Operações e Filtragem Espacial de Imagens

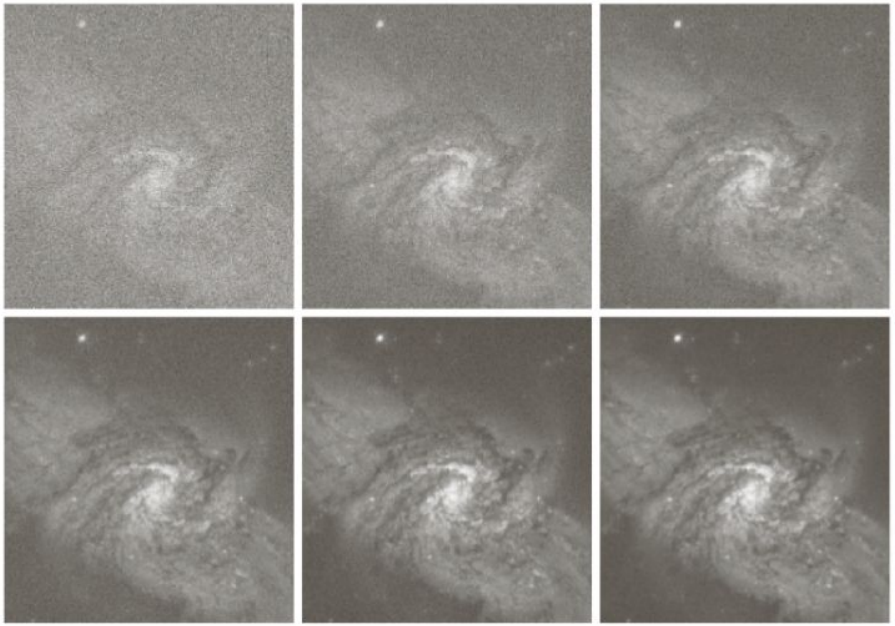
Prof. Vinícius de Oliveira

Brasília, 14 de Outubro de 2019

Operações em imagens digitais

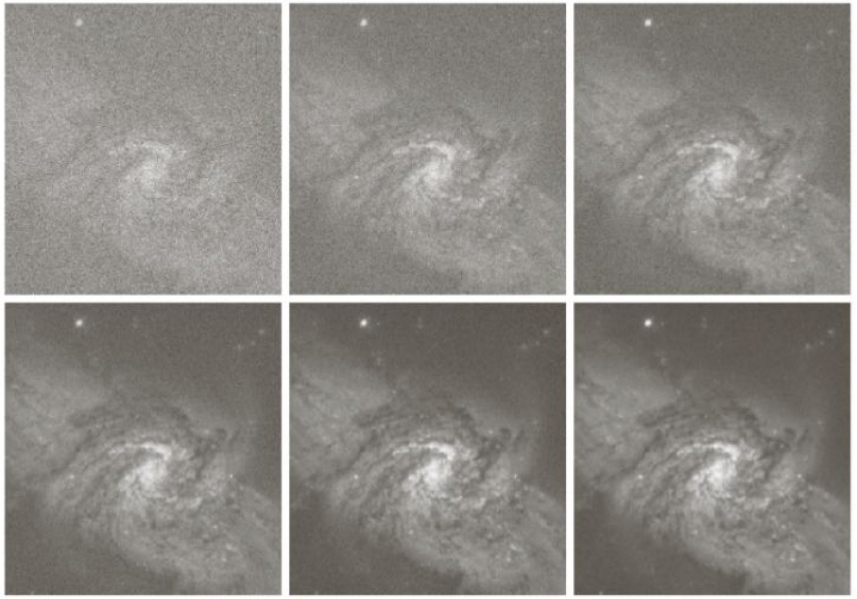
- Por elementos ou matrizes;
- Lineares ou não Lineares;
- Aritméticas (soma, subtração, divisão, muliplicação);
- Lógicas;
- Entre outras.

Soma



a b c
d e f

FIGURE 2.26 (a) Image of Galaxy Pair NGC 3314 corrupted by additive Gaussian noise. (b)–(f) Results of averaging 5, 10, 20, 50, and 100 noisy images, respectively. (Original image courtesy of NASA.)

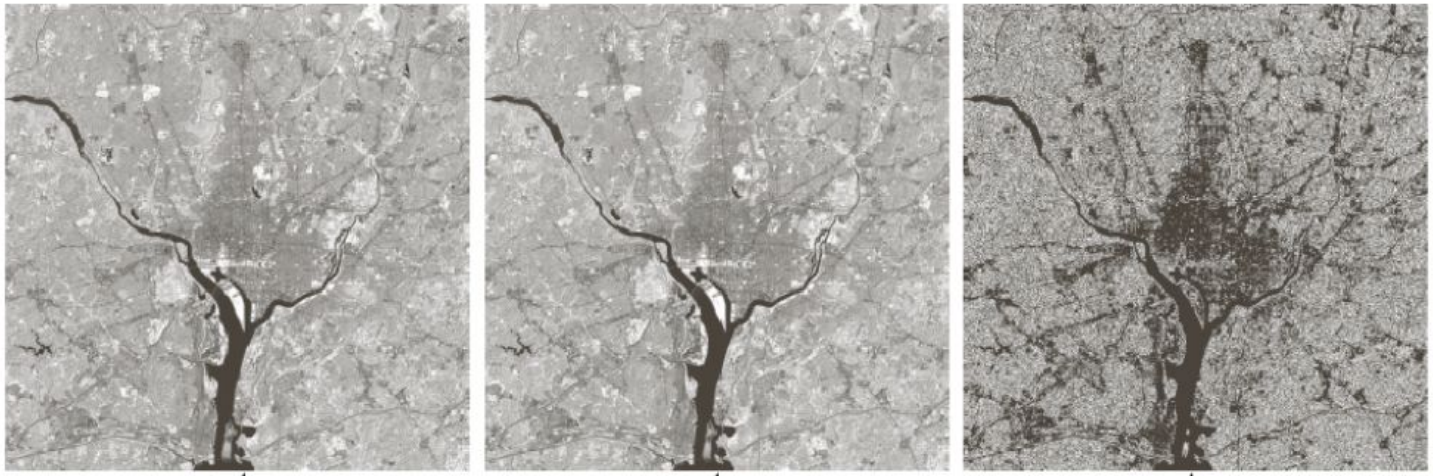


a b c
d e f

FIGURE 2.26 (a) Image of Galaxy Pair NGC 3314 corrupted by additive Gaussian noise. (b)–(f) Results of averaging 5, 10, 20, 50, and 100 noisy images, respectively. (Original image courtesy of NASA.)

•Este processo funciona se o ruído presente possui média zero e é descorrelacionado;

Diferença



original

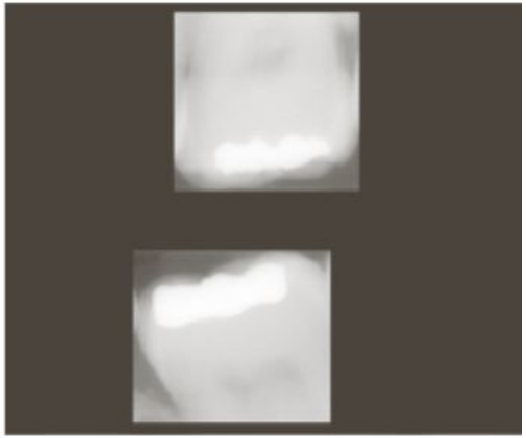
Zerando o bit menos significativo

Diferença normalizada

Multiplicação - ROI

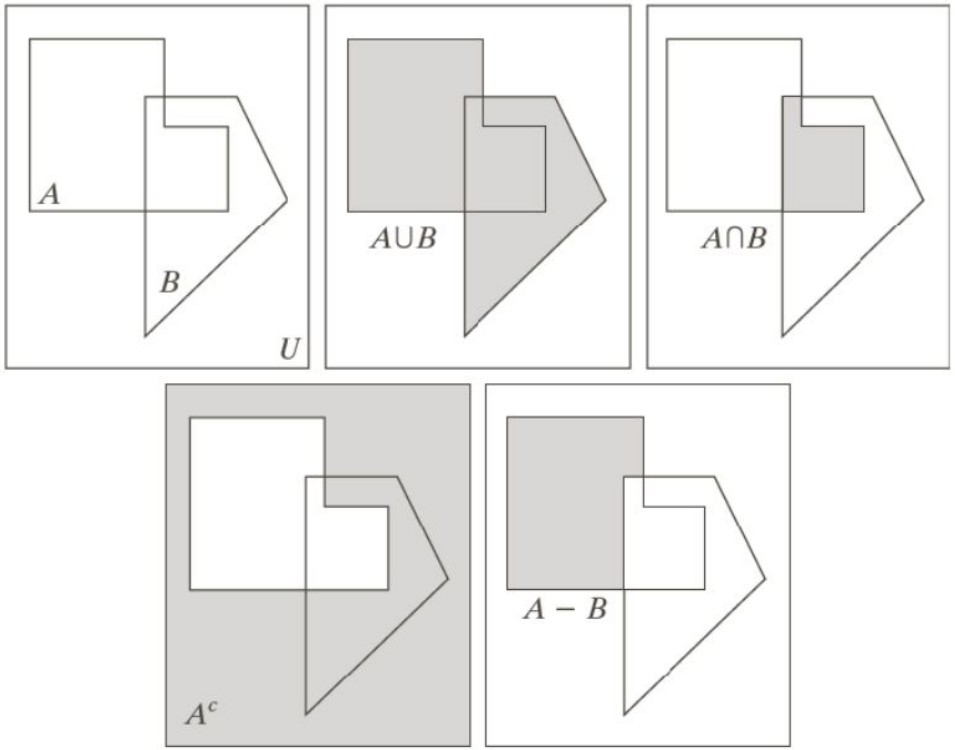
original

Máscaras englobando as Regiões de interesse



Multiplicação do original pela imagem com másca

Operações Lógicas



a b c
d e

FIGURE 2.31
(a) Two sets of coordinates, A and B, in 2-D space. (b) The union of A and B. (c) The intersection of A and B. (d) The complement of A. (e) The difference between A and B. In (b)–(e) the shaded areas represent the member of the set operation indicated.

Operações Lógicas

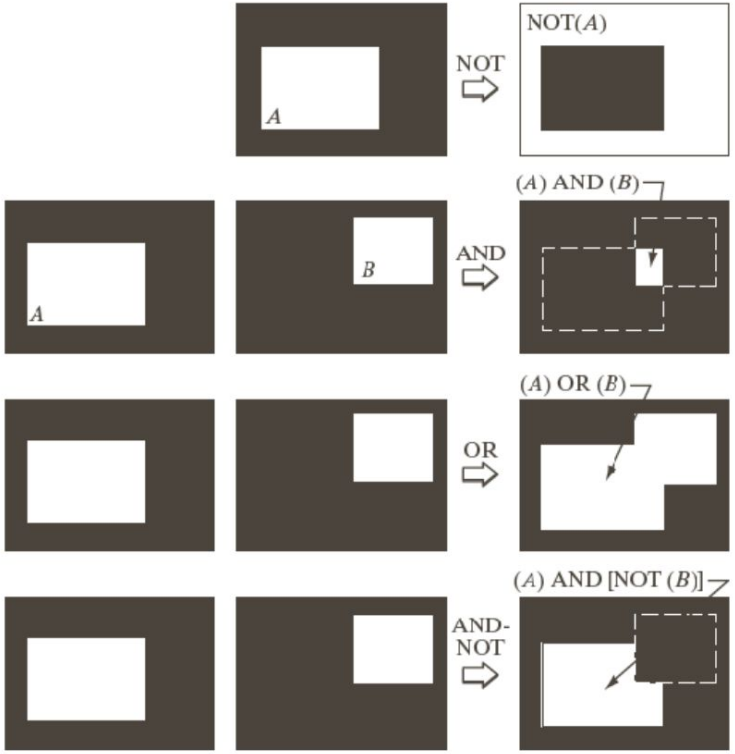


FIGURE 2.33
Illustration of logical operations involving foreground (white) pixels. Black represents binary 0s and white binary 1s. The dashed lines are shown for reference only. They are not part of the result.

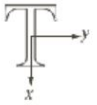
Original
f(x,y)

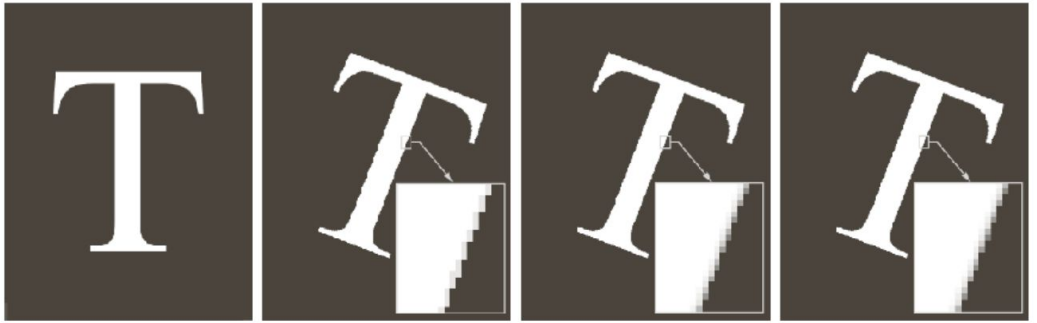
Imagem negativa
255-f(x,y)

União do original com uma imagem constante = {max(a,b)}

Operações Geométricas

TABLE 2.2
Affine transformations based on Eq. (2.6.–23).

Transformation Name	Affine Matrix, T	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = w$	
Scaling	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = c_x v$ $y = c_y w$	
Rotation	$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v\cos\theta - w\sin\theta$ $y = v\cos\theta + w\sin\theta$	
Translation	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$	$x = v + t_x$ $y = w + t_y$	
Shear (vertical)	$\begin{bmatrix} 1 & 0 & 0 \\ s_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v + s_v w$ $y = w$	
Shear (horizontal)	$\begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = s_h v + w$	

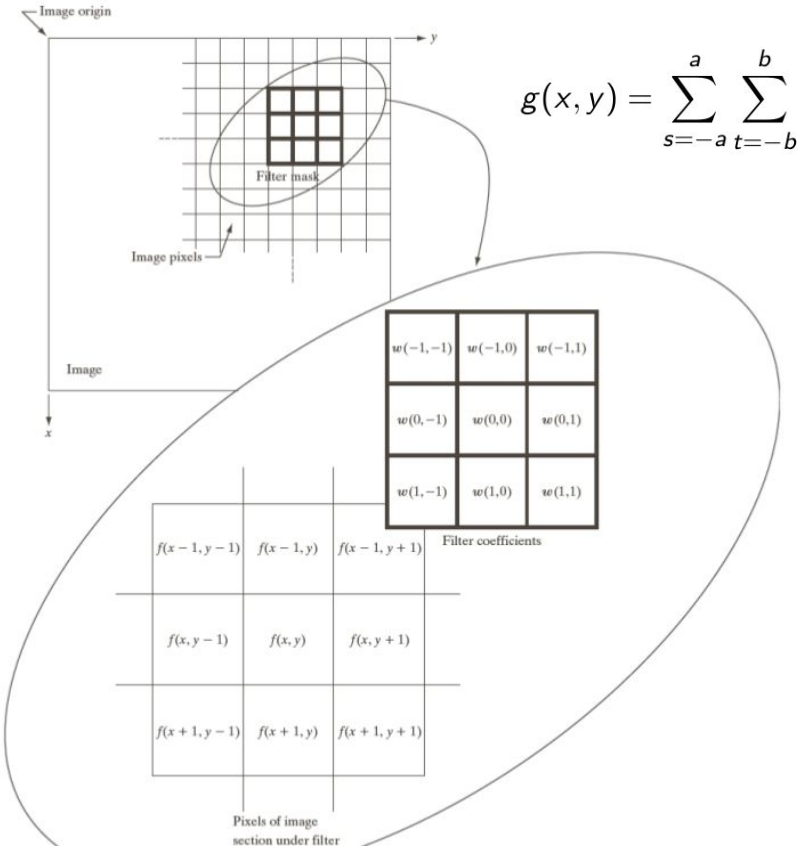


a b c d

FIGURE 2.36 (a) A 300 dpi image of the letter T. (b) Image rotated 21° clockwise using nearest neighbor interpolation to assign intensity values to the spatially transformed pixels. (c) Image rotated 21° using bilinear interpolation. (d) Image rotated 21° using bicubic interpolation. The enlarged sections show edge detail for the three interpolation approaches.

Filtragem Espacial

Filtragem Espacial



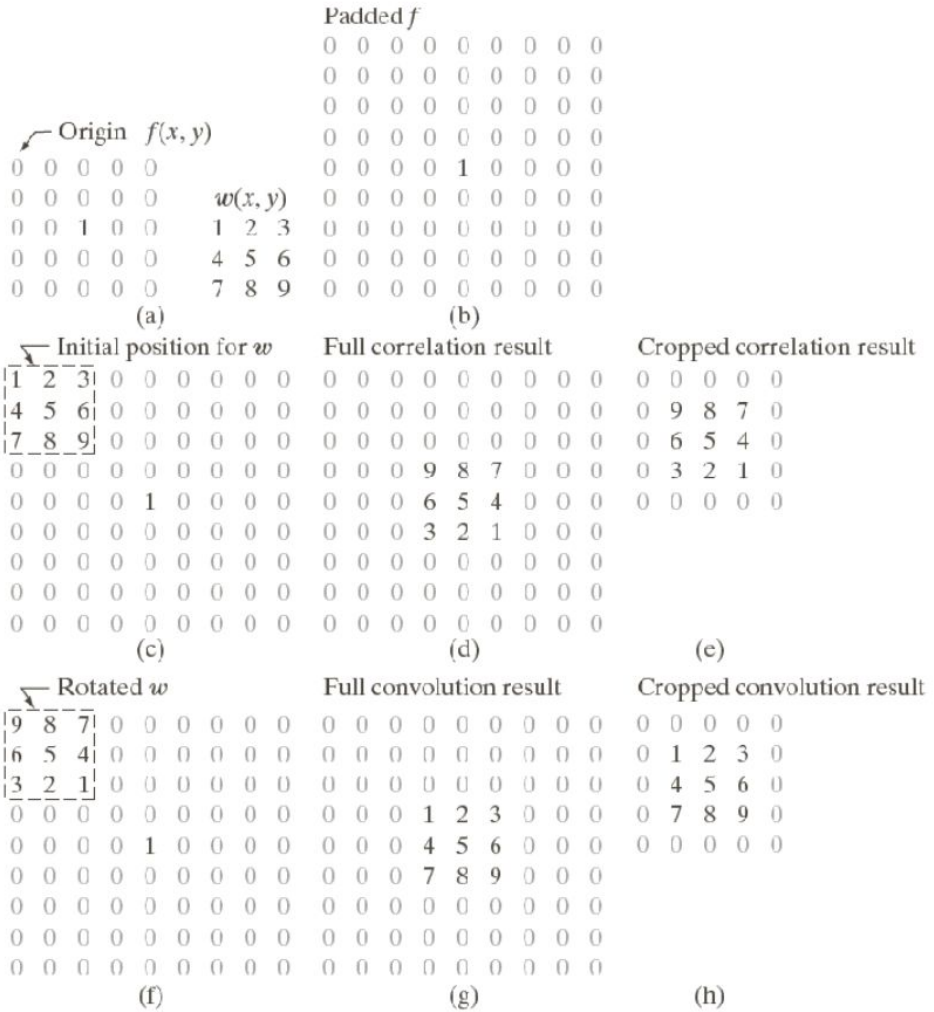
$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

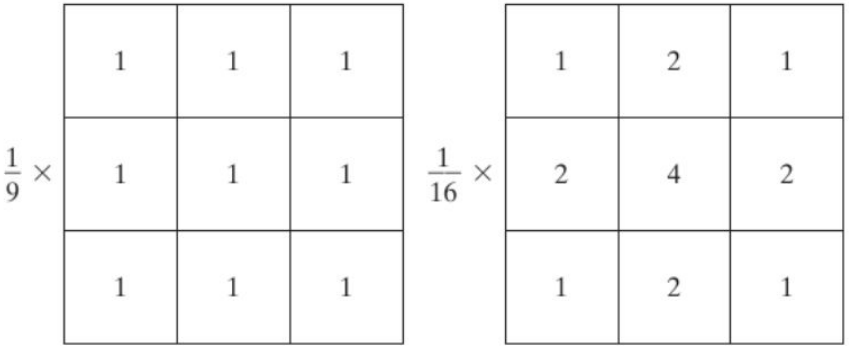
$$R = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \ldots + w(0,0)f(x,y) + \ldots + w(1,0)f(x+1,y0) + w(1,1)f(x+1,y+1)$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$$R = w_1 z_1 + w_2 z_2 + \ldots w_m n z_m n = \sum_{i=1}^{mn} w_i z_i$$

Filtragem Espacial:





$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

$\frac{1}{16} \times$

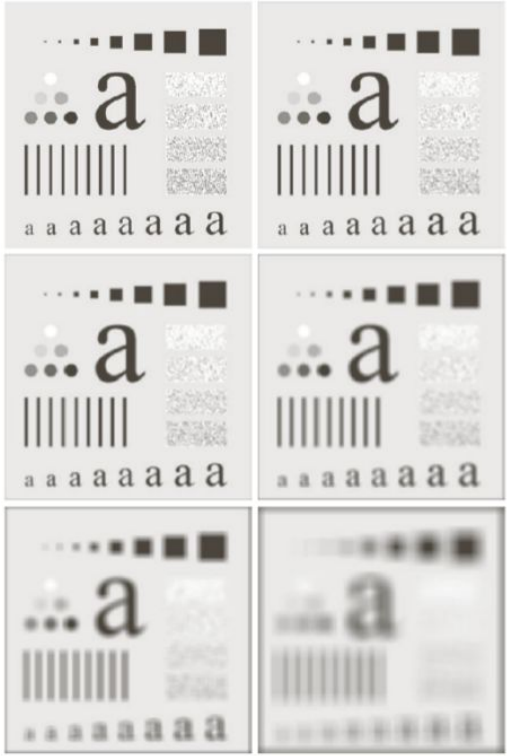
1	2	1
2	4	2
1	2	1

Filtragem Espacial:

$$R = \frac{1}{9} \sum_{i=1}^{mn} z_i$$

$$R = \frac{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)}{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t)}$$

Eliminando Detalhes



Filtros: $m = 3, 5, 9, 15,$ e 35.

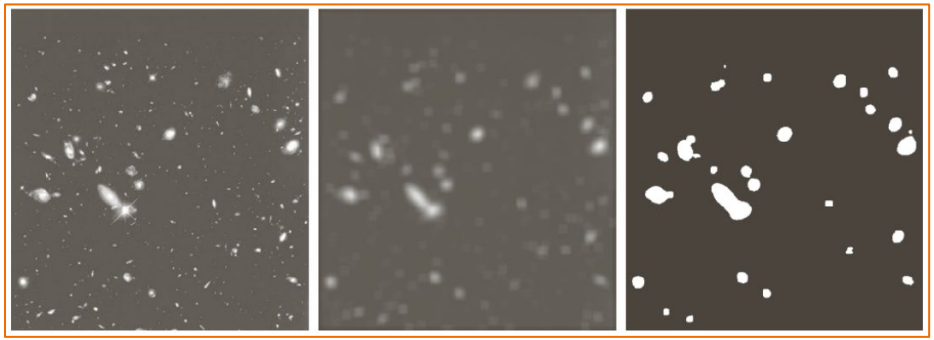
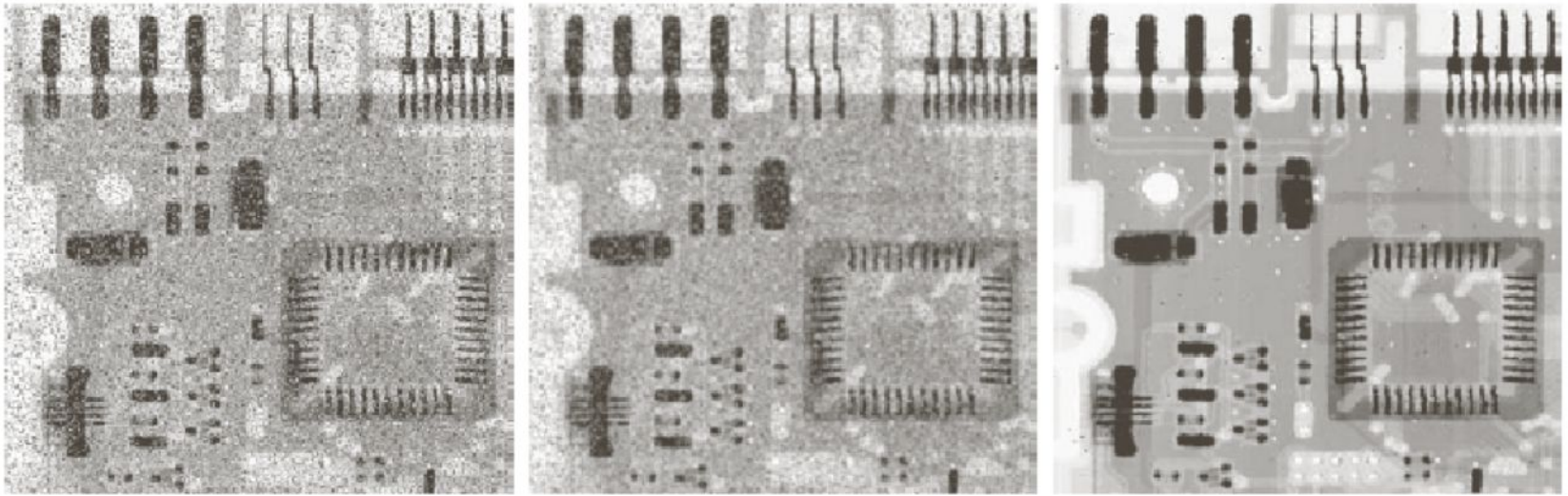


Imagem de 528 x 48. Filtragem com máscara de 15x15, seguida de thresholding.

Filtro da Mediana

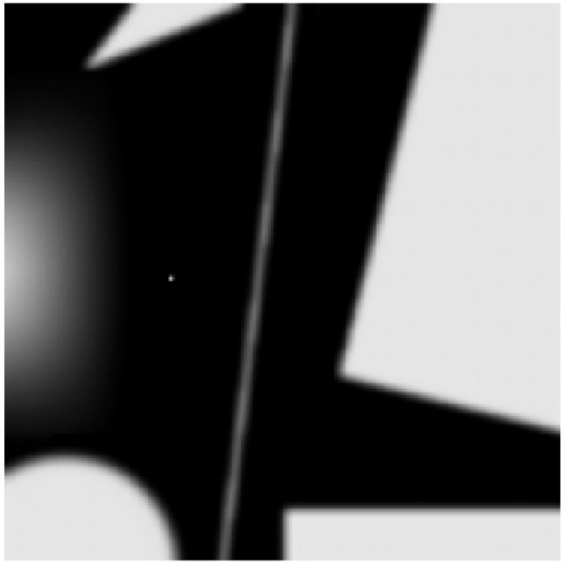


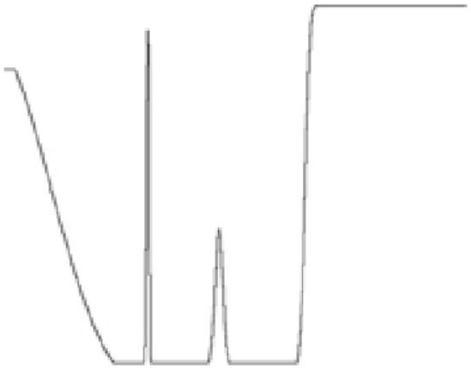
a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3×3 averaging mask. (c) Noise reduction with a 3×3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Filtros de Aguçamento

- 1a Derivada (discreta): $\frac{\partial f}{\partial x} = f(x+1) - f(x)$
- 2a Derivada (discreta): $\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$





Laplaciano

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

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$$

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

$$\frac{\partial^2 f}{\partial x^2} = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$

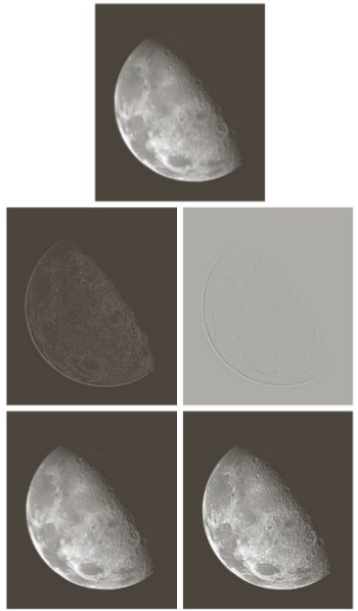
Laplaciano - Máscaras Espaciais

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

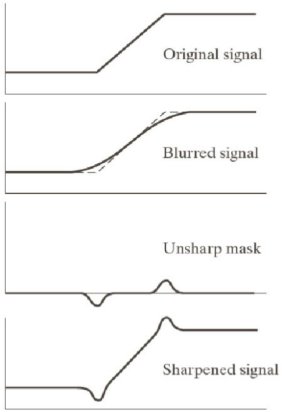
−1	−1	−1
−1	8	−1
−1	−1	−1

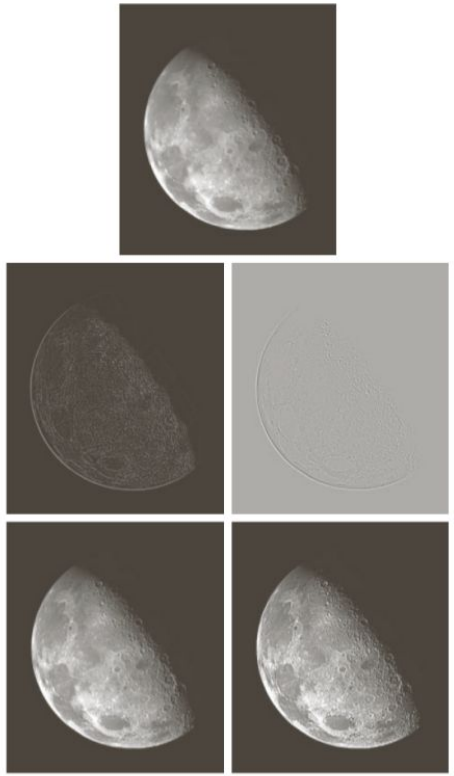


(a) imagem borrada, (b) Laplaciano sem escala, (b) Laplaciano com escalonamento, (c) Laplaciano.

Unsharp Masking

1. $\bar{f}(x,y) = \text{conv}(h_{LP}(x,y), f(x,y)) = h_{LP}(x,y) * f(x,y)$
2. $g_{mask} = f(x,y) - \bar{f}(x,y)$
3. $g(x,y) = f(x,y) + k \cdot g_{mask}(x,y)$





Roberts e Sobel

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0		0	−1	
0	1		1	0	
−1	−2	−1	−1	0	1
0	0	0	−2	0	2
1	2	1	−1	0	1

a
b c
d e

FIGURE 3.41
A 3 × 3 region of an image (the zs are intensity values).
(b)–(c) Roberts cross gradient operators.
(d)–(e) Sobel operators. All the mask coefficients sum to zero, as expected of a derivative operator.

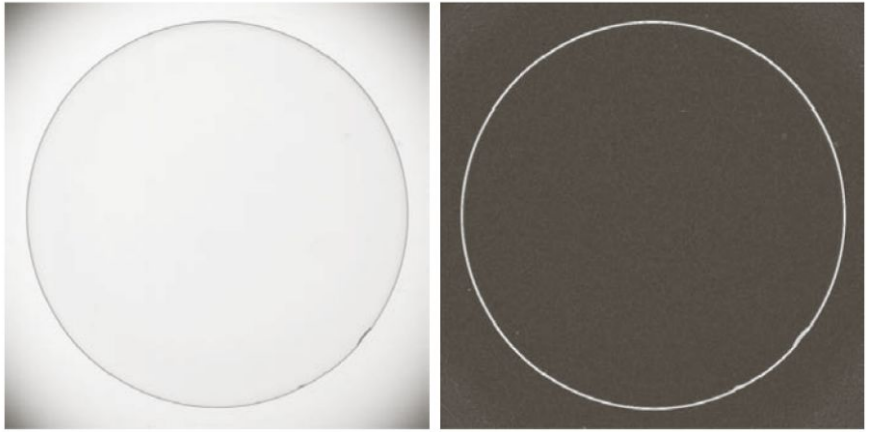
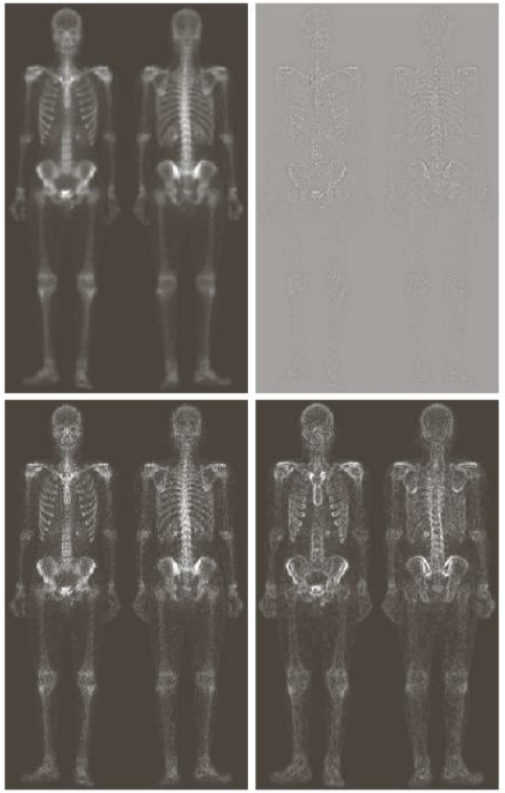


Imagem original e filtro de Sobel.

Roberts e Sobel

original



Laplaciano do original

Original + Laplaciano

Sobel do original

a b
c d

FIGURE 3.43
(a) Image of whole body bone scan.
(b) Laplacian of (a). (c) Sharpened image obtained by adding (a) and (b). (d) Sobel gradient of (a).

Fim!

Referências

[1] GONZALEZ, Rafael C.; WOODS, Richard E. Image processing. Digital image processing, v. 2, p. 1, 2007.

[2] Al Bovik, Handbook of Image and Video Processing, Academic Press.