Computer vision T-61.5070 (5 cr) P

Spring 2008

Lectures: Jorma Laaksonen
Exercises: Mats Sjöberg

Lecture notes

Lecture #1 22.1.2008

1. GENERAL INFORMATION .............................. 11
   1.1 Passing the course ................................. 11
   1.2 Enrollment ....................................... 11
   1.3 Notices ......................................... 11
   1.4 Lectures ........................................ 12
   1.5 Exercises ....................................... 12
   1.6 Exceptions in lecture and exercise times ......... 13
   1.7 Book ............................................ 14
   1.8 Additional material ................................ 15
   1.9 Exams ........................................... 15
   1.10 Obligatory course assignment .................... 16
   1.11 Feedback from the course ....................... 17

2. Introduction ........................................... 17
   2.1 What computer vision stands for? (1) .......... 17
   2.2 What for is computer vision needed? ............. 17
   2.3 Why is computer vision difficult? (1/2) ....... 18
   2.4 What are the essential parts of a CV system? .... 19
   2.5 Image representation and analysis (1/3) ......... 20

3. Digital image ........................................ 21
   3.1 Basic properties and definitions (2) ............ 21
   3.2 Mathematical tools and notations (2/2) ........ 22
   3.3 2-dimensional Fourier transform (2/2) ........ 22
   3.4 Convolution theorem (2,1/3,2/4) ............... 23
   3.5 Image as a stochastic process (2,1/3,2/3) .......... 25
   3.6 Image as a linear system (2,1/3,2/3) .......... 27
   3.7 Digitization of images (2/2) ..................... 29
   3.8 Metric properties of a digital image (2,3,1) ...... 29
   3.9 Noise in images (2,3,5,2,3,6) ................. 33

4. 3D vision ............................................. 34
   4.1 Difficulties of 3D vision (9/11) ................. 34
   4.2 Strategies of 3D vision (9/11) .................. 35
   4.3 Man’s theory (9,1/4,11,11) .................... 36
   4.4 Active versus passive computer vision (9,12/11,12) 37

Lecture #2 25.1.2008

4.5 3D projection geometry (9,2/1/11,2/1) ............ 38
4.6 Geometry of single perspective camera (9,2/1/11,3/1) 39
4.7 Two cameras (stereo) (9,2/5/11,5) ............... 43
4.8 Shape from stereo vision (9,2/5/11,5) ............ 45
4.9 Point correspondence in stereo vision (9,2/1/11,6,1) 45
4.10 Active acquisition of range images (9,2/2/11,6,2) ... 49
4.11 Radiometry in 3D vision (9,3/1,11,7,4,35) ........ 50
4.12 Shape from X (10,12) ................................ 51
4.13 Shape from motion (10,12/11,12) ................ 52
4.14 Shape from texture (10,12/11,12) ............... 54

Lecture #3 31.1.2008

4.15 Models of 3D world (10,2/1,12,2) ................ 55
4.16 Line labeling algorithm (10,2/2,12,2) ............ 56
4.17 More models of 3D world (10,2/4,5/12,2,4,5) .... 57
4.18 On recognition of 3D objects (10,3/12,3) ....... 58
4.19 Goad’s algorithm (10,3/2,12,3,2) ............... 59

4.20 Model-based 3D recognition from intensity images (10,3,3/12,3,3) 60
4.21 2D view-based representations for 3D (10,4/12,4) .... 62

5. Data structures ........................................ 64
   5.1 Introduction (33,4) ................................ 64
   5.2 Traditional data structures (33,4) ............... 65
   5.3 Hierarchical data structures (33,4) ............... 66
   5.4 Co-occurrence matrix (33,2/1,4) ............... 66
   5.5 Chain structures (32,2/4) ........................ 67
   5.6 Topological data structures (32,3/4,2) ........... 68
   5.7 Relational database structures (32,4/4,2) ....... 69
   5.8 Hierarchical data structures (33,4,3) ............. 70

Lecture #4 12.2.2008

6. Preprocessing .......................................... 72
   6.1 Brightness value changes in single pixels (4,1,5) 72
   6.2 Geometric coordinate transformations (2,4,5,2,1) 74
   6.3 Brightness interpolation (2,4,5,2,2) ............ 77
   6.4 Local pre-processing (3,4,5,1,1) ............... 78

6.5 Addtional constraints for local averaging (3,4,1/7) ... 79
6.6 Local neighborhood in edge detection (3,4,2/5,2) ... 81
6.7 Edge detection by derivative approximation (3,4,2) ... 82
6.8 Mann-Hildreth edge detector (3,4,3) ............... 83
6.9 Scale-space methods (3,4,3,5,4) .................. 85
6.10 Canny edge detector (3,4,5,5,3) .................. 86
6.11 Parametric edge models (3,4,6,5,3) ............... 88
6.12 Edges in multi-channel images (3,4,7,5,3) ....... 88
6.13 Other local neighborhood operations (3,4,8,3,9) .... 89
6.14 Corner and interest point detection (3,4,8,5,3,10) ... 89
6.15 Adaptive local pre-processing (3,4,9) ............ 93
6.16 Frequency domain image restoration (4,4,5,3,5,4) ... 94

Lecture #5 15.2.2008

7. Morphology ........................................... 95
   7.1 Basic notions and operations (11,1/3,1) ......... 95
   7.2 Dilution (2,2/1,3,31) ............................ 96
   7.3 Erosion (2,2/1,3,32) ............................. 96
   7.4 Some properties of dilution and erosion ........... 98

7.5 Opening and closing (3,3,4/3,3,4) ................ 101
7.6 Grayscale dilation and erosion (3,4,2/3,4) .......... 101
7.7 Skeletons and medial axis ................................ 102
7.8 Hit-or-miss, thinning, thickening (3,3,3,3,3,3,3,3,3,3,3,3,3) 105
7.9 Geometric elements .................................. 106
7.10 Quench function and ultimate erosion (3,5,3,4,3,5,4) 107
7.11 Ultimate erosion and distance functions (3,5,4,3,5,3,5,4,5,5) 108
7.12 Geodesic transformations (3,5,6,3,5,6) ........... 109
7.13 Morphological reconstruction (3,5,7,3,5,7) ........ 109
7.14 Granulometry (3,5,6,3,5,6,6) .................. 111
7.15 Morphological segmentation, watersheds (3,5,11,3,7,5,7) 112

Lecture #6 26.2.2008

8. Texture ................................................ 113
   8.1 Properties of natural textures .................... 113
   8.2 Statistical texture descriptors (14,1,1,3,3,4,1) .... 115
   8.3 Co-occurrence matrix (14,1,1,3,1,2) ............ 115
   8.4 Co-occurrence matrices - an example ............. 118
   8.5 Haralick features from co-occurrence matrix .... 119

8.6 Edge frequency (14,1,3,1,5,1,3) .................. 120
8.7 Run length statistics (14,1,4,1,5,4) ............... 122
8.8 Laws’ texture energy measures (14,1,5,1,5,1,5) .... 123
8.9 Other statistical measures (14,1,6,1,5,1,6,6) ....... 125
8.10 Syntactic texture descriptors (14,2,1,5,2,1) ....... 126
8.11 Graph grammars (14,2,2,1,5,2,2) ................. 127
8.12 Primitive grouping and hierarchical textures (14,1,5,2,2,1,3) 128
8.13 Hybrid texture description methods (14,3,1,5,3) .... 129
8.14 Application areas for texture analysis (14,4,1,5,4) .. 130

Lecture #7 4.3.2008

9. Segmentation ........................................... 131
   9.1 Thresholding methods in segmentation (9,2,6,2,6) ... 132
   9.2 Edge-based segmentation (9,2,6,2,6) ............. 136
   9.3 Border detection as graph searching (5,2,6,2,4) .... 142

Lecture #8 18.3.2008

9.4 Hough transforms (5,2,6,6,2,6) .................... 146
9.5 Region-based segmentation (5,3,6,3) ............... 151
1.6 Exceptions in lecture and exercise times
There will be some exceptions in lecture and exercise times:

- Friday 25.1, lecture instead of exercise
- Tuesday 5.2, no lecture
- Friday 15.2, lecture instead of exercise
- Tuesday 19.2, no lecture
- Friday 7.3, no exercise (exam period)
- Tuesday 11.3, no lecture (exam period)
- Friday 21.3, no exercise (Easter holiday)
- Tuesday 25.3, no lecture (Easter holiday)
- Tuesday 8.4, no lecture
- Friday 25.4, lecture instead of exercise
- Tuesday 29.4, exercise instead of lecture

1.7 Book
Milan Sonka, Vaclav Hlavac and Roger Boyle: Image Processing, Analysis and Machine Vision. Two editions are available and can be used, either:

A photocopy sample copy of the book (currently 2nd ed.) is available for short loans in a gray drawer in secretary Tarja Ahman's room B335.

In the 2nd ed. book, chapters 12 and 13 are skipped. In the 3rd ed. book, chapters 7 and 14 are skipped.

1.8 Additional material
Lecture notes and exercise papers with answers will be distributed by Edita. All the same material will also be available as PDF files for download.

1.9 Exams
There will be at least three exams: on Monday 12 May, one in the autumn and one in December 2008 or January 2009.

Use www.topi to register for the exam!

In the exam, there will be five tasks, each worth of 6 points, so the maximum will be 30 points. 11 points will suffice for passing the course. One of the tasks is a long textual question, one is based on some exercises and one consists of six short questions.

1.10 Obligatory course assignment
An obligatory course assignment has to be completed and accepted by the course assistant for passing course. The assignment will be graded as accepted/rejected. Further instructions concerning the practices will be given by the assistant. Monday 26 May 2008 is the deadline for submitting the assignment.

One cannot participate in the exams after May 2008 unless the obligatory course assignment has been passed.

Further instructions will be available at:

In all questions related to the exercise work, please contact the course assistant (mailtomats.sjoberg@hkk.fi).

1.11 Feedback from the course
After the lectures have ended, one can give feedback on the course at:
2. Introduction

2.1 What computer vision stands for? (1)
- qualitative / quantitative explanation of images
- structural / statistical recognition of objects

2.2 What for is computer vision needed?
- quality control in manufacturing
- medical diagnostics
- robot control
- surveillance cameras
- analysis of remote sensing (satellite) imagery
- intelligence/espionage applications
- image databases
- optical character recognition
- biometrics

2.3 Why is computer vision difficult? (1.2)
- loss of information in 3D → 2D projection
- interpretation of data by a model is problematic
- noise is inherently present in measurements
- there is way too much data
- measured brightness is weakly related to world’s properties
- most methods rely on local analysis of a global view

2.4 What are the essential parts of a CV system?
- low-level image processing
  - noise reduction
  - sharpening
  - edge detection
  - scale, rotation and location normalization
  - compression
  - feature extraction
- high-level "understanding"
  - model fitting
  - hypothesis testing
  - classification
  - feedback to preprocessing

2.5 Image representation and analysis (1.3)

Many different intermediate image content representations can be used.

2.6 Some useful vocabulary
- heuristic / heuristics = badly justified, but useful
- a priori information = something known, eg. by an expert
- syntactic = structure described with symbols and rules
- semantic = content or meaning described or explained
- top down = starting whom the whole, moving towards details
- bottom up = starting from details, moving towards the whole

3. Digital image

3.1 Basic properties and definitions (2.1)
- continuous / discrete / digital image
- intensity / depth image
- monochromatic / multispectral image
- photometry: intensity, brightness, gray levels
- colorimetry: analysis of color (wavelength) information
- resolution: spatial / spectral / radiometric / temporal

3.2 Mathematical tools and notations (2.1.2.3.1.2)
- 2-dimensional Dirac distribution $\delta(x, y)$:
  \[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(x, y) dx dy = 1, \quad \delta(x, y) = 0, \forall x, y \neq 0. \]
- 2-dimensional convolution $f \ast h$:
  \[ g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y-b) h(x-a, y) \, dx db \]
  \[ = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x-a, y) h(x, y-b) \, dx db \]
  \[ = (f \ast h)(x, y) = (h \ast f)(x, y) \]
- 2-dimensional sampling:
  \[ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(a, b) \delta(x-a, y-b) \, dx db = f(x, y) \]

3.3 2-dimensional Fourier transform (2.1.3.3.2.4)
- forward and backward (inverse) transforms:
  \[ \mathcal{F}(f(x, y)) = F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi i xu + yv} \, dx dy \]
  \[ \mathcal{F}^{-1}(F(u, v)) = f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{2\pi i xu + yv} \, du dv \]
- linearity of the Fourier transform:
  \[ \mathcal{F}(af(x, y) + bf(x, y)) = aF(u, v) + bF(u, v) \]
- translation of the origin:
  \[ \mathcal{F}(f(x-a, y-b)) = F(u, v)e^{-2\pi i (au+bv)} \]
  \[ \mathcal{F}(f(x, y)e^{2\pi i (ax+by)}) = F(u-a, v-b) \]
- symmetry, if $f(x, y) \in \mathbb{R}$:
  \[ F(-u, -v) = F^*(u, v) = \text{Real}(F(u, v)) - i \text{Imag}(F(u, v)) \]
3.4 Convolution theorem (2.13/3.2.4)
- duality of the convolution:
  \[ F(f * h)(x,y) = F(f)(u,v) H(u,v) \]
- is equivalent to
  \[ \text{if } g(x,y) = f(x,y) * h(x,y) \]
  \[ \text{then } G(u,v) = F(u,v) H(u,v) \]
- and
  \[ \text{if } g(x,y) = f(x,y) h(x,y) \]
  \[ \text{then } G(u,v) = F(u,v) * H(u,v) \]

3.5 Image as a stochastic process (2.14/2.3.3)
- entropy \( H = -\sum \pi(a) \log_2 \pi(a) \)
- average value \( \mu_x(y,\omega) = E(f(x,y)|\omega) = \int_{-\infty}^{\infty} z \pi(z|x,y,\omega) d\omega \)
- stationarity \( \Rightarrow \mu_x(x,y|\omega) = \mu_x(y|\omega) \)
- crosscorrelation \( R_{fg}(a,b|\omega) = \int_{-\infty}^{\infty} f(x,y+b,\omega) g(x,y+a,\omega) dxdy \)
  \[ F(R_{fg}(a,b)) = F(f)(u,v) G(u,v) \quad \text{(stat.)} \]
- autocorrelation \( R_{ff}(a,b|\omega) = \int_{-\infty}^{\infty} f(x,y+b,\omega) f(x,y+a,\omega) dxdy \)
  \[ F(f)(u,v) = |F(f)(u,v)|^2 \quad \text{(stat.)} \]
- \( f(x,y) \)’s power spectrum = spectral density \( S_f(u,v) = F(f)(u,v) \)
- ergodicity \( \Rightarrow \mu_x(x,y|\omega) = \mu_x(x,y) \)

3.6 Image as a linear system (2.15/3.2.1)
- linear operator \( L \):
  \[ L(a_j + b_j) = a_j L(j) + b_j L(f) \]
- image representation by a point spread function:
  \[ g(x,y) = L(f(x,y)) \]
  \[ = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(a,b) L(b-x-a, -b) dxdy \]
  \[ = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(a,b) h(x-a, y-b) dxdy \]
  \[ G(u,v) = F(f)(u,v) H(u,v) \]

3.8 Metric properties of a digital image (2.3.1)
- distance \( D(p,q) \) is a metric if:
  1) \( D(p,q) = 0 \iff p = q \) (identity)
  2) \( D(p,q) > 0 \iff p \neq q \) (non-negativity)
  3) \( D(p,q) = D(q,p) \) (symmetry)
  4) \( D(p,q) \leq D(p,r) + D(r,q) \) for triangular inequality
- distances \( D(p,q) \) between points \( p = (i,j) \) and \( q = (h,k) \):
  \[ D_{L_1}(i,j, h,k) = \sqrt{(i-h)^2 + (j-k)^2} \]
  \[ D_{L_2}(i,j, h,k) = |(i-h) + |j-k| \]
  \[ D_{L_1}(i,j, h,k) = \max(|i-h|, |j-k|) \]
  \[ D_{L_1}(i,j, h,k) = \max(|i-h|^2, |j-k|^2) \]
  \[ (\sqrt{2}-1) \min(|i-h|, |j-k|) \]

Distance function aka the chamfering algorithm
1) Pixel \( p \) in the object: \( F(p) := 0 \), otherwise \( F(p) := \infty \)
2) Scanning top to bottom, left to right, causal 4- or 8-neighborhood AL:
  \[ F(p) := \min_{q \in AL} (F(p), D(p,q) + F(q)) \]
3) Scanning bottom to top, right to left, causal 4- or 8-neighborhood BR:
  \[ F(p) := \min_{q \in BR} (F(p), D(p,q) + F(q)) \]

Adjacency of pixels
- 4- or 8-neighbors of pixels
- segmentation into regions on basis of adjacency
- path between pixels: simple/non-simple/closed
- contiguous pixels have a path between them
- being contiguous: reflective, symmetric and transitive
- simple contiguous = no holes, multiple contiguous = has holes
- connectivity paradoxes

Segmentation, borders/boundaries and edges
- segmentation: region / object / background / holes
- border/boundary is related to binary images
- edges are local properties of grayscale images: strength and direction
- crack edge: interpolational difference between 4/8-neighbor pixels

Topological properties
- rubber sheet and rubber band operations and invariants
- convex hull and its deficits: lakes and bays

Histograms (2.3.2)
3.9 Noise in images (2.3.5/2.3.6)
- white / Gaussian
- additive: \( f(x, y) = g(x, y) + \nu(x, y) \)
- multiplicative: \( f = g(1 + \nu) \approx g \nu \)
- quantization noise
- impulse noise = salt and pepper noise

4.4 Active versus passive computer vision (9.1.2/11.1.2)
- classical computer vision viewpoint: static, passive cameras
- robot systems can make use of active perception
- a system can actively acquire information it needs
- many ill-posed vision tasks become well-posed

Other dichotomies
- qualitative versus quantitative vision
- purposive vision versus precise description techniques

4. 3D vision

4.1 Difficulties of 3D vision (9.1/11)
- the camera projects 3D to 2D and unique inversion doesn’t exist
- complicated correspondence between measured intensities and the scene
- objects occlude themselves and each other
- noise and time complexity of algorithms

General questions (9.1/11.1)
- a priori knowledge about image characteristics being searched
- selection of the form of presentation, its influence on interpretations
- image interpretation: mapping from internal structures to the world

4.2 Strategies of 3D vision (9.1.1/11.1)
- bottom-up reconstruction
  - the most general solution for any problem
  - biological motivation
  - Marr, 1982
- top-down recognition
  - model-based vision
  - a special solution for a specific problem
  - engineering point of view
- 2D substitutes
  - aspect-based 2D approaches with qualitative features
  - alignment of 2D views

4.3 Marr’s theory (9.1.1/11.1.1)
The three levels of information processing:
- computational theory: logic or strategy for performing a task
- representation and algorithm: details on data and its processing
- implementation: programs and hardware

Stages of a bottom-up vision system according to Marrin
- 2D image: input data
- primal sketch: detection of significant intensity changes (edges)
- 2D sketch: reconstruction of a depth map
- 3D representation: movement to object-centered description
  - the last stage matches a top-down step
  - a priori knowledge can be used for regularization

4.5 3D projection geometry (9.2.1/11.2.1)
- 3D world is mapped to a 2D plane
- in perspective projection, parallel lines meet in the epipole

4.6 Geometry of single perspective camera (9.2.2/11.3.1)
- all points along an optical ray project to the same point
- single perspective camera system includes four coordinate systems:
  - world coordinates \( X = (X, Y, Z, W)^T \)
  - camera coordinates \( X_c = (X_c, Y_c, Z_c, W)^T \)
  - image Euclidean coordinates \( u = (u_x, v_x, w)^T \)
  - image affine coordinates \( u = (u_x, v_x, w)^T \)

Transformations between coordinate systems
A scene point \( X \) is transformed to camera coordinates with the extrinsic camera calibration parameters shift \( t \) and rotation \( R \) in the 3-dimensional non-homogeneous case:
\[
X_c = R(X - t)
\]
The same can be written in homogeneous form where \( X_c \) and \( X \) are 4-dimensional by augmentation of \( "1" \):
\[
X_c = \begin{bmatrix}
R & -Rt \\
0^T & 1
\end{bmatrix} X
\]
The camera coordinate point $X_c$ is projected to the image plane in Euclidean coordinates by the non-homogeneous equation as:

$$u = \frac{X_c f}{Z_c}, \quad v = \frac{Y_c f}{Z_c}$$

In homogeneous coordinates it equals to:

$$\mathbf{u} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{X}_c$$

(The homography or collineation symbol $\simeq$ means that the equation holds up to unknown scale.)

It will be easier to assume first that $f = 1$ and introduce its true value later. Then:

$$\mathbf{u} \simeq \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{X}_c$$

41

4.9 Point correspondence in stereo vision (9.2.11/11.6.1)

The correspondence of points in camera pair views is constrained by:

- epipolar constraint
- uniqueness constraint (almost always)
- symmetry constraint
- photometric (intensity) compatibility constraint
- geometric similarity constraints
- disparity smoothness constraint
- feature compatibility (same discontinuity) constraint
- disparity search range / small disparity values
- disparity gradient limit
- ordering constraint (almost always)

45

4.7 Two cameras (stereopsis) (9.2.5/11.5.1)

- in a general setting the two cameras can see each other
- the line connecting the cameras is called the baseline
- the cameras and the object point determine the epipolar plane
- the intersection of the image plane with the baseline and the rays from $X$ determine the epipolar lines $l$ and $l'$
- in the rectified configuration the cameras have parallel axes

42

4.8 Shape from stereo vision (9.2.5/11.5.5)

- projection geometry: perspective / orthographic

- $z$ can be solved from similar right-angled triangles:

$$\frac{u}{f} = \frac{-h + x}{z}, \quad \frac{u'}{f} = \frac{h - x}{z} \quad \Rightarrow \quad z = \frac{2hf}{h' - u} = \frac{2hf}{d}$$

43

Matching based of correlation

Multiple approaches exist:

- best match for each pixel without any special interest points
- each pixel block is associated with the best matching pixel block
- the resulting disparity function may be sparse, can be made denser
- edge detection can be applied prior to correlation
- gradual refinement of the resolution
- projection of a dot pattern on the scene

46

PMF algorithm

- matching based on candidate pairs of similar visual feature points
- set of feature points (eg. edges) are extracted from each image
- epipolar constraint: the y coordinate is the same in matching points
- uniqueness constraint: one-to-one matching

47

- cyclopean separation $S(A, B)$ is the distance between A and B:

$$S(A, B) = \sqrt{(\frac{ax - bx}{2} - \frac{by + ay}{2})^2 + (a_y - b_y)^2}$$

- the difference $D(A, B)$ in disparity between matches A and B is:

$$D(A, B) = (a_x - bx) - (by - b_y) = x_l - x_r$$

- the disparity gradient $\Gamma(A, B)$ should be as small as possible:

$$\Gamma(A, B) = \frac{D(A, B)}{S(A, B)} = \frac{x_l - x_r}{\sqrt{(x_l + x_r)^2 + (a_y - b_y)^2}}$$

- small disparity gradient $\Rightarrow$ coherence between A and B
4.10 Active acquisition of range images (9.2.12/11.6.2)
- shape from X techniques are generally passive
- range image / depth map can be obtained with explicit methods
- laser light / reflection, delay, phase shift
- laser stripe finders

- laser stripe is projected on the object
- object or stripe is shifted and/or rotated
- radar images, ultra sound images
- Moiré interference pattern gives relative distance information

4.11 Radiometry in 3D vision (9.3/11.7.1.3.4.5)
- radiometry, photometry, shape from shading
- humans can perceive distances from intensity changes
- measured intensity depends on surface reflectance and direction
- light source(s) have effect on the measured intensity
- radiometric methods are generally quite unreliable
- reflectance function \( R(\Omega) \) in spherical angle coordinates \( \Omega \)
  - Lambertian matte
  - specular (mirror) surface
- surface distance \( z(x,y) \), surface gradient space
  - normal vector of the surface: \( f(x,y), q(x,y) \) = \( \langle \hat{n}_x, \hat{n}_y \rangle \)
- reflectance map \( R(x,y) \) (\( x(\tau,y), q(x,y) \))
- shade smoothness constraint: \( (\nabla f(x,y))^2 + (\nabla q(x,y))^2 \) small
- good match of \( f(x,y) \) and \( R(x,y) \) reveals \( z(x,y) \)

4.12 Shape from X (10.12)
- shape from stereo
- shape from shading
- shape from motion
- shape from optical flow
- shape from texture
- shape from focus
- shape from defocus
- shape from vergence
- shape from contour

4.13 Shape from motion (10.1.12.1.1)
- human eyes and brain use motion information very efficiently
- Ullman's experiment with virtual coaxial cylinders
- static background is generally assumed
- objects need to be rigid, i.e., their shape must not change
- objects can move and rotate, they have six degrees of freedom
- sequence of images captures motion of objects
- assume we can match \( N \) correspondence points \( (x,y) \) in all images
- images from different times are equivalent to different projections
- 3 projections \( \times 4 \) matched points \( \Rightarrow 1 \) interpretation

4.14 Shape from texture (10.1.2/12.1.2)
- human eyes and brain use texture information very efficiently
- texture primitives or texel are distorted \( \Rightarrow \) texture gradient

4.15 Models of 3D world (10.2/12.2)
- 3D models of the world have two very different uses:
  - reconstruction of the model by the actual object (1)
  - recognition of the actual object by the model (1)
- model creation can be compared with CAD systems
- wire models vs. surface models vs. volumetric models
- the completeness and uniqueness of presentation

4.16 Line labelling algorithm (10.2.2/12.2.2)
- Roberts 1965, Clowes 1971, Huffman 1971
- for modelling of (only) blocks world
- each 3D edge is a meeting of two planar faces
- each 3D vertex is a meeting of three planar faces
- each 3D vertex can be seen in four different types of junction

- 22 different 2D to 3D vertex interpretations exist
- all interpretations of all detected 2D edges can be listed
- both edges of an edge need to have same interpretation (convex/concave)
- global coherence of interpretation for all edges and surfaces
4.17 More models of 3D world (10.2.4.5/12.2.4.5)
- Constructive Solid Geometry (CSG): cuboid, cylinder, sphere, cone and half-space
- Volumetric models: voxels or super-quadrics:
  \[
  \left(\frac{x}{a_1}\right)^2 + \left(\frac{y}{a_2}\right)^2 + \left(\frac{z}{a_3}\right)^2 = 1
  \]
- Generalized cylinders
  ![Generalized Cylinder Diagram]
- Surface models: surfaces/hedges/graph
- Surface triangulation, e.g., Delaunay triangulation
- Surface modelling with quadric model:
  \[
  \sum_{a_{ijkl}} a_{ijkl}x^iy^jz^k = 0
  \]

4.18 On recognition of 3D objects (10.3/12.3)
- Top-down: sensor data matched with existing model
- Sensor data is often limited to part of the object
- Also matching needs to be based on partial object model
- Part of the model is used to formulate a matching hypothesis
- Matching can be performed on data or feature level

4.19 Good's algorithm (10.3.2/12.3.2)
- Recovered coordinates (location and rotation) of a known 3D object
- The object known as wire model; edges are detected in the image
- Distance to the camera is known (and therefore also the size)
- The object is fully visible in narrow field of view
- The 5 degrees of freedom of the camera are quantized
- Matching is done edge by edge
- Cameras location gets more precise on each iteration
- Matching choices → branching points
- No choices left → backtracking
- Preprocessing of the model can be used to speed up the process

4.20 Model-based 3D recognition from intensity images (10.3.3/12.3.3)
- Description of curved surfaces more difficult than linear ones
- One image doesn't provide enough information
- A partial model can be created from the image
- The partial model can be compared with stored full models
- Surface features, typically, eg, curvature
- Surface characterization, partitioning of the surface
- Invariances to projection, rotation and shift

4.21 2D view-based representations for 3D (10.4.12.4)
- View-centered representation (as compared to object-centered)
- Characteristic images stored for all different viewpoints
- 2D projections of all surfaces and vertices
- Creation of an aspect graph

4.22 Geons as 2D view-based representation (10.4.3/12.4.3)
- Geons (GEOmetrical IONS): 36 enumerated models with following attributes:
  - Edge: straight / curved
  - Symmetry: rotational / reflective / asymmetric
  - Size variation: constant / expanding / varying
  - Spine: straight / curved

Other techniques (10.4.4/12.4.4)
- Use of multiple stored 2D views
- 2D reference views
- Creation of virtual view

5. Data structures

5.1 Introduction (3.1/4.1)
- Data structures pass information from one abstraction level to another
- Different information abstractions call for different data structures
- Data structures and algorithms are always coupled
  - "Iconic" pixel image
  - Edges detected in image to match object borders
  - Image segmented in regions
  - Geometric representations
  - Relational models
5.2 Traditional data structures (3.2/4.2)
- matrices
  - spatial and neighborhood relations
  - binary / grayscale / multichannel
  - use of different resolutions leads to hierarchic structure
  - co-occurrence matrix
- chain codes
- topological descriptions
- relational structures

5.3 Hierarchic data structures (3.3/4.3)
- matrix pyramids and tree pyramids
- quadrees

5.4 Co-occurrence matrix (3.2.1/4.2.1)
- 2-dimensional generalization of histogram
- joint distribution of grayscale values of neighboring pixels
  \[ C(x_1, y_1) = \#(f(x_1, y_1) = z_1, f(x_2, y_2) = z_2, (x_1, y_1) \neq (x_2, y_2)) \]
- if relation \( r \) is \( = \), then \( C(r, z) \) is histogram
- typically \( r \) is a shift: \( x_2 = x_1 + \Delta x, y_2 = y_1 + \Delta y \)
- often \( r \) is assumed symmetric → \( C_r \) is symmetric
- measurement of edges of specific orientation and values
- texture analysis
- intermediate representation for feature extraction

5.5 Chain structures (3.2.2/4.2.2)
- used often for describing object boundaries
- chain-coded object is not bound to any specific location
- chain code also Freeman code aka F-code
  - directions between adjacent 4 or 8-neighbor pixels
  - starting point has to be fixed
- vectors between chain-coded pixels can also be longer than one
- chains can be either closed or open
- rotations are easy to implement with chain codes

5.6 Topological data structures (3.2.3/4.2.3)
- graphs \( G = (V, E) \)
- nodes \( V = \{v_1, v_2, \ldots, v_n\} \)
- edges \( E = \{e_1, e_2, \ldots, e_m\} \)
- degree of a node
- weighted or evaluated graph: costs associated to nodes and edges
- region adjacency graph and region map

5.7 Relational database structures (3.2.4/4.2.4)

<table>
<thead>
<tr>
<th>Nr.</th>
<th>Object</th>
<th>Color</th>
<th>Start row</th>
<th>Start column</th>
<th>Inside of</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>sun</td>
<td>white</td>
<td>5</td>
<td>40</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>sky</td>
<td>blue</td>
<td>0</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>cloud</td>
<td>gray</td>
<td>20</td>
<td>180</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>tree trunk</td>
<td>brown</td>
<td>95</td>
<td>75</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>tree crown</td>
<td>green</td>
<td>53</td>
<td>63</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>hall</td>
<td>light green</td>
<td>97</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>pond</td>
<td>blue</td>
<td>100</td>
<td>160</td>
<td>6</td>
</tr>
</tbody>
</table>

5.8 Hierarchical data structures (3.3/4.3)

Pyramids (3.3.1/4.3.1)
- matrix or M pyramids
  - series of matrices \( M_0, M_{i-1}, \ldots, M_0 \)
  - \( M_0 \) = original image
  - \( M_0 = 1 \) pixel
  - \( M_{2N} = \frac{1}{N} \) of \( M \)
- tree or T pyramids
  - graph where nodes are placed in layers
  - each layer matches a matrix in M pyramid
  - each node has 4 sibling nodes
  - method for calculating values in parent nodes is needed
  - total number of nodes \( N^2(1 + \frac{1}{4} + \frac{1}{16} + \cdots) = 1.23N^2 \)

Quadrees (3.3.2/4.3.2)
- resemble T pyramids, but:
  - only heterogeneous nodes are divided
  - non-balanced tree
  - sensitive to small changes in input images
  - bounding to object coordinates
  - paths can be coded as symbol strings

6. Pre-processing
- pixel image → pixel image
- processing remains at low abstraction level
- data enhancement for later processing stages
- different pre-processing techniques:
  - changing brightness value of a single pixel
  - geometric transformations
  - local neighborhood methods
  - frequency domain operations
  - or another taxonomy:
    - image enhancement
    - image restoration
6.3 Brightness interpolation (4.2.2/5.2.2)
- inverse mapping \((x', y') = T^{-1}(x, y)\)
- nearest-neighbor interpolation: \(f_s(x, y) = g_s(\text{round}(x), \text{round}(y))\)
- linear interpolation:
\[
I = \text{round}(x), \quad k = \text{round}(y), \quad a = x - I, \quad b = y - k
\]
\[
f_l(x, y) = (1 - a)(1 - b)g(I, k) + a(1 - b)g(I + 1, k) + \cdots
\]
- bi-cubic interpolation

6.4 Local pre-processing (4.3.1/5.3.1)
- masks, convolutions, filtering/derivation
- smoothing and gradient operators
- linear and non-linear methods
- edge-preserving smoothing
- sequential smoothing, noise suppression \(\sigma^2/n\)
- spatial averaging
\[
h = \frac{1}{n} \begin{bmatrix}
1 & 1 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1 
\end{bmatrix} \quad h = \frac{1}{n} \begin{bmatrix}
2 & 2 & 1 \\
1 & 2 & 1 \\
1 & 1 & 1 
\end{bmatrix}
\]

6.5 Additional constraints for local averaging (4.3.1/7)
- only for limited grayvalue range
- only for limited range of grayvalue changes
- only for small gradient magnitudes
- in proportion to inverse of gradient magnitude
\[
\delta(i, j, m, n) = \frac{1}{[g(m, n) - g(i, j)]}
\]
\[
h(i, j, m, n) = 0.5 \frac{\delta(i, j, m, n)}{\sum_{(m,n)} \delta(i, j, m, n)}
\]

Rotating mask in averaging (4.3.1/5.3.1)
- the neighborhood that produces the lowest variance is selected

Non-linear methods
- median filtering
- filterings based on ranks and order statistics
- non-linear mean filters
- homomorphic filtering
6.6 Local neighborhood in edge detection (4.3.2/5.3.2)

- gradient has direction and magnitude

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

\[ \psi = \arctan \left( \frac{\partial g}{\partial y} \right) \frac{\partial g}{\partial x} \]

- Laplacian

\[ \nabla^2 g(x,y) = \frac{\partial^2 g(x,y)}{\partial x^2} + \frac{\partial^2 g(x,y)}{\partial y^2} \]

- image sharpening, unsharp masking

\[ f(i,j) = g(i,j) + \alpha \left( g(i,j) - f(i,j) \right) \]

- approximation of derivatives

- zero-crossings of the second derivative

- parametric fitting

81

6.7 Edge detection by derivative approximation (4.3.2)

- Robert's

\[
\begin{bmatrix}
1 & 0 & -1 \\
0 & 1 & 0 \\
-1 & 0 & 1
\end{bmatrix}
\]

- Prewitt

\[
\begin{bmatrix}
1 & 0 & -1 \\
1 & 1 & 1 \\
-1 & 0 & 1
\end{bmatrix}
\]

- Sobel

\[
\begin{bmatrix}
1 & 2 & 1 \\
0 & 0 & 0 \\
-1 & -2 & -1
\end{bmatrix}
\]

- Laplace

\[
\begin{bmatrix}
0 & 1 & 0 \\
-1 & 2 & -1 \\
0 & 0 & 1
\end{bmatrix}
\]

82

6.8 Marr-Hildreth edge detector (4.3.3)

- edge detection from second derivative zero-crossings

- image smoothing by a Gaussian kernel

\[ G(x,y;\sigma) = e^{-\frac{x^2+y^2}{2\sigma^2}} \]

- calculation of Laplacian image

\[ \nabla^2[G(x,y;\sigma) \ast f(x,y)] \]

- association order of the operators is changed: Laplace of Gaussian, LoG

\[ \nabla^2[G(x,y;\sigma)] \ast f(x,y) \]

- algebraic solution of the second derivative

\[ G^2(r;\sigma) = \frac{1}{2\sigma^2}(1-e^{-\frac{r^2}{2\sigma^2}}) \]

\[ h(r;\sigma) = (1-e^{-\frac{r^2}{2\sigma^2}}) e^{-\frac{r^2}{2\sigma^2}} \]

83

- “Mexican Hat” function

- proper mask size is approximately \(6\sigma \times 6\sigma \cdots 10\sigma \times 10\sigma \)

- operator can be separated in \(x\) and \(y\)-directions

- resembles the operation of the human eye

- \(\nabla^2 G\) can be approximated by the difference of two Gaussians, DoG

84

6.9 Scale-space methods (4.3.4/5.3.4)

- smoothing parameter, e.g., \(\sigma\), is varied to produce a family of images

- 1) curves can be analyzed at multiple scales

- 2) scale-space filtering, \(I(x,y)\), to a set of \(F(x,y;\sigma)\) images

- convolution with Gaussian function (1-dim. case)

\[ C(x) = e^{-\frac{x^2}{2\sigma^2}} \]

- \( F(x,y;\sigma) = C(x) \ast f(x,y) \)

- edges from second derivative's zero-crossings

\[ \frac{\partial^2 F(x,y;\sigma)}{\partial x^2} = 0, \quad \frac{\partial^2 F(x,y;\sigma)}{\partial y^2} \neq 0 \]

- different qualitative information with different \(\sigma\) interval tree

85

6.10 Canny edge detector (4.3.5/5.3.5)

- optimal for step-shape edges in additive white noise

- detection: edges are not missed, no spurious responses

- localization: located and actual positions near each other

- uniqueness: single edge doesn't produce multiple responses

1. image \(f\) is convolved with \(\sigma\)-scale Gaussian function

2. local edge's normal direction is estimated in each pixel

\[ n = \frac{\nabla(G \ast f)}{|\nabla(G \ast f)|} \]

3. 2nd derivative's zero-crossings are located in the normal direction

\[ \frac{\partial^2}{\partial n^2} G \ast f = 0 \quad \text{non-maximal suppression} \]

86

4. edge thresholding with hysteresis

- generalization of thresholding with high and low thresholds

- weak edge pixels are supported by strong nearby edge pixels

- only strong changes are detected, increased signal-to-noise ratio

5. edge information is collected with different \(\sigma\) values

- edge feature synthesis from small \(\sigma\) to large \(\sigma\)

- differences between prediction and reality give true information

87

6.11 Parametric edge models (4.3.6/5.3.6)

- a facet model is estimated for each pixel, e.g.,

\[ g(x,y) = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy + a_6 y^2 + a_7 x^2 y + a_8 xy^2 + a_9 y^3 \]

- least-squares methods in matching

- extreme points and values of derivatives are solved from the parameters

- sub-pixel localization

88

6.12 Edges in multi-channel images (4.3.7/5.3.7)

- edges can be solved for each channel separately

- scalar value can be obtained from the sum or maximum value

- channel difference or ratio can also be used

- Roberts gradient has a \(2 \times 2 \times n\)-sized generalization

- only magnitude, no direction information is produced
6.13 Other local neighborhood operations (4.38/5.3.9)
- Some methods fall under morphological operations
- Narrow lines detected with matched masks of different orientation
\[ f(i,j) = \max_k (h_k \cdot g(i,j)) \]

\[
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

\[ h_1 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}, \quad h_2 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix} \]

\[ h_3 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}, \quad h_4 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix} \]

6.16 Frequency domain image restoration (4.4/5.3.8, 5.4)
- Correction of degradations caused by image formation
- Deterministic / stochastic methods
- Image degradation model:
\[ g(i,j) = |f(i,j) + v(i,j)| \]
\[ G(u,v) = H(u,v)F(u,v) + N(u,v) \]
- Camera or object motion
- Wrong focus
- Atmospheric turbulence
- Inverse滤波器
\[ F(u,v) = \frac{\hat{G}(u,v)}{|\hat{G}(u,v)|^2 + \sigma^2} \]
\[ F(u,v) = \frac{\hat{N}(u,v)}{|\hat{N}(u,v)|^2 + \sigma^2} \]

7. Morphology
- Processing of binary images with logical set operations
- Image is treated as a set of pixels
- Generalizations to gray-level images exist

Application areas: (11.1)
- Pre-processing: reduction of binary noise
- Shape extraction or enhancement
- Qualitative description of objects

Operations:
- Dilation & erosion
- Opening & closing
- Hit-or-miss: thinning & thickening
- Conditional operations

7.1 Basic notions and operations (11.1/13.1)
- Point set \( E^2 = \text{Euclidean 2-dimensional space} \)
- Discrete point set \( Z^2 \) or \( Z^3 \)
- Subset \( C \subseteq \mathbb{Z} \), intersection \( \cap \), union \( \cup \)
- Empty set \( \emptyset \), complement \( C' \), difference \( X \setminus Y = X \cap Y^c \)
- Symmetrical set or rational set or transpose \( B = (-b : b \in B) \)
- Operator \( \Psi() \) has a dual operator \( \Psi^*() : \Psi(X) = \Psi^*(X)^c \)
- Structuring element, isotropic, structuring element
- Origin / reference point / current pixel
Translation or shift
\[ X_0 = \{ p \in \mathbb{Z}^2 : p = x + h \text{ for some } x \in X \} \]

Quantitative morphological operations (11.2/13.2)
- compatibility with translation \[ \Psi(X_0) = \Psi(X)_h \]
- compatibility with change of scale \[ \Psi(X) = \lambda \Psi(\frac{1}{\lambda} X) \]
- local knowledge \[ \Psi(X \cap Z') \cap Z' = \Psi(X) \cap Z' \]
- upper semi-continuity

7.2 Dilation \( \oplus \) (fill, grow) (11.3.1/13.3.1)
- expands the image, fills gaps
- \( X \oplus B = \{ p \in \mathbb{Z}^2 : p = x + b, x \in X \text{ and } b \in B \} \)

7.3 Erosion \( \ominus \) (shrink, reduce) (11.3.2/13.3.2)
- makes the image smaller, removes details
- \( X \ominus B = \{ p \in \mathbb{Z}^2 : p + b \in X \text{ for all } b \in B \} \)

7.4 Some properties of dilation and erosion
Duality
\[ (X \ominus Y)^c = X^c \ominus Y \]

Combination laws
\[ (X \cap Y) \oplus B \subseteq (X \oplus B) \cap (Y \oplus B) \]
\[ B \oplus (X \cap Y) \subseteq (X \oplus B) \cap (Y \oplus B) \]
\[ B \oplus (X \cup Y) = (X \cup Y) \oplus B \]
\[ B \oplus (X \cup Y) = (X \oplus B) \cup (Y \oplus B) \]
\[ (X \ominus Y) \ominus B = (X \ominus B) \cap (Y \ominus B) \]
\[ B \ominus (X \cap Y) \subseteq (X \ominus B) \cap (Y \ominus B) \]
\[ B \ominus (X \cup Y) = (X \ominus B) \cup (Y \ominus B) \]

Association laws
\[ (X \oplus B) \oplus D = X \oplus (B \oplus D) \]
\[ (X \ominus B) \ominus D = X \ominus (B \ominus D) \]

7.5 Opening \( \circ \) and closing \( \bullet \) (11.3.4/13.3.4)
- combinations of dilation and erosion
- opening removes non-connected points
- closing fills in holes and gaps
- area is preserved approximately in the operations
- \( X \circ B = (X \ominus B) \oplus B \]
- \( X \bullet B = (X \oplus B) \ominus B \]
- operations are each other's duals: \( (X \bullet B)^c = X^c \circ B \)
- opening and closing are idempotent operations
- one may say that \( X \) is open/closed with respect to \( B \)

7.6 Gray-scale dilation and erosion (11.4/13.4)
- above operations work with binary images only
- generalizations to gray-scale images exist
- gray-scale dilation as max operation
\[ (f \oplus k)(x) = \max\{f(x - z) + k(z), z \in K, x - z \in F\} \]
- gray-scale erosion as min operation
\[ (f \ominus k)(x) = \min\{f(x + z) - k(z), z \in K\} \]
- Point set \( A \subseteq \mathbb{Z}^n, n = 3 \)
- \( A \)'s support \( F = \{ x \in \mathbb{Z}^n \mid x \in \mathbb{Z}^n \} \)
- \( A \)'s supporting \( T[A](x) = \max\{a(x,y) \mid y \in A\} \)
- \( f(x) \)'s umbra \( U[f] = \{ x, y \mid f(x) \leq f(y) \} \)
- gray-scale dilation \( f \circ k = T[U[f] \oplus k] \)
- gray-scale erosion \( f \ominus k = T[U[f] \ominus k] \)

7.7 Skeletons and maximal ball
Homotopic transforms (11.5.1/13.5.1)
- don't change topological relations
- homotopic tree, that shows neighborhood relations, remains the same

Skeletons (11.5.2/13.5.2)
- medial axis transform
- grassfire metaphor
- formation with maximal balls
- the result can be non-homotopic
- homotopic skeleton can be extracted with morphological thinnings
- easy to understand in Euclidean world – discrete world is difficult

Maximal ball \( B(p, r) \)
- unit ball \( B \) or \( 1B \) contains the origin and points in distance \( 1 \) from it
- \( nB \) is \( B \)'s \((n - 1)\)th successive dilation with itself
\[ nB = B \oplus \cdots \oplus B \]
- ball \( B(p, r) \), shape \( B \) located in \( p \) with radius \( r \), is maximal if
  \( B(p, r) \subseteq X \) and
  \( \exists \ r \) cannot be a larger ball \( B' \) so that \( B(p, r) \subseteq B' \subseteq X \)
  \( \Rightarrow \) for all \( B' \) it holds \( B' \subseteq B \)
- skeleton by maximal balls
\[ S(X) = \{ p \in \mathbb{Z}^n : \exists r \geq 0, B(p, r) \text{ is } X \text{'s maximal ball} \} \]
\[ S(X) = \bigcap_{i=0}^{\infty} (X \ominus nB) \setminus (X \ominus nB + B) \]
7.8 Hit-or-miss $\otimes$, thinning $\circ$, thickening $\oplus$ (13.33.13.5.3)
- composite structuring element is an ordered pair $B = (B_1, B_2)$
- $X \ominus B = (x : B_1 \subset X \land \exists B_2 \subset X)$
- $X \ominus B = (X \ominus B_1) \cap (X \ominus B_2) = (X \ominus B_1) \setminus (X \ominus B_2)$
- $X \ominus B = X \cup (X \ominus B)$
- thinning and thickening are dual transformations
  $(X \ominus B)^c = X^c \ominus B^c ; \quad B^c = (B_2, B_1)$
- sequential thinnings/thickenings with Goltz algebras
  $X \ominus (B_0) \ominus (B_0) = (((X \ominus B_0) \ominus B_0) \cdots \ominus B_0) \cdots)
  X \ominus (B_0) \ominus (B_0) = (((X \ominus B_0) \ominus B_0) \cdots \ominus B_0) \cdots)
- homotopic skeleton is ready when thinning is idempotent

7.9 Goltz algebras
- thinning with $L$ element (4-neighbors)
  $l_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$
- thinning with $E$ element (4-neighbors)
  $e_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$
- thinning with $M$ element (4-neighbors)
  $m_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$
- thinning with $D$ and thickening with $D'$ element (4-neighbors)
  $a_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$
- thinning with $C$ element (4-neighbors)
  $c_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

7.10 Quench function and ultimate erosion (11.5.4/13.54)
- quench function $q_{X}(p)$:
  $X = \cup_{p \in X} (p + q_{X}(p)B)$
- $q_{X}(p)$'s regional maxima points = ultimate erosion $\text{Ult}(X)$
- ultimate erosion can be used to extract markers in objects
- original object can be reconstructed from markers
- market set $B \subset A = = \text{reconstruction} \mu_{B}(B)$
- ultimate erosion can be expressed as
  $\text{Ult}(X) = \cup_{p \in A} (X \ominus \{p \in (n + 1)B\})$

7.11 Ultimate erosion and distance functions (11.5.5/13.55)
- distance function $d_{X}(p)$ is $p$'s distance from $X$:
  $\forall p \in X \quad d_{X}(p) = \min(n \in N, p \notin (X \ominus nB))$
- ultimate erosion is the set of $d_{X}(p)$'s regional maxima points
- maximal ball skeleton is the set of $d_{X}(p)$'s local maxima points
- each connected component $X_i$ of set $X$ has an influence zone
  $Z(X_i) = \{p \in Z, \forall i \neq i, d(p, X_i) \leq d(p, X_i)\}$
- skeleton by influence zones (SKIZ) is the set of boundary pixels of the influence zones $Z(X_i)$$

7.12 Geodesic transformations (11.5.6/13.56)
- geodesic transformations are restricted inside subset $X$
- interval distances $d_{X}(x, y)$ measured along paths inside $X$
- a geodesic ball located at $p$ with radius $n$ inside $X$
  $B_{X}(p, n) = \{p' \in X : d_{X}(p, p') \leq n\}$
- $X$'s geodesic dilation $d_{X}(Y)$ with $n$-radius ball inside $X$
  $d_{X}(Y) = \cup_{p \in Y} B_{X}(p, n) = \{p' \in X : \exists p \in Y, d_{X}(p, p') \leq n\}$
- corresponding geodesic erosion $e_{X}(Y)$
  $e_{X}(Y) = \{p \in X, \forall y \in Y, d_{X}(p, y) = n\}$
- result of a geodesic operation is always a subset of $X$
- geodesic dilation with unit ball $d_{X}(Y) = (Y \ominus B) \cap X$
- geodesic dilation with $n$-radius ball $d_{X}(Y) = (Y \ominus B_{X}) \cap X$

7.13 Morphological reconstruction (11.5.7/13.57)
- geodesic dilations can be used to implement reconstruction
- start with marker set $X$ inside object $Y$
- dilation with geodesic ball grows $Y$ while restricting it inside $X$
- only components of $X$ that contain a marker are reconstructed
- many markers inside one component $\Rightarrow$ geodesic SKIZ inside the component
- reconstruction can be generalized for grayscale images
- binary scale image is interpreted as a stack binary images obtained by thresholding

7.14 Granulometry (11.6/13.6)
- granulometry measures the sizes of objects or particles
- a size histogram is created that describes distribution of particle sizes
- particle sizes resolved by openings/erosions with an increasing ball
- $\psi_{i}(X)$ is $X$ after opening with $i$-sized ball
  $\psi_{i}(X) = X \ominus i$ for $i \in \mathbb{N}$
- pattern spectrum or granulometric curve $PS_{i}(X)(x)$:
  $PS_{i}(X)(x) = m_{\psi_{i}(x)} - m_{\psi_{i}(x)} \forall m > 0$
- granulometric function $G_{i}(X)(x)$:
  $G_{i}(X)(x) = \min(n > 0, x \notin \psi_{n}(X))$
  $PS_{i}(X)(n) = \text{card}(p, G_{i}(X)(p) = n)$

7.15 Morphological segmentation, waterheads (11.7/13.7)
- morphological segmentation is suitable for binary particles
- markers are first extracted inside the particles
- watershed method is then used for reconstructing the particles
- areas between the waterheads are "basins of increasing water"
- geodesic influence zones and SKIZ can produce correct segments
- watershed segmentation may produce a better result
8. Texture

Some examples of textured real-world surface images:

---

8.1 Properties of natural textures
- surface shape / surface structure / surface image
- physical origin is often 3-dimensional
- texture analysis uses 2-dimensional images
- effect of lighting and light direction?
- direction/orientation of the texture or is it unoriented?
- texture primitives / texture elements, textures
- spatial relations between primitives, dependency on the scale
- tone and structure
- fine / coarse texture, weak / strong texture
- can there exist constant texture in a natural image?
- statistical and structural descriptions, hybrid descriptions
- human eye's ability to recognize textures, textures

8.2 Statistical texture descriptions (14.1.1.15.1.1)
- formation of a statistical feature vector
- one feature vector can describe a large area or a single pixel
- use of pixel-wise feature vectors:
  - comparison between neighboring pixels, clustering
  - averaging inside areas of nearly constant values, segmentation
- generally statistics of second order
- methods based on spatial frequencies
- autocorrelation function
  \[ C_{xy}(p,q) = \frac{MN}{(M-p)(N-q)} \sum_{i=1}^{M-p} \sum_{j=1}^{N-q} f(i,j) f(i+p,j+q) \]

8.3 Co-occurrence matrices (14.1.2.15.1.2)
- 2-dimensional generalizations of 1-dimensional histograms
- second order statistics of two nearby pixel values
- parameters: distance \( d \), angle \( \phi \)
- symmetric / asymmetric definition
  \[ P_{xy}(a,b) = \left\{ \begin{array}{l}
  k = m = 0, f(k,l) = a, f(m,n) = b \\
  k = m = d, l = n = -d, f(k,l) = a, f(m,n) = b \\
  \end{array} \right. \]

8.4 Co-occurrence matrices – an example

Gray-scale image, 4 intensity levels:

\[
\begin{array}{cccc}
0 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 \\
0 & 2 & 2 & 2 \\
2 & 2 & 3 & 3 \\
\end{array}
\]

Co-occurrence matrices:

\[
P_{xy} = \begin{pmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix} \quad P_{xy'} = \begin{pmatrix} 2 & 1 & 3 & 0 \\ 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 2 \\ 0 & 0 & 2 & 0 \end{pmatrix}
\]

8.5 Haralick features from co-occurrence matrix
- energy
  \[ \sum_{ab} P_{xy}(a,b) \]
- entropy
  \[ \sum_{ab} P_{xy}(a,b) \log P_{xy}(a,b) \]
- maximum probability
  \[ \max_{ab} P_{xy}(a,b) \]
- contrast
  \[ \sum_{ab} (a-b)^2 P_{xy}(a,b) \]
- inverse difference moment
  \[ \sum_{ab} \frac{P_{xy}(a,b)}{a+b} \]
- correlation
  \[ \frac{\sum_{ab} (a-b)(a'-b') P_{xy}(a,b)}{\sum_{ab} (a-b)^2 P_{xy}(a,b)} \]

8.6 Edge frequency (14.1.3.15.1.3)
- average gradient magnitude can be calculated with varying scale \( d \):
  \[ g(d) = |f(i,j) - f(i+d_j)| + |f(i,j) - f(i-d_j)| + |f(i,j) - f(i+j) - f(i+j-d_j)| + |f(i,j) - f(i+j+d_j)| + |f(i,j) - f(i+j-d_j)| \]
- compare with autocorrelation function: minima = maxima
- first and second order edge statistics can be characterized:
  - coarseness: finer texture ~ higher number of edge pixels
  - contrast: higher contrast ~ stronger edges
  - randomness: entropy of edge magnitude histogram
  - directivity: histogram of edge directions
• more edge statistic features
  - linearity: sequential edge pairs with same direction
  - periodicity: parallel edge pairs with same direction
  - size: parallel edge pairs with opposite directions

8.9 Other statistical methods (14.1.6–7, 15.1.6–8)
  • fractal texture description
  • wavelets, Gabor transforms, wavelet energy signatures
  • morphological methods: erosion, opening
  • texture transform \( f(x,y) \rightarrow g(x,y) \)
  • autoregression texture models
  • peak and valley method
  • Markov random fields

8.7 Run length statistics (14.1.4, 15.1.4)
  • \( B(a,r) \): number of runs/primatives of length \( r \) and value \( a \) in \( M \times N \) image
    - total number of runs: \( K = \sum_{a}^{p} \sum_{r=1}^{R} B(a,r) \)
    - short primitives emphasis: \( \frac{k}{a} \sum_{a}^{p} \sum_{r=1}^{R} B(a,r)^2 \)
    - long primitives emphasis: \( \frac{k}{a} \sum_{a}^{p} \sum_{r=1}^{R} B(a,r)r^2 \)
    - gray-level uniformity: \( \frac{k}{a} \sum_{a}^{p} \sum_{r=1}^{R} B(a,r)^2 \)
    - primitive length uniformity: \( \frac{k}{a} \sum_{a}^{p} \sum_{r=1}^{R} B(a,r)r \)
    - primitive percentage: \( \frac{k}{a} \sum_{a}^{p} \sum_{r=1}^{R} B(a,r) \)

8.8 Laws' texture energy measures (14.1.5, 15.1.5)
  • Laws' texture energy masks can measure
    - grayvalues
    - edges
    - spots
    - waves
  - three one-dimensional masks:
    \( L_0 = (1,2,1), \ E_0 = (-1,0,1), \ S_0 = (-1,2,-1) \)
  - their one-dimensional convolutions:
    \( L_0 \ast L_0 = L_0 = (1,4,6,4,1) \)
    \( L_0 \ast E_0 = E_0 = (-1,-2,0,2,1) \)
    \( L_0 \ast S_0 = S_0 = (-1,0,2,0,-1) \)
    \( S_0 \ast S_0 = R_0 = (1,-4,6,-4,1) \)
    \( E_0 \ast S_0 = W_0 = (1,-2,0,2,-1) \)

8.11 Graph grammars (14.2.2, 15.2.2)
  • comparison between 2D texture primitive graphs
  • recognition of a set of visual primitives
  • thresholding of distances between texture primitives
  • formation of a graph describing the texture
  • comparison between a graph of input image and stored grammar models
    1) 3D chains of the graph compared with the grammar
    2) stochastic grammar of graphs
    3) direct graph comparison

8.12 Primitive grouping and hierarchical textures (14/15.2.3)
  • many textures are in fact hierarchical
  • can be studied in different scales
  • bottom-up texture primitive grouping
    - detection of homogeneous texture regions
8.13 Hybrid texture description methods (14.3.15.3)
- combinations of statistical and syntactic approaches
- weak textures:
  - division of the image into homogeneous regions
  - statistical analysis of region shapes and sizes
- strong textures:
  - spatial relations between texture primitives
  - primitive sizes one pixel or larger
- hierarchical multi-level description of textures

Thresholding can be based on
- grayvalues
- gradient
- texture
- motion
- something else

Threshold detection methods (5.1.1/6.1.1)
- histogram analysis and filtering, possibly many scales
- is the total area of the objects known?
- uni-, bi- or multi-modal histogram?
- local maxima, minimum distance between them
- histogram extracted from small gradient pixels only
- uni-modal histogram from large gradient pixels

Optimal thresholding (5.1.2/6.1.2)
- model of the distribution needed
- fitting of normal distributions in the histogram
- iterative selection of the parameters
- initial guess for the background from image corners

Segmentation of multi-spectral images (5.1.3/6.1.3)
- each channel segmented separately
  - channel-wise histogram peaks with lower and upper limits
  - union of boundaries from all channels
  - iterative division of the created regions
- multi-dimensional histograms
- classification of n-dimensional pixels

Thresholding in hierarchical data structures (5.1.4/)
- computational efficiency
- removal of noise
- lowering in data pyramid onto higher-resolution level
- detection of important pixels on all levels in 3x3-size
- threshold is fixed on higher levels
- same or updated threshold is used on lower levels

9. Segmentation
- splitting image into semantically meaningful regions
- complete segmentation
  - disjoint regions correspond uniquely with objects in the image
  - information from higher-level processing stages
- partial segmentation
  - similarity between pixels and regions, homogeneity
- segmentation methods
  - thresholding: global knowledge concerning the whole image
  - edge-based segmentation
  - region-based segmentation
  - template matching

9.1 Thresholding methods in segmentation (5.1/6.1)
- complete segmentation in $S$ regions $R_1, \ldots, R_S$:
  $$ R = \bigcup_{i=1}^{S} R_i \text{ and } R_i \cap R_j = \emptyset, \forall i \neq j $$
- selecting of a global threshold $T$
- values larger / smaller than $T$ are background / object
- difficult to find an efficient global solution
- $\Rightarrow$ segmentation in partial images
- background or object can also be a range of values
- creation of an edge image with a narrow range of values
- many simultaneous value ranges

9.2 Edge-based segmentation (5.2/6.2)
- edge-based methods are important historically and in practice
- edges need to be developed to boundaries and regions
- importance of a priori knowledge
- comparison between detected edges and model predictions
- methods and topics:
  - thresholding of gradient magnitude or something
  - non-maximal suppression
  - hysteresis
  - relaxation
  - border tracing
  - use of location information
  - region construction from borders
Edge relaxation (5.2.2/6.2.2)

- comparison of edge information between pixels
- iteration until coherence between neighboring pixels reached
- crack edges

![Crack Edges Diagram]

- types of crack edges

<table>
<thead>
<tr>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>04</td>
<td>05</td>
<td>06</td>
<td>07</td>
</tr>
<tr>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
</tr>
</tbody>
</table>

- each crack edge has a confidence value $0 \leq c^{(0)}(c) \leq 1$

137

- classification of crack edge types
  
  $a \geq b \geq c$, $m = \max(a, b, c)$
  
  $type(0) = \max(a, b, c)$
  
  $type(1) = (m-a)(m-b)/(m-c)$
  
  $type(2) = a(m-b)/(m-c)$
  
  $type(3) = abc$

- modification of each crack edge confidence value

  - 0-0, 0-2, 0-3, $c^{(0)}(c)$ decreases
  - 0-1, $c^{(0)}(c)$ increases little
  - 1-2, $c^{(0)}(c)$ increases very much
  - 1-2, 1-3, $c^{(0)}(c)$ increases quite much
  - 2-0, 2-3, 3-3, no change

- iteration should not be too long, possibly non-linear rounding, towards 0 and 1

Border detection in gray-scale images

- much more difficult than for binary images
- gradient or other edge image is created first
  
  - current boundary direction is continued to locate more edge pixels
  
  - gradient directions are compared between neighboring pixels
  
  - gray value tells whether we are inside or outside the object
- turns and weak edges cause difficulties
- closed boundary can remain undetected
- heuristic search
  
  - starting from the strongest edges
  
  - continuation in backward and forward directions

141

9.3 Border detection as graph searching (5.2.4/6.2.4)

- some amount of a priori knowledge needed
- starting and end points assumed known for detecting optimal path between them
- directed weighted graph

![Directed Weighted Graph]

- list of all open nodes, each node listed at most once
- full path cost estimate $f(n_i) = g(n_i) + \tilde{h}(n_i)$
- the lowest cost estimate $f(n_i)$ expanded to new node
- if forward direction is known, one tries to follow it
- straightening of the path and image with geometric warping?

142

Optimal and heuristic search

- generally $g(n_i)$ is the real cost up to that node
- estimate $\tilde{h}(n_i)$ can affect the speed of search
- if $\tilde{h}(n_i) = 0$, optimal result is guaranteed
- if $\tilde{h}(n_i) > h(n_i)$, some result is obtained fast
- if $0 < \tilde{h}(n_i) < h(n_i)$, optimal result, if $c(n_p, n_q) \geq \tilde{h}(n_q) - \tilde{h}(n_p)$
- if $\tilde{h}(n_i) = h(n_i)$, optimal result with minimal computation
- cost function $f()$ can contain
  
  - strength of edges, inverse of gradient
  
  - difference between gradient directions of succeeding nodes
  
  - distance to a priori assumed location
  
  - distance to end point

143

Node pruning for search speedup

- is the total cost estimate too high?
- is the average cost per path length too high?
- favoring of the largest minimum or smallest maximum along path?
- favoring of the smallest increase?
  
  - breadth first -- depth first
  
  - a lower bound of the cost is obtained: $\tilde{h}(n_i)$

Search for a closed boundary

- selecting one pixel as both starting and end node
- starting into opposite directions
- paths meet (hopefully) on the opposite side

144
Border detection as dynamic programming (5.2.5/6.2.5)

- principle of optimality: all subpaths of an optimal path are also optimal
- one can use a set of starting and end nodes
- entering direction and accumulated cost stored in each node

\[
C^{(p+1)}(k) = \min \{ C^{(p)}(i) + g^p(i,k) \} \\
C^{(p+1)}(k) = \min_{i=1}^{n} \{ C^{(p)}(i) + g^p(i,k) \} \\
\min_{i=1}^{n} (C(x_1, x_2, \ldots, x^N)) = \min_{i=1}^{n} (C(x^N))
\]

9.4 Hough transforms (5.2.6/6.2.6)

- search for shapes in parameter space
- arbitrary curve equation \( f(x, y) = 0 \), e.g., \( x \cos \theta + y \sin \theta - a = 0 \)
- rather few than many parameters, only object's size and shift
- limiting the search with a priori knowledge, e.g., edge direction

\[
\begin{align*}
\rho^2 &= (x_1 - a)^2 + (y_1 - b)^2 \\
a &= x_1 - R \cos(\psi(x)) \\
b &= y_1 - R \sin(\psi(x)) \\
\psi(x) &\in [\psi(x) - \Delta \psi, \psi(x) + \Delta \psi]
\end{align*}
\]

9.5 Region-based segmentation (5.3/6.3)

- important to define homogeneity criterion for a region
- homogeneity inside regions, heterogeneity between them

Region merging (5.3.1/6.3.1)

- initial state: separate pixels, each in its own segment
- combining first neighboring pixels with same grayscale values
- conditional merging of adjacent regions

Fuzzy Hough transform (6.2.6/7)

- first the generalized Hough transform as earlier
- reference point of the fuzzy model is solved first
- exact locations of the borders are specified iteratively
- previous video frame produces the initial model for the next frame

Benefits of Hough transforms

- applicable also for partially occluded objects
- many objects can be detected concurrently
- tolerant for noise
- parallel implementation is possible
Region splitting (5.3.2/6.3.2)
- starting from the whole image in one segment
- segments are split into smaller ones according to some criteria
- criteria can include eg. histogram peaks and existing prominent edges

Splitting and merging (5.3.3/6.3.3)
- quadtrees
  - how merging of nodes 03, 1, 30 and 310 is implemented in Fig 5.47/6.46?
  - use of overlapping trees
    - each child linked to the most probable parent
    - content of the parents recalculated after reassignment of children

Single-pass split and merge
- top-to-down left-to-right scan of the image plane
- 12 templates of size 2x2 pixels
- criterion can be eg. mean of 4 pixels variance
- each pixel is given a segment label from a neighbor, or a new one
- possible contradictions are solved online or afterwards
- segments are merged if they are homogeneous
  \[ H(R_1 \cup R_2) = \text{TRUE} \]
  \[ |m_1 - m_2| < T \]
- sensitive to the order of operation, i.e. scanning pattern

Watershed segmentation (5.3.4/6.3.4)
- analogous with geographical watersheds and water basins
- edge/border areas assumed to have larger values than inner parts
- “water” is allowed to rise, i.e. the used threshold is increased
- pixels are merged in the basin areas
- too low watersheds are raised with “flood dams”

Region growing post-processing (5.3.5/6.3.5)
- bottom-up segmentation results are seldom optimal as such
- many different heuristic methods can be used
- comparison between output of region growing and detected edges

9.6 Segmentation from template matching (5.4/6.4)
- correlating with a partial image
- different matching criteria:
  \[ C_1(u,v) = \max_{(i,j) \in F} \left| f(i + u, j + v) - h(i, j) \right| \]
  \[ C_2(u,v) = \frac{1}{\sum_{(i,j) \in F} \left| f(i + u, j + v) - h(i, j) \right|} \]
  \[ C_3(u,v) = \frac{1}{\sum_{(i,j) \in F} \left| f(i + u, j + v) - h(i, j) \right|^2} \]
- correlating in the Fourier plane?
- order of matching is important, termination of summation?
- processing on different resolutions
- more precise search around points that match well

10. Shape description
- 3D or 2D shape described
- description for (qualitative) recognition / (quantitative) analysis
- characterizations of the methods
  - input representation form: boundary / area
  - object reconstruction ability
  - incomplete shape description ability
  - mathematical / heuristic techniques
  - statistical / syntactic descriptions
  - invariances to shift, rotation, scaling and resolution changes

10.1 Methods and stages in image analysis (6/8)

10.2 Region identification from pixel labels (6.1/8.1)
- two-pass algorithm
  - if pixel label exists above or left, it is used
  - if label does not exist, new one is assigned
  - if above and left have different labels, regions are marked for combination
  - second pass combines regions that have more than one label
  - can be formed directly from run-length encoding
  - can be formed from quadtree representation

10.3 Boundary-based description (6.2/8.2)
- coordinates for boundary representation: \( x_0, y_0, x_1, y_1 \)
- geometric representations
  - boundary length
  - direction histogram
  - curvature \( \sim \) number of turns

4/8 chain codes, difference code, what is the starting point?
Fourier descriptors (6.2.3/8.2.3)

- Fourier transform of the boundary coordinates
  \[ x(t) = \sum_{n=0}^{N} T_n e^{i \omega t}, \quad t = 2\pi n/L \]
  \[ T_n = \frac{1}{L} \int_0^L x(s) e^{-i 2\pi n s/L} ds \]

- discrete case
  \[ a_n = \frac{1}{L-1} \sum_{m=1}^{L-1} x_m e^{-i 2\pi n m/L} \]
  \[ b_n = \frac{1}{L-1} \sum_{m=1}^{L-1} y_m e^{-i 2\pi n m/L} \]

- rotation invariance
  \[ r_n = (|a_n|^2 + |b_n|^2)^{1/2} \]
- scale invariance
  \[ u_n = a_n/b_n \]
- tangent coordinates

Boundary description with segment sequences (6.2.4/8.2.4)

- polygonal representation by split&merge
- tolerance interval method

3D shape invariants (6.2.7/8.2.7)

- 3D descriptions that are invariant to changes in projection
  - for example: cross ratio of four collinear points
    \[ l = \frac{(x_1-x_3)(x_2-x_4)}{(x_1-x_4)(x_2-x_3)} \]

10.4 Region-based description (6.3/8.3)

- description of the region as a whole or in parts
- skeletons, division of regions
- characteristics of the descriptions:
  - shift and rotation invariant descriptions
  - invariant to small changes in region shapes
  - intuitive techniques
  - many descriptions fit mostly for structural/syntactic recognition

Other contour-based shape descriptions (6.2.6/8.2.6)

- Hough transforms
- moments
- fractal descriptions
- morphological methods
- geometrical correlation function
- shape recognition with neural networks
Simple scalar descriptors (6.3.1/8.3.1)

- area can be calculated from chain code coordinates:
  \[ A = \frac{1}{2} \sum_{i=0}^{N} (y_{i+1} - y_{i+1}) \]

- Euler’s number (Genus, Euler-Poincare) \( \nu = S - N \)
- horizontal and vertical projections, height and width from them
- eccentricity: ratio between the maximum dimension and its perpendicular dimension
- sphericity: \( A/(4\pi r^2) \)
- circularity: maximum of the ratio of the area and surrounding rectangle
- direction can be calculated from moments: \( \theta = \frac{1}{2} \tan^{-1} \frac{2\mu_{10}}{\mu_{01}} \)
- compactness \( r^2/A \)

169

Moments in shape description (6.3.2/8.3.2)

- moments
  \[ m_{pq} = \sum_{i,j} \phi_i \phi_j f(i,j) \]
- central moments
  \[ \rho_{pq} = \sum_{i,j} (i - \mu_{10})(j - \mu_{01}) f(i,j) \]
- scaled central moments
  \[ \eta_{pq} = \frac{\rho_{pq}}{\mu_{10} \mu_{01}} \]
- normalized unscaled central moments
  \[ \tilde{\eta}_{pq} = \frac{\eta_{pq}}{\mu_{00}} \]
- Hu’s moment invariants
  \[ \begin{align*}
  \psi_1 &= \theta_1 + \theta_2 \\
  \psi_2 &= (\theta_1 - \theta_2)^2 + 4\theta_3 \\
  \psi_3 &= (\theta_1 - \theta_2)^2 + (\theta_2 - \theta_3)^2 \\
  \psi_4 &= (\theta_1 + \theta_2 + \theta_3 + \theta_4)^2 \end{align*} \]
- boundary moments from the distance to center of mass: \( m_b = \frac{1}{N} \sum_{i=1}^{N} z(i)^p \)

170

Convex hull of region (6.3.3/8.3.3)

Region concavity tree

171

Region representation with a skeleton (6.3.4/8.3.4)

- skeleton, medial axis transform, thinning
- skeleton extraction by thinning
  - \( H(R) \): inner boundary of region \( R \)
  - \( L(R) \): outer boundary of region \( R \)
  - \( S(R) \): \( R \) & neighbors \( \in H(R) \cup L(R) \)

\[ R_{new} = S(R_{old}) \cup [R_{old} - H(R_{old})] \cup [H(S(R_{old})) \cap R_{old}] \]

172

11. Object recognition

- many machine vision tasks involve object recognition
- structural versus statistical versus soft computing methods

11.1 Knowledge representation (7.1/9.1)

- simple methods for complex data
- descriptions, features
- grammars, languages
- predicate logic
- production rules
- fuzzy logic
- semantic nets
- frames, scripts

176
11.2 Statistical pattern recognition (7.2/9.2)
- classification of quantitative object descriptions
- object classes, classification, classifiers
- classification function, discrimination function
- pattern, pattern space, pattern vector
- feature, feature space, feature vector
- (linear) separability, clustering
- minimum distance principle
- error criterion, optimal Bayes classifier
- training set, validation set, testing set
- probability density estimation methods
- direct optimization / regression methods
- support vector machines
- clustering: K-means, ISODATA

11.3 Neural network classifiers (7.3/9.3)
- supervised / unsupervised learning
- parametric / semi-parametric / non-parametric methods
- prototype-based classifiers, support vector machines
- perceptron, non-linear feed-forward networks
- error back-propagation
- competitive learning, self-organizing map
- recognition as an optimization task, Hopfield net
- hybrid classifiers

11.4 Syntactic pattern recognition (7.4/9.4)
- classification of qualitative object descriptions
- primitives and the relational structure between them
- rules of thumb concerning primitives
  - small number, but enough for appropriate object representation
  - easily segmentable and recognizable
  - should correspond with significant elements of the object
- main groups of grammars:
  - general, context-sensitive, context-free, regular
- nondeterministic, stochastic, fuzzy
- top-down / bottom-up matching
- pruning of the search tree, backtracking
- syntactic classifier learning, grammar inference: enumeration, induction

11.5 Recognition as graph matching (7.5/9.5)
- exact matching of graphs, isomorphism
  - graph-graph
  - graph-subgraph
  - subgraph-subgraph
  - graph partitioning
- non-exact matching
  - similarity measures between two graphs
  - Levenshtein distance between strings
  - deletions, insertions and substitutions

11.6 Optimization techniques (7.6/9.6)
- parameters used for object description need to be optimized
- difficult due to typically non-linear objective functions
  \[ f : D \rightarrow R \quad f_{\text{min}}(x) = \min_{x \in D} f(x) \quad f_{\text{max}}(x) = \max_{x \in D} f(x) \]
- natural and real number parameters
- mutual dependencies between parameters
- iterative optimization methods
- high probability of studding in local extrema points
- genetic algorithms
- simulated annealing

12. Image understanding
- image interpretation, scene analysis
- even humans need practicing
- the highest and most difficult stage of computer vision
- interaction between lower and higher level processing stages needed
- top-down hypotheses, formulation, testing, correction

Topics:
- different control strategies
- active contour models
- pattern recognition in image understanding
- scene labeling and constraint propagation
- semantic segmentation and understanding
12.1 Control strategies (8.1/10.1)
- controlling the interaction between processing stages
- parallel / serial execution
- bottom-up / top-down in data and abstraction hierarchy
- non-hierarchical blackboard / daemon control
- hybrid approaches

12.3 Point distribution models, PDMs (8.3/10.3)
- PDMs can be used for semi-parametric shape representation
- set of $M$ similar training shapes
- $N$ landmark points extracted from boundary of each training shape
- each boundary produces a $2N$-dimensional point distribution vector $x = (x_1, y_1, x_2, y_2, \ldots, x_N, y_N)^T$
- point distribution vector can be translated, scaled and rotated
  $T_{\delta t_{\theta_{x_y}}}(x) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix}$
- point distribution vector $x$ aligned with model $x^*$ minimizing
  $\min_{\delta t_{\theta_{x_y}}} E = ||x - T_{\delta t_{\theta_{x_y}}}(x^*)||$

12.4 Principal component analysis, PCA (8.3/3.2/10)
- Hotelling / Karhunen-Loève transform, KLT
- PCA can be used for fitting point distribution models
- dimensionality reduction for a high-dimensional data set
- eigenvectors of the data’s covariance matrix used in linear transform
- linear transform is as $y = A(x - m_x)$
- in PCA, rows of the transform matrix $A$ the eigenvectors $e_{ij}$ of $C_x$
  according to the eigenvalue $C_{x e_{ij}} = \lambda e_{ij}$
- $C_x$ is the data set’s covariance matrix and $m_x$ is its mean
  $C_x = E((x - m_x)(x - m_x)^T)$
- inverse transform is as $x = A^T y + m_x$
- squared reconstruction error $E(||x^* - x||^2)$ is minimized by PCA

12.5 Example: metacarpal bones, PCA+PDM (8.3/3.2/10)

12.6 Pattern recognition in image understanding (8.4/10.5)
- formation of a statistical feature vector for each pixel
- pixel classification / clustering
- utilization of context information
  - noise reduction, e.g. by median filtering
  - second classification of each pixel and its neighborhood
  - merging of homogeneous regions before classification
  - feature extraction from pixel neighborhoods
  - combination of spectral and spatial information
12.7 Scene labelling and constraint propagation (8.5/10.7)
- aiming at consistent interpretation of the image
- discrete / probabilistic labeling
- regions, attributes, relations
- regions $R_i, i = 1, \cdots, N$, labels $\Omega = \{\omega_1, \cdots, \omega_R\}$
- moving from local constraints to image level
- relaxation in constraint propagation
- discrete relaxation
  - attributes are discrete Boolean values: $\text{is} / \text{is not}$
  - first all regions are given all labels
  - impossible labels are removed one by one

Discrete relaxation: example (8.5.1/10.7.1)

a. window (W) is rectangular
b. table (T) is rectangular
c. drawer (D) is rectangular
d. phone (P) is above table
e. drawer is inside table
f. background (B) is adjacent to the border

193

Probabilistic relaxation (8.5.2/10.7.2)
- produces always some solution
- support for label $\omega_k$ in region $O_i$ at iteration step $k$
  \[ Q^k(O_i = \omega_k) = \sum_{j=1}^{R} c_j q_j^k(O_j = \omega_k) \]
  \[ = \sum_{j=1}^{R} c_j \frac{P(O_j = \omega_k | X_j)}{P(O_j = \omega_k) \prod_{j=1}^{R} O_j = \omega_k} \]
- linear relaxation
  \[ P^k(O_i = \omega_k) = P(O_i = \omega_k | X_i) \]
  \[ P^{k+1}(O_i = \omega_k) = P^k(O_i = \omega_k) \prod_{j=1}^{R} O_j = \omega_k \]
- non-linear relaxation
  \[ P^{k+1}(O_i = \omega_k) = \frac{1}{K} P^k(O_i = \omega_k) q^k(O_i = \omega_k) \]
  \[ K = \sum_{i=1}^{R} P^k(O_i = \omega_k) q^k(O_i = \omega_k) \]

Relaxation as optimization problem

Maximization $F$:

\[ F = \sum_{i=1}^{R} \sum_{j=1}^{N} P(O_i = \omega_j) \sum_{j=1}^{R} c_j \frac{P(O_j = \omega_k | X_j)}{P(O_j = \omega_k) \prod_{j=1}^{R} O_j = \omega_k} \]

\[ \sum_{i=1}^{R} P(O_i = \omega_k) = 1 \quad \forall i \quad P(O_i = \omega_k) > 0 \quad \forall i, k \]

Image interpretation as tree search (8.5.3/10.7.3)
- number of image regions = number of layers in search tree
- leaves of the tree correspond to different full image labelings

195

12.8 Semantic image segmentation (8.6/10.8)
- region adjacency graph and its dual
- iterative updating of data structures
- semantic region growing
- merging of adjacent regions
- aiming at maximizing objective function $F$
- always the most probable interpretation is fixed

197

13. Motion analysis
- a collection of diverse problem settings and algorithms
  - detection of motion
  - detection of moving object
  - extraction of 3D properties of the object
- assumptions concerning the object's motion
  - the maximal speed is known
  - the maximal acceleration is small
  - the motion is uniform / the object is rigid
  - mutual correspondence between reference points

198

13.1 Differential motion analysis methods (15.1/16.1)
- difference image
  \[ d(i,j) = \begin{cases} 
0 & \text{if } |f_1(i,j) - f_2(i,j)| \leq \epsilon \\
1 & \text{otherwise} 
\end{cases} \]
- object-background, object-another object, object-object, noise
- cumulative difference image
  \[ d_{cum}(i,j) = \sum_{k=1}^{n} \omega_k |f_1(i,j) - f_2(i,j)| \]
- static reference image and its composition from pieces

199

13.2 Optical flow (15.2/16.2)
- it is assumed that
  - each point's illumination is constant
  - neighboring points have similar gray values
- modeling $f(.)$ by using Taylor's series
  \[ f(x + dx, y + dy, t + dt) = f(x,y,t) + f_x dx + f_y dy + f_t dt + O(P) \]
- locating matching image areas with different $t$
  \[ f(x + dx, y + dy, t + dt) = f(x,y,t) \Rightarrow -f_t = f_x dx + f_y dy \]
we aim at solving the speed vector for each pixel
\[ c = (x, y) = (u, v) \quad \Rightarrow \quad -f_1 = f_2 u + f_3 v = c \nabla f \]

- smoothness conditions incorporated with Lagrange coefficient λ
\[ E^2(x, y) = (f_2 u + f_3 v + f_1^2 \lambda + (c_1^2 + c_2^2 + c_3^2) \]

- solution
\[ u = \tilde{u} - f_2 P \frac{u}{D x}, \quad v = \tilde{v} - f_3 P \frac{v}{D x}, \quad D = \lambda^2 + f_2^2 + f_3^2 \]

- relaxation with Gauss-Seidel iteration
\[ u^{i+1}(i, j) = u^i(i, j) - \frac{f_2(i, j)}{D(i, j)} \frac{P_{i+1}}{D(i, j)} \]
\[ v^{i+1}(i, j) = v^i(i, j) - \frac{f_3(i, j)}{D(i, j)} \frac{P_{i+1}}{D(i, j)} \]

### 13.3 Optical Flow in Motion Analysis (15.2.4/16.2.4)

- four elementary movement types
  - translation at constant distance
  - translation in depth: approaching / drawing away
  - rotation with axes aligned with view axis
  - rotation with axes perpendicular to view axis

- perspective image \((x', y') = \left( \frac{x d f}{D x}, \frac{y d f}{D y} \right)\)
- focus of expansion, FOE: \(x_{\text{FOE}} = \left( \frac{d}{D x}, \frac{d}{D y} \right)\)
- \(D(i, j) = \text{2D distance from the FOE in the image plane}\)
- speed in the image plane \(V(t) = \frac{d D}{d t}\)
\[ \frac{D}{V} = \frac{z(t)}{u(t)} \]
- z distance can be solved for any pixel
\[ z_2(i) = \frac{z_1(i)}{\text{FOE}(i) D(i)} \]
- for all points it holds that
\[ x(t) = \frac{x(i) u(t) D(i)}{V(t)} \quad \text{and} \quad y(t) = \frac{y(i) v(t) D(i)}{V(t)} \]

### 13.4 Correspondence of Interest Points (15.3/16.3)

- interest points are detected and traced in video frames
- a priori knowledge about the maximal speed
- sparse field of speed vectors is formed
- selection of interest points
  - specular pixels: edges, corners
  - eg., Moravec detector
  - or Zuniga-Hartley/Kitchen-Rosenfeld detector
- matching of the interest points
  - first row: matching \(x_0 \rightarrow y_0\)
  - each point pair has probability of match \(P_{\text{match}}\)
  - consistency between the closest neighbor pairs, relaxation
- explicit markers used eg., in crash test dummies
- 3D dynamical programming can be applied in matching

### 14. EXAM GUIDE

#### Second Edition book
The importance of the Second Edition book's chapters in Spring 2008's teaching and exam:
- Chapters 1-6 belong to the course's central content.
  - Section 5.5 was not treated
  - Section 6.2 was not treated in detail
- Chapter 7 belongs to pattern recognition and neural networks course. The most important sections for Computer Vision course are 7.1, 7.4 and 7.5.
  - Section 7.5.1’s algorithms were not treated
- Chapter 8 is central content.
  - Section 8.6.2 was not treated
  - Section 8.7 was not treated

- Chapters 9-10 were treated superficially compared to the amount of text in the book. Lectures slides have references to book sections and give a hint what parts were treated and which were not.
- Chapter 11 is central content.
- Chapters 12-13 belong to digital image processing course and are not included in Computer Vision course's exam.
- Chapters 14-15 are central content.
  - Section 14.6 was not treated
  - Sections 15.3.3-15.4.1 were not treated
  - Chapter 16 was not treated

You may have in the exam a pen, paper and a calculator capable for trigonometric and logarithmic calculations. No tablet nor formula books are needed.

#### Third Edition book
The importance of the Third Edition book's chapters in Spring 2008's teaching and exam:
- Chapters 1-6 belong to the course's central content.
  - Sections 2.4-2.5, 3.2.5-3.2.9, 3.4.3, 5.3.11, 6.5 were not treated
- Chapter 7 presents material mostly beyond the course requirements.
  - Only sections 7.2-7.2.1 were treated
- Chapter 8 belongs to the course's central content.
  - Section 8.2.7 was not treated in detail
- Chapter 9 belongs to pattern recognition and neural networks course.
  - The most important sections for Computer Vision course are 9.1, 9.4 and 9.5.
  - Section 9.5.1's algorithms were not treated

- Chapter 10 is central content.
  - Sections 10.2, 10.4, 10.6, 10.8.2, 10.9-10.10 were not treated
- Chapters 11-12 were treated superficially compared to the amount of text in the book. Lectures slides have references to book sections and give a hint what parts were treated and which were not.
- Chapter 13 is central content.
- Chapter 14 belongs to digital image processing course and is not included in Computer Vision course's exam.
- Chapters 15-16 are central content.
  - Sections 15.1.6-15.1.8, 16.4-16.6 were not treated

You may have in the exam a pen, paper and a calculator capable for trigonometric and logarithmic calculations. No tablet nor formula books are needed.