Image Visualization
Image Visualization
• Image Representation & Visualization
• Basic Imaging Algorithms
• Shape Representation and Analysis
• Image Representation & Visualization
• Basic Imaging Algorithms
• Shape Representation and Analysis
What is an image?

- An image is a well-behaved uniform dataset.
- An image is a two-dimensional array, or matrix of pixels, e.g., bitmaps, pixmaps, RGB images.
- A pixel is square-shaped.
- A pixel has a constant value over the entire pixel surface.
- The value is typically encoded in 8 bits integer.

\[ D_s = (\{p_i\}, \{C_i\}, \{f_i\}, \{\Phi_i\}) \]
Image Data Representation

- Pixel values typically represent gray levels, colours, heights, opacities etc
- Remember digitization implies that a digital image is an approximation of a real scene
Image Processing and Visualization

- Image processing follows the visualization pipeline, e.g., image contrast enhancement following the rendering operation.
- Image processing may also follow every step of the visualization pipeline.
• Image Representation & Visualization
• Basic Imaging Algorithms
• Shape Representation and Analysis
Basic Image Processing

- Image enhancement operation is to apply a **transfer function** on the pixel luminance values.

- Transfer function is usually based on image histogram analysis.

- High-slope function enhances image contrast.
- Low-slope function attenuates the contrast.
Basic Image Processing

- The basic image processing is the contrast enhancement through applying a transfer function

- Transfer function
  - The original image:
    \[ f(x) = x \]
  - Linear normalization
    \[ f(x) = \frac{x - l_{\text{min}}}{l_{\text{max}} - l_{\text{min}}} \]
  - Nonlinear transfer
Image Enhancement

Linear Transfer

Non-linear Transfer
Image Histograms

- The histogram of an image shows us the distribution of grey levels in the image.
- Massively useful in image processing, especially in segmentation.
Histogram Equalization

- All luminance values cover the same number of pixels.
- Histogram equalization method is to compute a transfer function such as the resulted image has a near-constant histogram.

\[ f(x) = (size-1) \sum_{i=0}^{x} h[i] \]
Histogram Equalization

Original Image

After equalization
Noise and Images

- Noise can be described as rapid variation of high amplitude
- Or regions where high-order derivatives of $f$ have large values
- Noise is usually the high frequency components in the Fourier series expansion of the input signal
Noise Model

• We can consider a noisy image to be modeled as follows:

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

• where \( f(x, y) \) is the original image pixel, \( \eta(x, y) \) is the noise term and \( g(x, y) \) is the resulting noisy pixel

• If we can estimate the model the noise in an image is based on this will help us to figure out how to restore the image
There are many different models for the image noise term $\eta(x, y)$:

- Gaussian
  - Most common model
- Rayleigh
- Erlang
- Exponential
- Uniform
- Impulse
  - *Salt and pepper* noise
Filtering to Remove Noise

We can use spatial filters of different kinds to remove different kinds of noise. The *arithmetic mean* filter is a very simple one and is calculated as follows:

\[
\hat{f}(x, y) = \frac{1}{mn} \sum_{(s, t) \in S_{xy}} g(s, t)
\]

This is implemented as the simple smoothing filter:

\[
\begin{array}{ccc}
\frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\
\frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\
\frac{1}{9} & \frac{1}{9} & \frac{1}{9}
\end{array}
\]

Blurs the image to remove noise.
Smoothing

Noise image

After filtering
Fourier Series

For any continuous function $f(x)$ with period $T$ (or $x=[0,T]$), the Fourier series expansion are:

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \sin(\omega_n x) + \sum_{n=1}^{\infty} b_n \cos(\omega_n x)$$

$$\omega_n = n \frac{2\pi}{T}$$

$$a_n = \frac{2}{T} \int_{0}^{T} f(t) \sin(\omega_n t) dt$$

$$b_n = \frac{2}{T} \int_{0}^{T} f(t) \cos(\omega_n t) dt$$

The higher the order $n$ or the frequency, the smaller the amplitudes $a_n$ and $b_n$
Fourier Series
Fourier Transform

When $T \to \infty$, $w$ is continuous, amplitudes are also continuous.

\[
A(w) = \int_{0}^{\infty} f(t) \sin(wt) \, dt
\]

\[
B(w) = \int_{0}^{\infty} f(t) \cos(wt) \, dt
\]

\[
F(w) = (A(w), B(w))
\]
Fourier Transform
The Discrete Fourier Transform of \( f(x, y) \), for \( x = 0, 1, 2...M-1 \) and \( y = 0,1,2...N-1 \), denoted by \( F(u, v) \), is given by the equation:

\[
F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi \left( \frac{ux}{M} + \frac{vy}{N} \right)}
\]

for \( u = 0, 1, 2...M-1 \) and \( v = 0, 1, 2...N-1 \).
Discrete Fourier Transform (DFT)

The DFT of a two dimensional image can be visualized by showing the spectrum of the images component frequencies.

Scanning electron microscope image of an integrated circuit magnified ~2500 times

Fourier spectrum of the image
Convolution Theorem

\[(f(x) * g(x)) = \int_{-\infty}^{\infty} f(t)g(x - t)dt\]

\[(f(x) * g(x)) \leftrightarrow F \cdot G\]

\[(f * g)_i = \sum_{k=0}^{N} f_k g_{N+i-k}\]

Frequency filtering is equivalent to the convolution with a filter function \(g(x)\)
1. Compute the Fourier transform $F(w_x, w_y)$ of $f(x, y)$

2. Multiply $F$ by the transfer function $\Phi$ to obtain a new function $G$, e.g., high frequency components are removed or attenuated.

3. Compute the inverse Fourier transform $G^{-1}$ to get the filtered version of $f$

$$f \rightarrow F$$
$$G = F \cdot \Phi$$
$$f = G^{-1}$$
Frequency Filtering

Frequency filter function $\Phi$ can be classified into three different types:

1. **Low-pass filter**: increasingly damp frequencies above some maximum $w_{\text{max}}$
2. **High-pass filter**: increasingly damp frequencies below some minimal $w_{\text{min}}$
3. **Band-pass filter**: damp frequencies with some band $[w_{\text{min}}, w_{\text{max}}]$

To remove noise, low-pass filter is used
Smoothing Frequency Domain Filters

- Smoothing is achieved in the frequency domain by dropping out the high frequency components.
- The basic model for filtering is:
  \[ G(u,v) = H(u,v)F(u,v) \]
- where \( F(u,v) \) is the Fourier transform of the image being filtered and \( H(u,v) \) is the filter transform function.
- Low pass filters – only pass the low frequencies, drop the high ones.
Gaussian smoothing

The most-used low-pass filter is the Gaussian function

\[ F(e^{-a x^2}) = \sqrt{\frac{\pi}{a}} e^{-\pi^2 \omega^2 / a} \]
The transfer function of a Gaussian lowpass filter is defined as

\[ H(u,v) = e^{-D^2(u,v)/2D_0^2} \]
Edge Detection

Original Image

Edge Detection
Edge Detection

- Edges are curves that separate image regions of different luminance
- Edges are locations that have high gradient

\[ |\nabla I(x,y)| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \]

\[ \frac{\partial I}{\partial x}(i,j) = I_{i+1,j} - I_{i,j} \]

\[ \frac{\partial I}{\partial y}(i,j) = I_{i,j+1} - I_{i,j} \]
Edge Detection

Edges detection using derivatives
Edge Detection Operators

Roberts Operator

\[ R(i, j) = \sqrt{ (l_{i+1,j+1} - l_{i,j})^2 + (l_{i+1,j} - l_{i,j+1})^2 } \]

Sobel Operator:
good on noise

\[ \frac{\partial l}{\partial x}(i, j) = l_{i+1,j-1} + 2l_{i+1,j} + l_{i+1,j+1} - l_{i-1,j-1} - 2l_{i-1,j} - l_{i-1,j+1} \]

\[ \frac{\partial l}{\partial y}(i, j) = l_{i+1,j+1} + 2l_{i,j+1} + l_{i-1,j+1} - l_{i+1,j-1} - 2l_{i,j-1} - l_{i-1,j-1} \]

These are the first-order derivative. Finding edge is to find the high value through thresholding segmentation.
Edge Detection Operators

Laplacian-based operator: good on producing thin edge

Second-order derivative. Finding edge is to find the zero-crossing or minimum.

\[ |\Delta l(x, y)| = \left| \frac{\partial^2 l}{\partial x^2} + \frac{\partial^2 l}{\partial y^2} \right| \]

\[ \Delta l(i, j) = 4l_{i,j} - l_{i+1,j} - l_{i-1,j} - l_{i,j+1} - l_{i,j-1} \]
Derivative based edge detectors are extremely sensitive to noise.
The Laplacian of Gaussian filter uses the Gaussian for noise removal and the Laplacian for edge detection.
• Image Representation & Visualization
• Basic Imaging Algorithms
• Shape Representation and Analysis
Shape Representation and Analysis

- Filtering high-volume, low level datasets into low volume dataset containing high amounts of information
- Shape is defined as a compact subset of a given image
- Shape is characterized by a boundary and an interior
- Shape properties include
  - geometry (form, aspect ratio, roundness, or squareness)
  - Topology (genus, number)
  - Texture (luminance, shading)
Segmentation

- Segment or classify the image pixels into those belonging to the shape of interest, called foreground pixels, and the remainder, also called background pixels.

- Segmentation results in a binary image.

- Segmentation is related to the operation of selection, i.e., thresholding.
Segmentation

Find soft tissue

Find hard tissue
Connected Components

Find non-local properties

Algorithm: start from a given foreground pixels, find all foreground pixels that are directly or indirectly neighbored.
Morphological image processing (or morphology) describes a range of image processing techniques that deal with the shape (or morphology) of features in an image.

Morphological operations are typically applied to remove imperfections introduced during segmentation, and so typically operate on bi-level images.
To close holes and remove islands in segmented images

a: original image
b: segmentation
c: close holes
d: remove island
Morphological Operations

- **Dilation**: translate a structuring element (e.g., disc, square) over each foreground pixel of the segmented image.
  - Dilation thickens thin foreground regions, and fill holes and close background gaps that have a size smaller than the structuring element R.

- **Erosion**: the opposite operation of dilation.
  - Erosion is to thin the foreground components, remove island smaller than the structuring element R.
Morphological Operations

Original image

Dilation by 3*3 square structuring element

Dilation by 5*5 square structuring element

Original image

Erosion by 3*3 square structuring element

Erosion by 5*5 square structuring element
Morphological Operations

- Compound Operations
  - More interesting morphological operations can be performed by performing combinations of erosions and dilations

- Morphological closing
  - dilation followed by an erosion

- Morphological opening
  - erosion followed by a dilation operation
Examples

Original Image

Image After Opening

Image After Closing
Distance Transform
The distance transform DT of a binary image I is a scalar field that contains, at every pixel of I, the minimal distance to the boundary \( \partial \Omega \) of the foreground of I.

\[
DT(p) = \min_{q \in \partial \Omega} |p - q|
\]
Distance Transform

- Distance transform can be used for morphological operation
- Consider a contour line $C(\delta)$ of DT

$$C(\delta) = \{ p \in \mathbb{R}^2 | DT(p) = \delta \}$$

- $\delta = 0$ ...
- $\delta > 0$ ...
- $\delta < 0$ ...
Distance Transform

- The contour lines of DT are also called level sets
Feature Transform

- Find the closest boundary points, so called feature points

Given a:
Feature point is b

Given p:
Feature points are q1 and q2
Skeletonization: the Goals

- Geometric analysis: aspect ratio, eccentricity, curvature and elongation
- Topological analysis: genus
- Retrieval: find the shape matching a source shape
- Classification: partition the shape into classes
- Matching: find the similarity between two shapes
Skeletons are the medial axes
Or skeleton $S(\Omega)$ was the set of points that are centers of maximally inscribed disks in $\Omega$
Or skeletons are the set of points situated at equal distance from at least two boundary feature points of the given shape

$$S(\Omega) = \{p \in \Omega \mid \exists q, r \in \partial \Omega, |p - q| = |p - r|\$$
Skeletonization
Skeleton Computation

Feature Transform Method:
Select those points whose feature transform contains more than two boundary points.

Works well on continuous data
Fails on discrete data
Skeleton Computation

Using distance field singularities:
Skeleton points are local maxima of distance transform
Summary

• Basic Imaging Algorithms
  – Image Enhancement
  – Histogram Equalization
  – Noise and Images
  – Spatial Filtering
  – Fourier Transform
  – Frequency Filtering
  – Edge Detection

• Shape Representation and Analysis
  – Segmentation
  – Connected Components
  – Morphological Operations
  – Distance Transform
  – Feature Transform
  – Skeletonization