhança-4: Um pixel $f(x,y)$ possui 4 vizinhos horizontais e verticais como ilustrado.

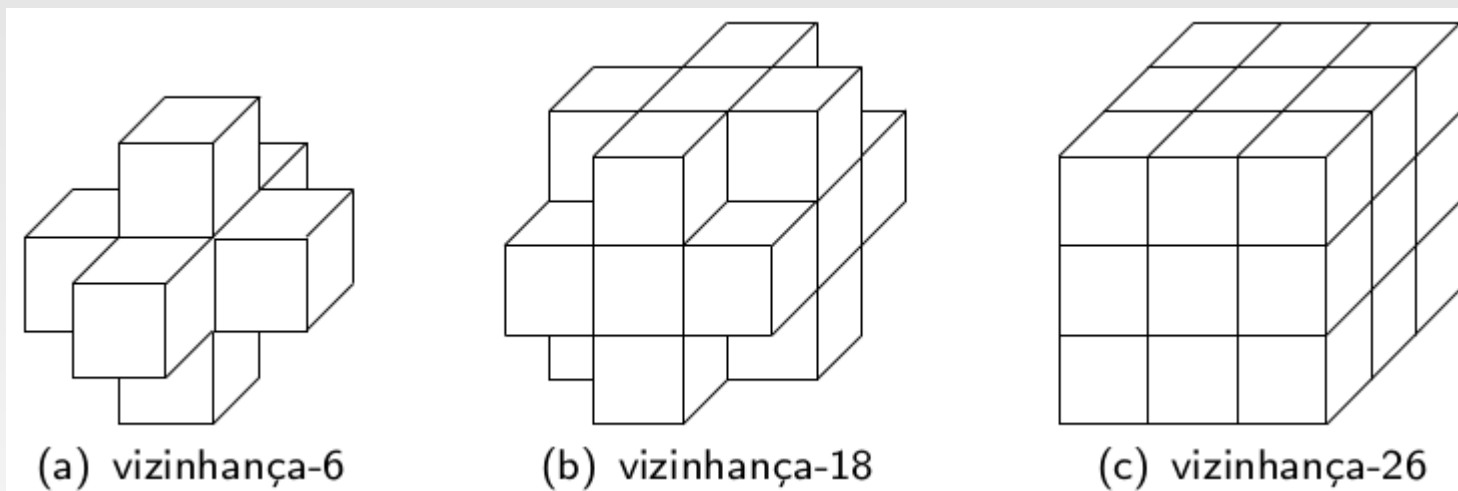


- Vizinhança-8: Um pixel $f(x,y)$ possui 8 vizinhos horizontais, verticais e diagonais como ilustrado.



Vizinhança

- Estendendo esse conceito para imagens tridimensionais temos:



Conectividade

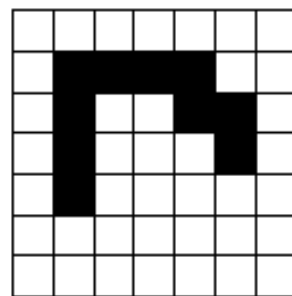
- Dois objetos são conexos se são vizinhos e similares (de acordo com um critério de vizinhança e similaridade adotado).
 - Critérios de vizinhança: Vizinhança-4, Vizinhança-8, etc.
 - Critérios de similaridade: intensidade, cor, textura.

Adjacência

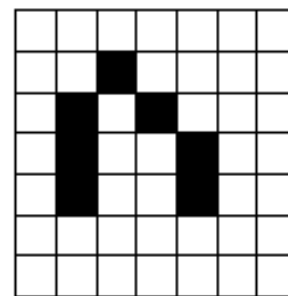
- Dois elementos (podem ser subconjuntos de pixels) são adjacentes se forem conexos em pelo menos um elemento.

Caminho

- É uma sequência de pixels distintos onde cada pixel é adjacente ao seguinte pixel na sequência.
- Os pixels adjacentes podem considerar vizinhança-4, vizinhança-8, etc.
- O número de elementos na sequência representa o comprimento do caminho
- Na Figura (a) o comprimento considerando caminho-4 é 10. Na Figura (b) é 7.



(a) caminho-4



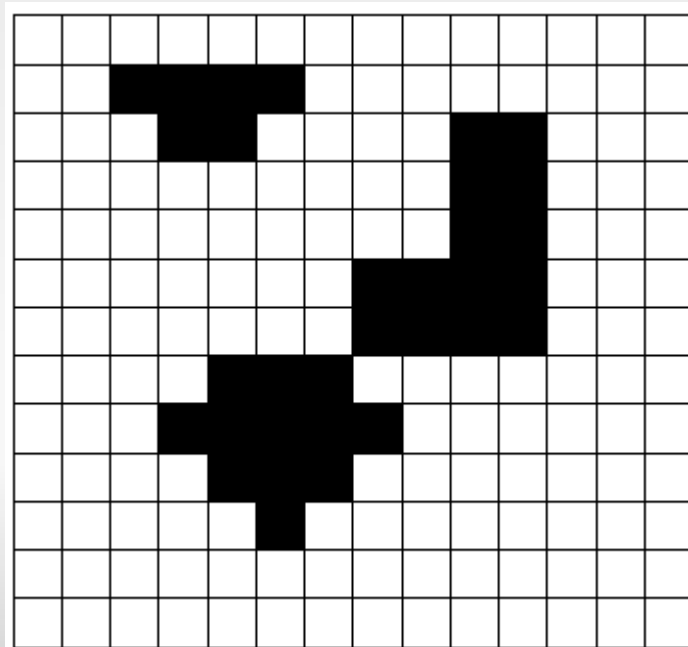
(b) caminho-8

Componentes Conexos

- É um subconjunto de elementos C da imagem que são conexos entre si.

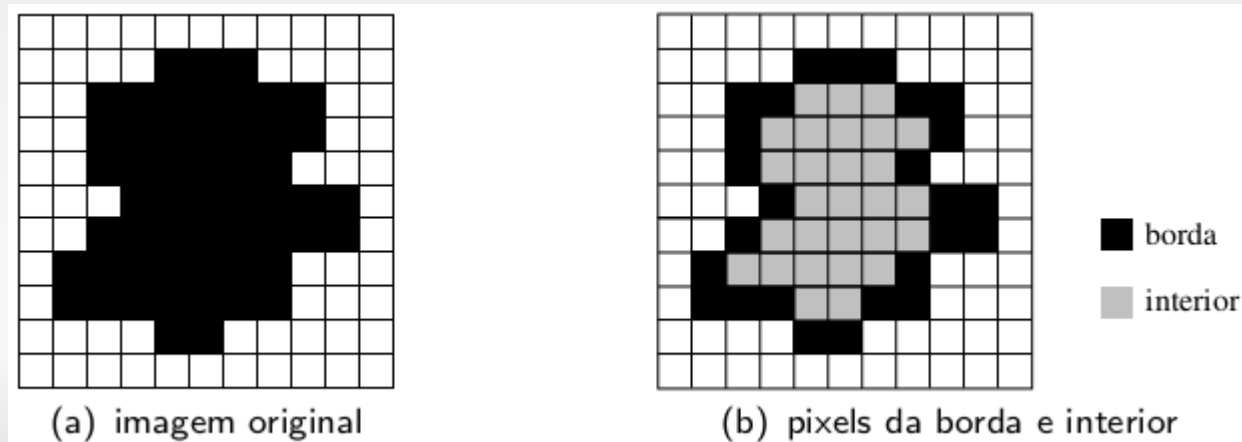
Exemplo:

- 3 componentes conexos com vizinhança-4
- 2 componentes conexos com vizinhança-8



Bordar e Interior

- A borda corresponde ao conjunto de pontos no contorno do componente conexo.
- A borda de um componente conexo S em uma imagem bidimensional é o conjunto de pixels pertencentes ao componente e que possuem vizinhança-4 com um ou mais pixels externos a S .
- O interior é o conjunto de pixels de S que não estão em sua borda.



Medidas de Distancia

- Existem várias formas de computar a distância entre dois pixels ou componentes de uma imagem.
- Dados os pixels f_1 , f_2 e f_3 , com coordenadas (x_1, y_1) , (x_2, y_2) e (x_3, y_3) respectivamente, qualquer métrica de distância D deve satisfazer todas as seguintes propriedades:
 - (i) $D(f_1, f_2) \geq 0$ ($D(f_1, f_2) = 0$ somente se $f_1 = f_2$)
 - (ii) $D(f_1, f_2) = D(f_2, f_1)$
 - (iii) $D(f_1, f_3) \leq D(f_1, f_2) + D(f_2, f_3)$

Distância Euclidiana

- A distância Euclidiana entre f_1 e f_2 é definida:

$$D_E(f_1, f_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

- Os pixels com distância menor ou igual a algum valor d formam um **disco** de radio d centrado em f_1 (exemplo com $D \leq 3$)

			3			
	$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$	
	$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$	
3	2	1	0	1	2	3
	$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$	
	$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$	
			3			

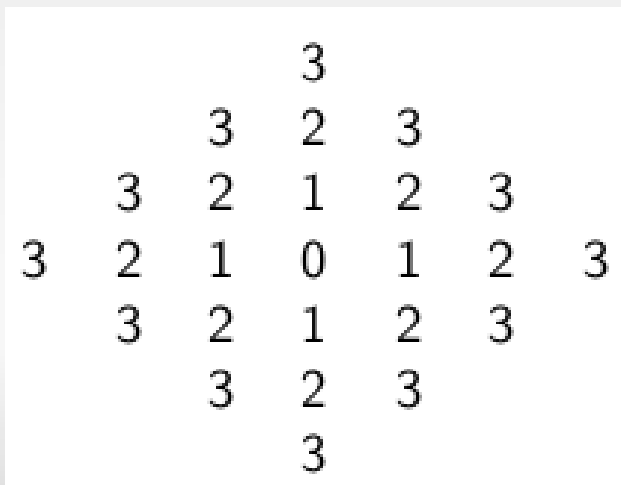
- Está próxima ao caso contínuo mas requer esforço computacional elevado e pode produzir valores fracionários

Distância D4

- Também denominada city-block, definida por:

$$D4 (f1 , f2) = |x1 - x2 | + |y1 - y2 |$$

- Os pixels com uma distância D4 de f1 menor ou igual a algum valor d formam um losango centrado em f1 (Exemplo: figura com $D4 \leq 3$)
- Notar que pontos com distância 1 são pixels com vizinhança-4 do ponto central .



•A distância D4 entre dois pixels f1 e f2 é igual ao comprimento do caminho mais curto entre esses pixels, considerando-se a vizinhança-4.

Distância D8

- Também denominada chessboard, definida por:
$$D8 (f1 , f2) = \max(|x1 - x2 |, |y1 - y2 |)$$
- Os pixels com uma distância D8 de f1 menor ou igual a algum valor d formam um **quadrado** centrado em f1 (Exemplo: figura com $D8 \leq 3$)
- Notar que pontos com distância 1 são pixels com vizinhança-8 do ponto central .

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

A distância D8 corresponde ao caminho-8 mais curto entre esses pontos.

Operações Aritméticas e Lógicas

Usando uma imagem

- Modificar a intensidade de cada pixel
- Operações de limiarização
- Invertir uma imagem

Usando duas imagens

Operações Aritméticas

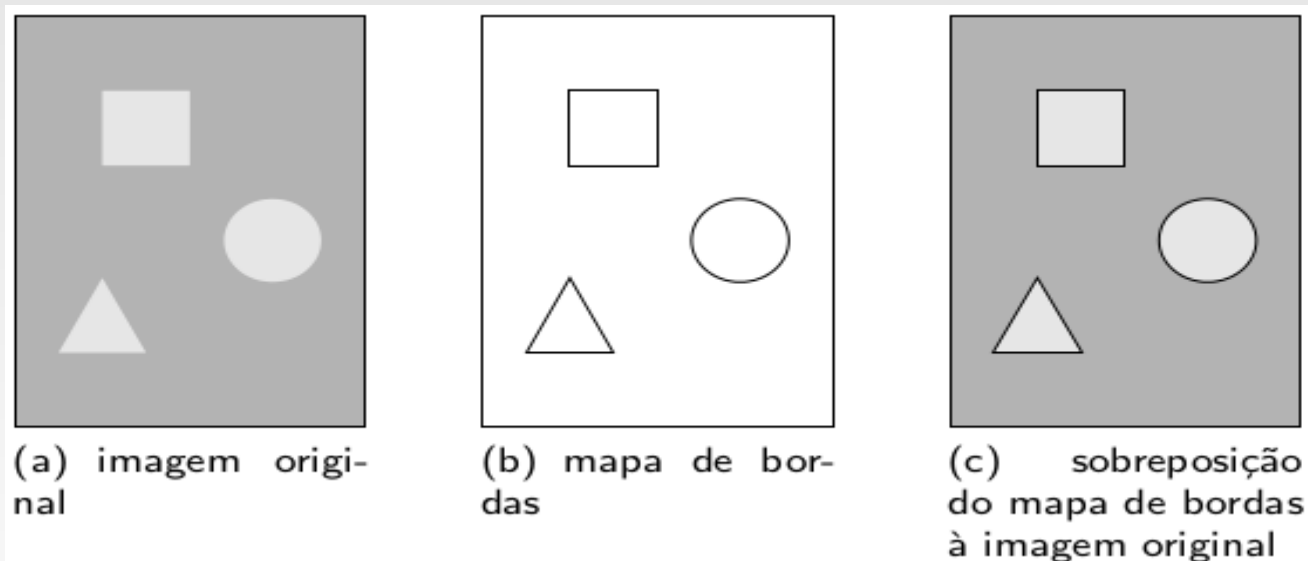
- Dadas duas imagens f_1 e f_2 , as operações aritméticas entre dois pixels $f_1(x,y)$ e $f_2(x,y)$ são denotadas assim:

Adição	$f_1(x, y) + f_2(x, y)$
Subtração	$f_1(x, y) - f_2(x, y)$
Multiplicação	$f_1(x, y) \cdot f_2(x, y)$
Divisão	$f_1(x, y) / f_2(x, y)$

- O resultado dessas operações pode ser fora do intervalo usado.
- Por exemplo, '+': valores mto grandes, '-': valores negativos. Portanto pode-se realizar uma transformação para manter os valores dentro do intervalo usado.
- Outro exemplo é a divisão que pode produzir valores fracionários (devem ser convertidos a inteiros) e deve-se evitar a divisão por zero (pode-se adicionar 1: teríamos 1-256 no lugar de 0-255)

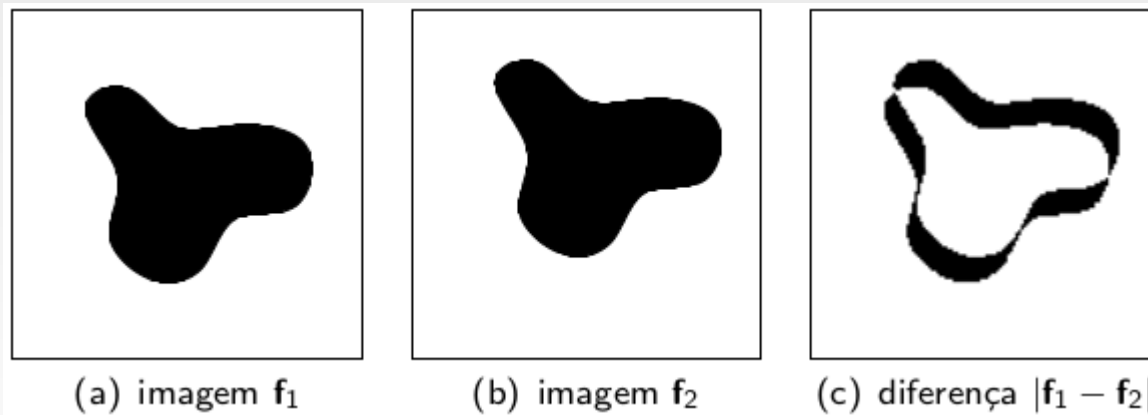
Adição de Imagens

- Pode ser usada para sobrepor o conteúdo de uma imagem em outra.
- Se combinarmos a figura a e b obtemos c.



Subtração de Imagens

- Tem diversas aplicações como a identificação de diferenças entre imagens.
- O movimento de objetos pode ser medido com a subtração, as regiões dos objetos que permanecem inalteradas são eliminadas (objetos movidos são claramente mostrados)



Multiplicação e Divisão

- Uma das principais aplicações da multiplicação ou divisão é no ajuste de brilho (para corrigir problemas durante a aquisição).
- Outras aplicações desses operadores são a filtragem de imagens no domínio da frequência (será visto proximamente) e na modelagem de ruído.

Operações Lógicas

- As principais operações lógicas são:

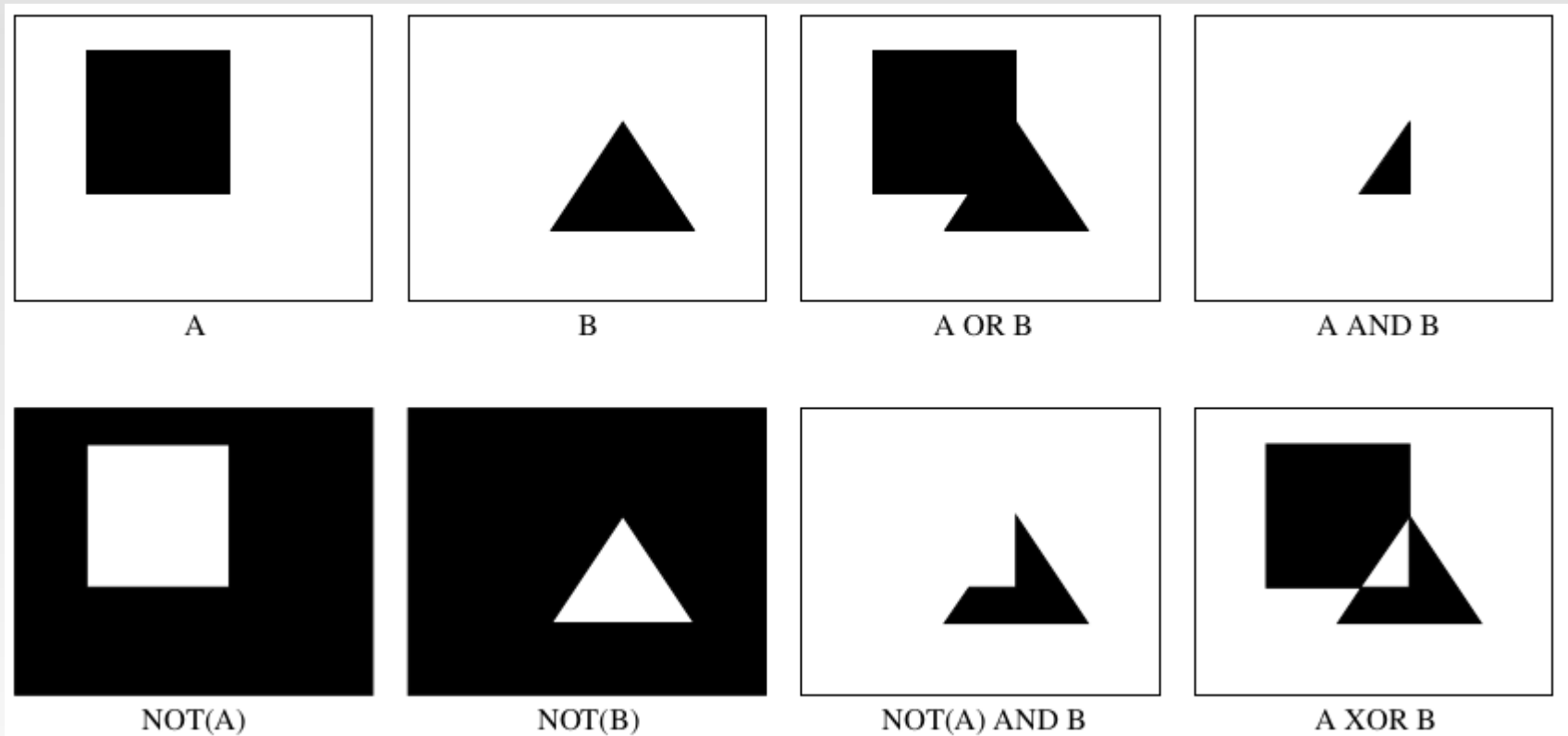
AND	$f_1(x, y) \text{ AND } f_2(x, y)$
OR	$f_1(x, y) \text{ OR } f_2(x, y)$
XOR	$f_1(x, y) \text{ XOR } f_2(x, y)$
NOT	$NOT(f_1(x, y))$

- Podem ser combinadas para formar expressões mais complexas.
- Operações lógicas podem ser aplicadas apenas a imagens binárias (em contraste com as operações aritméticas).
- Terminologia: Pixels pretos (0)->Objeto, pixels brancos (255)->fundo

Operações Lógicas

- AND produz 1 na saída quando os pixels correspondentes nas duas imagens de entrada possuem 1.
- XOR produz 1 quando apenas um dos pixels (não ambos) é 1.
- OR produz 1 quando pelo menos um dos pixels das imagens é 1
- NOT inverte o valor de cada pixel da imagem

Operações Lógicas

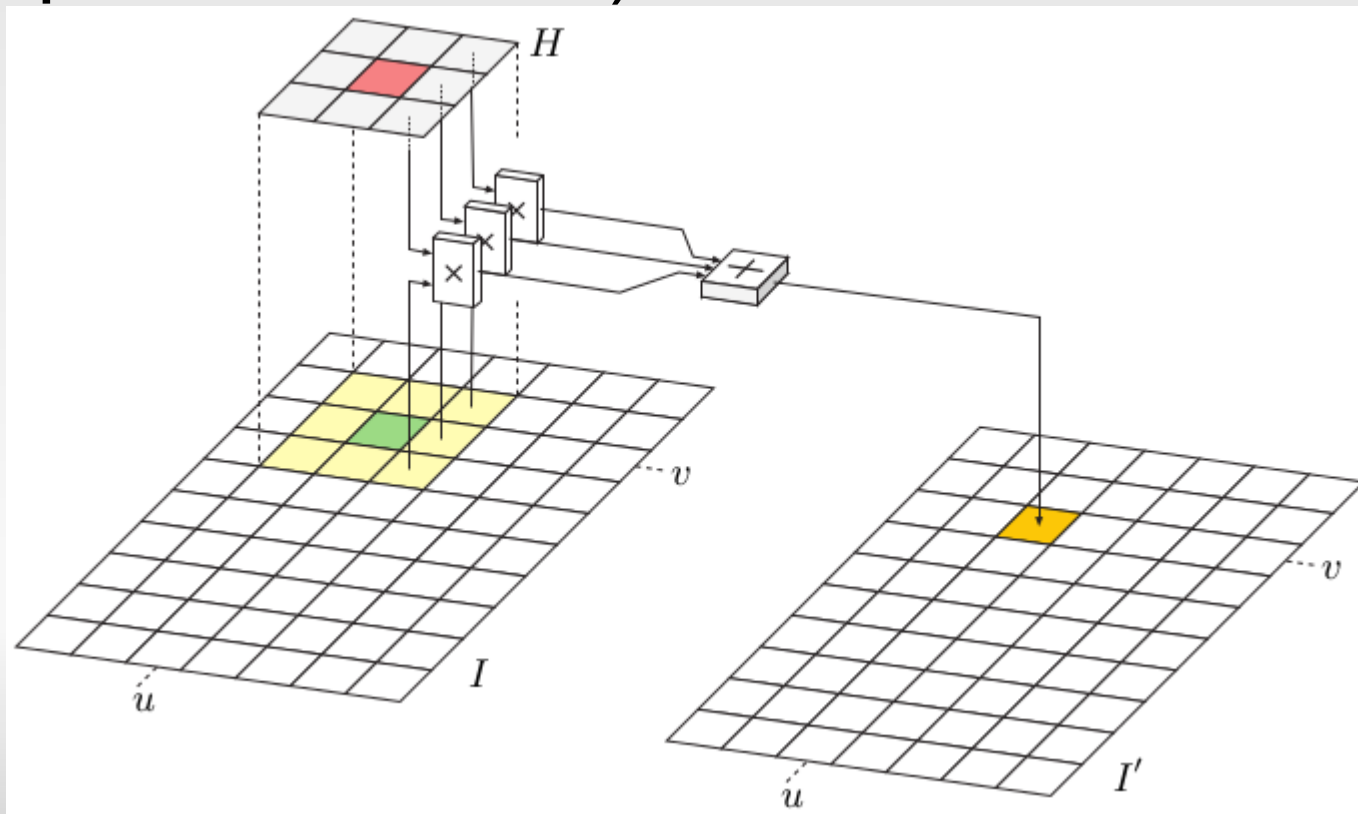


Usando uma máscara

(img de dimensoes pequenas usualmente 3x3
ou 5x5)

Outros usos

- Além do processamento pixel a pixel, podemos realizar operações lógicas e aritméticas em processamentos orientados a vizinhança como por exemplo usando janelas deslizantes (será visto posteriormente)



Bibliografia

- PEDRINI, H. e SCHWARTZ, W. R., "Análise de Imagens Digitais", São Paulo, Thomson, 2008, 508p e slides.
- FALCÃO, A.
(<http://www.ic.unicamp.br/~afalcao/mo443/>)
- Digital Image Processing, An Algorithmic Introduction using Java. De Wilhelm Burger e Mark James Burge

Várias imagens foram extraídas do material mencionado acima com fines didáticos.