Lecture 2 – Pointwise processing

Ch. 2.6-2.6.4
3.1-3.3 in Gonzales & Woods

Filip Malmberg
filip.malmberg@it.uu.se
Centre for Image analysis
Uppsala University
We want to create an image which is "better" in some sense.

- For example
- Image restoration (reduce noise)
- Image enhancement (enhance edges, lines etc.)
- Make the image more suitable for visual interpretation
- **Image enhancement does NOT increase image information**
Image processing

- can be performed in the:

• **Spatial domain**
  - **Pointwise processing → Lecture 2**
    - Works per pixel
  - **Spatial filtering → Lecture 3**
    - Works on small neighborhood

• **Frequency domain → Lecture 4**
Problem solving using image analysis: fundamental steps

- **Image acquisition**
- **Preprocessing, enhancement**
- **Segmentation**
- **Feature extraction, description**
- **Classification, interpretation, recognition**
- **Result**

Knowledge about the application
Overview

i. repetition

ii. image arithmetics

iii. intensity transfer functions

iv. histograms and histogram equalization
Last lecture

- Digitization
  - Sampling in space (x,y)
  - Sampling in amplitude (intensity)
- Pixel/Voxel
- How often should you sample in space to see details of a certain size?
Bit depth

- Number of bits that are used to store the intensity information
  - Images are typically of 8- or 16-bit
    - 1 bit = $2^1 \rightarrow 2$ steps (0, 1)
    - 2 bit = $2^2 \rightarrow 4$ steps
    - 8 bit = $2^8 \rightarrow 256$ steps
    - 16 bit = $2^{16} \rightarrow 65536$ steps

2 gray levels, 1 bit/pixel

64 gray levels, 6 bit/pixel

256 gray levels, 8 bit/pixel
I. Image arithmetics in the spatial domain
Image arithmetics

- \( A(x,y) = B(x,y) \circ C(x,y) \) for all \( x,y \).
  - \( B, C \rightarrow \) images with the same (spatial) dimensions
    - \( \rightarrow \) images + constant value
  - can be
    - Standard arithmetic operation: +, -, *, /
    - Logical operator (binary images): AND, OR, XOR,...

- Any pitfalls?
Arithmetics with binary images

image1 - image2

image2 - image1

min value
max value
Arithmetics with binary images

\[ \text{image1} - \text{image2} \]

\[ \text{image2} - \text{image1} \]

\( \blacksquare \) min value

\( \square \) max value
Arithmetics with binary images

- min value
- max value
Arithmetics with greyscale images
Logical operations on binary images

<table>
<thead>
<tr>
<th>INPUT</th>
<th>OUTPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INPUT</th>
<th>OUTPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Applications

- **Noise reduction** using image mean or median

\[ I = \frac{1}{n} \sum_{k=1..n} I_k \]
Applications

- Change detection using subtraction

Has anything changed?
Applications

- Change detection using subtraction

2015  2016

direct difference
Applications

- Change detection using subtraction
Applications

- Change/motion detection using subtraction
Applications

- **Background removal**
  
  \textit{image - background image}

- Creating a background image

Max or median of the pixel intensities at all positions.
Applications

• **Background removal - result**
Applications

- Subtracting a background image/correcting for uneven illumination
II. Intensity transfer functions
Intensity transfer functions

\[ g(x, y) = T f(x, y) \]

i. **linear** (neutral, negative, contrast, brightness)

ii. **smooth** (gamma, log)

iii. **arbitrarily**

\[ n = T(o) \]
The negative transformation

\[ g(x, y) = \max - f(x, y) \]

- For eight bit image:
  \[ g(x, y) = 2^8 - 1 - f(x, y) \]

The rules of how to transfer values from the old image to the new one.
The negative transformation

\[ g(x, y) = \max - f(x, y) \]

- For eight bit image:
  \[ g(x, y) = 2^8 - 1 - f(x, y) \]

The rules of how to transfer values from the old image to the new one.

<table>
<thead>
<tr>
<th>old value</th>
<th>new value</th>
</tr>
</thead>
<tbody>
<tr>
<td>255</td>
<td>0</td>
</tr>
<tr>
<td>254</td>
<td>1</td>
</tr>
<tr>
<td>253</td>
<td>2</td>
</tr>
<tr>
<td>252</td>
<td>3</td>
</tr>
<tr>
<td>251</td>
<td>4</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>0</td>
<td>255</td>
</tr>
</tbody>
</table>
The negative transformation

\[ g(x, y) = \max - f(x, y) \]

- For eight bit image:
  \[ g(x, y) = 2^8 - 1 - f(x, y) \]

<table>
<thead>
<tr>
<th>Old Value</th>
<th>New Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>255</td>
<td>0</td>
</tr>
<tr>
<td>254</td>
<td>1</td>
</tr>
<tr>
<td>253</td>
<td>2</td>
</tr>
<tr>
<td>125</td>
<td>130</td>
</tr>
<tr>
<td>130</td>
<td>125</td>
</tr>
<tr>
<td>110</td>
<td>145</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>251</td>
<td>252</td>
</tr>
<tr>
<td>252</td>
<td>255</td>
</tr>
</tbody>
</table>

The rules of how to transfer values from the old image to the new one.
The negative transform

- Example

Original

Negative
The negative transform

original digital mammogram

image negative to (visually) enhance white or gray details embedded in dark regions
The negative transformation

- Careful with color images
Brightness

\[ g(x, y) = f(x, y) + C \]
Brightness

\[ g(x, y) = f(x, y) + C \]

\[ g(x, y) = f(x, y) - C \]
Brightness

\[ g(x, y) = f(x, y) + C \quad g(x, y) = f(x, y) - C \]
Contrast

\[ \theta = \pi/4 \]

\[ \theta > \pi/4 \]

\[ g(x, y) = f(x, y) \times C \]

\[ C > 1 \]
Contrast

\[ \theta = \pi/4 \]

\[ \theta > \pi/4 \]

\[ \theta < \pi/4 \]

\[ g(x, y) = f(x, y) \times C \]

\[ C > 1 \]

\[ g(x, y) = f(x, y) \times C \]

\[ C < 1 \]
Contrast

\[ \theta = \frac{\pi}{4} \]
\[ \theta > \frac{\pi}{4} \]
\[ \theta < \frac{\pi}{4} \]

\[ g(x, y) = f(x, y) \times C \]
\[ C > 1 \]

\[ g(x, y) = f(x, y) \times C \]
\[ C < 1 \]
Examples

- Decrease the brightness by 10
  \[ g(x, y) = f(x, y) - 10 \]

- Decrease the contrast by 2
  \[ g(x, y) = f(x, y) \times 0.5 \]
Examples

- Decrease the brightness by 10

\[ g(x, y) = f(x, y) - 10 \]

- Decrease the contrast by 2

\[ g(x, y) = f(x, y) \times 0.5 \]
Gamma transformation

\[ g(x, y) = C \times f(x, y)^\gamma \]

- Computer monitors have \( \gamma \approx 2.2 \)
- Eyes have \( \gamma \approx 0.45 \)
- Microscopes should have \( \gamma = 1 \)
$\gamma = 0.25$

$\gamma = 1$
Log transformations

- Log transformation to visualize patterns in the dark regions of an image

\[ g(x, y) = C \log(1 + f(x, y)) \]
Arbitrary transfer functions

- Only one output per input.
- Possibly non-continuous.
- Usually no inverse
III. Histograms and histogram equalization
Image histogram

- A gray scale histogram shows how many pixels there are at each intensity level.

  - width = 340 px
  - height = 370 px
  - bit-depth = 8 bits → 0..255
Exercise

Gray level histogram

- width = 4px
- height = 4px
- bit-depth = 3 bits
Exercise

- width = 4px
- height = 4px
- bit-depth = 3 bits
Image histogram

- Gray-level histogram shows intensity distribution
Beware

- Intensity histogram says nothing about the spatial distribution of the pixel intensities
Pair images and histograms!
Use of histogram

- Thresholding \(\rightarrow\) decide the best threshold value
- *works well with bi-modal* histograms
- *does not work with uni-modal* histograms
- Analyze the brightness and contrast of an image
- Histogram equalization
Analyze the brightness

- Increase brightness - shift histogram to the right
- Decrease brightness - shift histogram to the left

Greylevel transform:
up \rightarrow \text{increased brightness}
down \rightarrow \text{decreased brightness}
Analyze the contrast

- Decreased contrast - compressed histogram.
- When contrast is increased - the histogram is stretches.

Greylevel transform:
\(< 45^\circ \rightarrow \text{decreased contrast}\
\> 45^\circ \rightarrow \text{increased contrast}
Cumulative histogram

- Easily constructed from the histogram

\[ c_j = \sum_{i=0}^{j} h_i \]
Cumulative histogram

- Slope
  - Steep → intensely populated parts of the histogram
  - Gradual → in sparsely populated parts of the histogram
Cumulative histogram
Histogram equalization

- **Idea:** Create an image with evenly distributed greylevels, for visual contrast enhancement
- **Goal:** Find the transformation that produces the most even histogram $\rightarrow$ cumulative histogram curve
- **Equalization** flattens the histogram, linearizes cumulative histogram
- **Automatic contrast enhancement**
Histogram equalization

original image

result of histogram equalization

The contrast transform
Hist eq: small example

- Intensity: 0 1 2 3 4 5 6 7
- Number of pixels: 10 20 12 8 0 0 0 0

- \( p(0) = \frac{10}{50} = 0.2, \ cdf(0) = 0.2 \)
- \( p(1) = \frac{20}{50} = 0.4, \ cdf(1) = 0.6 \)
- \( p(2) = \frac{12}{50} = 0.24, \ cdf(2) = 0.84 \)
- \( p(3) = \frac{8}{50} = 0.16, \ cdf(3) = 1 \)
- \( p(r) = 0/50 = 0, \ r = 4, 5, 6, 7 \ cdf(r) = 1 \)
Hist eq: small example

- $T(0) = 7 \times (p(0)) = 7 \times 0.2 = 1.4 \approx 1$
- $T(1) = 7 \times (p(0) + p(1)) = 7 \times 0.6 = 3.6 \approx 4$
- $T(2) = 7 \times (p(0) + p(1) + p(2)) \approx 6$
- $T(3) = 7 \times (p(0) + p(1) + p(2) + p(3)) \approx 7$
- $T(r) = 7, \ r = 4, 5, 6, 7$

Intensity

\[
\begin{array}{cccccccc}
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\end{array}
\]

Number of pixels

\[
\begin{array}{cccccccc}
0 & 10 & 0 & 0 & 20 & 0 & 12 & 8 \\
\end{array}
\]
More examples of hist eq

Transformations for image 1-4. Note that the transform for figure 4 (dashed) is close to the neutral transform (thin line).
Local histogram equalization

**FIGURE 3.23** (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a $7 \times 7$ neighborhood about each pixel.
Histogram equalization

- Useful when much information is in a narrow part of the histogram

- Drawbacks:
  - Amplifies noise in large homogenous areas
  - Can produce unrealistic transformations
  - Information might be lost, no new information is gained
  - Not invertible, usually destructive
- Does not work well in all cases!
• Histogram equalization is not always “optimal” for visual quality
• Histogram eq: the result depends on the amount of different intensities
Histogram matching

• In histogram equalization a flat distribution is the goal

• In histogram matching the distribution of another image is the goal

For an image, I, find the transformation, T, that gives the histogram some ideal shape, s.
Image 1 histogram matched to image 2
Image 2 histogram matched to image 1
Histogram matching

- Compute the histograms for image $I_1$, $I_2$
- Calculate the cumulative distribution function $F_1()$, $F_2()$
- For each gray level $G_1 [0,255]$ find gray level $G_2$ for which $F_1(G_1)=F_2(G_2)$
- Histogram matching function $M(G_1) = G_2$

![Images with histograms and cumulative distribution functions]
Summary

• Many common tasks can be described by image arithmetics.

• Histogram equalizations can be useful for visualization.

• Watch out for information leaks!

A few things to think about....

• What is the relation between image arithmetics and linear transfer functions?

• What can you know about an image from the histogram?

• If you have an 8-bit image, A; how will the 8-bit image B=255*(A+1) look like (exactly!)?

• What conclusions can you draw from the histogram if the first/last column is really high?

• Can you get better resolution by combining multiple images of the same sample?

Suggested problems:
2.22, 2.18, 2.9, 3.1, 3.6

Next lecture:
Spatial filtering (Ch. 3.4-3.8)
Bit plane slicing

- Pixels $\rightarrow$ digital numbers composed of bits
- Computer $\rightarrow$ Binary number system
- Basic unit, bit

$194 = 11000010$

- 8- and 16- bits are common for file formats.

But how many bits are necessary?

Next slides:
- The eight bit planes for an image.
- The same image using 7, ..., 0 bit planes.