Edge Linking and Boundary Detection
3 Edge Linking

- Local processing
  - link all points in a local neighbourhood (3x3, 5x5, etc.) that are considered to be similar
    - similar response strength of a gradient operator
    - similar direction of the gradient vector
3 Edge Linking

- Local processing
  - link all points in a local neighbourhood (3x3, 5x5, etc.) that are considered to be similar
    - similar response strength of a gradient operator
    - similar direction of the gradient vector

- Global processing
  - Hough transform
3 Edge Linking

- Hough Transform
  - a method for finding global relationships between pixels
  - example: finding straight lines in an image
    - apply edge enhancing filter (e.g., Laplace)
    - set a threshold for deciding what is a true "edge pixel"
    - extract the pixels that are on a straight line using the Hough transform

original image  edge enhanced image  thresholded edge image
3 Edge Linking

- Hough Transform

\[ y_i = ax_i + b \]

\[ b = -ax_i + y_i \]
3 Edge Linking

Hough Transform

- divide parameter space into accumulator cells $A(p,q)$
  - initialise all cell values $A(p,q) = 0$
- for all selected pixels $(x_i, y_i)$ in the image
  - for all $a$-values $a_p$ ("draw a line in $ab$-space")
    - calculate the corresponding $b$-value $b_q = -a_p x_i + y_i$
    - increment accumulator value $A(p,q) \leftarrow A(p,q) + 1$
- $A(p,q) = Q \leftrightarrow Q$ points in $xy$-space lying on the line $y = a_p x + b_q$
- complexity = (number of $a$-increments) $\times$ (number of points)

result: the largest value $A(p',q') = \max\{A(p,q)\}$
gives the line connecting the largest number of pixels
3 Edge Linking

Hough Transform

- In reality, we have a problem with \( y = ax + b \) because \( a \) reaches infinity for vertical lines. → use \( x \cos \theta + y \sin \theta = \rho \) instead.

- Different variations of the Hough transform can also be used for finding other shapes of the form \( g(\mathbf{v}, \mathbf{c}) = 0 \):
  - \( \mathbf{v} \) is a vector of coordinates.
  - \( \mathbf{c} \) is a vector of coefficients.

- Possible to find any kind of simple shape.
  - E.g., circle: \((x - c_1)^2 + (y - c_2)^2 = c_3^2\) (3D parameter space)
3 Edge Linking

- Hough Transform – Example
  - 5 points
  - points of intersection
    - A ↔ line through 1, 3, 5
    - B ↔ line through 2, 3, 4
3 Edge Linking

- Hough Transform-based Edge Linking
  - compute the image gradient
  - threshold the image gradient
  - specify subdivisions in the $\rho \theta$ space
  - examine the counts of accumulator cells for high pixel concentrations
  - examine continuity between the pixels in a cell
    - continuity typically based on computation of the distance between disconnected pixels identified during traversal of a set of pixels according to a given accumulator cell
3 Edge Linking

- Hough Transform-based Edge Linking
  - aerial infrared image
  - thresholded gradient
  - Hough transform
  - set of pixels in the 3 max. accumulator cells linked over gaps < 6 pixels
3 Edge Linking

- Hough Transform-based Edge Linking

from http://www.cs.tu-bs.de/rob/lehre/bv/HNF.html
3 Edge Linking

- Hough Transform-based Edge Linking

7 maxima from http://www.cs.tu-bs.de/rob/lehre/bv/HNF.html
3 Edge Linking

Hough Transform-based Edge Linking

from http://www.cs.tu-bs.de/rob/lehre/bv/HNF.html
Thresholding
4 Thresholding

- **Global**
  - based on some kind of histogram: grey-level, edge, feature etc.
  - lighting conditions are extremely important, will only work under very controlled circumstances
  - fixed threshold (the same value is used in the whole image)

- **Local (or Adaptive Thresholding)**
  - depends on the position in the image
  - the image is divided into overlapping sections which are thresholded one by one
4 Thresholding

- Global Thresholding – Histograms

  to love…

  …and to hate
4 Thresholding

Global Thresholding – Illumination

- solutions
  - calibration of the imaging system
    - measure illumination pattern $g$
on a white reflective surface
    - normalize images $f$
      obtained by dividing through $g$
  - local/adaptive thresholding
  - use a percentile filter with very large mask
to estimate illumination
4 Thresholding

- Automatic Thresholding Algorithm
  1. select an initial estimate for $T$
  2. segment the image using $T$ which produces 2 groups:
     - $G_1$, pixels with value $>T$ and $G_2$, with value $<T$
  3. compute $\mu_1$ and $\mu_2$ as average pixel value of $G_1$ and $G_2$
  4. new threshold: $T=1/2(\mu_1+\mu_2)$
  5. repeat steps 2 to 4 until $T$ stabilizes

- very easy + very fast
- assumptions: normal dist. + low noise
4 Thresholding

- Automatic Thresholding
4 Thresholding

- Optimal Thresholding
  - based on the shape of the image histogram

\[ p(z) = P_1 p_1(z) + P_2 p_2(z) \]

- mixture distribution
- if \( P_1 = P_2 \) then optimum threshold is at \( T \)
4 Thresholding

- Otsu’s Method
  - Maximize between-class variance (10.3.3 GW)
  - Based on the histogram (L-1 grey levels)
  - Find Threshold \( k \) for which \( \eta \) is maximized:

\[
\eta = \frac{\sigma_B^2}{\sigma_G^2}
\]

\[
\sigma_B^2 = P_1 (m_1 - m_G)^2 + P_2 (m_2 - m_G)^2
\]

\[
\sigma_G^2 = \sum_{i=0}^{L-1} (i - m_G)^2 p_i
\]

\[
m_G = \sum_{i=0}^{L-1} i \cdot p_i
\]

\[
m_1(k) = \frac{1}{P_1(k)} \sum_{i=0}^{k} i \cdot p_i
\]

\[
m_2(k) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} i \cdot p_i
\]
4 Thresholding

- Adaptive Thresholding

![Diagram showing adaptive thresholding process with images and graphs illustrating the changes in threshold values T.]
4 Thresholding

Chow-Kaneko Method

- subdivide image into overlapping regions
- test bimodality for each region
  - fit a bimodal Gaussian to the histograms of the bimodal regions
  - select optimal threshold for each bimodal region
- compute thresholds for the non-bimodal regions by interpolating the thresholds of the bimodal regions
- interpolate thresholds for each pixel (2nd interpolation)
- can yield excellent results
4 Thresholding

Remarks on Thresholding

- problem: can leave “holes” in segmented objects
  - solution: post-processing with morphological operators
- thresholding is a special case of pixel classification
  - 2 classes: 0 or 1 (above or below threshold)

Histogram Improvement Using Boundary Information

- determine edges using the image gradient and Laplacian
  - discard pixels not on an edge (gradient < threshold)
  - derive segmentation threshold from a histogram of the edge pixels (gradient > threshold)
- similar height of histogram peaks for object and background
- peaks tend to be better separated (deeper valleys)