Basics and Hands-on of Computer Vision

Intensive course at University of Helsinki – day 3

Jorma Laaksonen

Aalto University
Department of Computer Science
Espoo

16.5.2018

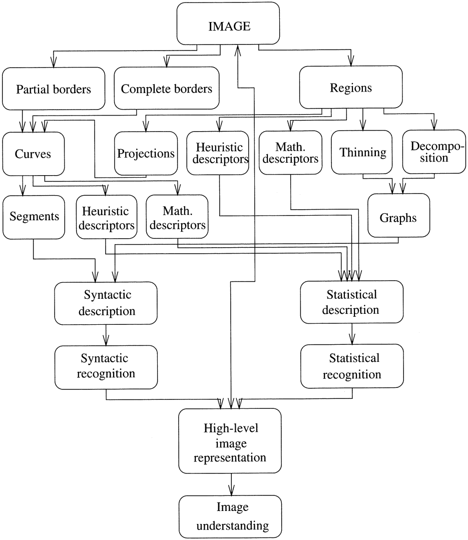
Contents

Contents

12. 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

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

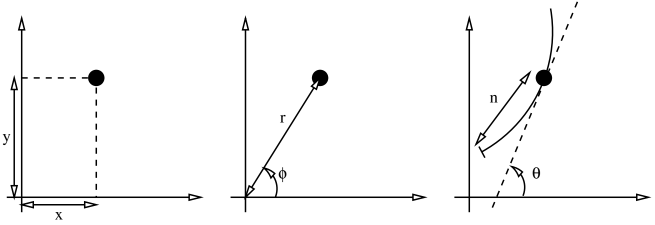


12.2 Region identification from pixel labels (8.1/6.1)

- 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-lenght encoding
- can be formed from quadtree representation

12.3 Boundary-based description (8.2/6.2)

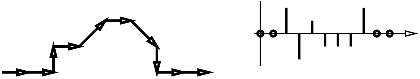
- coordinates for boundary representation: xy, $r\phi$ tai $n\theta$



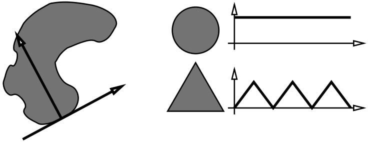
- 4/8 chain codes, difference code, what is the starting point?

- geometric representations
  - boundary length
  - direction histogram
  - curvature $\sim$ number of turns

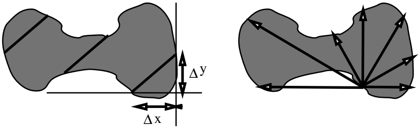
– bending energy $BE = \frac{1}{L} \sum_{k=1}^{L} c^2(k)$



– signature, normal distance to opposite border point



– choird distribution



Fourier descriptors (8.2.3/6.2.3)

- Fourier transform of the boundary coordinates

$$z(t) = \sum_n T_n e^{int} \quad t = 2\pi s/L$$

$$T_n = \frac{1}{L}\int_0^L z(s)e^{-i(2\pi/L)ns}\, ds$$

- discrete case

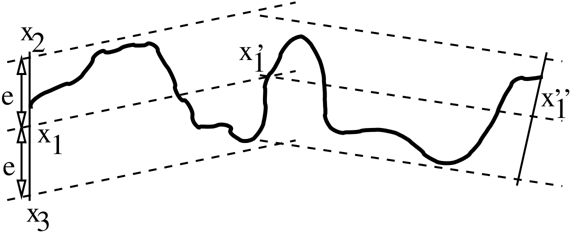
$$a_n = \frac{1}{L-1}\sum_{m=1}^{L-1} x_m e^{-i(2\pi/(L-1))nm}$$

$$b_n = \frac{1}{L-1}\sum_{m=1}^{L-1} y_m e^{-i(2\pi/(L-1))nm}$$

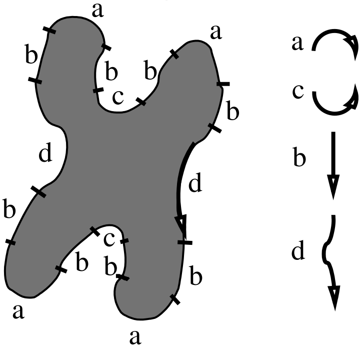
- rotation invariance $\quad r_n = (|a_n|^2 + |b_n|^2)^{1/2}$
- scale invariance $\quad w_n = r_n/r_1$
- tangent coordinates

Boundary description with segment sequences (8.2.4/6.2.4)

- polygonal representation by split&merge
- tolerance interval method



- recursive boundary splitting
- division in constant curvature pieces



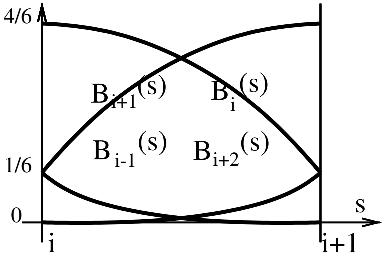
- scale-space methods
- curvature primal sketch

B-spline representation (8.2.5/6.2.5)

- piecewise polynomial curves

$$\mathbf{x}(s) = \sum_{i=0}^{n} \mathbf{v}_i B_i(s)$$

- most often 3rd degree polynomials



$$C_0(t) = \frac{t^3}{6}$$

$$C_1(t) = \frac{-3t^3 + 3t^2 + 3t + 1}{6}$$

$$C_2(t) = \frac{3t^3 - 6t^2 + 4}{6}$$

$$C_3(t) = \frac{-t^3 + 3t^2 - 3t + 1}{6}$$

12.4 Region-based description (8.3/6.3)

- 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

Simple scalar descriptors (8.3.1/6.3.1)

- area can be calculated from chain code coordinates:

$$A = \frac{1}{2} \left| \sum_{k=0}^{n-1} (i_k j_{k+1} - i_{k+1} j_k) \right|$$

- Euler's number (Genus, Euler-Poincaré) $\nu = S - N$
- horizontal and vertical projections, height and width from them
- eccentricity: ratio between the maximum dimension and its perpendicular dimension
- elongatedness: $A/(2d)^2$
- rectangularity: maximum of the ratio of the area and surrounding rectangle
- direction can be calculated from moments: $\theta = \frac{1}{2} \tan^{-1} \frac{2\mu_{11}}{\mu_{20} - \mu_{02}}$
- compactness: l^2/A

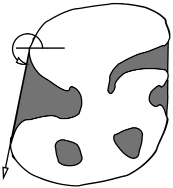
Moments in shape description (8.3.2/6.3.2)

- moments $$m_{pq} = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} i^p j^q f(i,j)$$
- central moments $$\mu_{pq} = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} (i - \frac{m_{10}}{m_{00}})^p (j - \frac{m_{01}}{m_{00}})^q f(i,j)$$
- scaled central moments $$\eta_{pq} = \frac{\mu'_{pq}}{\mu'_{00}{}^{\frac{p+q}{2}+1}} \quad \mu'_{pq} = \frac{\mu_{pq}}{\alpha^{p+q+2}}$$
- normalized un-scaled central moments $$\vartheta_{pq} = \frac{\mu_{pq}}{\mu_{00}{}^{\frac{p+q}{2}+1}}$$
- Hu's moment invariants

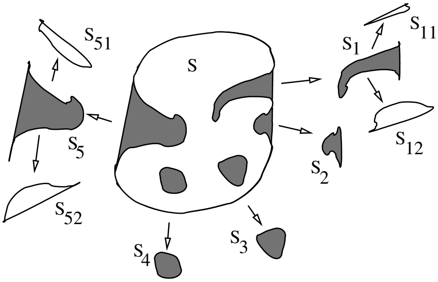
$$\varphi_1 = \vartheta_{20} + \vartheta_{02}$$
$$\varphi_2 = (\vartheta_{20} - \vartheta_{02})^2 + 4\vartheta_{11}^2$$
$$\varphi_3 = (\vartheta_{30} - 3\vartheta_{12})^2 + (3\vartheta_{21} - \vartheta_{03})^2$$
$$\varphi_4 = (\vartheta_{30} + \vartheta_{12})^2 + (\vartheta_{21} + \vartheta_{03})^2$$

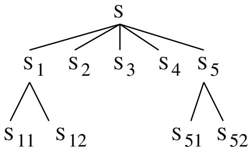
- boundary moments from the distance to center of mass: $m_r = \frac{1}{N}\sum_{i=1}^{N} z(i)^r$

Convex hull of region (8.3.3/6.3.3)



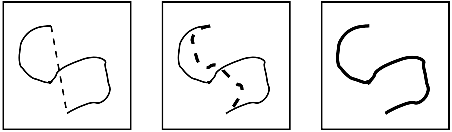
Region concavity tree





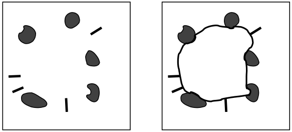
14.2 Active contour models aka snakes (7.2/8.2)

- minimization of the spline model's energy, iterative search

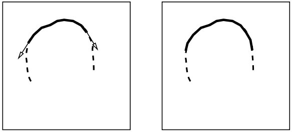


- total energy $E^*_{\text{snake}} = \int_0^1 E_{\text{snake}}(\mathbf{v}(s))ds$
- internal energy $E_{\text{int}} = \alpha(s)|\frac{d\mathbf{v}}{ds}|^2 + \beta(s)|\frac{d^2\mathbf{v}}{ds^2}|^2$
- image energy $E_{\text{image}} = w_{\text{line}}E_{\text{line}} + w_{\text{edge}}E_{\text{edge}} + w_{\text{term}}E_{\text{term}}$
- line energy $E_{\text{line}} = f(x, y)$
- edge energy $E_{\text{edge}} = -|\nabla f(x, y)|^2$
- termination energy $E_{\text{term}} = \frac{\partial\psi}{\partial\mathbf{n}_R} = \frac{\partial^2 g/\partial\mathbf{n}_R^2}{\partial g/\partial\mathbf{n}}$
- boundary conditions $E_{\text{con}}(\mathbf{v}(s))$

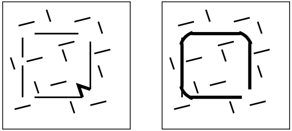
- stabilization



- snake stretching and fitting



- inflating balloon



14.3 Point distribution models, PDMs (10.3/8.3)

- 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

$$\mathbf{x} = (x_1, y_1, x_2, y_2, \ldots, x_N, y_N)^T$$

- point distribution vector can be translated, scaled and rotated

$$\mathcal{T}_{s,\theta,t_x,t_y}(\mathbf{x}) = s \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}$$

- point distribution vector $\mathbf{x}^2$ aligned with model $\mathbf{x}^1$ minimizing

$$\min_{s,\theta,t_x,t_y} E = \|\mathbf{x}^1 - \mathcal{T}_{s,\theta,t_x,t_y}(\mathbf{x}^2)\|$$

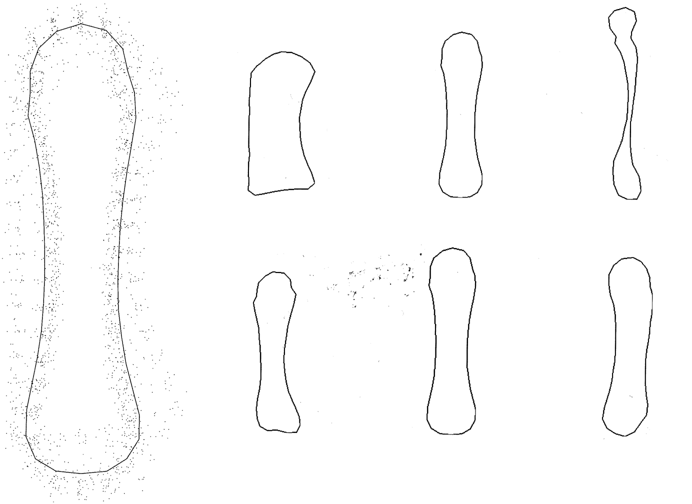
14.4 Principal component analysis, PCA (3.2.10/8.3)

- 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 $\mathbf{y} = \mathbf{A}(\mathbf{x} - \mathbf{m}_\mathbf{x})$
- in PCA, rows of the transform matrix $\mathbf{A}$ are eigenvectors $\mathbf{e}_{\mathbf{x},i}$ of $\mathbf{C}_\mathbf{x}$
- according to the eigenequation $\mathbf{C}_\mathbf{x}\mathbf{e}_{\mathbf{x},i} = \lambda_{\mathbf{x},i}\mathbf{e}_{\mathbf{x},i}$
- $\mathbf{C}_\mathbf{x}$ is $\mathbf{x}$ data set's covariance matrix and $\mathbf{m}_\mathbf{x}$ is its mean

$$\mathbf{C}_\mathbf{x} = E\{(\mathbf{x} - \mathbf{m}_\mathbf{x})(\mathbf{x} - \mathbf{m}_\mathbf{x})^T\}$$

- inverse transform is as $\mathbf{x}' = \mathbf{A}^T\mathbf{y} + \mathbf{m}_\mathbf{x}$
- squared reconstruction error $E\{\|\mathbf{x}' - \mathbf{x}\|^2\}$ is minimized by PCA

14.5 Example: metacarpal bones, PCA+PDM (3.2.10/8.3)



Contents

9.7 Skeletons and maximal ball

Homotopic transforms (13.5.1/11.5.1)

- don't change topological relations
- homotopic tree, that shows neighborhood relations, remains the same

Skeletons (13.5.2/11.5.2)

- medial axis transform
- grassfire metaphora
- 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

$$nB = \underbrace{B \oplus \cdots \oplus B}_{n \text{ times}}$$

- ball $B(p,r)$, shape B located in p with radius r, is maximal if
  - $B(p,r) \subseteq X$ and
  - there cannot be a larger ball B' so that $B(p,r) \subset B' \subseteq X$
  - for all B' it holds $B \subseteq B' \subseteq X \Longrightarrow B' = B$
- skeleton by maximal balls:

$$S(X) = \{p \in X : \exists r \geq 0, B(p,r) \text{ is } X\text{'s maximal ball}\}$$

$$S(X) = \cup_{n=0}^{\infty}\big((X \ominus nB) \setminus (X \ominus nB) \circ B\big)$$

9.8 Hit-or-miss $\otimes$, thinning $\oslash$, thickening $\odot$ (13.3.3,13.5.3)

- composite structuring element is an ordered pair $B = (B_1, B_2)$
- $X \otimes B = \{x \ : \ B_1 \subset X \text{ and } B_2 \subset X^C\}$
- $X \otimes B = (X \ominus B_1) \cap (X^C \ominus B_2) = (X \ominus B_1) \setminus (X \oplus \breve{B}_2)$
- $X \oslash B = X \setminus (X \otimes B)$
- $X \odot B = X \cup (X \otimes B)$
- thinning and thickening are dual transformations

$$(X \odot B)^C = X^C \oslash B^*, \quad B^* = (B_2, B_1)$$

- sequential thinnings / thickenings with Golay alphabets

$$X \oslash \{B_{(i)}\} = ((((X \oslash B_{(1)}) \oslash B_{(2)}) \cdots \oslash B_{(i)}) \cdots)$$
$$X \odot \{B_{(i)}\} = ((((X \odot B_{(1)}) \odot B_{(2)}) \cdots \odot B_{(i)}) \cdots)$$

- homotopic skeleton is ready when thinning is idempotentic

9.9 Golay alphabets

- thinning with L element (4-neighbors)

$$L_{(1)} = \begin{bmatrix} 0 & 0 & 0 \\ * & 1 & * \\ 1 & 1 & 1 \end{bmatrix}, \quad L_{(2)} = \begin{bmatrix} * & 0 & * \\ 1 & 1 & 0 \\ 1 & 1 & * \end{bmatrix}, \cdots$$

- thinning with E element (4-neighbors)

$$E_{(1)} = \begin{bmatrix} * & * & * \\ 0 & 1 & 0 \\ * & 0 & * \end{bmatrix}, \quad E_{(2)} = \begin{bmatrix} * & 0 & * \\ 0 & 1 & * \\ * & 0 & * \end{bmatrix}, \cdots$$

- thinning with M element (4-neighbors)

$$M_{(1)} = \begin{bmatrix} * & 0 & * \\ * & 1 & * \\ 1 & 1 & 1 \end{bmatrix}, \quad M_{(2)} = \begin{bmatrix} * & 0 & * \\ 1 & 1 & 0 \\ 1 & 1 & * \end{bmatrix}, \cdots$$

- thinning with D and thickening with D^t element (4-neighbors)

$$D_{(1)} = \begin{bmatrix} * & 0 & * \\ 0 & 1 & 1 \\ * & 0 & * \end{bmatrix}, \quad D_{(2)} = \begin{bmatrix} 0 & 0 & * \\ 0 & 1 & 1 \\ * & 1 & 1 \end{bmatrix}, \cdots$$

- thickening with C element (4-neighbors)

$$C_{(1)} = \begin{bmatrix} 1 & 1 & * \\ 1 & 0 & * \\ * & * & * \end{bmatrix}, \quad C_{(2)} = \begin{bmatrix} * & 1 & 1 \\ * & 0 & 1 \\ * & * & * \end{bmatrix}, \cdots$$

9.10 Quench function and ultimate erosion (13.5.4/11.5.4)

- quench function $q_X(p)$:

$$X = \cup_{p \in S(X)} (p + q_X(p)B)$$

- $q_X(p)$'s regional maxima points = ultimate erosion Ult(X)
- ultimate erosion can be used to extract markers in objects
- original object can be reconstructed from markers
- market set $B \subseteq A \Longrightarrow$ reconstruction $\rho_A(B)$
- ultimate erosion can be expressed as

$$\text{Ult}(X) = \cup_{n \in \mathcal{N}} \big((X \ominus nB) \setminus \rho_{X \ominus nB}(X \ominus (n+1)B)\big)$$

9.11 Ultimate erosion and distance functions (11.5.5/13.5.5)

- distance function $\text{dist}_X(p)$ is p's distance from X^C:

$$\forall p \in X \quad \text{dist}_X(p) = \min\{n \in \mathcal{N}, p \notin (X \ominus nB)\}$$

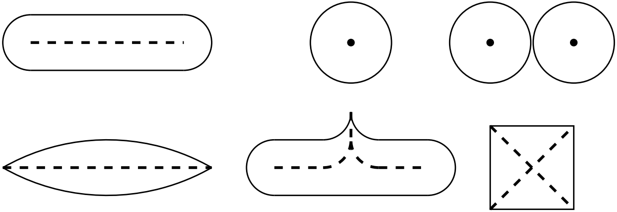
- ultimate erosion is the set of $\text{dist}_X(p)$'s regional maxima points
- maximal ball skeleton is the set of $\text{dist}_X(p)$'s local maxima points
- each connected component X_i of set X has an influence zone

$$Z(X_i) = \{p \in \mathcal{Z}^2,\ \forall i \neq j,\ d(p, X_i) \leq d(p, X_j)\}$$

- skeleton by influence zones (SKIZ) is the set of boundary pixels of the influence zones $\{Z(X_i)\}$

Skeleton construction from medial axis

- medial axis: same minimum distance to at least two boundaries



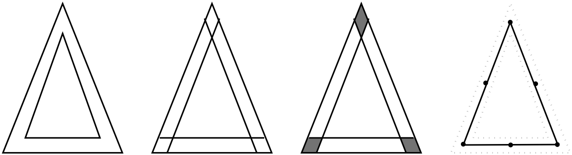
- distance stored in the skeleton pixel

Region graph construction

- pixel types: end points, node points, normal points
- end and node points $\longrightarrow$ graph nodes
- normal points $\longrightarrow$ graph arcs

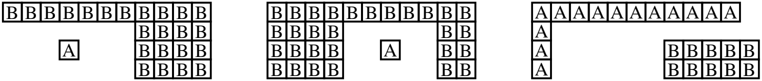
Region decomposition into shape primitives (8.3.5/6.3.5)

- region is segmented in primary convex sub-regions or kernels
- mutual relations of the sub-regions are described with a graph
- each graph node contains the following information:
  - type of the node (primary sub-regions or kernel)
  - number of vertices
  - area
  - main axis direction
  - center of gravity



Region neighborhood graph (8.3.6/6.3.6)

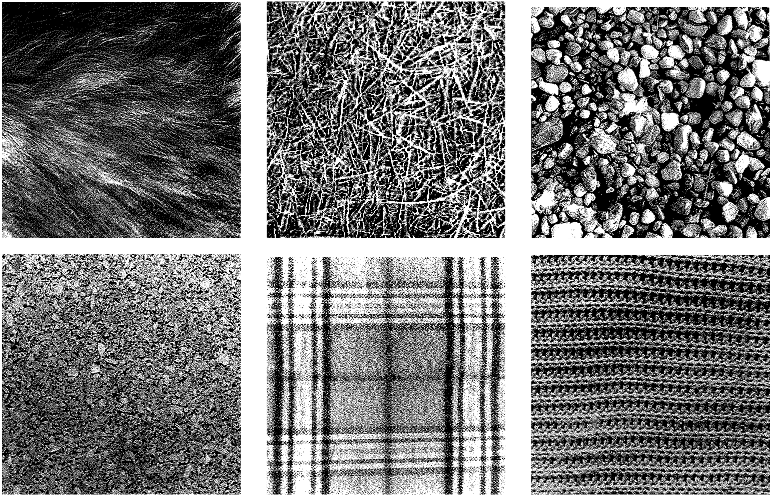
- representation of the relations of sub-regions of a region (or image)
- sub-regions don't need to be adjacent
- expressions for spatial relations:
  - to the left/right of, above/below
  - close to, between
- examples of definitions of "A is to the left of B"
  - all A_i are to the left of all B_i
  - at least one A_i is to the left of some B_i
  - A's center of gravity is to the left of that of B
  - previous AND A's rightmost pixel is to the left of B's rightmost pixel



Contents

10. Texture

Some examples of textured real-world surface images:



10.1 Properties of natural textures

- surface shape / surface structure / surface image
- physical origin is very often 3-dimensional
- texture analysis uses 2-dimensional images
- effect of lighting and light directions?
- direction/orientation of the texture or is it unoriented?
- texture primitives / texture elements, **texels**
- 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, **textons**

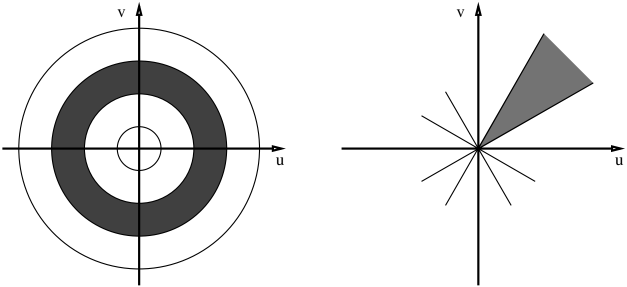
10.2 Statistical texture descriptions (15.1.1,14.1.1)

- 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

$$C_{ff}(p,q) = \frac{MN \sum_{i=1}^{M-p} \sum_{j=1}^{N-q} f(i,j) f(i+p, j+q)}{(M-p)(N-q) \sum_{i=1}^{M} \sum_{j=1}^{N} f^2(i,j)}$$

$$C_{ff}(r) = C_{ff}(p,q), \quad r^2 = p^2 + q^2$$

- optical Fourier transform
- discrete Fourier or Hadamard transform
- partitioning of Fourier spectrum for feature calculation



- for example, 28 spatial frequency-domain features

10.3 Co-occurrence matrices (15.1.2,14.1.2)

- 2-dimensional generalizations of 1-dimensional histograms
- second order statistics of two nearby pixel values
- parameters: distance d, angle ϕ
- symmetric / asymmetric definition

$$
\begin{aligned}
P_{0^\circ,d}(a,b) =& |\{[(k,l),(m,n)] \in D : \\
& k-m=0, |l-n|=d, f(k,l)=a, f(m,n)=b\}| \\
P_{45^\circ,d}(a,b) =& |\{[(k,l),(m,n)] \in D : \\
& (k-m=d, l-n=-d) \vee (k-m=-d, l-n=d), \\
& f(k,l)=a, f(m,n)=b\}| \\
P_{90^\circ,d}(a,b) =& |\{[(k,l),(m,n)] \in D : \\
& |k-m|=d, l-n=0, f(k,l)=a, f(m,n)=b\}| \\
P_{135^\circ,d}(a,b) =& |\{[(k,l),(m,n)] \in D : \\
& (k-m=d, l-n=d) \vee (k-m=-d, l-n=-d), \\
& f(k,l)=a, f(m,n)=b\}|
\end{aligned}
$$

10.4 Co-occurrence matrices – an example

Gray-scale image, 4 intensity levels:

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

Co-occurrence matrices:

$$P_{0^\circ,1} = \begin{pmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix} \qquad P_{135^\circ,1} = \begin{pmatrix} 2 & 1 & 3 & 0 \\ 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 2 \\ 0 & 0 & 2 & 0 \end{pmatrix}$$

10.5 Haralick features from co-occurrence matrix

- energy $\sum_{a,b} P^2_{\phi,d}(a,b)$
- entropy $\sum_{a,b} P_{\phi,d}(a,b) \log P_{\phi,d}(a,b)$
- maximum probability $\max_{a,b} P_{\phi,d}(a,b)$
- contrast $\sum_{a,b} |a-b|^\kappa P^\lambda_{\phi,d}(a,b)$
- inverse difference moment $\sum_{a,b;a\neq b} \frac{P^\lambda_{\phi,d}(a,b)}{|a-b|^\kappa}$
- correlation $\frac{\sum_{a,b}[abP_{\phi,d}(a,b)]-\mu_x\mu_y}{\sigma_x\sigma_y}$

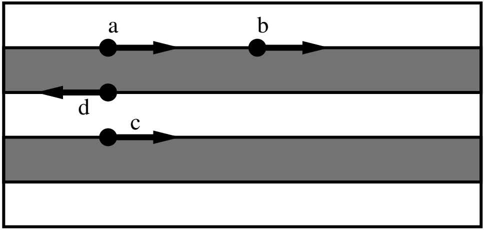
10.6 Edge frequency (15.1.3,14.1.3)

- average gradient magnitude can be calculated with varying scale d:

$$
\begin{aligned}
g(d) = & |f(i,j) - f(i+d,j)| + |f(i,j) - f(i-d,j)| + \\
& |f(i,j) - f(i,j+d)| + |f(i,j) - f(i,j-d)|
\end{aligned}
$$

- compare with autocorrelation function: minima $\rightleftharpoons$ maxima
- first and second order edge statistics can be characterized:
  - coarseness: finer texture $\sim$ higher number of edge pixels
  - contrast: higher contrast $\sim$ 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



10.8 Laws' texture energy measures (15.1.5,14.1.5)

- Laws' texture energy masks can measure
  - grayvalues
  - edges
  - spots
  - waves
- three one-dimensional masks:

$$L_3 = (1, 2, 1), \quad E_3 = (-1, 0, 1), \quad S_3 = (-1, 2, -1)$$

- their one-dimensional convolutions:

$$\begin{aligned} L_3 * L_3 &= L_5 = (1, 4, 6, 4, 1) \\ L_3 * E_3 &= E_5 = (-1, -2, 0, 2, 1) \\ L_3 * S_3 &= S_5 = (-1, 0, 2, 0, -1) \\ S_3 * S_3 &= R_5 = (1, -4, 6, -4, 1) \\ E_3 * S_3 &= W_5 = (1, -2, 0, 2, -1) \end{aligned}$$

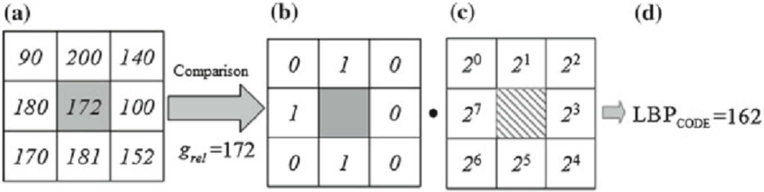
- two-dimensional outer-products of the one-dimensional masks, eg.:

$$L_5^T \times S_5 = \begin{bmatrix} -1 & 0 & 2 & 0 & -1 \\ -4 & 0 & 8 & 0 & -4 \\ -6 & 0 & 12 & 0 & -6 \\ -4 & 0 & 8 & 0 & -4 \\ -1 & 0 & 2 & 0 & -1 \end{bmatrix}$$

- energy (squared sum) of the response is calculated after convolution
- 25 masks can be used to create 25-dimensional feature vector

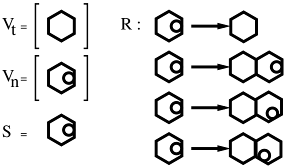
Local Binary Patterns

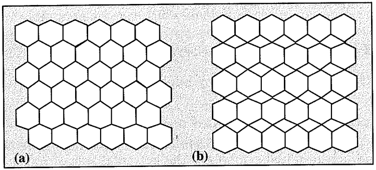
- Local Binary Patterns (LBP) is an efficient texture feature
- used also for face recognition and other computer vision tasks
- each pixel's binarized 8-neighborhood is used as its LBP code
- local binarization based on the pixel's own value
- creates 256-dimensional histogram
- invariant to illumination changes like shadows



10.10 Syntactic texture descriptions (15.2.1,14.2.1)

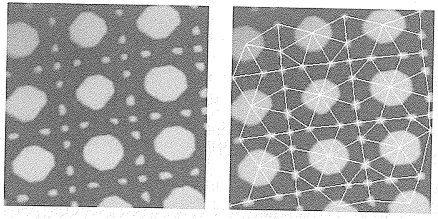
- description of a surface with a set of texture primitives and rules
- real-world textures are non-deterministic
- shape chain grammars
  - texture synthesis
  - terminal symbols V_t
  - non-terminal symbols V_n
  - start symbol S
  - set of rules R





10.11 Graph grammars (15.2.2,14.2.2)

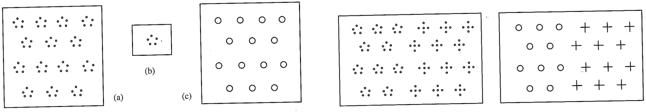
- comparison between 2D texture primitive graphs
- recognition of a set of visual primitives
- thresholding of distances between texture primitives
- formation of a graph describing the texture
- comparison between graph of input image and stored grammar models
    1) 1D chains of the graph compared with the grammar
    2) stochastic grammar of graphs
    3) direct graph comparison



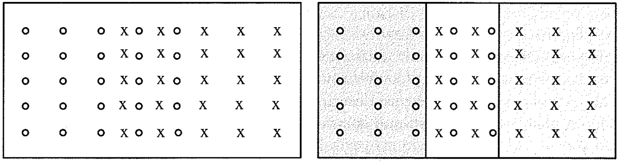
17

10.12 Primitive grouping and hierarchical textures (15/14.2.3

- many textures are in fact hierarchical
- can be studied in different scales
- bottom-up texture primitive grouping



- detection of homogeneous texture regions



10.13 Hybrid texture description methods (15.3,14.3)

- 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

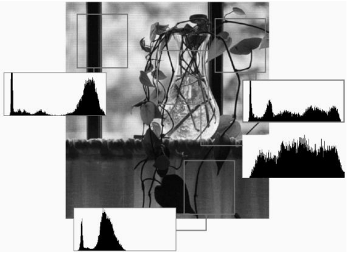
Contents

11. Wavelets and multiresolution processing

Wavelets are sometimes a better method for image analysis than the Fourier transform. While the Fourier transform is a global method for the whole image, wavelets act more locally and reveal the location of structures that can be found in the image.

One application area of wavelets is *multiresolution processing* of images, where one image is studied in varying sizes.

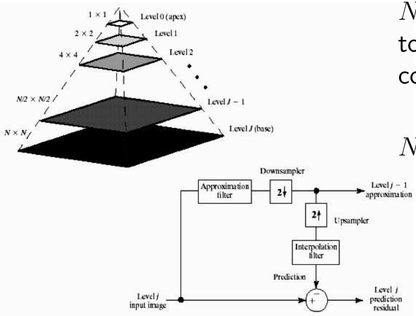
11.1 Image pyramids and subband coding (7.1)



In the following examples this image will be used. In the different areas of the image there are very different intensity distributions and frequency contents.

Image pyramids (7.1.1)

Image pyramids are a classical approach to multiresolution processing. The original image is typically lowpass filtered and downsampled (decimated) so that four original pixels are used to create one new pixel in the lower resolution image.

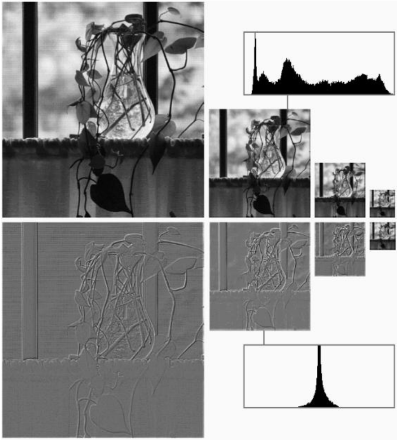


N^2-sized ($N = 2^J$) original image leads to an image pyramid with $P+1$ levels containing the total number of pixels:

$$N^2\left(1+\frac{1}{4}+\frac{1}{4^2}+\cdots+\frac{1}{4^P}\right) \le \frac{4}{3}N^2$$

An example of an image pyramid (7.1.1)

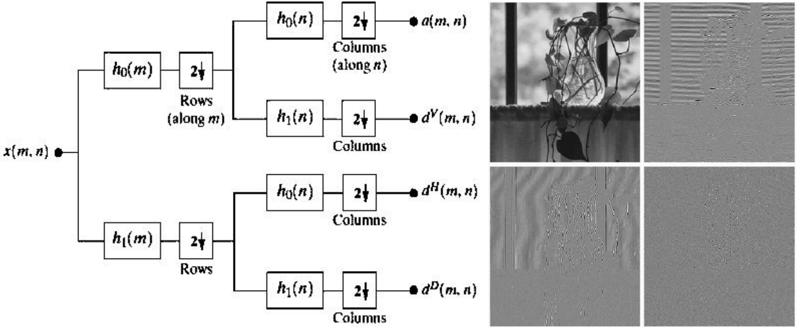
Approximation pyramid (Gaussian lowpass, downsampling)



Prediction residual pyramid (upsampling, interpolation, difference, Laplacian)

Two-dimensional subband coding (7.1.2)

The original discrete image $x(m,n)$ can be similarly decomposed first by rows and then by columns to one 2-D *approximation subband* $a(m,n)$ and one *vertical*, one *horizontal* and one *diagonal detail subbands* $d^V(m,n)$, $d^H(m,n)$ and $d^D(m,n)$:



An example of discrete one-dimensional wavelet transform (7.3.2)

Let's transform the series $f(0) = 1$, $f(1) = 4$, $f(2) = -3$, $f(3) = 0$ with DWT by using Haar's scaling - and wavelet functions. We have now thus $M = 4$ and $J = 2$ and we can choose $j_0 = 0$ so that we will solve for the (j, k) pairs $(0, 0)$, $(1, 0)$ and $(1, 1)$.

By sampling $\varphi_{j_0,k}(x)$ and $\psi_{j,k}(x)$ we get the *Haar transform matrix*:

$$\mathbf{H}_4 = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

The coefficients can be solved with row-wise matrix-vector products:

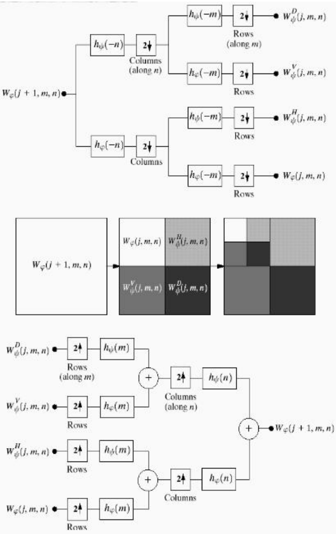
$$W_\varphi(0,0) = \frac{1}{2}[1 \cdot 1 + 4 \cdot 1 - 3 \cdot 1 + 0 \cdot 1] = 1$$

$$W_\psi(0,0) = \frac{1}{2}[1 \cdot 1 + 4 \cdot 1 - 3 \cdot (-1) + 0 \cdot (-1)] = 4$$

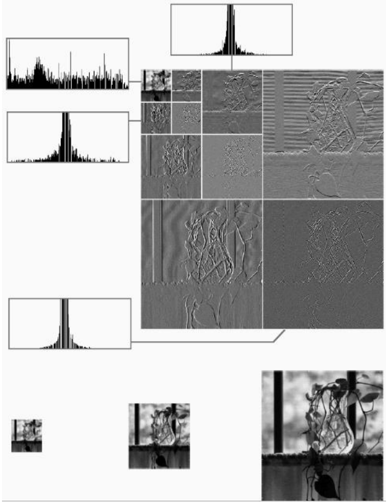
$$W_\psi(1,0) = \frac{1}{2}[1 \cdot \sqrt{2} + 4 \cdot (-\sqrt{2}) - 3 \cdot 0 + 0 \cdot 0] = -1.5\sqrt{2}$$

$$W_\psi(1,1) = \frac{1}{2}[1 \cdot 0 + 4 \cdot 0 - 3 \cdot \sqrt{2} + 0 \cdot (-\sqrt{2})] = -1.5\sqrt{2}$$

The block diagram of two-dimensional wavelet transform (7.5)



An example of Haar function in wavelet transform (7.1.3)



Contents

Hands-on work and home assignment

- Instructions in Google Drive
- Also hands-on work can be done at home
- Report all code and images of the hands-on and the home assignment in the same PDF
- Report also how long time the assignments took
- Email the PDF before midnight to the lecturer