An introduction to image processing

Universidad Nacional de Colombia
Sede Bogotá
2020-3
Anthony Huertas

Based on Ebroul Izquierdo:
Introduction to Computer Science

Road Map

- Digital image representation
- Sampling
- Quantization
- Algebraic Transformations
- Sub-sampling and pixel interpolation
- Local filtering (local operators and convolution)
- Gaussian and Laplacian filters

Digital image representation

Intensity function:

Digital image: $$I : D \rightarrow \Re^{+}$$

$f(x, y)$: images (grayscale, color, multi-spectral)

Digital image representation

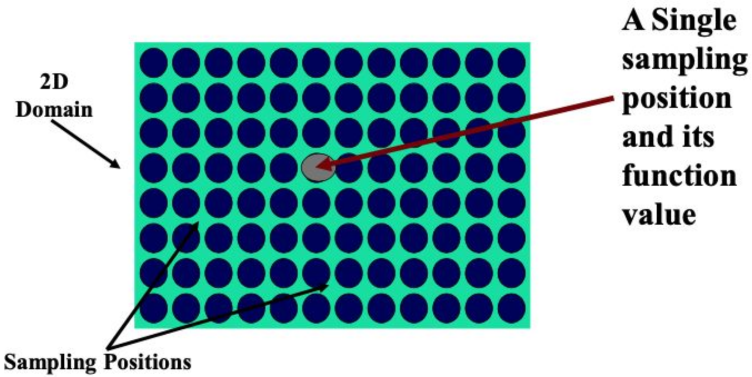


Image Formats

- GIF
- JPEG
- TIFF
- PNG
- PGM

Portable Grey Map

A simple format, that consists of two parts:

- Simple Header
- Pixel data

PGM Header

- Three parts normally separated by carriage returns and/or linefeeds
- The first "line" is a magic PGM identifier, it can be P2 or P5
- The next line consists of the width and height of the image as ascii numbers
- The last part of the header gives the maximum value of the intensity for the pixels, this allows the format to describe more than single byte (0..255) colour values

PGM Pixel data

- One value per pixel
- P2 corresponds to the ascii form of the data
- P5 corresponds to the binary form of the data

PGM Example

P2
24 7
9
000000000000000000000000
033330077770011110051550
030000070000010000050010
033300077700011100011110
030000070000010000010000
030000077770011110010000
000000000000000000000000

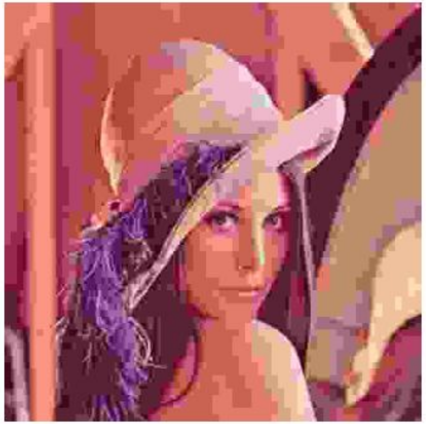
Sampling

Capture the intensity of the pixels in each position of the domain.

The resolution of the image heavily depends on the sampling process

Sampling

Full Resolution



1/8 Resolution

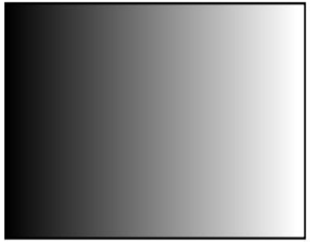
Quantization

Fit the intensity values on a integer scale(gray levels)

Then, appears the intensity resolution, this affects how we see the continuity of the image

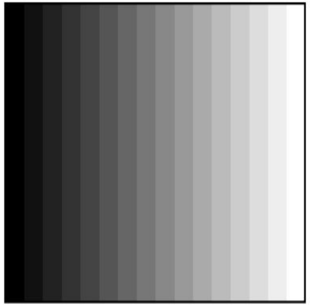
Quantization

Intensity resolution depends on the number of bits available



This figure shows a digital image quantized with 8 bits (256 gray levels). The image appears continuous

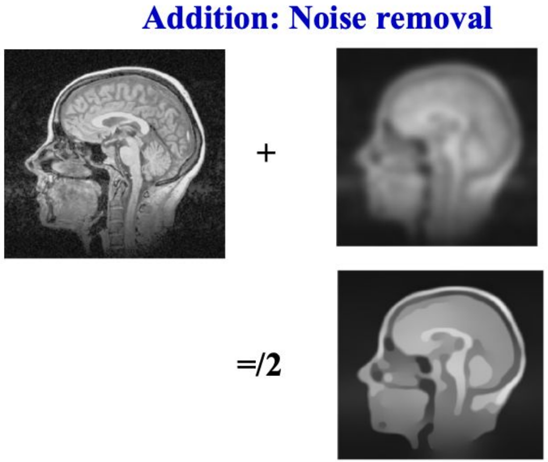
The same image quantized with only 4 bits (16 gray levels). Now the image brightness appears discontinuous



Algebraic transformations

- Addition, subtraction of two or more images
- Other linear
- Non linear

Algebraic transformations



Algebraic transformations

Subtraction: Background Segmentation

Algebraic transformations

Chromakey

Linear Operators

- Let H be an operator whose input and output are images.
- H is a linear operator, if for any two images f and g and two scalars a and b then
  $H(af+bg)=aH(f)+bH(g)$

Linear and Nonlinear Operators

Linear operators (average filter)

$$h = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Nonlinear operators (median filter)

- Replace the current point in the image by the median of the gray values in its neighborhood.

Edge Detection Using Subtraction

If an image is displaced (Translated) relative to another image, then the difference between them approximates the first derivative.

Edge Detection Using Subtraction

Original

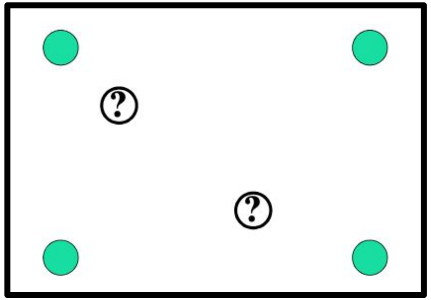
Gradient Image

Up-sampling and pixel interpolation

- To increase resolution
- Requires generation of additional pixels from available ones

Up-sampling and pixel interpolation

- Assume four pixels on a regular grid are known
- Pixels “inside” can be interpolated



Linear Interpolation

If f is a linear operator, then

$$f(x_1 + x_2, y) = f(x_1, y) + f(x_2, y)$$

Bilinear Interpolation

Use the simplest second order form (Hyperbolic Paraboloid)

$$f(x,y) = ax + by + cxy + d$$

Bilinear Interpolation

Interpolate linear on each horizontal edge
Interpolate linear in the vertical direction using obtained results

$$
\begin{aligned}
f(x,y) = {} & [f(1,0) - f(0,0)]x + [f(0,1) - f(0,0)]y \\
& + [f(1,1) + f(0,0) - f(0,1) - f(1,0)]xy + f(0,0)
\end{aligned}
$$

Convolution

Continuous case

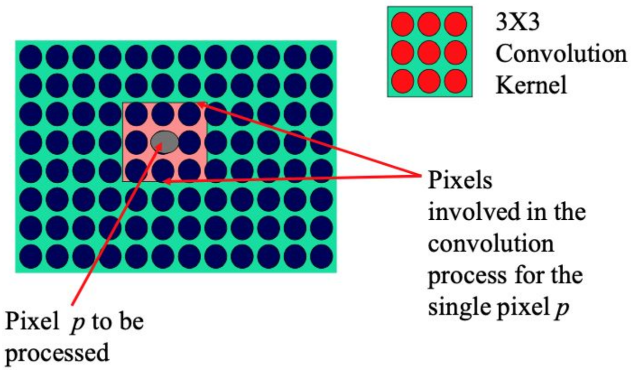
$$f(x,y) \otimes g(x,y) = \iint_{\infty} f(\alpha,\beta)\, g(x-\alpha, y-\beta)\, d\alpha\, d\beta$$

Discrete form

$$f_d(x,y) \otimes g_d(x,y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n)\, g(x-m, y-n)$$

Here f_d and g_d represent the discretized versions of f and g

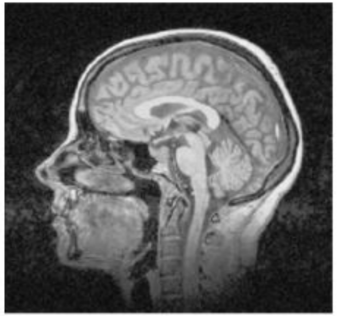
Convolution



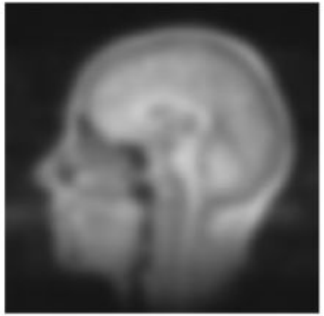
Convolution applications

- Deconvolution
  Remove effects of previously applied linear operations
- Noise Removal
  Filtering to separate noise from signal for estimating noise
  Feature detection
  Periodic noise removal
- Feature enhancement
  Using high pass filter, for instance

Convolution applications



Original



Signal Smoothing

Denoising

Computation of filtering

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

Original image $f(x,y)$

1	2	2	3	1	0	0
-1	7/9	11/9	1	4/9	-2/9	-1
0	2/3	1	8/9	4/9	1/9	-1
1	11/9	4/3	10/9	1	1	2
2	2	1	2	1	2	3

Filtered image $g(x,y)$

mask

1	1	1
1	1	1
1	1	1

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

Computation of filtering

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

Original image $f(x,y)$

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

Filtered image $g(x,y)$

mask

1	1	1
1	1	1
1	1	1

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

Mean filter

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Box filter

$$\frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Weighted average

Median filter

$$\underset{\text{Original}}{\begin{matrix}5 & 5 & 6\\ 3 & 4 & 5\\ 3 & 4 & 7\end{matrix}} \rightarrow \underset{\text{Sorting}}{(3,3,4,4,\boldsymbol{5},5,5,6,7)} \rightarrow \underset{\text{Filtered}}{\begin{matrix}5 & 5 & 6\\ 3 & 5 & 5\\ 3 & 4 & 7\end{matrix}}$$

Example

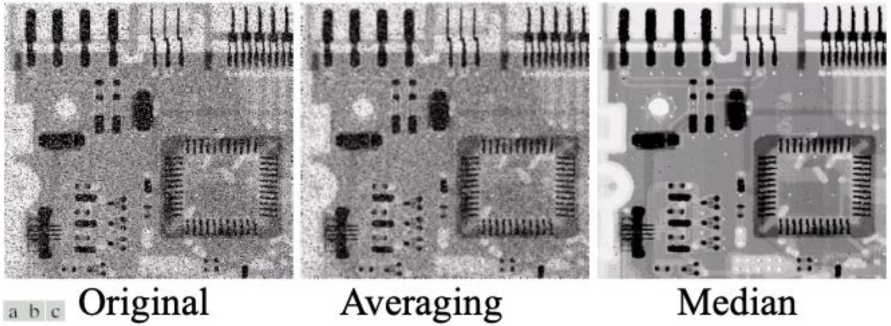


FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3×3 averaging mask. (c) Noise reduction with a 3×3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Order-Statistics Filters

Median filter

$$\hat{f}(x,y) = \underset{(x,y)\in S_{xy}}{\mathrm{median}}\{g(s,t)\}$$

Max and min filter

$$\hat{f}(x,y) = \max_{(x,y)\in S_{xy}}\{g(s,t)\}$$

Good for pepper

$$\hat{f}(x,y) = \min_{(x,y)\in S_{xy}}\{g(s,t)\}$$

Good for salt

Midpoint filter

$$\hat{f}(x,y) = \frac{1}{2}\left\{\max_{(x,y)\in S_{xy}}\{g(s,t)\} + \min_{(x,y)\in S_{xy}}\{g(s,t)\}\right\}$$

Alpha-trimmed mean filter
d/2 lowest and d/2 highest values are deleted

$$\hat{f}(x,y) = \frac{1}{mn-d}\sum_{(x,y)\in S_{xy}} g_r(s,t)$$

$d = 0$ → Arithmetic

$d = mn - 1$ → Median

Laplacian filter

The Laplacian is defined as

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

Derivation of Laplacian mask

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

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

$$\begin{aligned}\nabla^2 f &= \{f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1)\} - 4f(x,y) \\ &= 4 \times \{\bar{f}(x,y) - f(x,y)\}\end{aligned}$$

$$\bar{f}(x,y) = \frac{1}{4}\{f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1)\}$$

Laplacian mask

0	1	0	1	1	1
1	−4	1	1	−8	1
0	1	0	1	1	1

0	−1	0	−1	−1	−1
−1	4	−1	−1	8	−1
0	−1	0	−1	−1	−1

4-neighborhood 8-neighborhood

Enhancement Using Laplacian

$$\nabla^2 f = 4 \times \{\bar{f}(x,y) - f(x,y)\}$$

$$g(x,y) = \begin{cases} f(x,y) - \nabla^2 f(x,y) & \text{if the center coefficient of the Laplacian mask is negative} \\ & f(x,y) > \bar{f}(x,y) \Rightarrow -\nabla^2 f > 0 \Rightarrow g(x,y) > f(x,y) \\ & f(x,y) < \bar{f}(x,y) \Rightarrow -\nabla^2 f < 0 \Rightarrow g(x,y) < f(x,y) \\ f(x,y) + \nabla^2 f(x,y) & \text{if the center coefficient of the Laplacian mask is positive} \end{cases}$$

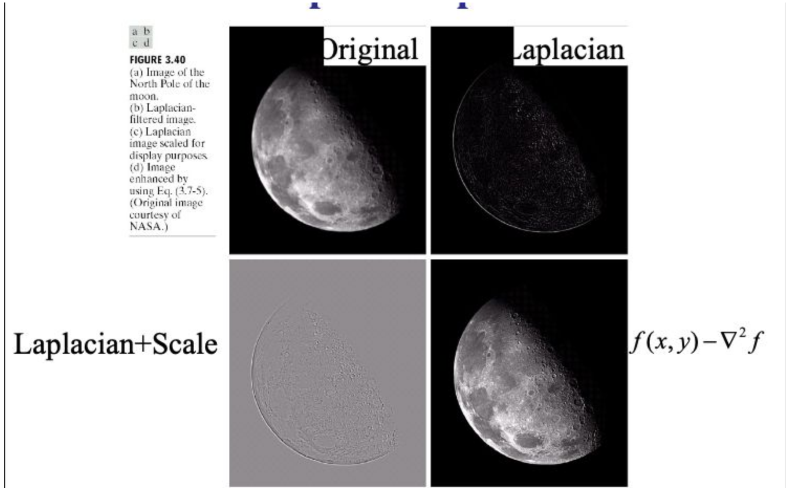
Simplification Using Laplacian

$$\nabla^2 f = \{f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1)\} - 4f(x,y)$$
$$= 4 \times \{\bar{f}(x,y) - f(x,y)\}$$

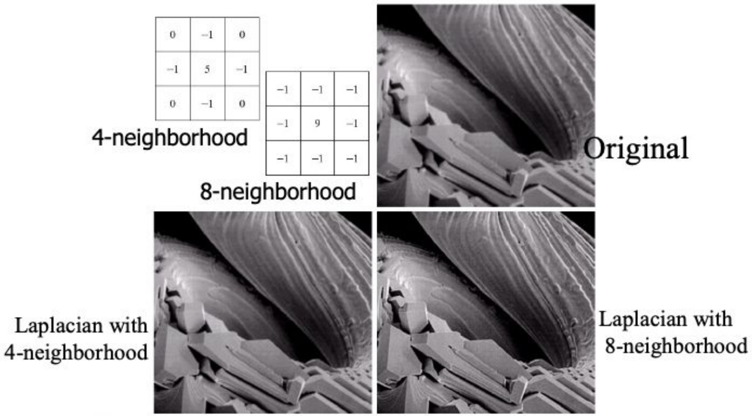
Simplification:

$$g(x,y) = f(x,y) - \nabla^2 f(x,y)$$
$$= f(x,y) - 4\{\bar{f}(x,y) - f(x,y)\}$$
$$= 5f(x,y) - 4 \times \bar{f}(x,y)$$

Laplacian example



Enhancement example



a b c
d e

FIGURE 3.41 (a) Composite Laplacian mask. (b) A second composite mask. (c) Scanning electron microscope image. (d) and (e) Results of filtering with the masks in (a) and (b), respectively. Note how much sharper (e) is than (d). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

An introduction to image processing

Universidad Nacional de Colombia
Sede Bogotá
2020-3
Anthony Huertas
Based on Ebroul Izquierdo:
Introduction to Computer Science