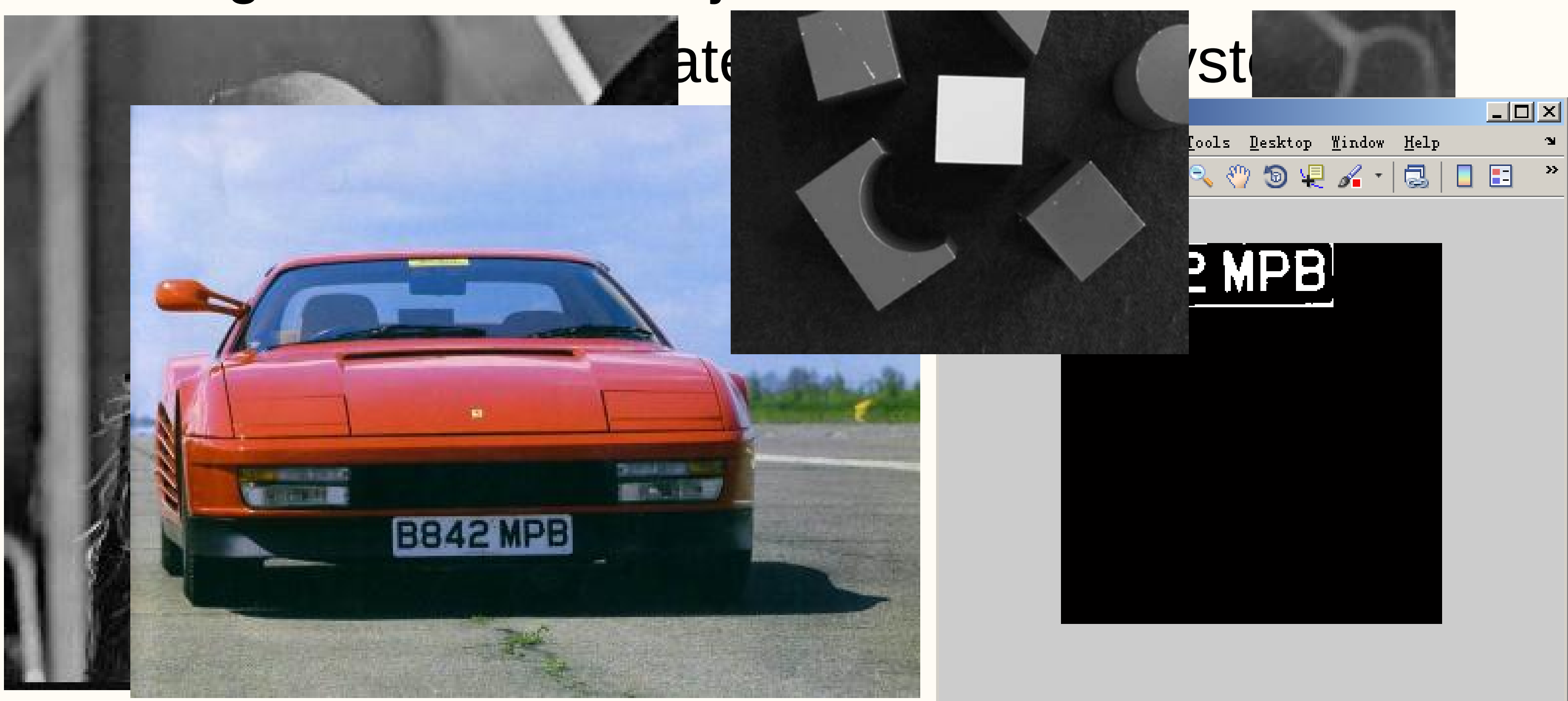
Chapter 4
Image Segmentation

Course Arrangement

- Theory: 32 hours
- Project & Practice: 16 hours
  - Segmentation of Objects



Why we need to learn “image segmentation” ?

Video of “cow tracking ”

Image capture

↓

Preprocessing

↓

Segmentation → Standard segmentation method / Advanced segmentation method

↓

Model fitting

↓

Model prediction

↓

Conclusion (understanding)

Fig. Flowchart of “cow tracking”

Outline

- 4.1 Standard segmentation method

- 4.2 Advanced segmentation method

4.1 Standard segmentation method

- 4.1.1 Thresholding
- 4.1.2 Edge-based segmentation
- 4.1.3 Region growing segmentation

4.1.0 Overview

- One of the most important steps leading to the analysis of processed image data
- Its main goal is to divide an image into parts that have a strong correlation with objects or areas of the real world contained in the image.

4.1.0 Overview

- Complete segmentation

  - set of disjoint regions uniquely corresponding with objects in the input image

  - cooperation with higher processing levels which use specific knowledge of the problem domain is necessary

- Partial segmentation - regions do not correspond directly with image objects

博学慎思 明辨笃行

4.1.0 Overview

- Image is divided into separate regions that are homogeneous with respect to a chosen property such as brightness, color, reflectivity, texture, etc.
- In a complex scene, a set of possibly overlapping homogeneous regions may result. The partially segmented image must then be subjected to further processing, and the final image segmentation may be found with the help of higher level information.

4.1.0 Overview

- Simple segmentation problems:
  - contrasted objects on a uniform background
  - simple assembly tasks, blood cells, printed characters, etc.
- Totally correct and complete segmentation of complex scenes usually cannot be achieved in this processing phase
- A reasonable aim is to use partial segmentation as an input to higher level processing.

4.1.0 Overview

- Segmentation problems(difficulties):
  - image data ambiguity
  - information noise
- Segmentation methods
  - global approaches, e.g. using histogram of image features
  - edge-based segmentation methods
  - region-based segmentation methods
- Characteristics used in edge detection or region growing
  - brightness
  - texture
  - velocity field
  - etc.

4.1.0 Overview

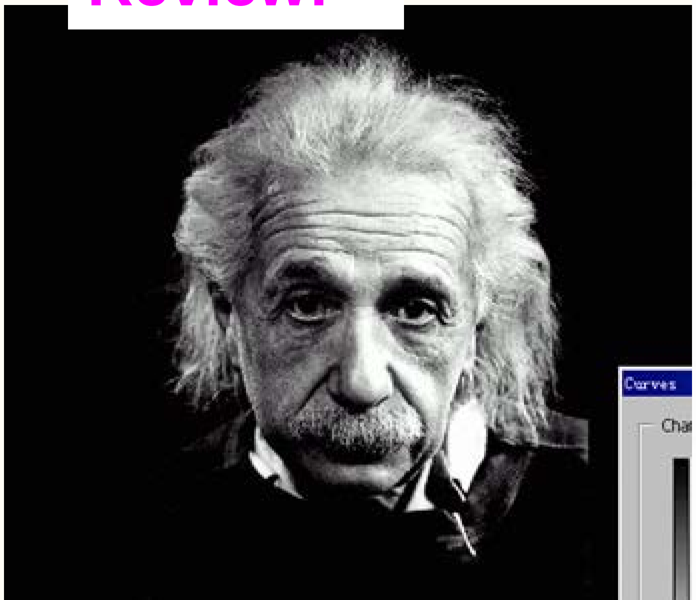
- Edge-based and region-based segmentation approaches solve a dual problem.
- Because of the different natures of the various edge- and region-based algorithms, they may be expected to give somewhat different results and consequently different information.
- The segmentation results of these two approaches can therefore be combined in a single description structure, e.g., a relational graph

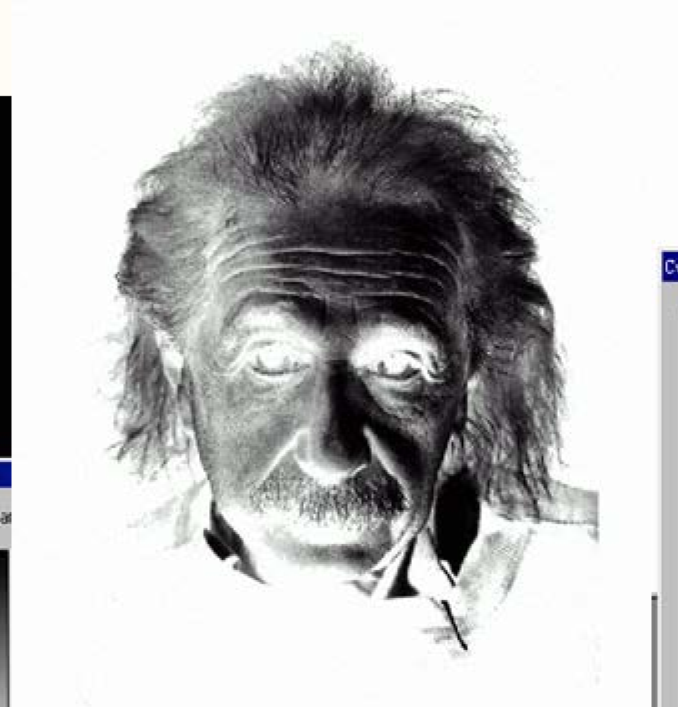
4.1.1 Thresholding

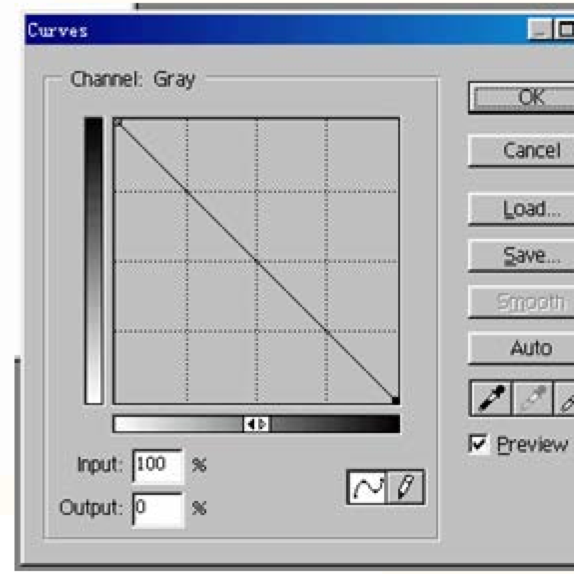
- Gray level thresholding is the simplest segmentation process.
- Many objects or image regions are characterized by constant reflectivity or light absorption of their surface.
- Thresholding is computationally inexpensive and fast.
- Thresholding can easily be done in real time using specialized hardware.

Review!

mation







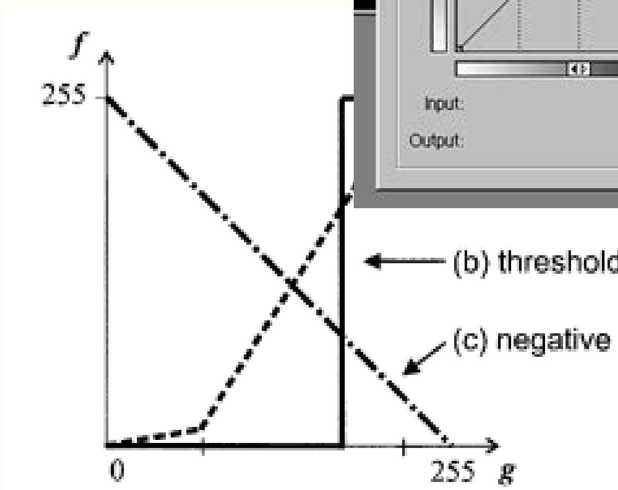


Figure 5.1: Perspective projection geometry examples.

Grey scale transformations are mostly used if the result is viewed by a human!

4.1.1 Thresholding

- Complete segmentation can result from thresholding in simple scenes.

$$R = \bigcup_{i=1}^{S} R_i \qquad R_i \cap R_j = \emptyset \qquad i \neq j \qquad (5.1)$$

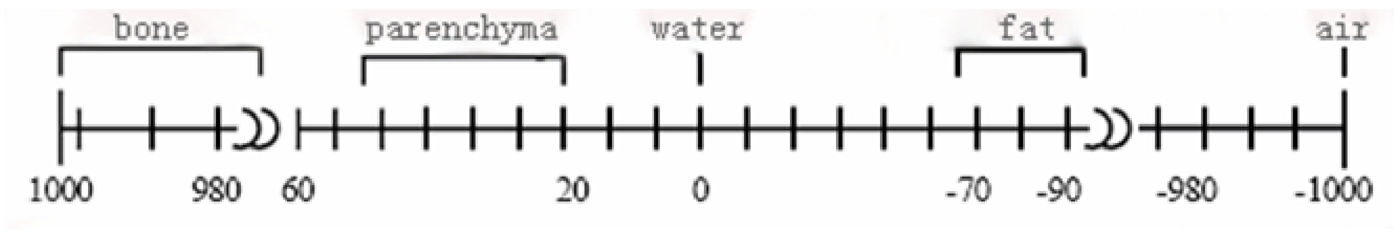
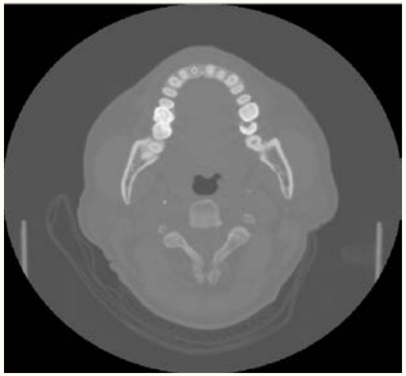
- 

Fig2 The Distribution Curve of CT Value of Different Ingredients of Human Body（Unit HU）



4.1.1 Thresholding

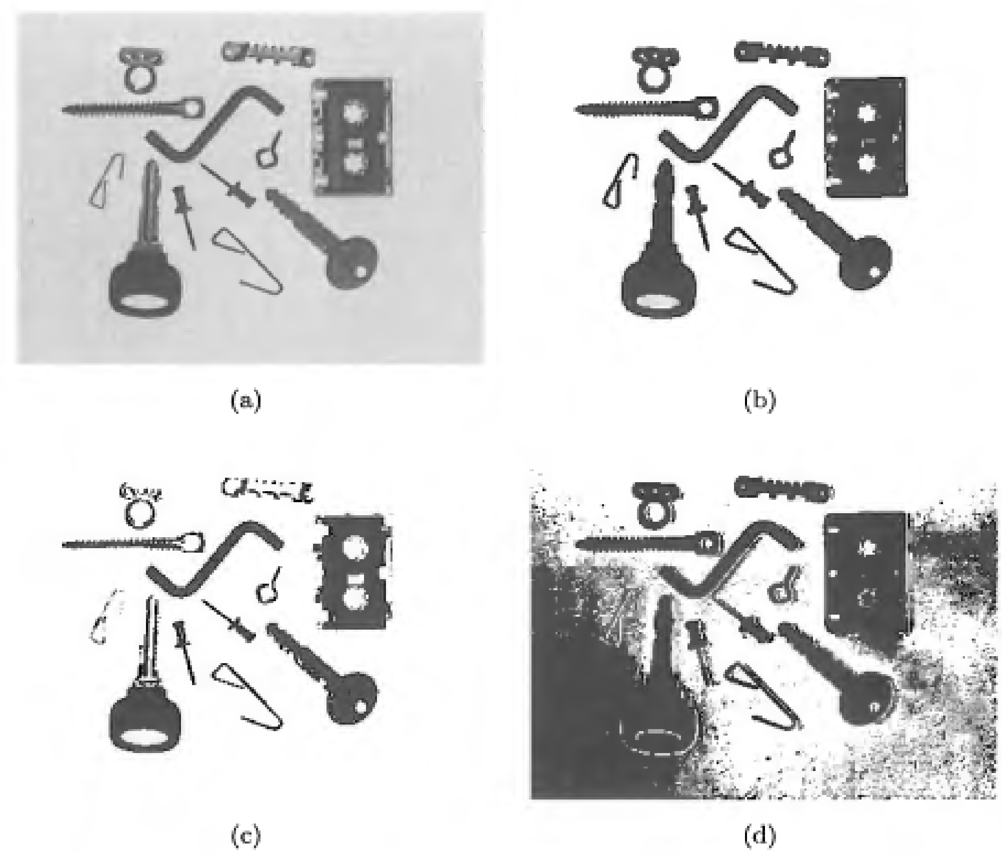


Figure 6.1: Image thresholding. (a) Original image. (b) Threshold segmentation. (c) Threshold too low. (d) Threshold too high.

明辨笃行
博学慎思

4.1.1 Thresholding

- Thresholding algorithm

  Search all the pixels f(i,j) of the image *f*. An image element g(i,j) of the segmented image is an object pixel if f(i,j) >= T, and is a background pixel otherwise.

4.1.1 Thresholding

- Correct threshold selection is crucial for successful threshold segmentation
- Threshold selection can be
  - interactive
  - can be the result of some threshold detection method
- Single global threshold ... successful only under very unusual circumstances

4.1.1 Thresholding

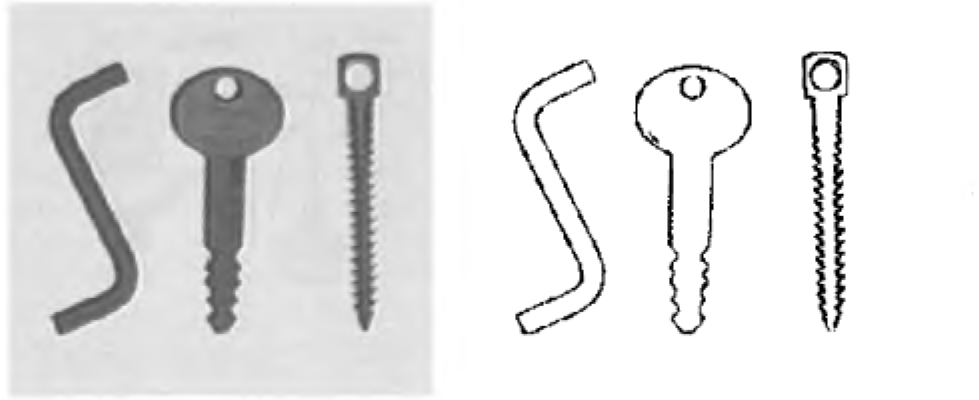
- **Variable thresholding** (also **adaptive thresholding**), in which the threshold value varies over the image as a function of local image characteristics, can produce the solution in these cases.

$$T = T(f, f_c) , \tag{6.4}$$

- image *f* is divided into subimages *fc*
- a threshold is determined independently in each subimage
- each subimage is then processed with respect to its local threshold.

4.1.1 Thresholding

- Thr
- **Ba**
  - – with gray otherwise



(a) (b)

Figure 6.2: Image thresholding modification. (a) Original image. (b) Border detection using band-thresholding.

$$
\begin{aligned}
g(i,j) &= 1 \quad \text{for } f(i,j) \in D \\
&= 0 \quad \text{otherwise}
\end{aligned}
\tag{5.5}
$$

- Can also serve as border detection

4.1.1 Thresholding

- **Multithresholding**
- resulting image is no longer binary

$$
\begin{aligned}
g(i,j) &= 1 \quad \text{for } f(i,j) \in D_1 \\
&= 2 \quad \text{for } f(i,j) \in D_2 \\
&= 3 \quad \text{for } f(i,j) \in D_3 \\
&= 4 \quad \text{for } f(i,j) \in D_4 \\
&\ldots \\
&= n \quad \text{for } f(i,j) \in D_n \\
&= 0 \quad \text{otherwise}
\end{aligned}
\tag{5.6}
$$

- **Semi-thresholding**
  - aims to mask out the image background leaving gray level information present in the objects

$$
\begin{aligned}
g(i,j) &= f(i,j) \quad \text{for } f(i,j) \geq T \\
&= 0 \quad \text{for } f(i,j) < T
\end{aligned}
\tag{5.7}
$$

4.1.1 Thresholding

- Thresholding can also be applied to
  - gradient
  - local texture
  - any other image decomposition criterion

4.1.1 -**Threshold detection methods**

- If some property of an image after segmentation is known a priori, the task of threshold selection is simplified, since the threshold is chosen to ensure this property is satisfied.

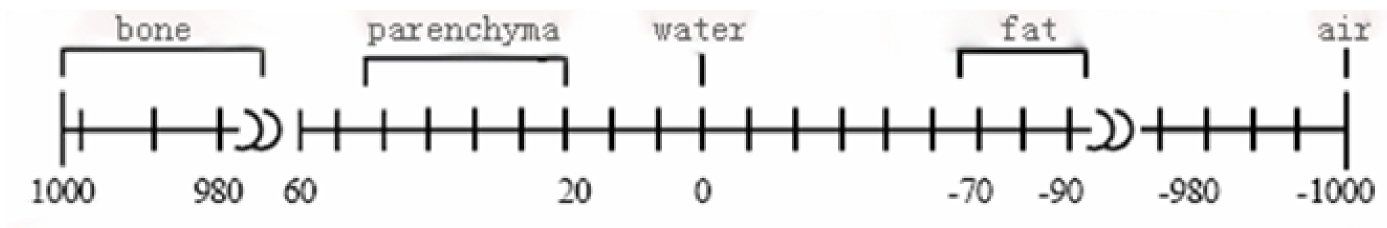
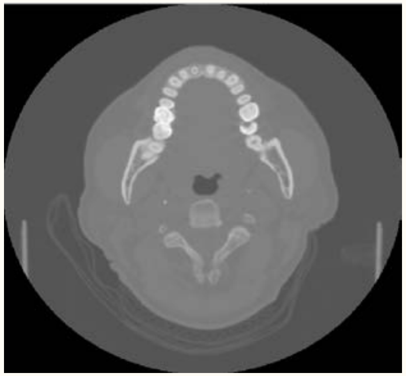


Fig2 The Distribution Curve of CT Value of Different Ingredients of Human Body（Unit HU）



4.1.1 -**Threshold detection methods**

- **P-tile-thresholding**
- choose a threshold T (based on the image histogram) such that 1/p of the image area has gray values less than T and the rest has gray values larger than T
- in text segmentation, prior information about the ratio between the sheet area and character area can be used

4.1.1 -**Threshold detection methods**

- More **complex methods of threshold detection**
- based on histogram shape analysis
- **bimodal histogram** - if objects have approximately the same gray level that differs from the gray level of the background

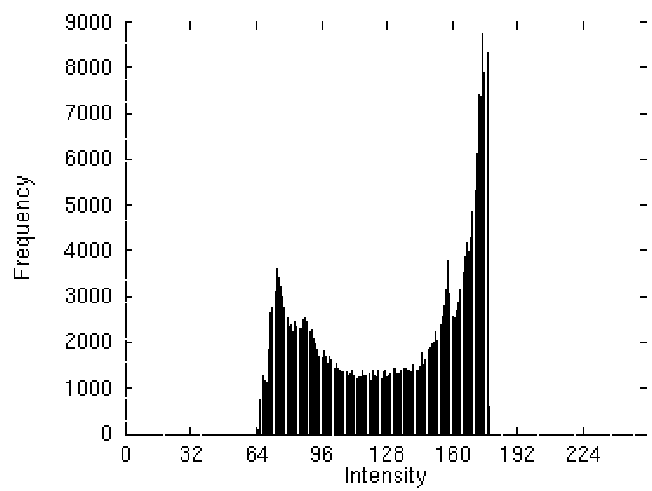
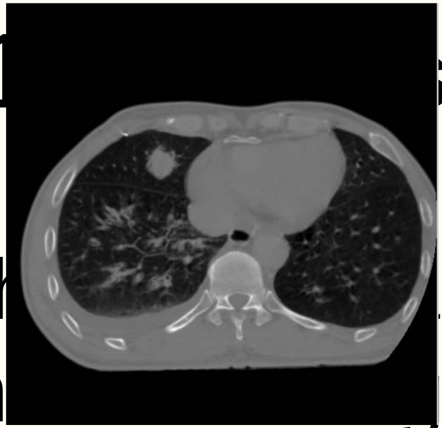
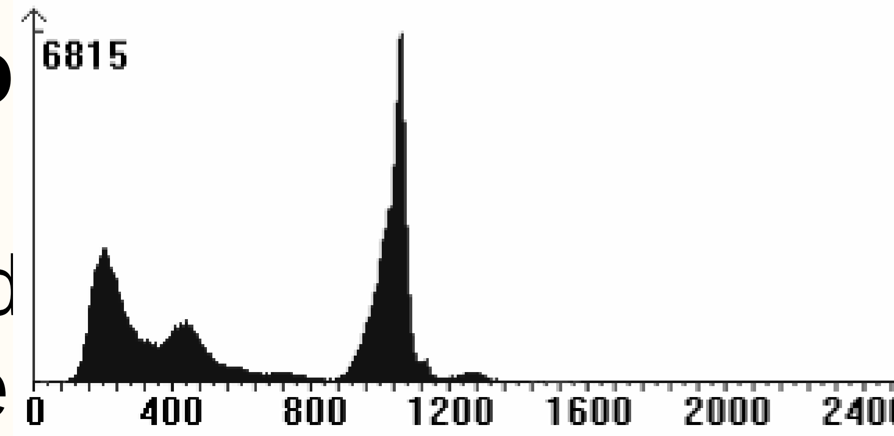


Figure 5.3 *A bimodal histogram.*

4.1 sho ls





- Th sed m gme ts
  – it makes intuitive sense to determine the threshold as the gray level that has a minimum histogram value between the two mentioned maxima
  – multimodal histogram - more thresholds may be determined at minima between any two maxima.

明辨笃行
博学慎思

4.1.1 -**Threshold detection methods**

- Bimodality of histograms
  - to decide if a histogram is bimodal or multimodal may not be so simple in reality
  - it is often impossible to interpret the significance of local histogram maxima

4.1.1 -**Threshold detection methods**

- Bimodal histogram threshold detection algorithms
  - **Mode method** - find the highest local maxima first and detect the threshold as a minimum between them
    - to avoid detection of two local maxima belonging to the same global maximum, a minimum distance in gray levels between these maxima is usually required
    - or techniques to smooth histograms are applied
- Histogram bimodality itself does not guarantee correct threshold segmentation

4.1.1 -**Optimal thresholding**

- based on approximation of the histogram of an image using a weighted sum of two or more probability densities with normal distribution

4.1.1 -**Optimal thresholding**

- The threshold is set as the closest gray level corresponding to the minimum probability between the maxima of two or more normal distributions, which results in minimum error segmentation

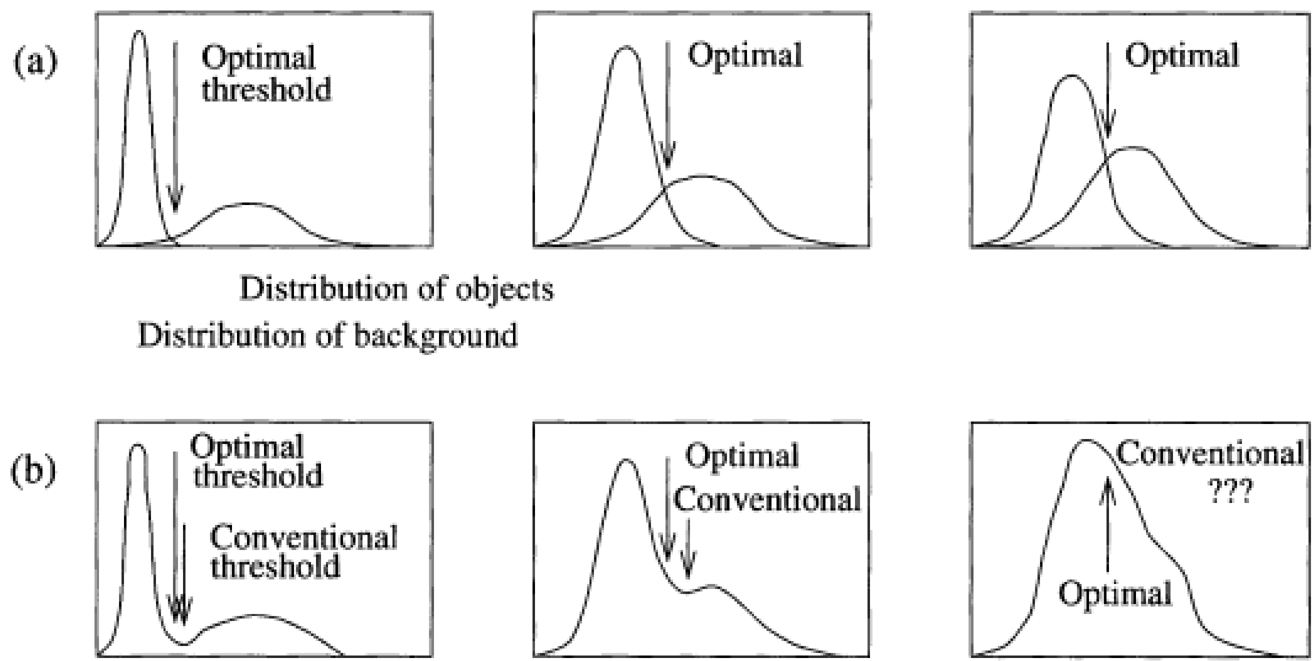


Figure 6.4: Gray-level histograms approximated by two normal distributions--the threshold is set to give minimum probability of segmentation error. (a) Probability distributions of background and objects. (b) Corresponding histograms and optimal threshold.

4.1.1 -**Optimal thresholding**

- Problems - estimating normal distribution parameters together with the uncertainty that the distribution may be considered normal.

Algorithm 6.2: Iterative (optimal) threshold selection

1. Assuming no knowledge about the exact location of objects, consider as a first approximation that the four corners of the image contain background pixels only and the remainder contains object pixels.
2. At step t, compute μ_B^t and μ_O^t as the mean background and object gray-level, respectively, where segmentation into background and objects at step t is defined by the threshold value T^t determined in the previous step [equation 6.9]

$$\mu_B^t = \frac{\sum_{(i,j)\in\text{background}} f(i,j)}{\#\text{background\_pixels}}\,, \qquad \mu_O^t = \frac{\sum_{(i,j)\in\text{objects}} f(i,j)}{\#\text{object\_pixels}}\,. \tag{6.8}$$

3. Set

$$T^{(t+1)} = \frac{\mu_B^t + \mu_O^t}{2}\,, \tag{6.9}$$

   $T^{(t+1)}$ now provides an updated background—object distinction.
4. If $T^{(t+1)} = T^{(t)}$, halt; otherwise return to step 2.

4.1.1 -**Optimal thresholding**

- The method pe   ige
  contrast conditi
- Example - Brai
- Applied to segr
  segmentation c

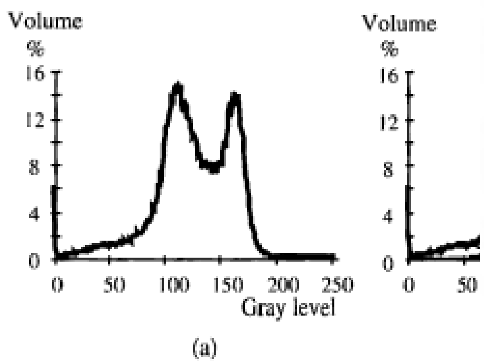


Figure 6.5: Segmentation of 3D T1-weigh
(a) Local gray-level histogram. (b) Fitted Gaussian distributions, global 3D image fit. (c) Gaussian distributions corresponding to WM, GM, and CSF. *Courtesy of R. J. Frank, T. J. Grabowski, The University of Iowa.*

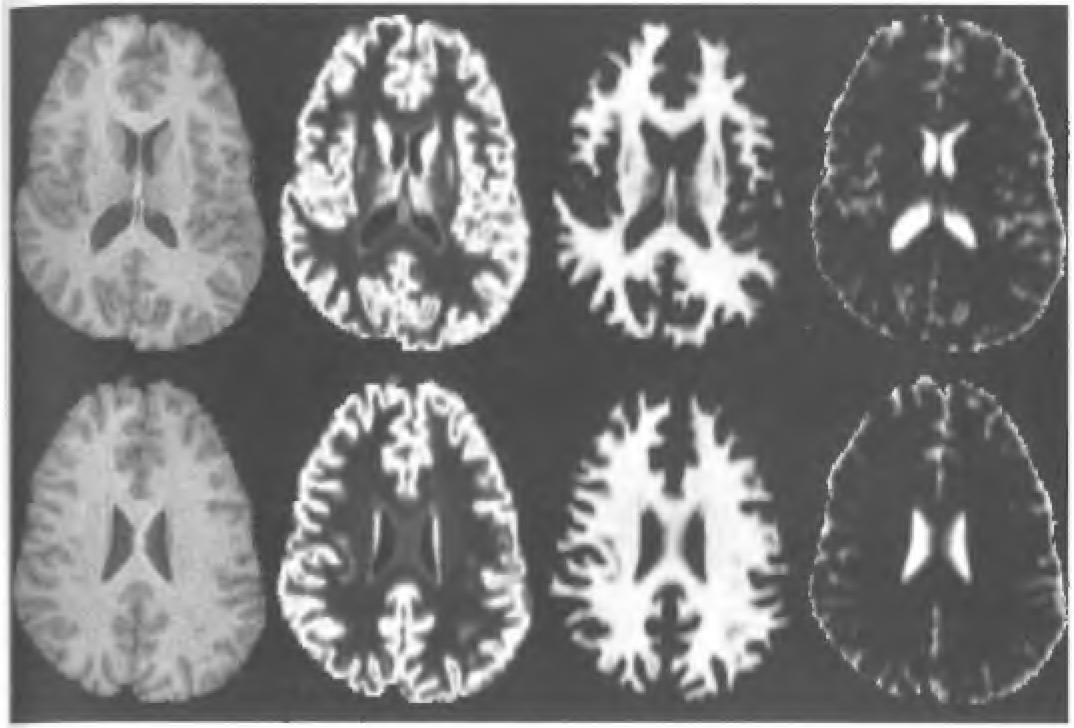


Figure 6.6: Optimal MR brain image segmentation. Left column: original T1-weighted MR images, two of 120 slices of the 3D volume. Middle left: Partial-volume maps of gray matter. The brighter the voxel, the higher is the partial volume percentage of gray matter in the voxel. Middle right: Partial-volume maps of white matter. Right column: Partial-volume maps of cerebro-spinal fluid. *Courtesy of R. J. Frank, T. J. Grabowski, The University of Iowa.*

明辨笃行
博学慎思

4.1.2 Edge-based segmentation

- Edge-based segmentation represents a large group of methods based on information about edges in the image
- Edge-based segmentations rely on edges found in an image by edge detecting operators -- these edges mark image locations of discontinuities in gray level, color, texture, etc.

4.1.2 Edge-based segmentation

- Image resulting from edge detection cannot be used as a segmentation result.
- Supplementary processing steps must follow to combine edges into edge chains that correspond better with borders in the image. An Example!
- The final aim is to reach at least a partial segmentation -- that is, to group local edges into an image where only edge chains with a correspondence to existing objects or image parts are present.

4.1.2 Edge-based segmentation

- The more prior information that is available to the segmentation process, the better the segmentation results that can be obtained.
- The most common problems of edge-based segmentation are
  - an edge presence in locations where there is no border, and
  - no edge presence where a real border exists.

Weak edge!

4.1.2 -Edge image thresholding

- Almost no zero-value pixels are present in an edge image, but small edge values correspond to non-significant gray level changes resulting from quantization noise, small lighting irregularities, etc.
- Selection of an appropriate global threshold is often difficult and sometimes impossible; p-tile thresholding can be applied to define a threshold.

4.1.2 -Edge image thresholding

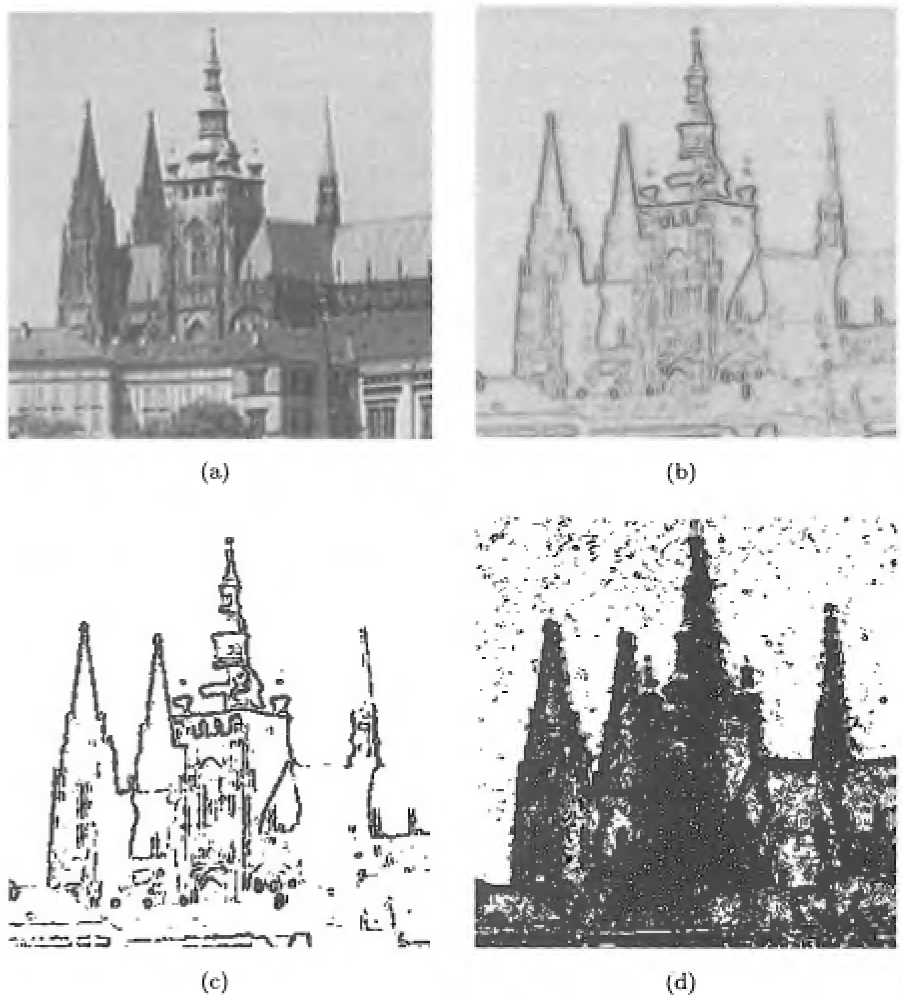


Figure 6.8: Edge image thresholding. (a) Original image. (b) Edge image (low contrast edges enhanced for display). (c) Edge image thresholded at 30. (d) Edge image thresholded at 10.

Review!

3.3.2 Edge detectors

- locate sharp changes in the intensity function
- edges are pixels where brightness changes abruptly.
- Calculus describes changes of continuous functions using derivatives; an image function depends on two variables - partial derivatives.
- A change of the image function can be described by a gradient that points in the direction of the largest growth of the image function.

Review!

3.3.2 Edge detectors

- Edges are often used in image analysis for finding region boundaries.

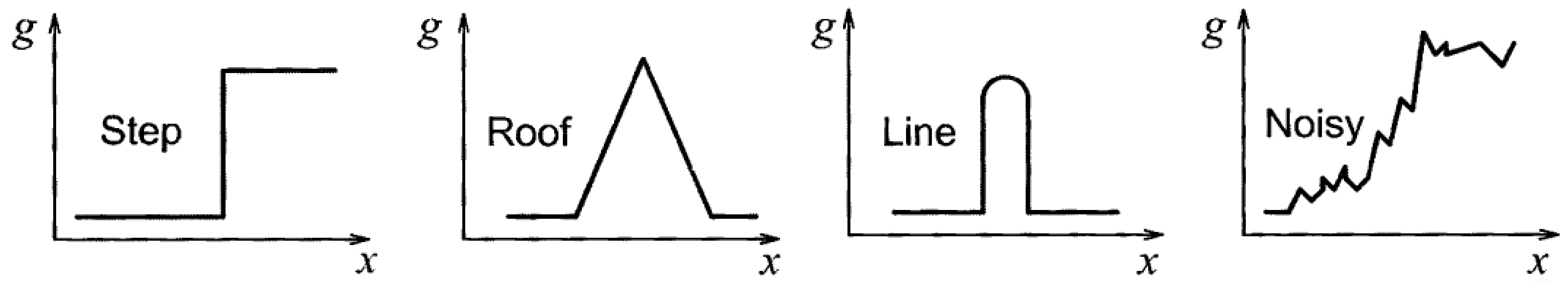
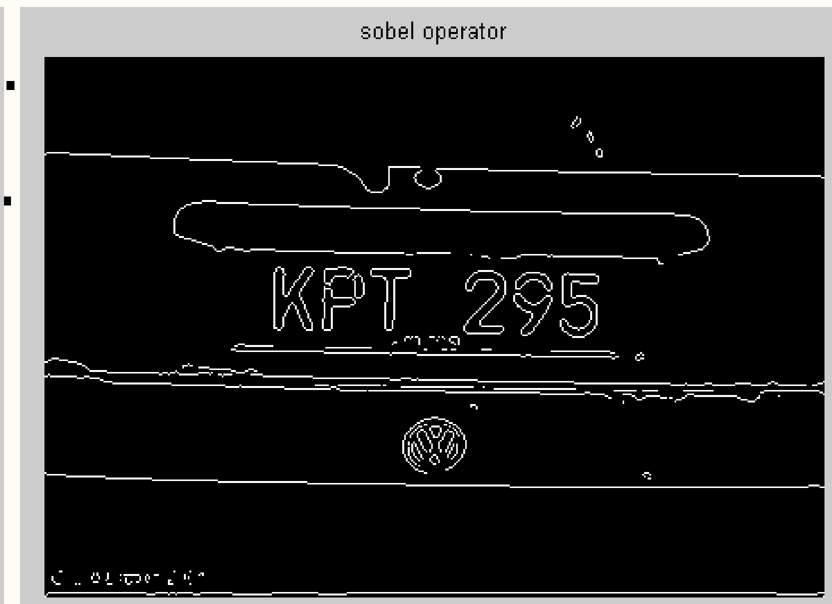


Figure 5.19: Typical edge profiles.

Review!

3.3.2 -other operator

- Roberts operator pp.135
- Prewitt operator pp.136
- Sobel operator pp.136



4.1.2 -Border tracing

- If a region border is not known but regions have been defined in the image, borders can be uniquely detected.
- First, let us assume that the image with regions is either binary or that regions have been labeled.
- An **inner** region border is a subset of the region
- An **outer** border is not a subset of the region.

Review!

2.1.3 -Metric properites of digital images

- **Border** R is the set of pixels within the region that have one or more neighbors outside R ... **inner** borders, **outer** borders exist.

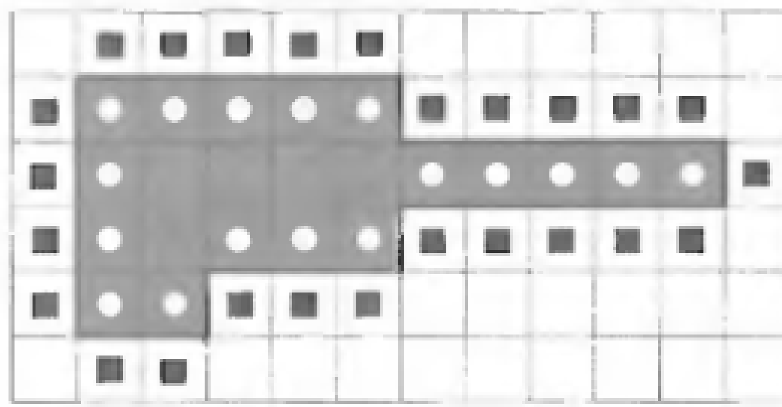


Figure 2.13: Inner borders of a region shown as white circles and outer borders shown as black squares. 4-neighborhood was considered.

4.1.2 -Border tracing

- The following algorithm covers inner boundary tracing in both 4-connectivity and 8-connectivity.

Algorithm 6.7: Inner boundary tracing

1. Search the image from top left until a pixel of a new region is found; this pixel P_0 then has the minimum column value of all pixels of that region having the minimum row value. Pixel P_0 is a starting pixel of the region border. Define a variable *dir* which stores the direction of the previous move along the border from the previous border element to the current border element. Assign
   (a) $dir = 3$ if the border is detected in 4-connectivity (Figure 6.14a),
   (b) $dir = 7$ if the border is detected in 8-connectivity (Figure 6.14b).
2. Search the 3×3 neighborhood of the current pixel in an anti-clockwise direction, beginning the neighborhood search in the pixel positioned in the direction
   (a) $(dir + 3) \bmod 4$ (Figure 6.14c),
   (b) $(dir + 7) \bmod 8$ if *dir* is *even* (Figure 6.14d),
   $(dir + 6) \bmod 8$ if *dir* is *odd* (Figure 6.14e).

   The first pixel found with the same value as the current pixel is a new boundary element P_n. Update the *dir* value.
3. If the current boundary element P_n is equal to the second border element P_1, and if the previous border element P_{n-1} is equal to P_0, stop. Otherwise repeat step 2.
4. The detected inner border is represented by pixels $P_0 \ldots P_{n-2}$.

4.1.2 -Border tracing

- The following algorithm covers inner boundary tracing in both 4-connectivity and 8-connectivity.

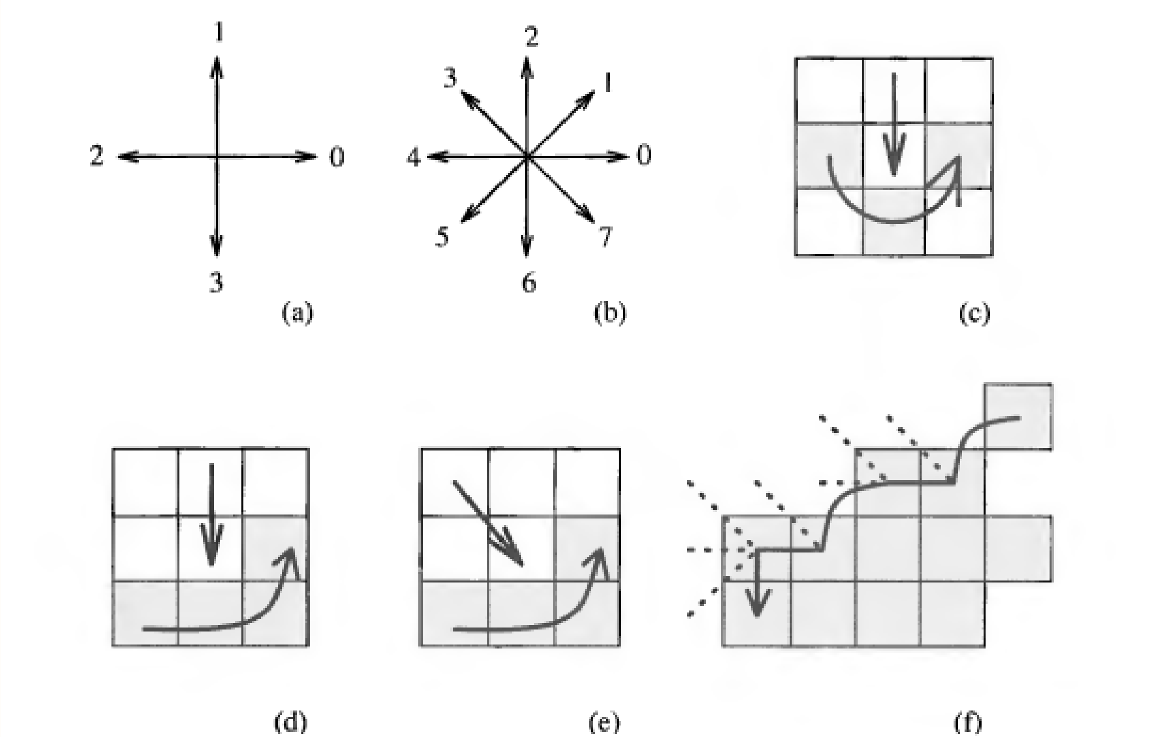


Figure 6.14: Inner boundary tracing. (a) Direction notation, 4-connectivity. (b) 8-connectivity. (c) Pixel neighborhood search sequence in 4-connectivity. (d), (e) Search sequence in 8-connectivity. (f) Boundary tracing in 8-connectivity (dotted lines show pixels tested during the border tracing).

4.1.2 -Border tracing

- The inner border is always part of a region but the outer border never is.
- Therefore, if two regions are adjacent, they never have a common border, which causes difficulties in higher processing levels with region description, region merging, etc.
- The inter-pixel boundary extracted, for instance, from crack edges is common to adjacent borders, nevertheless, its position cannot be specified in pixel co-ordinates.

4.1.2 -Border detection as graph searching

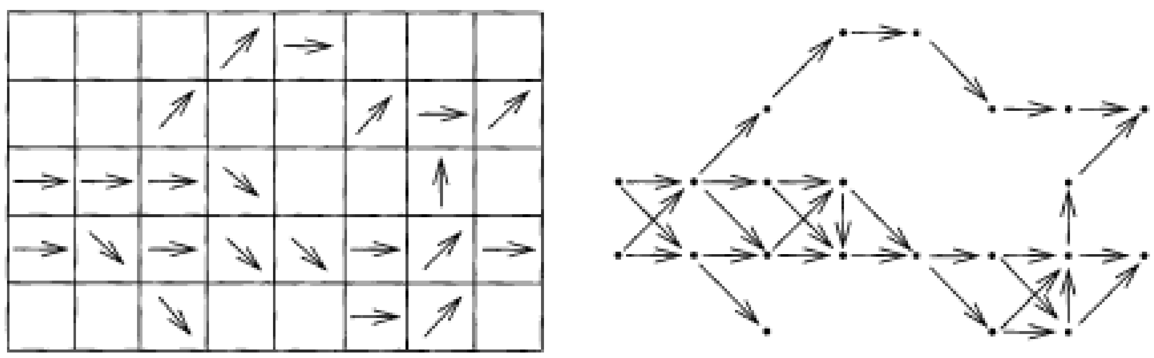
- Whenever additional knowledge is available for boundary detection, it should be used - e.g., known approximate starting point and ending point of the border
- Even some relatively weak additional requirements such as smoothness, low curvature, etc. may be included as prior knowledge.

4.1.2 -Border detection as graph searching

- A graph is a general structure consisting of a set of nodes n_i and arcs between the nodes [n_i,n_j]. We consider oriented and numerically weighted arcs, these weights being called **costs**.
- The border detection process is transformed into a search for the optimal path in the weighted graph.
- Assume that both edge magnitude and edge direction information is available in an edge image.

4.1.2 -Border detection as graph searching

- Figure 6.20a shows an image of edge directions, with only significant edges according to their magnitudes listed.
- Figure 6.20b shows an oriented graph constructed in accordance with the presented principles.



(a) (b)

Figure 6.20: Graph representation of an edge image. (a) Edge directions of pixels with above-threshold edge magnitudes. (b) Corresponding graph.

4.1.2 -Border detection as graph searching

- To use graph search for region border detection, a method of oriented weighted-graph expansion must first be defined.
- A cost function $f(x_i)$ must also be defined that is a cost estimate of the path between nodes n_A and n_B (pixels x_A and x_B) which goes through an intermediate node n_i (pixel x_i).
- The cost function $f(x_i)$ typically consists of two components; an estimate of the path cost between the starting border element x_A and x_i, and an estimate of the path cost between x_i and the end border element x_B.

4.1.2 -Border detection as graph searching

- The cost of the path from the starting point to the node *n_i* is usually a sum of costs associated with the arcs or nodes that are in the path.
- The cost function must be separable and monotonic with respect to the path length, and therefore the local costs associated with arcs are required to be non-negative.

Algorithm 5.12: Heuristic graph search

1. Expand the starting node n_A and put all its successors into an OPEN list with pointers back to the starting node n_A. Evaluate the cost function f for each expanded node.
2. If the OPEN list is empty, fail.
   Determine the node n_i from the OPEN list with the lowest associated cost $f(n_i)$ and remove it. If $n_i = n_B$, then trace back through the pointers to find the optimum path and stop.
3. If the option to stop was not taken in step (2), expand the specified node n_i, and put its successors on the OPEN list with pointers back to n_i. Compute their costs f. Go to step (2).

4.1.2 -Border detection as graph searching

- Since the range of border detection applications is quite wide, the cost functions described may need some modification to be relevant to a particular task.

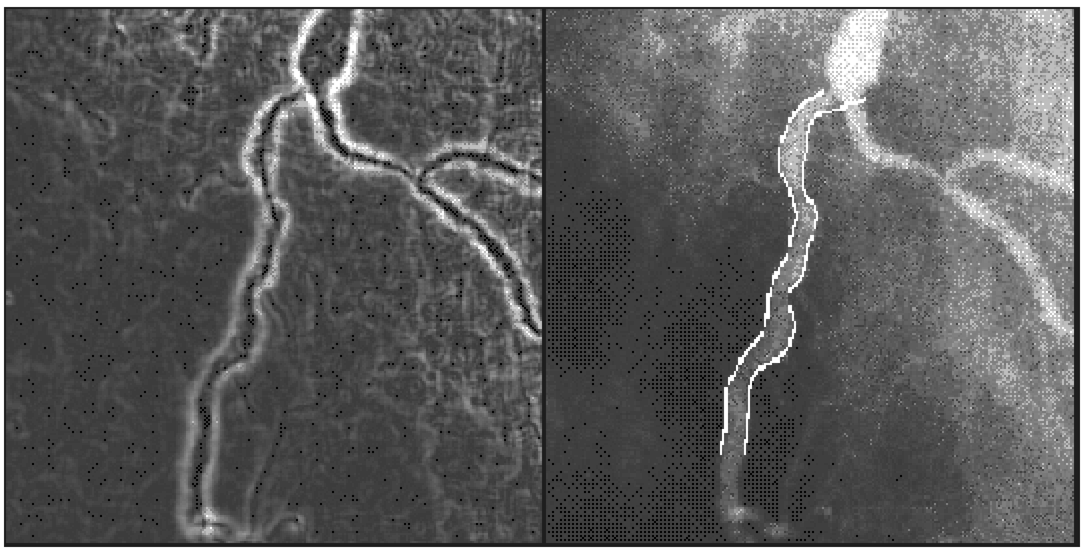


Figure 5.23 *Graph search applied to coronary vessel border detection: (a) Edge image (see Figure ?? for the original angiographic image), (b) determined vessel borders.*

明辨笃行
博学慎思

4.1.2 -**Hough transforms**

- If an image consists of objects with known shape and size, segmentation can be viewed as a problem of finding this object within an image.

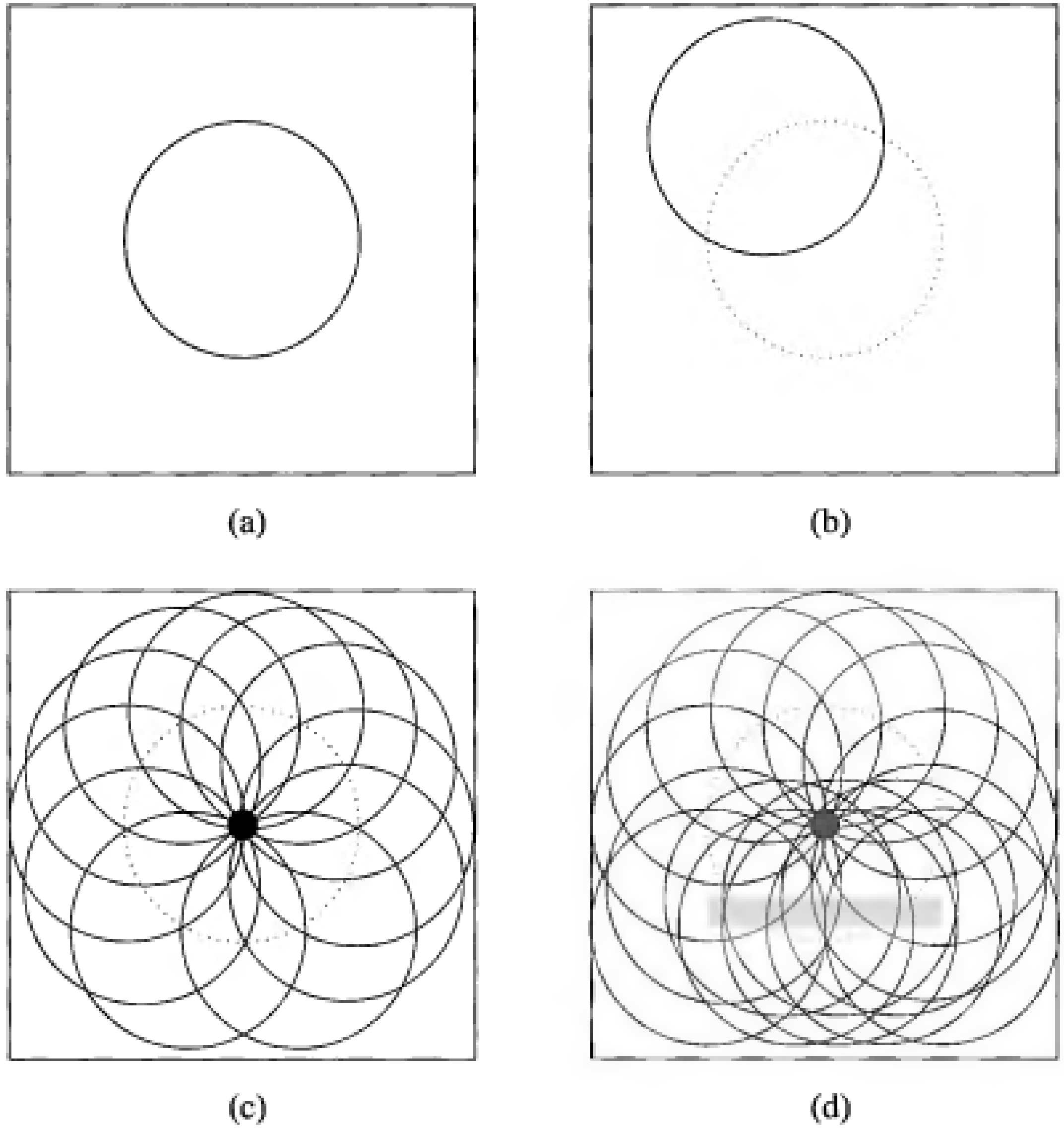
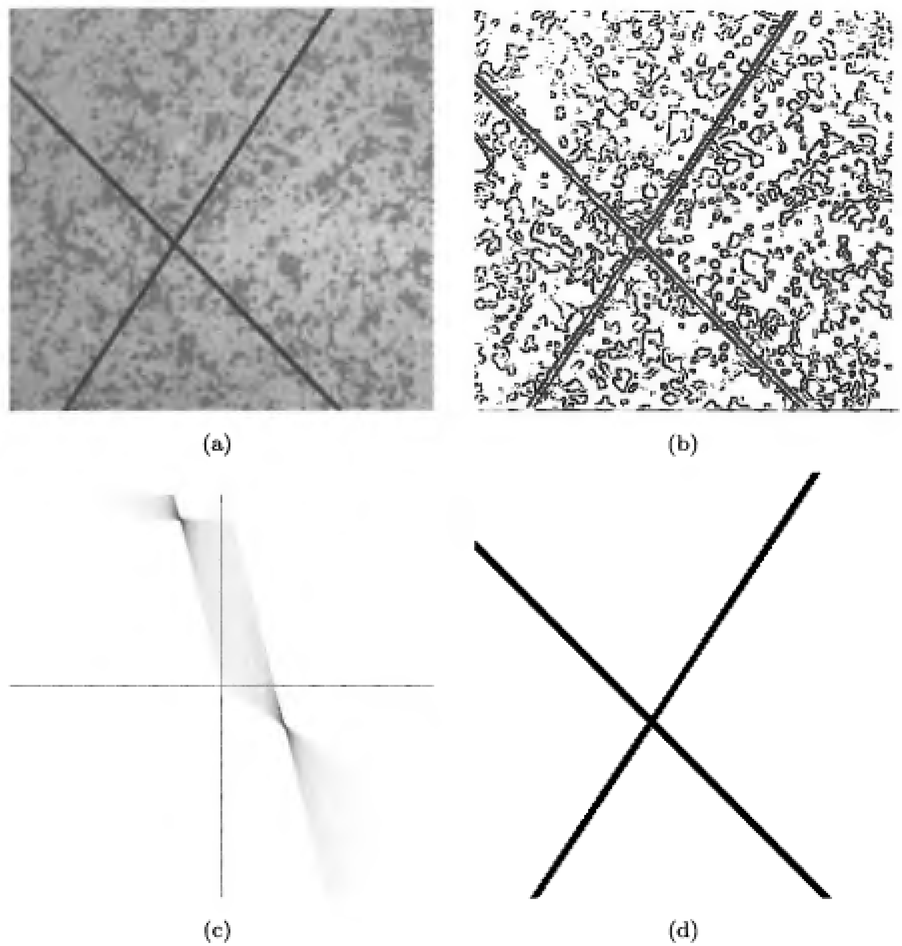


Figure 6.32: Hough transform—example of circle detection. (a) Original image of a dark circle (known radius r) on a bright background. (b) For each dark pixel, a potential circle-center locus is defined by a circle with radius r and center at that pixel. (c) The frequency with which image pixels occur in the circle-center loci is determined—the highest-frequency pixel represents the center of the circle (marked by •). (d) The Hough transform correctly detects the circle (marked by •) in the presence of incomplete circle information and overlapping structures. (See Figure 6.37 for a real-life example.)

4.1.2 -**Hough transforms**

- The ori… …d to
  detect s…
- A big a… …ness of
  segmen… …is not
  too sen…



(a) (b) (c) (d)

Figure 6.34: Hough transform—line detection. (a) Original image. (b) Edge image (note many edges, which do not belong to the line). (c) Parameter space. (d) Detected lines.

明辨笃行
博学慎思

4.1.2 -**Hough transforms**

- This means that any straight line in the image is represented by a single point in the k,q parameter space and any part of this straight line is transformed into the same point.

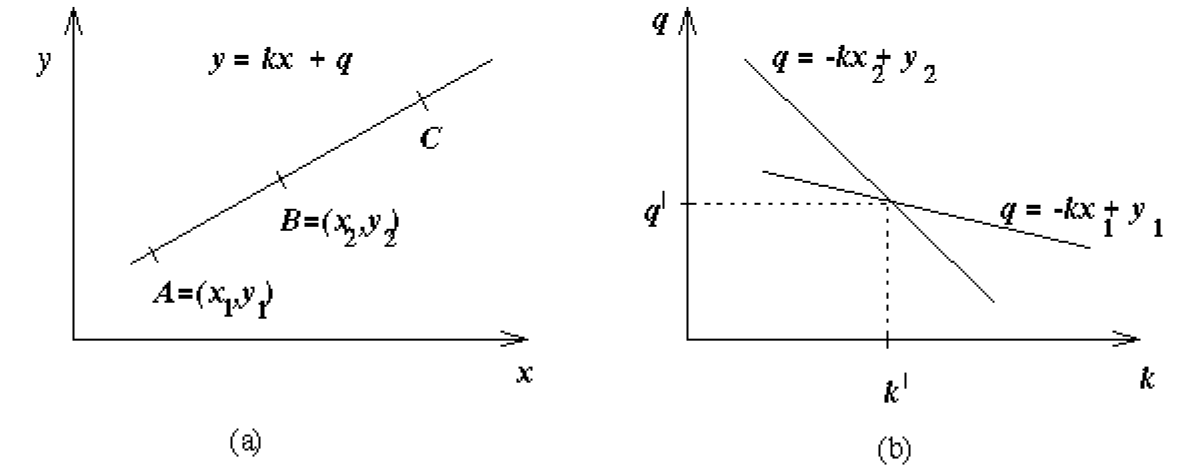


Figure 5.30 *Hough transform principles: (a) Image space, (b) k, q parameter space.*

4.1.2 -**Hough transforms**

- The main idea of line detection is to determine all the possible line pixels in the image, to transform all lines that can go through these pixels into corresponding points in the parameter space, and to detect the points (a,b) in the parameter space that frequently resulted from the Hough transform of lines y=ax+b in the image.

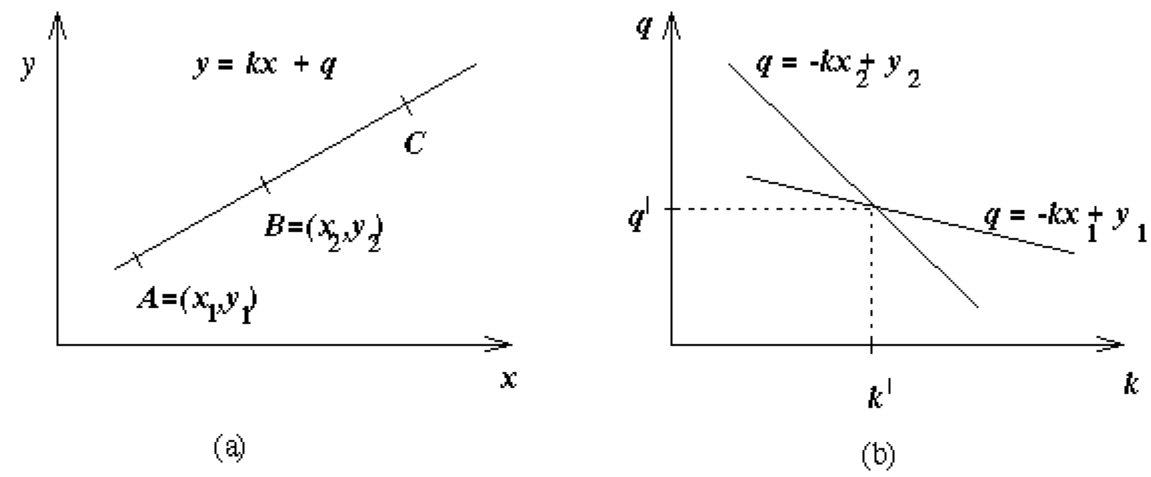


Figure 5.30 *Hough transform principles: (a) Image space, (b) k, q parameter space.*

4.1.2 -**Hough transforms**

- Then, all pixels with edge magnitude exceeding some threshold can be considered possible line pixels.

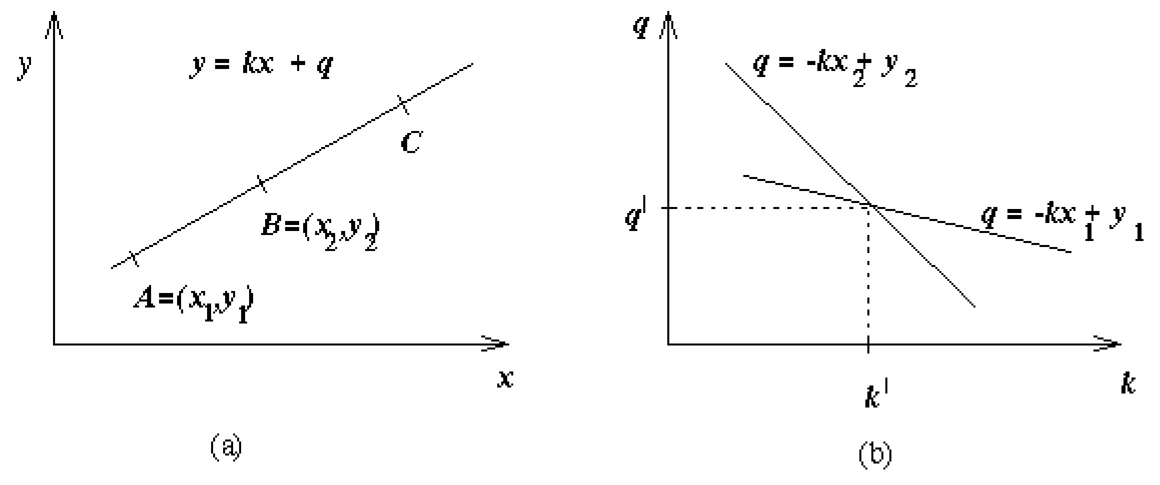


Figure 5.30 *Hough transform principles: (a) Image space, (b) k, q parameter space.*

4.1.2 -**Hough transforms**

- In the most general case, nothing is known about lines in the image, and therefore lines of any direction may go through any of the edge pixels. In reality, the number of these lines is infinite, however, for practical purposes, only a limited number of line directions may be considered.

4.1.2 -**Hough transforms**

- The possible directions of lines define a discretization of the parameter k.
- Similarly, the parameter q is sampled into a limited number of values.

4.1.2 -**Hough transforms**

- The parameter space is not continuous any more, but rather is represented by a rectangular structure of cells. This array of cells is called the **accumulator array** A, whose elements are **accumulator cells** A(k,q).
- For each edge pixel, parameters k,q are determined which represent lines of allowed directions going through this pixel. For each such line, the values of line parameters k,q are used to increase the value of the accumulator cell A(k,q).

4.1.2 -**Hough transforms**

- Clearly, if a line represented by an equation y=ax+b is present in the image, the value of the accumulator cell A(a,b) will be increased many times -- as many times as the line y=ax+b is detected as a line possibly going through any of the edge pixels.

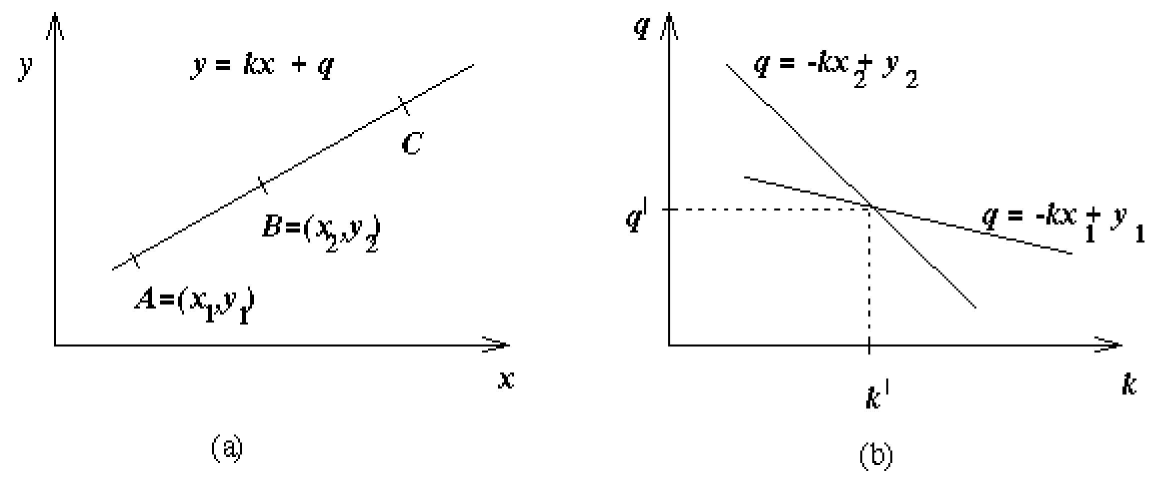


Figure 5.30 *Hough transform principles: (a) Image space, (b) k, q parameter space.*

4.1.2 -**Hough transforms**

- Lines existing in the image may be detected as high-valued accumulator cells in the accumulator array, and the parameters of the detected line are specified by the accumulator array co-ordinates.
- As a result, line detection in the image is transformed to detection of local maxima in the accumulator space.

4.1.2 -**Hough transforms**

- The parametric equation of the line $y=kx+q$ is appropriate only for explanation of the Hough transform principles -- it causes difficulties in vertical line detection ($k \to$ infinity) and in nonlinear discretization of the parameter k.
- The parameterization of the lines used in this demonstration are given by the polar form of a line here.

4.1.2 -**Hough transforms**

- If a line is represented as

$$s = x\cos\theta + y\sin\theta\,, \tag{6.25}$$

the Hough transform does not suffer from these limitations.

- Again, the straight line is transformed to a single point

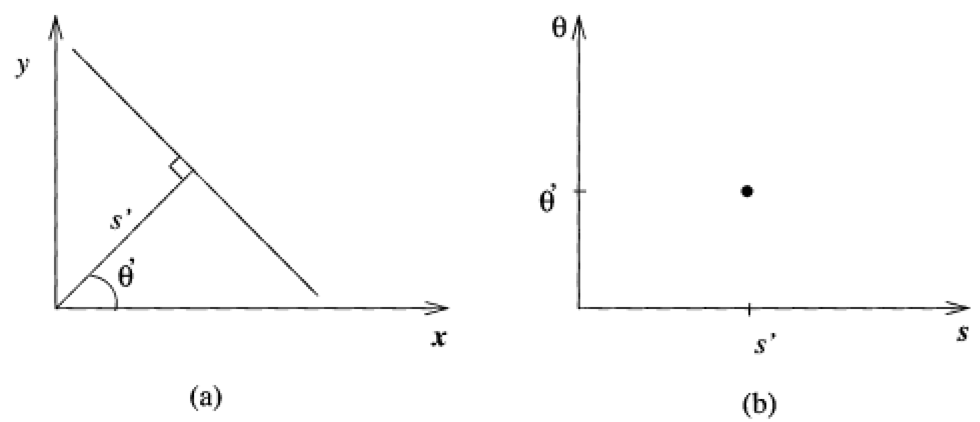
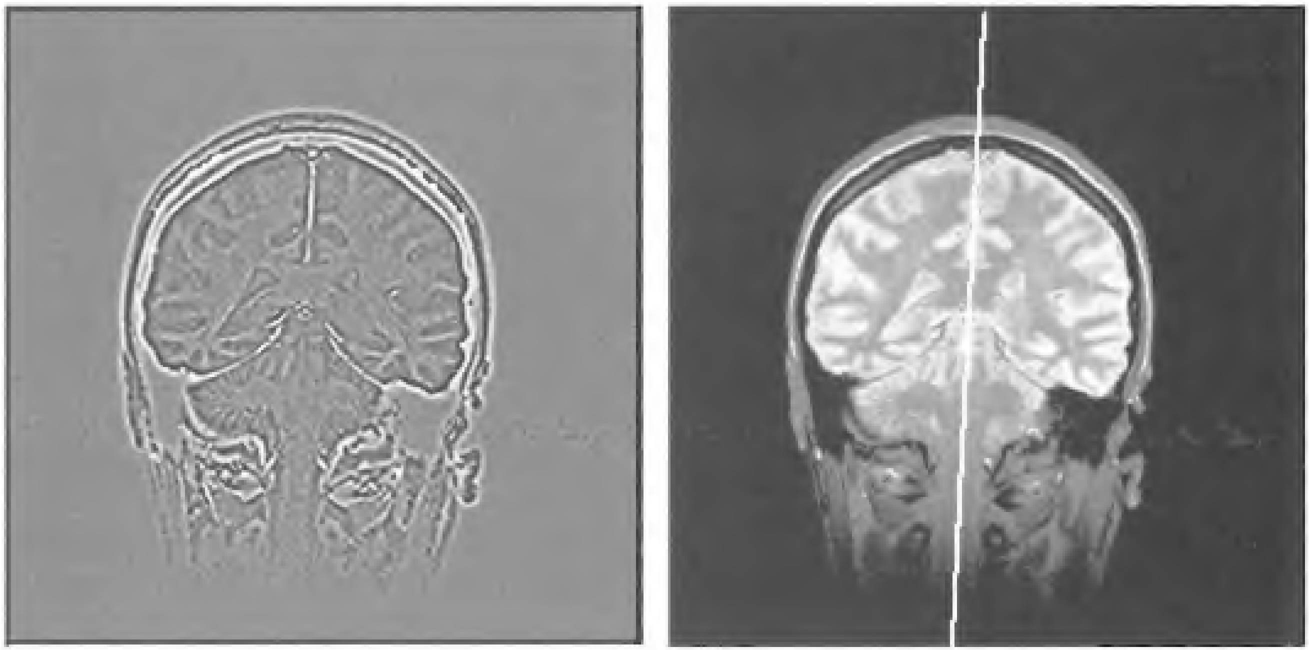


Figure 6.35: Hough transform in s, θ space. (a) Straight line in image space. (b) s, θ parameter space.

4.1.2 -**Hough transforms**



(a) (b)

Figure 6.36: Hough transform line detection used for MRI brain segmentation to the left and right hemispheres. (a) Edge image. (b) Segmentation line in original image data.

4.1.3 Region-based segmentation

- Edge-based segmentation: borders between regions
- Region-based segmentation: direct construction of regions
- It is easy to construct regions from their borders and it is easy to detect borders of existing regions.

4.1.3 Region-based segmentation

- Segmentations resulting from edge-based methods and region growing methods are not usually exactly the same.
- Combination of results may often be a good idea.
- Region growing techniques are generally better in noisy images where edges are extremely difficult to detect.
- Homogeneity of regions is used as the main segmentation criterion in region growing.

4.1.3 Region-based segmentation

- The criteria for homogeneity:
- gray level
- color, texture
- shape
- model
- etc.

4.1.3 Region-based segmentation

- Regions have already been defined

$$R = \bigcup_{i=1}^{S} R_i \qquad R_i \cap R_j = \emptyset \qquad i \neq j \qquad (5.1)$$

- Further assumptions:

$$H(R_i) = TRUE \qquad i = 1, 2, ..., S \qquad (5.31)$$

$$H(R_i \cup R_j) = FALSE \quad i \neq j, \quad R_i \;\; adjacent\ to \;\; R_j \qquad (5.32)$$

- Resulting regions of the segmented image must be both homogeneous and maximal.

Region Growing

- A simple approach to image segmentation is to start from some pixels (seeds) representing distinct image regions and to grow them, until they cover the entire image
- For region growing we need a rule describing a growth mechanism and a rule checking the homogeneity of the regions after each growth step

Region Growing – cont.d

- The growth mechanism – at each stage k and for each region Ri(k), i = 1,…,N, we check if there are unclassified pixels in the 8-neighbourhood of each pixel of the region border
- Before assigning such a pixel x to a region Ri(k),we check if the region homogeneity:

  P(Ri(k) U {x}) = TRUE , is valid

Region Growing – cont.d

- The arithmetic mean m and standard deviation *sd* of a class Ri having n pixels:

  $M = (1/n) \sum_{(r,c) \in R(i)} I(r,c)$

  $s.d = \text{Square root}((1/n) \sum_{(r,c) \in R(i)} [I(r,c)-M]^2)$

  Can be used to decide if the merging of the two regions R1,R2 is allowed, if

  $|M1 – M2| < (k)s.d(i)$, $i = 1, 2$, two regions are merged

Region Growing – cont.d

- Homogeneity test: if the pixel intensity is close to the region mean value

$$|I(r,c) - M(i)| <= T(i)$$

- Threshold Ti varies depending on the region Rn and the intensity of the pixel I(r,c).It can be chosen this way:

$$T(i) = \{ 1 - [s.d(i)/M(i)] \} T$$

Example for region growing!

4.1.3 Region-based segmentation

- Region merging

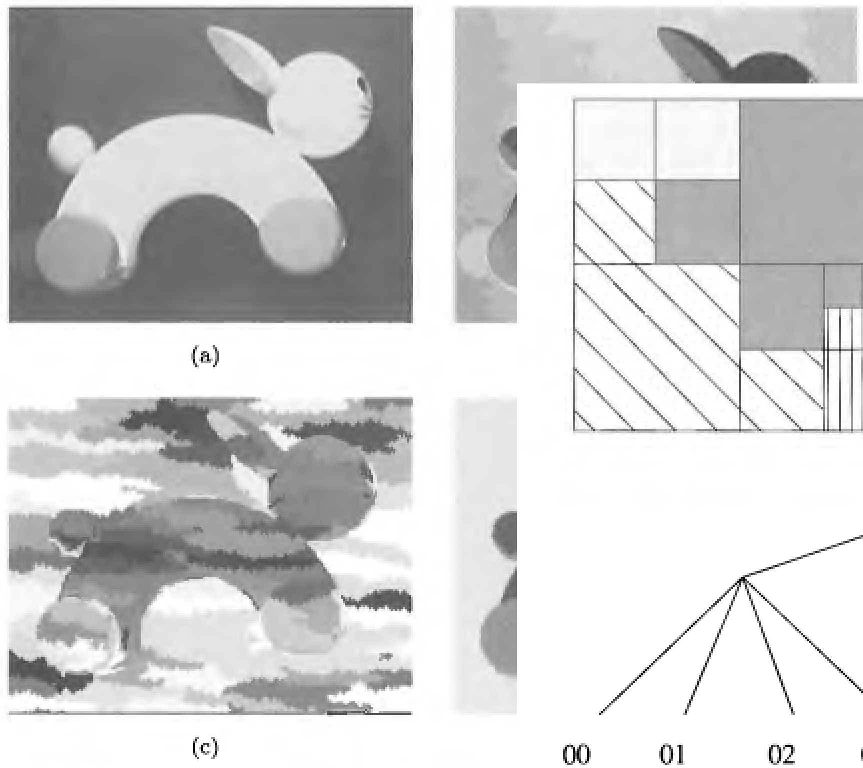


Figure 6.43: Region merging segmentation. (a) Origina
of the original image (in grayscale). (c) Recursive re
boundary melting. *Courtesy of R. Marik, Czech Technica*

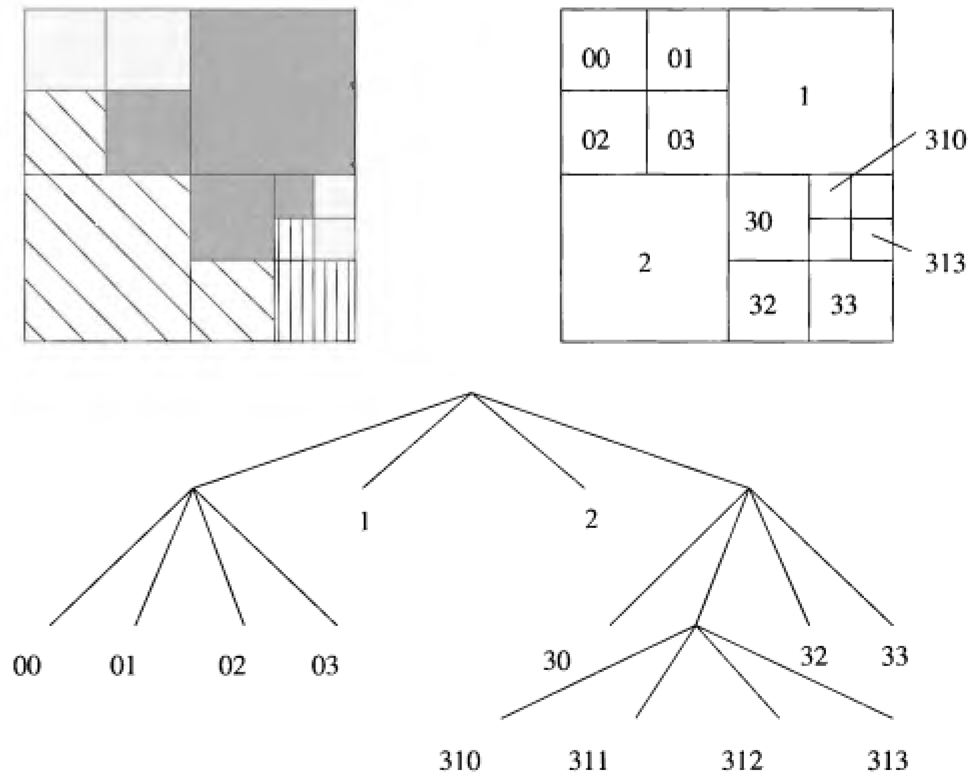


Figure 6.46: Segmentation quadtree.

Split / Merge

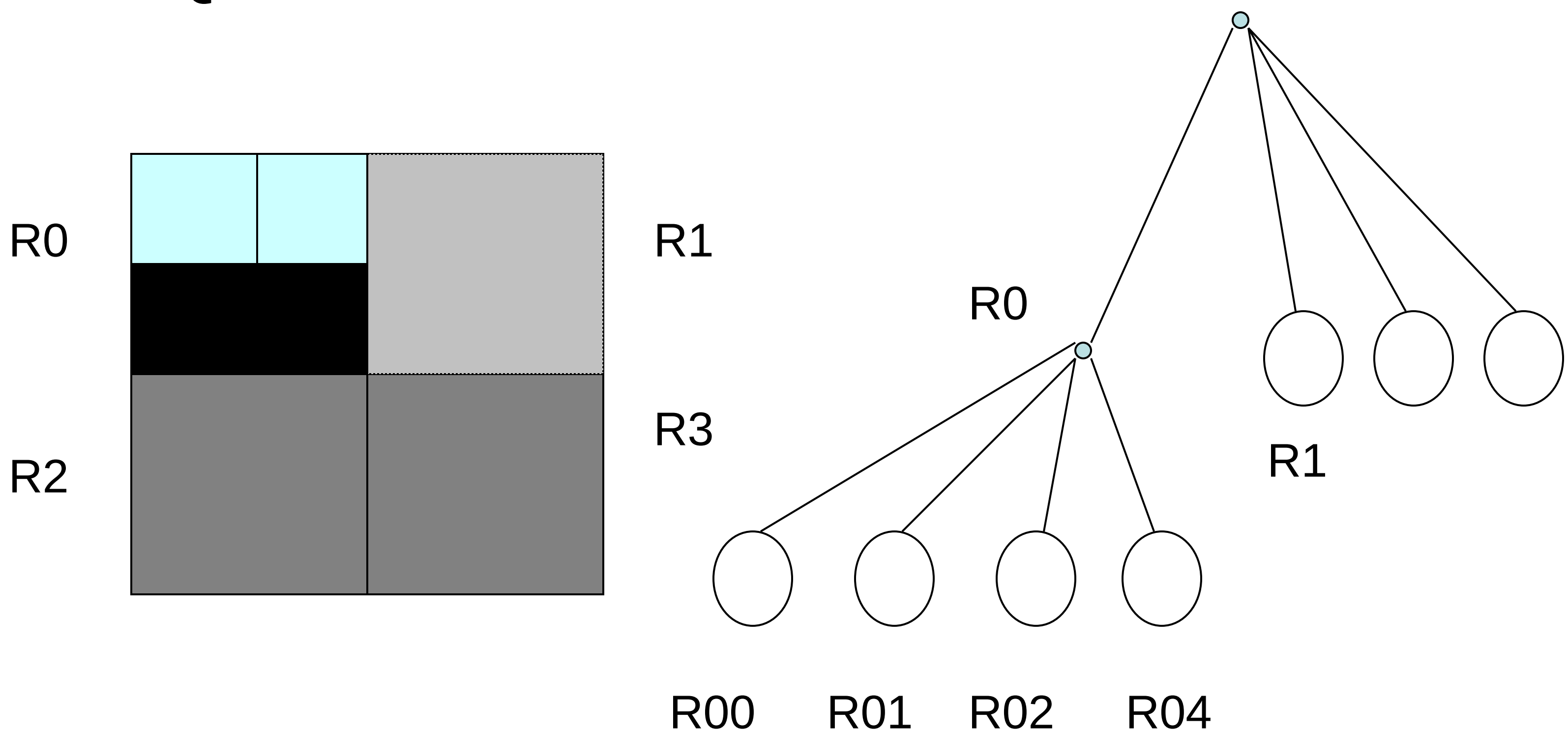
- The opposite approach to region growing is region shrinking (splitting).
- It is a top-down approach and it starts with the assumption that the entire image is homogeneous
- If this is not true , the image is split into four sub images
- This splitting procedure is repeated recursively until we split the image into homogeneous regions

Split / Merge

- If the original image is square N x N, having dimensions that are powers of 2(N = 2n):
- All regions produced but the splitting algorithm are squares having dimensions M x M , where M is a power of 2 as well (M=2m,M<= n).
- Since the procedure is recursive, it produces an image representation that can be described by a tree whose nodes have four sons each
- Such a tree is called a Quadtree.

Split / Merge

Quadtree



Split / Merge

- Splitting techniques disadvantage, they create regions that may be adjacent and homogeneous, but not merged.
- Split and Merge method – It is an iterative algorithm that includes both splitting and merging at each iteration:

Split / Merge

- If a region R is inhomogeneous (P(R)= False) then is split into four sub regions
- If two adjacent regions Ri,Rj are homogeneous (P(Ri U Rj) = TRUE), they are merged
- The algorithm stops when no further splitting or merging is possible

Split / Merge

- The split and merge algorithm produces more compact regions than the pure splitting algorithm

4.1.3 - Watershed Segmentation

- The concepts of **watersheds** and **catchment basins** are well known in topography.
- Watershed lines divide individual catchment basins.

4.1.3 - Watershed Segmentation

- Image data may be interpreted as a topographic surface where the gradient image gray-levels represent altitudes.
- Region edges correspond to high watersheds and low-gradient region interiors correspond to catchment basins.

4.1.3 - Watershed Segmentation

- Catchment basins of the topographic surface are homogeneous in the sense that all pixels belonging to the same catchment basin are connected with the basin's region of minimum altitude (gray-level) by a *simple path* of pixels that have monotonically decreasing altitude (gray-level) along the path.
- Such catchment basins then represent the regions of the segmented image.

4.1.3 - Watershed Segmentation

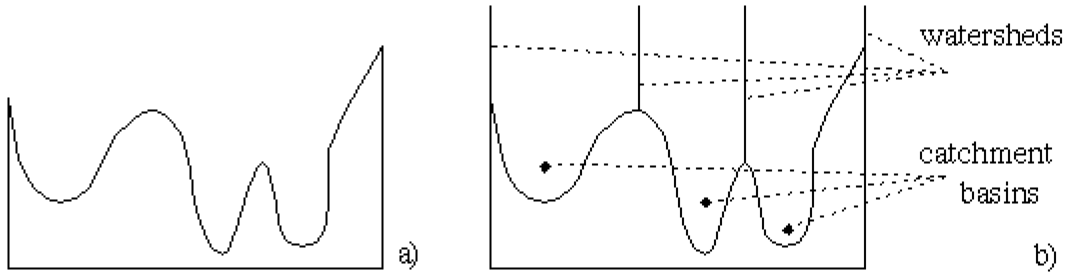


Figure 5.47 *One-dimensional example of watershed segmentation. (a) Gray level profile of image data. (b) Watershed segmentation – local minima of gray level (altitude) yield catchment basins, local maxima define the watershed lines.*

- Concept of watersheds and catchment basins is quite straightforward.
- Early watershed methods resulted in either slow or inaccurate execution.

4.1.3 - Watershed Segmentation

- Most of the existing algorithms start with extraction of potential watershed line pixels using a local 3 x 3 operation, which are then connected into geomorphological networks in subsequent steps. Due to the local character of the first step, these approaches are often inaccurate.
- A watershed transformation was also introduced in the context of mathematical morphology - computationally demanding and therefore time consuming.

4.1.3 - Watershed Segmentation

- Two basic approaches to watershed image segmentation.
- The first one starts with finding a *downstream* path from each pixel of the image to a local minimum of image surface altitude.
- The second approach is essentially dual to the first one; instead of identifying the downstream paths, the catchment basins fill from the bottom.

4.1.3 - Watershed Segmentation

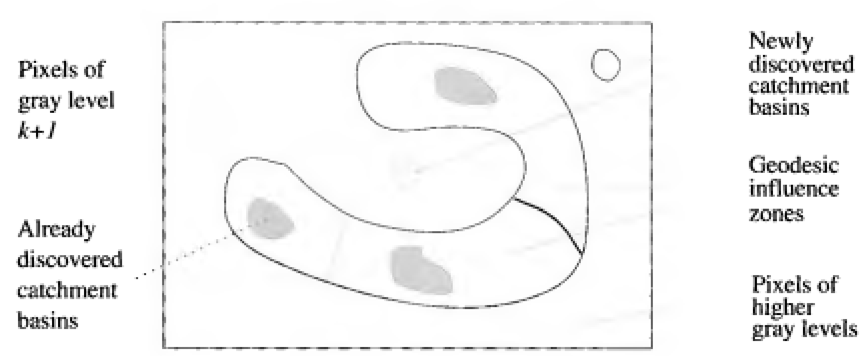


Figure 6.49: Geodesic influence zones of catchment basins.

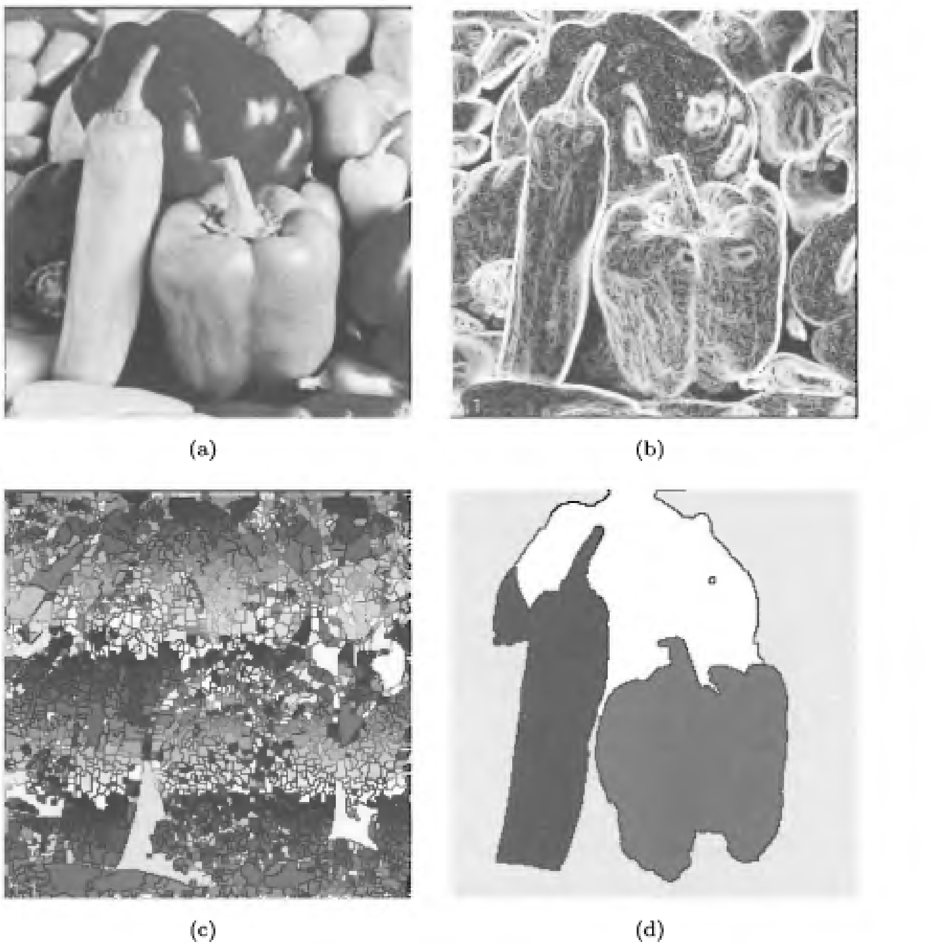


Figure 6.50: Watershed segmentation. (a) Original;. (b) Gradient image, 3 × 3 Sobel edge detection, histogram equalized. (c) Raw watershed segmentation. (d) Watershed segmentation using region markers to control oversegmentation. *Courtesy of W. Higgins, Penn State University.*

Knowledge points

- Thresholding segmentation
- Edge-based segmentation
  - Edge detectors
  - Hough transforms
- Region-based segmentation
  - Region Growing
  - Watershed Segmentation

Questions and Practices

- See "Practice 4 Image segmentation"