Digital Image Processong

Region and Shape Representation and Description

Christophoros Nikou

cnikou@cs.uoi.gr

Images taken from: R. Gonzalez and R. Woods. Digital Image Processing, Prentice Hall, 2008

Region and Shape Representation and Description

Well, but reflect; have we not several times acknowledged that names rightly given are the likeness and images of the things which they name?

Socrates

Contents

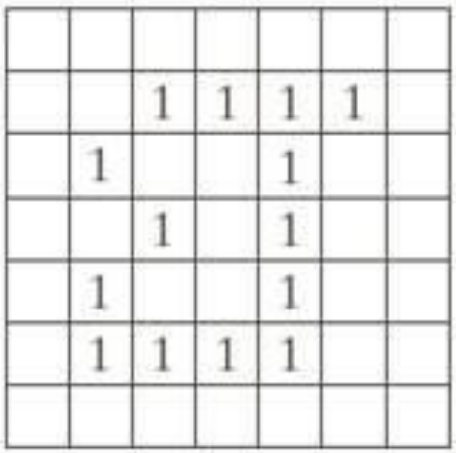
- After performing image segmentation, a region may be represented in terms of
  - external characteristics (boundaries).
  - internal characteristics (texture).
- A shape may be considered as a filled region with a unique value, e.g. $f(x,y)=1$.

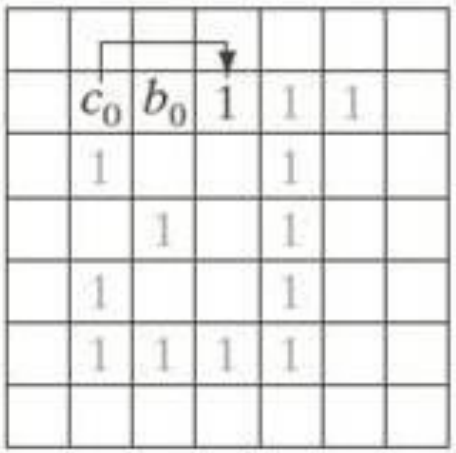
Boundary following

- Several algorithms require the points in an ordered clockwise (or counterclockwise) direction.
- We will describe an algorithm whose output is an ordered sequence of points.
  - Binary images (object and background points).
  - Images are padded with a border of zeros to avoid object merging with the image borders.
  - We limit the discussion to single regions. The extension is straightforward.

Boundary following (cont.)

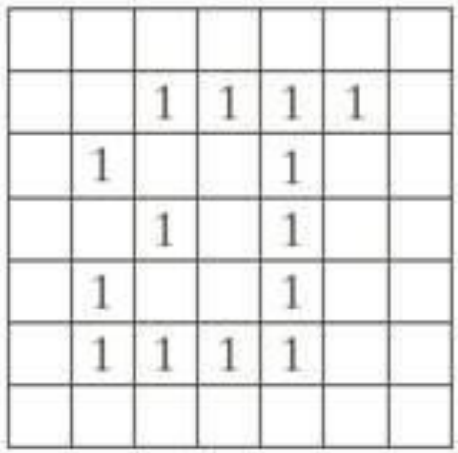
- Given a binary region R or its boundary, the algorithm for following the border of R:
  1. Let the starting point b_0, be the uppermost, leftmost point in the image labeled 1.
  2. Denote by c_0 the west neighbor of b_0. c_0 is always a background point.

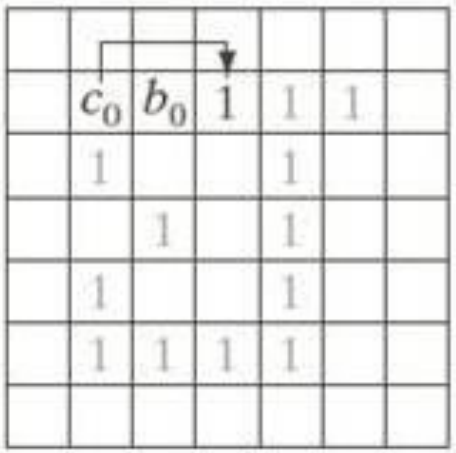




Boundary following (cont.)

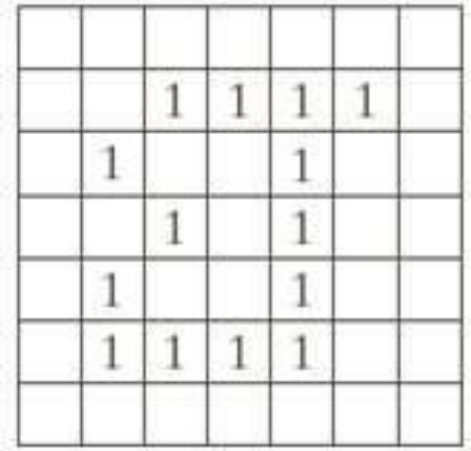
3. Examine the 8-neighbors of b_0, starting at c_0 and proceeding in a clockwise direction.
4. Let b_1 denote the first neighbor encountered whose value is 1.
5. Let c_1 denote the background point immediately preceding b_1 in the sequence.

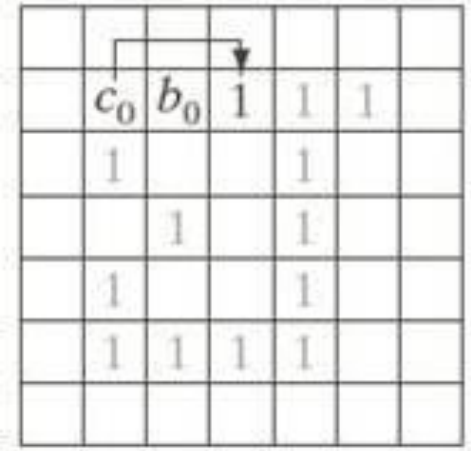


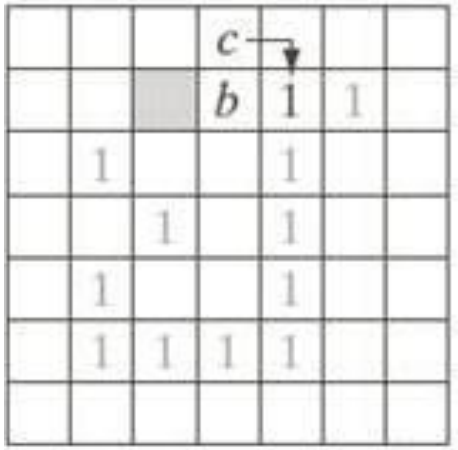


Boundary following (cont.)

6. Store the locations of b_0 and b_1 for use in Step 10.
7. Let $b=b_1$ and $c=c_1$.
8. Let the 8-neighbors of b, starting at c and proceeding in a clockwise direction, be denoted by $n_1, n_2, \ldots, n_8$. Find the first n_k labeled 1.



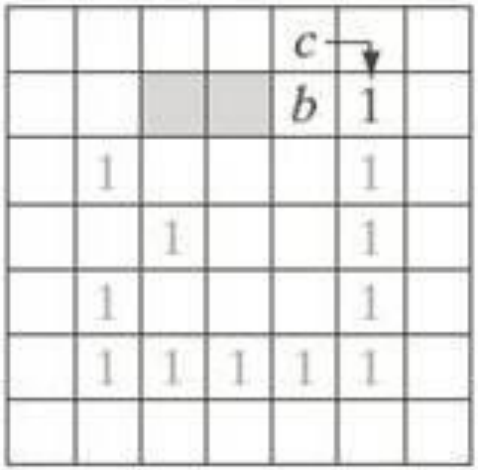


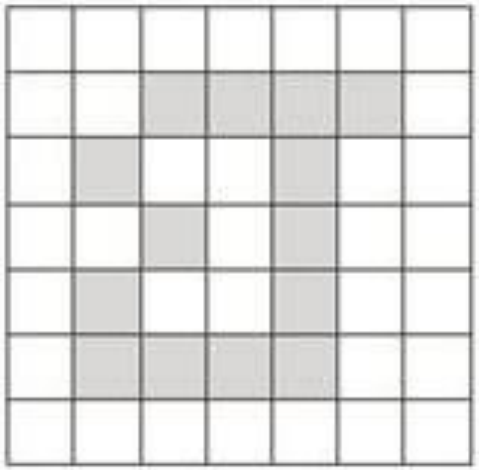


Boundary following (cont.)

9. Let $b=n_k$ and $c=n_{k-1}$.
10. Repeat steps 8 and 9 until $b=b_0$, that is, we have reached the first point and the next boundary point found is b_1.

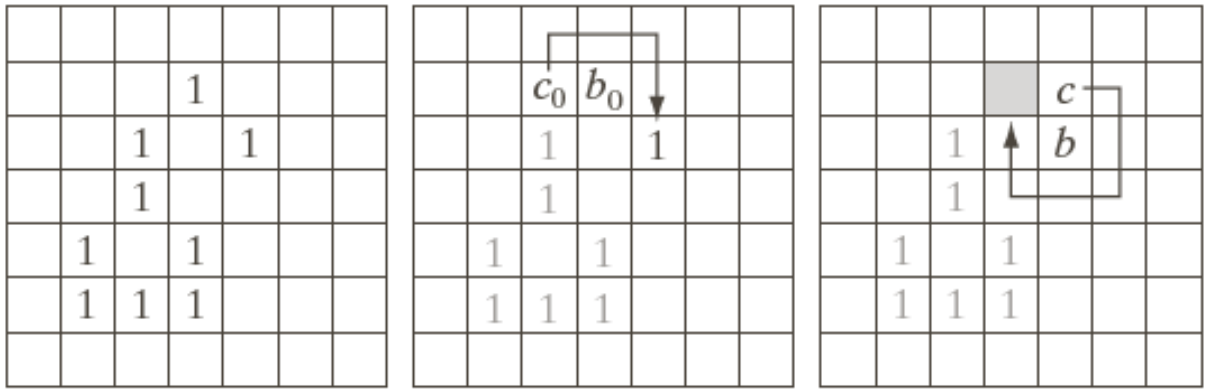
- The algorithm is due to G. Moore [1968]





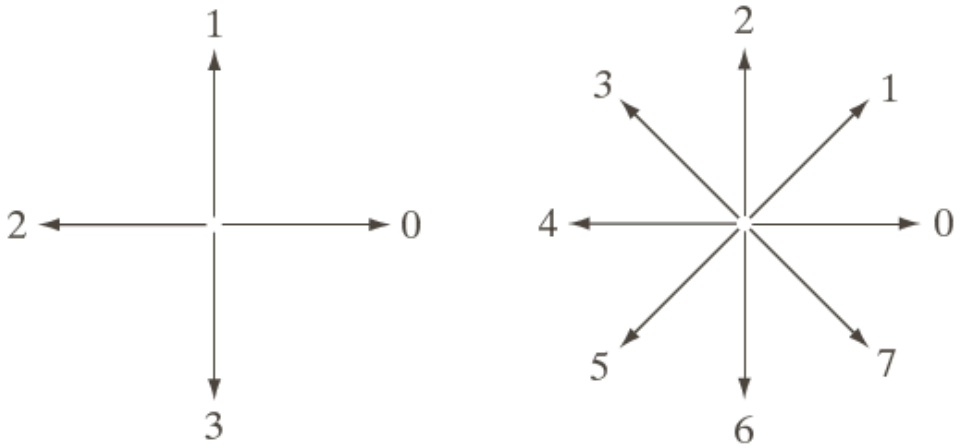
Boundary following (cont.)

- The need for the stopping rule "… and the next boundary point found is b_1" is shown below.
- We would only include the spur at the right if we stop when we reach the initial point without checking the next point.



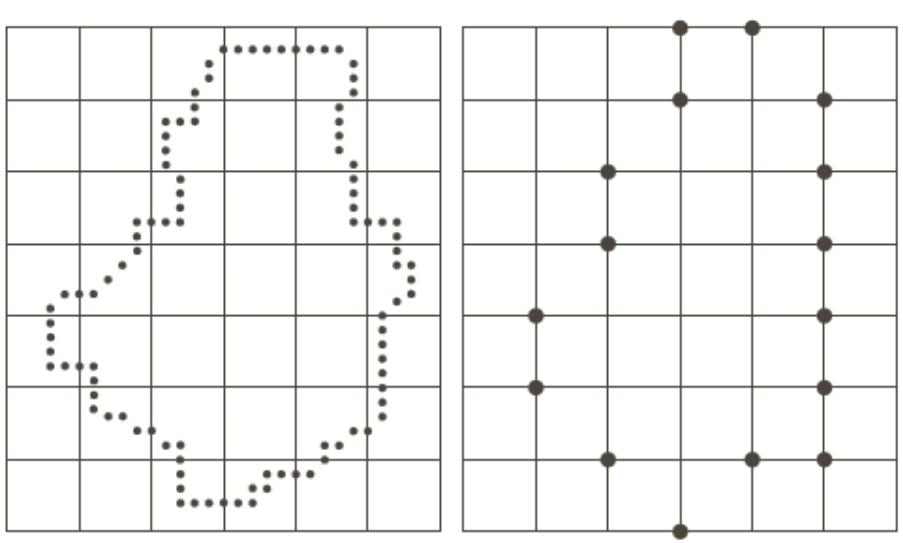
Chain codes

- Freeman codes [1961] represent a boundary by the sequence of straight line segments of specified length and direction.
- The direction is coded by a numbering scheme (4 or 8-connectivity).



Chain codes (cont.)

- To avoid noise degradation and long chains a resampling of the image grid is commonly used to describe the boundary at a coarser level.



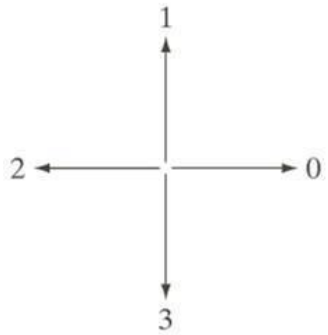
Chain codes (cont.)

- The chain code depends on the starting point.
- To normalize it, we treat the code as a circular sequence of direction numbers and redefine the starting point so that the resulting sequence forms an integer of minimum magnitude.
- To account for rotation, we use the first differences of the chain code instead of the code itself.

Chain codes (cont.)

- The first difference is obtained by counting the number of direction changes that separate two adjacent elements of the code.
- For instance, the first difference of the chain code 10103322 (e.g. in a counterclockwise direction) is 3133030.

1 -> 0 : 3 times
counter
clockwise
0 -> 1 : 1 time
1 -> 0 : 3
0 -> 3 : 3
3 -> 3 : 0
3 -> 2 : 3
2 -> 2 : 0

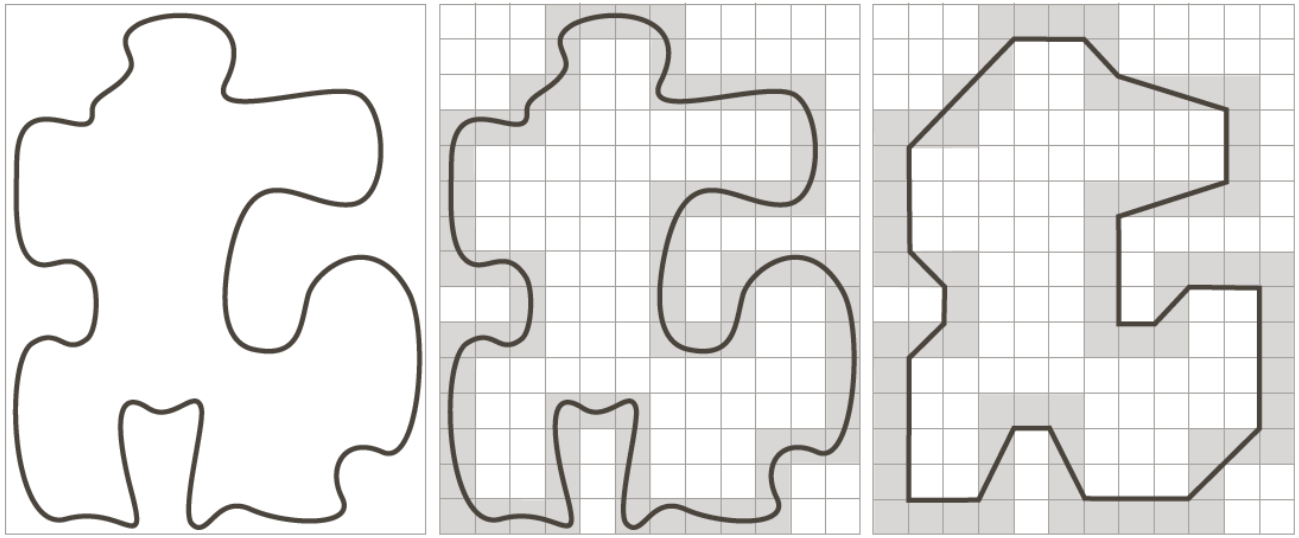


Minimum perimeter polygons

- A digital boundary may be approximated by a polygon with arbitrary accuracy.
- The goal is to capture the essence of the shape using the fewest possible segments.
- Non trivial and time-consuming problem.
- The minimum perimeter polygon (MPP) approximation is of modest complexity and provides good representations for image analysis applications.

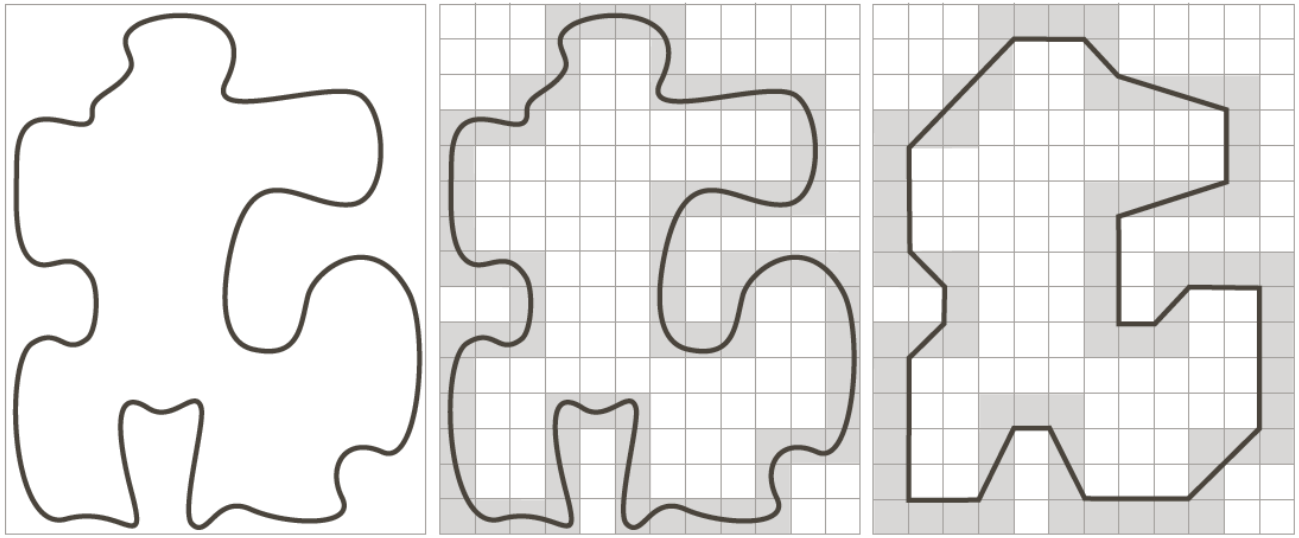
Minimum perimeter polygons (cont.)

- The idea is to enclose a boundary (think of it as a rubber band) by a set of concatenating cells.
- The boundary is allowed to shrink but it is constrained by the inner and outer walls of the bounding region defined by the cells.
- Ultimately, the shape shrinking provides the MPP.



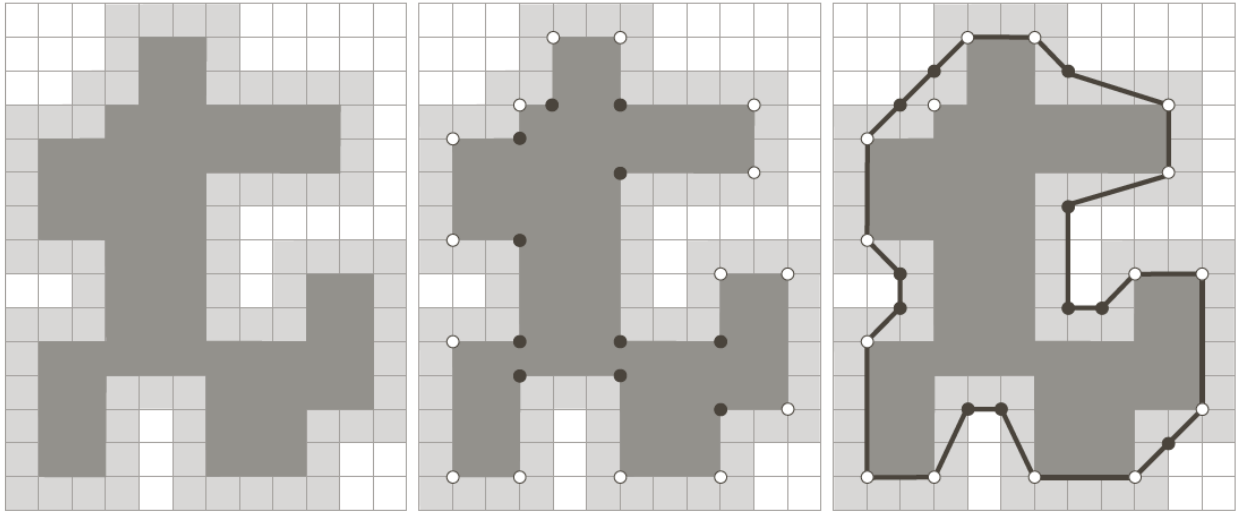
Minimum perimeter polygons (cont.)

- The vertices of the MPP coincide with corners of the inner or the outer wall.
- The size of the cells determines the accuracy of the representation.
- The objective is to use the largest possible cell size acceptable in a given application.



Minimum perimeter polygons (cont.)

- The shape of the object enclosed by the inner wall of the light gray cells is shown in dark gray.
- Traversing the boundary (counterclockwise) encounters convex (white dots) or concave (black dots) vertices.
- The vertices of the MPP coincide either with convex vertices in the inner wall or with the “mirrors” of the concave vertices in the outer wall.



Minimum perimeter polygons (cont.)

- Useful: the orientation of triplets of points:

$$a=(x_1,y_1),\ b=(x_2,y_2),\ c=(x_3,y_3)$$

$$A=\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}$$

$$\det(A)=\begin{cases} >0 & (a,b,c) \text{ is a counterclockwise sequence} \\ =0 & a,b,c \text{ are colinear} \\ <0 & (a,b,c) \text{ is a clockwise sequence} \end{cases}$$

- Convenient notation: $\text{sgn}(a,b,c)=\det(A)$

Minimum perimeter polygons (cont.)

- $\text{sgn}(a,b,c)>0$ indicates that point c lies on the positive side of the line passing from (a,b).
- $\text{sgn}(a,b,c)<0$ indicates that point c lies on the negative side of the line passing from (a,b).
- **Note:** $\text{sgn}(a,b,c)=\text{sgn}(c,a,b)$ because the direction of traversal is the same.
  - However, the geometrical interpretation is different. For example, $\text{sgn}(c,a,b)>0$ indicates that point b lies on the negative side of the line passing from (c,a).

Minimum perimeter polygons (cont.)

- Data preparation
  - List of coordinates of each vertex.
  - Label each vertex as W (convex) or B (concave).
  - List of the mirrors of B vertices.
  - Vertices must be in sequential order.
  - The first vertex V_0 is the uppermost leftmost vertex.
    - It is always a W vertex (proof skipped).
  - The algorithm uses a white crawler W_C and a black crawler B_C crawling along the convex (W) and mirrored concave (B) vertices respectively.

Minimum perimeter polygons (cont.)

- Initialization: $W_C = B_C = V_0$.
- V_L is the last vertex examined
- V_K is the current vertex being examined.
- $\text{sgn}(V_L, W_C, V_K)>0$ or V_K lies to the positive side of the line through (V_L, W_C).
  - The next MPP vertex is W_C,
  - $V_L = W_C$
  - Continue with the next vertex after V_L.

Minimum perimeter polygons (cont.)

- $\text{sgn}(V_L, W_C, V_K) \leq 0$ and $\text{sgn}(V_L, B_C, V_K) \geq 0$
  - V_K becomes a candidate MPP vertex.
  - If V_K is convex then $W_C = V_K$
  - Else $B_C = V_K$.
  - Continue with the next vertex in the list.
- $\text{sgn}(V_L, W_C, V_K) \leq 0$ and $\text{sgn}(V_L, B_C, V_K) < 0$
  - B_C becomes a candidate MPP vertex.
  - $V_L = B_C$
  - Reinitialize the algorithm by setting $W_C = B_C = V_L$
  - Continue with the next vertex in the list.

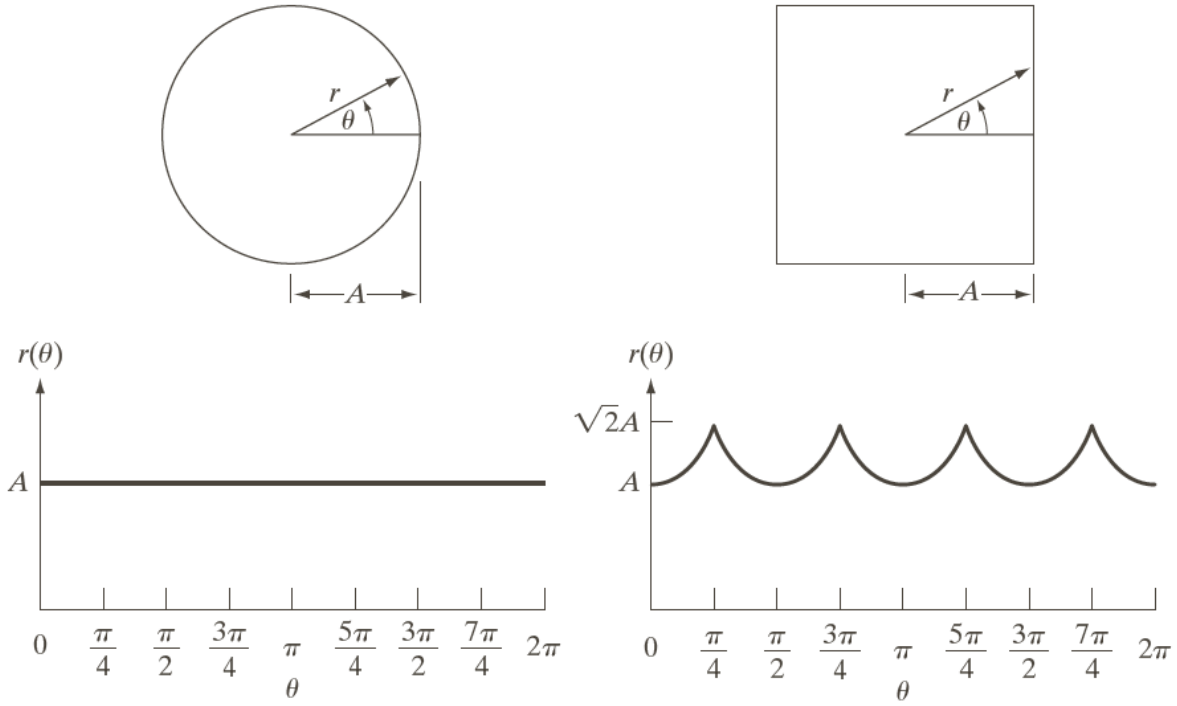
Minimum perimeter polygons (cont.)

a b c
d e f
g h i

FIGURE 11.8
(a) 566 × 566 binary image. (b) 8-connected boundary. (c) through (i), MMPs obtained using square cells of sizes 2, 3, 4, 6, 8, 16, and 32, respectively (the vertices were joined by straight lines for display). The number of boundary points in (b) is 1900. The numbers of vertices in (c) through (i) are 206, 160, 127, 92, 66, 32, and 13, respectively.

Signatures

- 1D functional representation of a boundary.
- A simple way s to plot the distance from the centroid as a function of the angle.



Signatures (cont.)

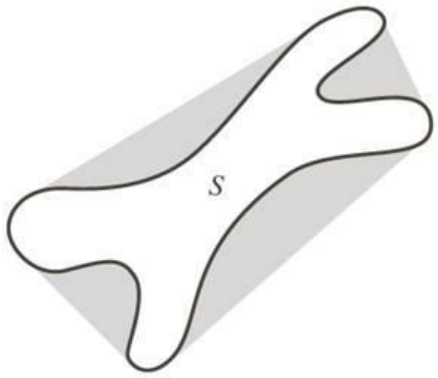
- Translation invariant but not rotation invariant.
  - We have to find a way to select the same starting point.
    - Select the point which is farthest from the centroid (not unique for a family of shapes).
    - Select the point on the largest eigen axis.
    - Obtain the chain code and use the first differences.
- Scale changes
  - Scale to [0, 1]. Depends on min and max of the function. Noise affects this type of scaling.
  - Divide each sample by the variance of the signature.

Signatures (cont.)

- Distance vs angle is not the only way to obtain a signature.
- Line tangent to the boundary vs a reference line.
  - Horizontal segments in the signature would correspond to straight lines as the tangent would be constant there.
- Slope density function
  - Histogram of tangent-angle values at segments of the boundary of a certain length.
  - Peaks at sections of boundary with constant tangent and valleys in sections with rapidly varying angles.

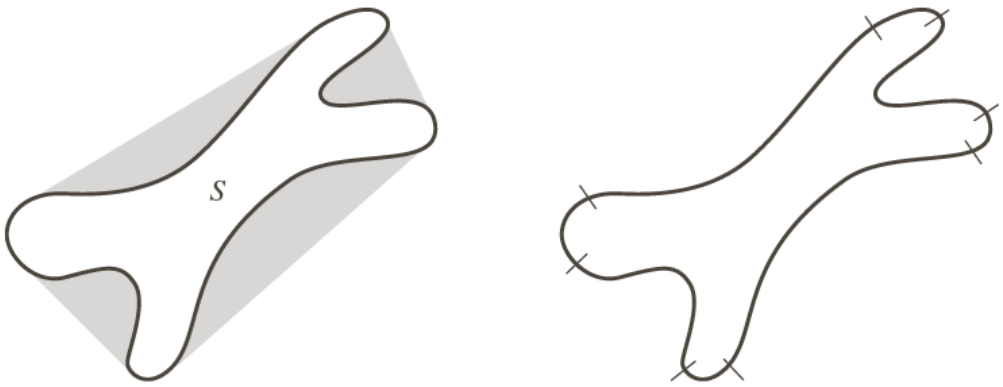
Boundary segments

- Decomposition into segments reduces the boundary complexity.
- Attractive approach in presence of concavities.
- We employ
  - The convex hull H of the region S:
    - The smallest convex set containing S.
  - The convex deficiency H-S of S .



Boundary segments (cont.)

- The boundary is partitioned by following the contour of S and marking the points at which a transition is made into or out of the convex deficiency.
- Smoothing may be necessary to remove irregularities (noise).



Skeleton by medial axis transformation

- It may be obtained by region thinning using morphological operators
  - The skeleton is not always connected.
- The **medial axis transformation** (MAT) [Blum 1967] of a region R with border B corrects this drawback:
  - For each point p in R, we find each closest neighbor in B.
  - If p has more than one neighbor, it is said to belong to the medial axis of R.
    - The definition of a distance is crucial (e.g. Euclidean).

Skeleton by medial axis transformation (cont.)

- MAT calculation is computationally expensive.
- Numerous algorithms have been proposed for improving the execution time.
- They are based on thinning algorithms iteratively deleting boundary points subject to the constraints that the deletion of points:
  - does not remove end points.
  - does not break connectivity.
  - does not cause excessive erosion of the region.

Skeleton by medial axis transformation (cont.)

- Such an iterative algorithm, consists of two thinning steps and it is based on
  - The number of nonzero neighbors $N(p_1)$ of a pixel p.
  - The number of 0-1 transitions $T(p_1)$ in the ordered sequence $p_2, p_3, \ldots, p_8, p_9, p_2$.

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

- Example:

  $N(p_1) = 4$, $T(p_1)=3$.

0	0	1
1	p_1	0
1	0	1

Skeleton by medial axis transformation (cont.)

- Step 1 flags a boundary pixel p_1 for deletion if the following conditions are satisfied:
  a) $2 \leq N(p_1) \leq 6$
  b) $T(p_1) = 1$
  c) $p_2 * p_4 * p_6 = 0$
  d) $p_4 * p_6 * p_8 = 0$

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Skeleton by medial axis transformation (cont.)

- In Step 2, conditions a) and b) remain the same, but conditions c) and d) are changed:

a) $2 \le N(p_1) \le 6$

b) $T(p_1) = 1$

c) $p_2 * p_4 * p_8 = 0$

d) $p_2 * p_6 * p_8 = 0$

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Skeleton by medial axis transformation (cont.)

- Step 1 is applied to every contour pixel and if all conditions are satisfied the pixel is flagged for deletion.
- However, it is not deleted until all contour pixels have been processed in order not to change the structure of the data during execution of Step 1.
- Then, Step 2 is applied to the remaining border pixels in exactly the same manner.
- The procedure is repeated until no points are deleted.

Skeleton by medial axis transformation (cont.)

a) $2 \leq N(p_1) \leq 6$

– The condition is violated when contour pixel p_1 has only one or seven neighbors.

- One neighbor implies that p_1 is the endpoint of a skeleton and obviously it should not be deleted.
- Deleting p_1 if it has seven neighbors would cause erosion into the region.

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Skeleton by medial axis transformation (cont.)

b) $T(p_1) = 1$

– The condition is violated when it is applied to points on a stroke 1 pixel thick.

– It prevents breaking thin segments.

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Skeleton by medial axis transformation (cont.)

c) $p_2 * p_4 * p_6 = 0$

d) $p_4 * p_6 * p_8 = 0$

– The conditions are satisfied simultaneously by the minimum set of values (p_4=0 *or* p_6=0) or (p_2=0 *and* p_8=0).

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

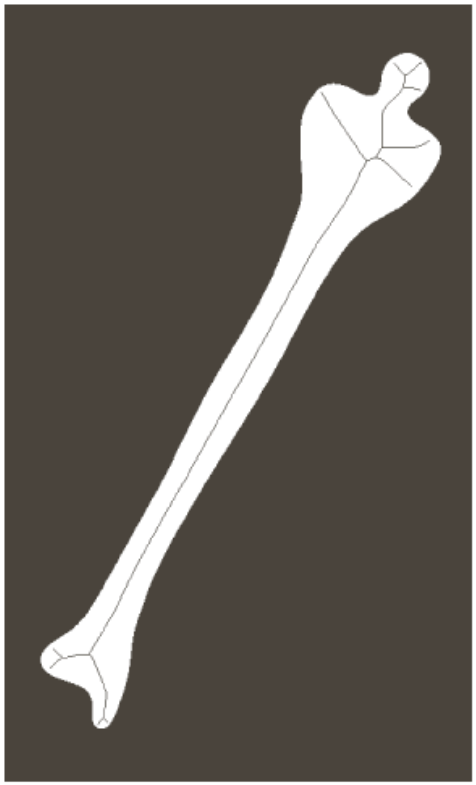
Skeleton by medial axis transformation (cont.)

- A point p_1 that satisfies these conditions and simultaneously the conditions a) and b) is either
  - an east or south boundary point.
  - a northwest corner point in the boundary.
- Therefore, p_1 is not a point of the skeleton and it should be removed.

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Skeleton by medial axis transformation (cont.)

– Similar assumptions hold for Step 2 concerning north or west boundary points and southeast corner points.



Boundary descriptors

- Geometric descriptors
- Shape numbers
- Fourier descriptors
- Statistical moments

Geometric boundary descriptors

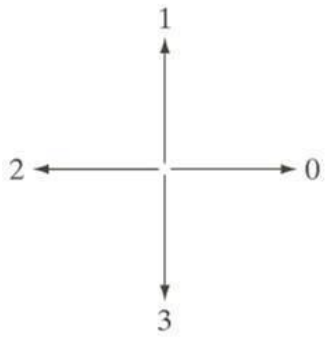
- Length
- Diameter $\text{Diam}(B) = \max_{i,j}[\text{Dist}(p_i, p_j)]$
- Major axis (connecting the two extreme points of the diameter)
- Minor axis (perpendicular to the major axis)
  - The two axes define the *basic rectangle* completely enclosing the boundary
- Eccentricity
  - ratio of the lengths between the major and the minor axis
- Curvature

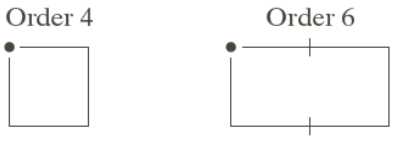
Shape numbers

- Recall that the chain code depends on the starting point.
  - We select the smallest number in its representation.
- The **shape number** of a boundary is the first difference of smallest magnitude.
  - First differences make it invariant to rotation.
- The order n of a shape number is defined as the number of digits in its representation.

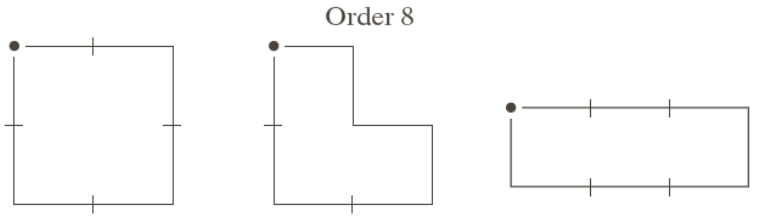
Shape numbers (cont.)

- Examples. All closed shapes of order n=4, 6 and 8.
- First differences are computed by treating the chain as a circular sequence.





	Order 4	Order 6
Chain code:	0 3 2 1	0 0 3 2 2 1
Difference:	3 3 3 3	3 0 3 3 0 3
Shape no.:	3 3 3 3	0 3 3 0 3 3



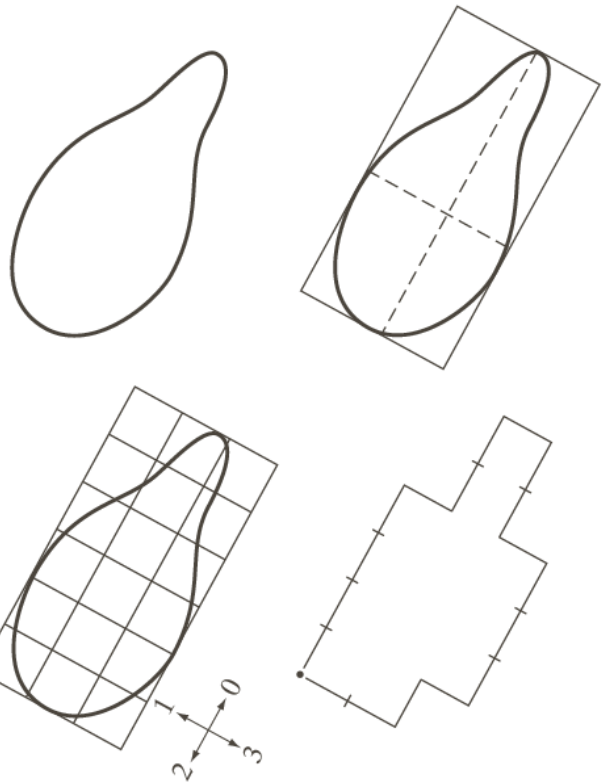
	Order 8		
Chain code:	0 0 3 3 2 2 1 1	0 3 0 3 2 2 1 1	0 0 0 3 2 2 2 1
Difference:	3 0 3 0 3 0 3 0	3 3 1 3 3 0 3 0	3 0 0 3 3 0 0 3
Shape no.:	0 3 0 3 0 3 0 3	0 3 0 3 3 1 3 3	0 0 3 3 0 0 3 3

Shape numbers (cont.)

- Although first differences are invariant to rotation the coded boundary depends on the orientation of the grid.
- Solution: align the chain-code grid with the sides of the basic rectangle.
  - Compute the basic rectangle and the order n of the shape.
  - Approximate the basic rectangle with a rectangle of order n (a rectangle with perimeter equal to n), e.g. if $n=12$, all the rectangles with a perimeter 12 are in $\{2\text{x}4, 3\text{x}3, 1\text{x}5\}$.
  - Select the one that best matches the eccentricity of the basic rectangle of the shape.
  - Establish the grid on the new rectangle.

Shape numbers (cont.)

- Suppose that $n=12$ for the specified boundary.



- Basic rectangle

- The closest rectangle is of size $3x6$
- The grid is aligned to it.

Chain code:	0	0	0	0	3	0	0	3	2	2	3	2	2	2	1	2	1	1
Difference:	3	0	0	0	3	1	0	3	3	0	1	3	0	0	3	1	3	0
Shape no.:	0	0	0	3	1	0	3	3	0	1	3	0	0	3	1	3	0	3

Fourier descriptors

- The two coordinates are treated as a complex number and the DFT is computed.

$$s(k) = x(k) + jy(k),\ k = 0,1,2,...,K-1$$

$$a(u) = \sum_{k=0}^{K-1} s(k) e^{-j\frac{2\pi u}{\mathrm{K}}k},\ u = 0,1,2,...,K-1$$

$$s(k) = \sum_{u=0}^{K-1} a(u) e^{j\frac{2\pi k}{\mathrm{K}}u},\ k = 0,1,2,...,K-1$$

Fourier descriptors (cont.)

- Approximation by the first $P<K$ first DFT coefficients, $a(u)=0$ for $u>P$:

$$\hat{s}(k)=\frac{1}{P}\sum_{u=0}^{P-1}a(u)e^{j\frac{2\pi k}{P}u},\ k=0,1,2,...,K-1$$

- Notice that the number of points is always K, that is the initial number of points is represented by fewer frequencies.
- Recall from DFT that high frequency components account for fine detail and low frequencies capture the global shape.

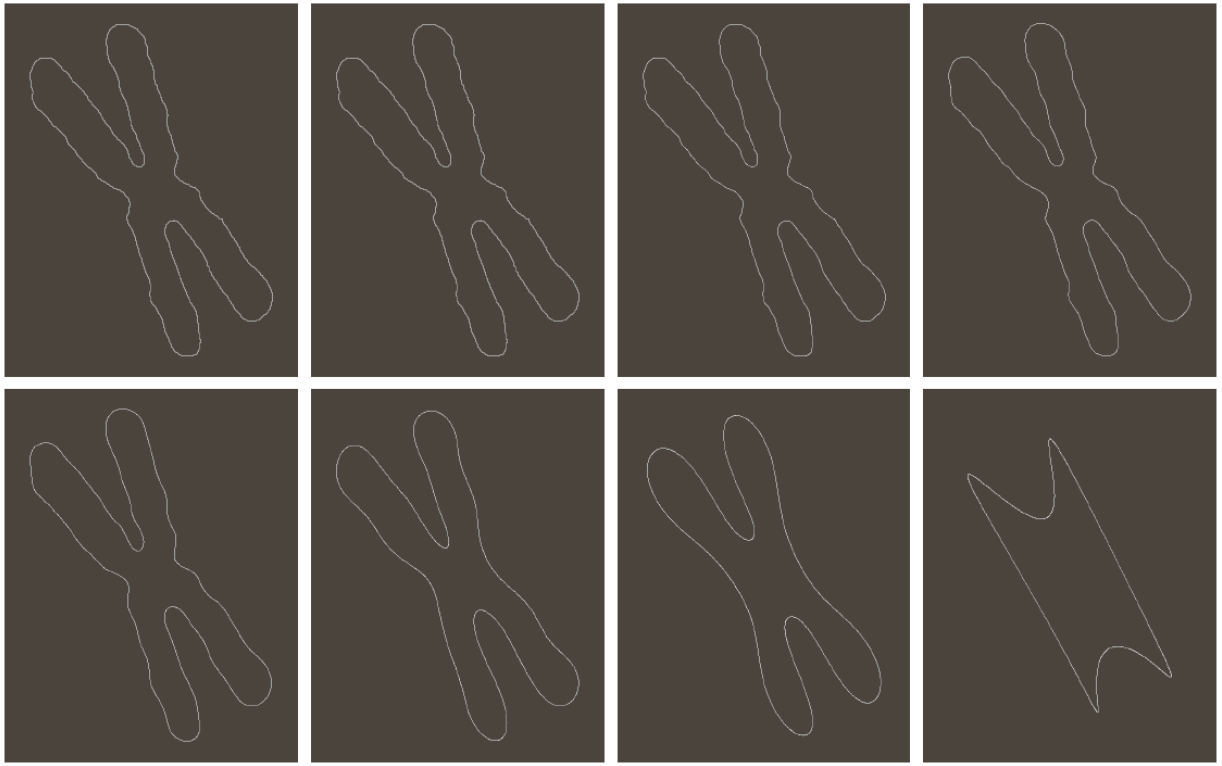
Fourier descriptors (cont.)

- The properties of the DFT hold.

Transformation	Boundary	Fourier Descriptor
Identity	$s(k)$	$a(u)$
Rotation	$s_r(k) = s(k)e^{j\theta}$	$a_r(u) = a(u)e^{j\theta}$
Translation	$s_t(k) = s(k) + \Delta_{xy}$	$a_t(u) = a(u) + \Delta_{xy}\delta(u)$
Scaling	$s_s(k) = \alpha s(k)$	$a_s(u) = \alpha a(u)$
Starting point	$s_p(k) = s(k - k_0)$	$a_p(u) = a(u)e^{-j2\pi k_0 u/K}$

- A change in the starting point affects all descriptors in a different but known way as the term multiplying $a(u)$ depends on u.

Fourier descriptors (cont.)

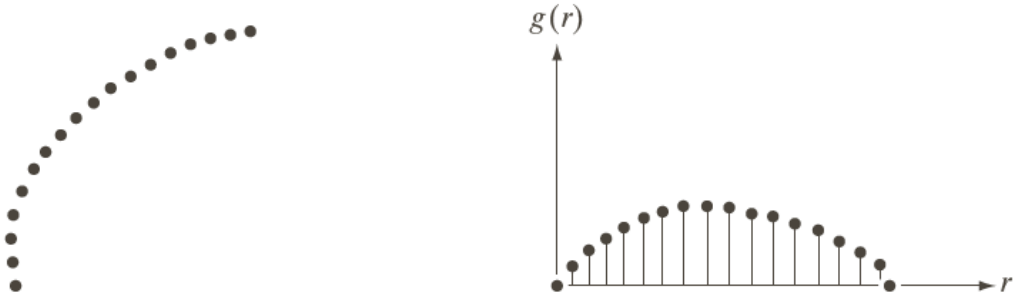


a b c d
e f g h

FIGURE 11.20 (a) Boundary of human chromosome (2868 points). (b)–(h) Boundaries reconstructed using 1434, 286, 144, 72, 36, 18, and 8 Fourier descriptors, respectively. These numbers are approximately 50%, 10%, 5%, 2.5%, 1.25%, 0.63%, and 0.28% of 2868, respectively.

Statistical moments

- A boundary segment may be represented as a 1D function $g(r)$ by connecting the end points and rotating the line segment to be horizontal.
- Normalizing $g(r)$ to unit area we can treat it as a histogram.



Statistical moments (cont.)

$$\mu_n(r) = \sum_{i=0}^{K-1} (r_i - m)^n g(r_i)$$

$$m = \sum_{i=0}^{K-1} r_i g(r_i)$$

- The second moment measures the spread of the curve around the mean
- The third moment measures the symmetry with respect to the mean.
- ….

Regional descriptors

- Simple descriptors
  - Geometric descriptors similar with the boundary case and descriptors including pixel intensities
- Topological descriptors
- Texture descriptors
  - Statistical approaches
  - Structural approaches
  - Spectral approaches
- Moment invariants

Simple region descriptors

- The **area** of a region S:

$$\mathrm{A}(S) = \sum_x \sum_y f(x, y)$$

$f(x,y)=1$ on the pixels belonging to the region and zero otherwise.

- The **perimeter** of a region S:

$$P(S) = \sum_i \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2}$$

where (x_i,y_i) are the coordinates of the i-th pixel of the boundary of the region.

Simple region descriptors (cont.)

- The **compactness** of a region S:

$$C(S) = \frac{4\pi A(S)}{P^2(S)} = \frac{A(S)}{P^2(S)/4\pi}$$

- The denominator is the area of a circle whose perimeter is $P(S)$.
- It measures the ratio of area of the shape and the circle that can be traced with the same perimeter.
- For a perfectly circular region $A(S)=1$.

Simple region descriptors (cont.)

- The **dispersion** of a region S:

$$I(S) = \frac{\pi \max_i \left(\left(x_i - \bar{x} \right)^2 + \left(y_i - \bar{y} \right)^2 \right)}{A(S)}$$

- It measures the ratio of the major chord length to the area.
- The numerator defines the area of the maximum circle enclosing the region.

Simple region descriptors (cont.)

- An alternative definition of **dispersion** of a region S measures the ratio of the maximum to the minimum radius:

$$I(S) = \frac{\max_i \left((x_i - \bar{x})^2 + (y_i - \bar{y})^2 \right)}{\min_i \left((x_i - \bar{x})^2 + (y_i - \bar{y})^2 \right)}$$

- It is the ratio between the radius of the maximum circle enclosing the region and the minimum circle that may be contained in the region.

Simple region descriptors (cont.)

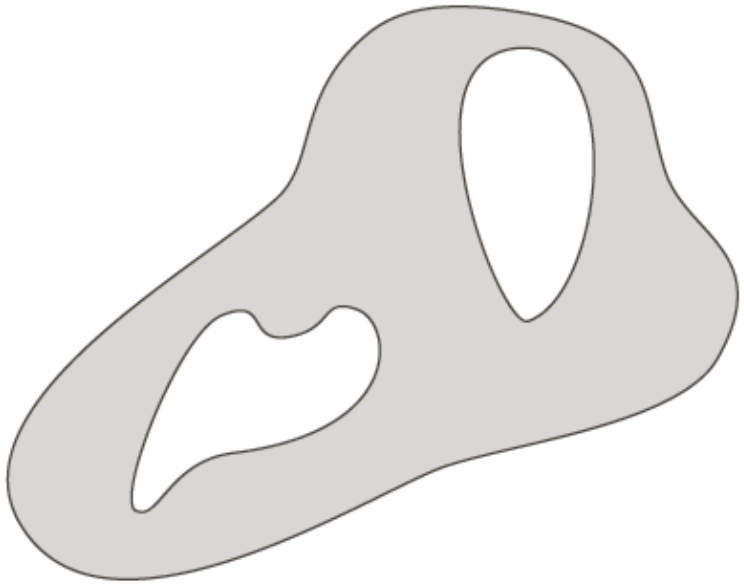
- Other regional descriptors include statistics on the intensity levels of the pixels in the region
  - mean
  - median
  - minimum
  - Maximum
  - Number of pixels with values above or below the mean

Topological descriptors

- Topology is the study of properties that are unaffected by any deformation.
  - Rubber-sheet distortions.
    - No tearings or joinings of the region.
- These properties do not depend on the notion of distance or a distance measure.
  - Number of connected components C.
  - Number of holes H.
- **Euler number**:

$$E = C - H$$

Topological descriptors (cont.)



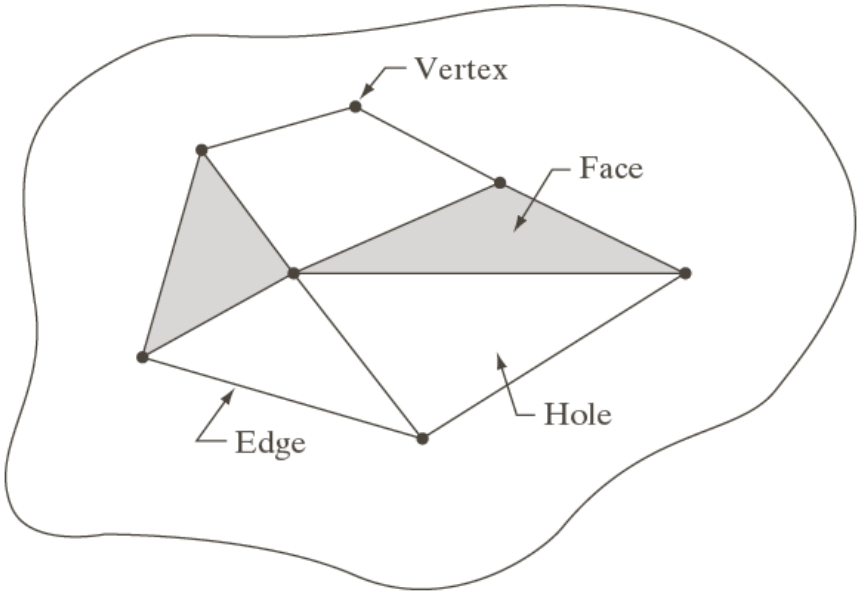
$$E = C - H = 1 - 2 = -1$$

Topological descriptors (cont.)

- Regions represented by straight line segments (*polygonal networks*) have a particular simple interpretation in terms of the Euler number.
- Classification of interior regions into faces and holes.
- The number of vertices (V) and the number of edges (Q) and the number of faces (F) give the **Euler formula**:

$$V - Q + F = C - H = E$$

Topological descriptors (cont.)



$$C - H = 1 - 3 = -2$$

$$V - Q + F = 7 - 11 + 2 = -2$$

Texture

- Basic texture representation
  - Statistical approaches
    - Histogram
    - Co-occurrence matrices.
  - Structural approaches
    - relational descriptors based on rules (we will look at them at the end of the section)
  - Spectral methods.

Histogram approaches

- Statistical moments of the histogram of a region.
  - Let $p(z)$ be the histogram of an image region with L possible gray levels z_i, $i=1,\ldots L$.
  - n^{th} moment:

$$\mu_n(z) = \sum_{i=0}^{L-1} (z_i - m)^n \, p(z_i), \quad m = \sum_{i=0}^{L-1} z_i \, p(z_i),$$

    - The third moment measures the skewness and fourth moment measures flatness of the histogram.

Histogram approaches (cont.)

– Contrast:

$$R(z) = 1 - \frac{1}{1 + \sigma^2(z)}$$

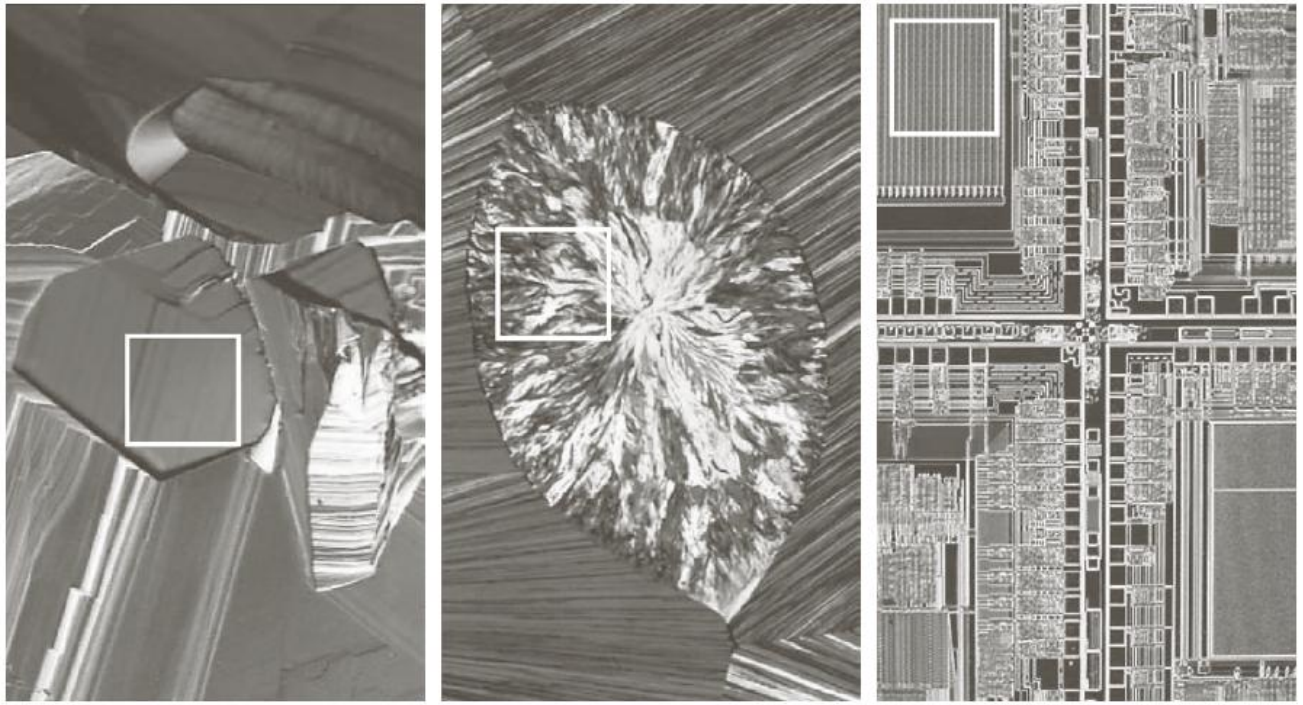
– Uniformity (maximum for uniform images):

$$U(z) = \sum_{i=0}^{L-1} p^2(z_i)$$

– Entropy:

$$E(z) = -\sum_{i=0}^{L-1} p(z_i) \log\left[p(z_i) \right]$$

Histogram approaches (cont.)



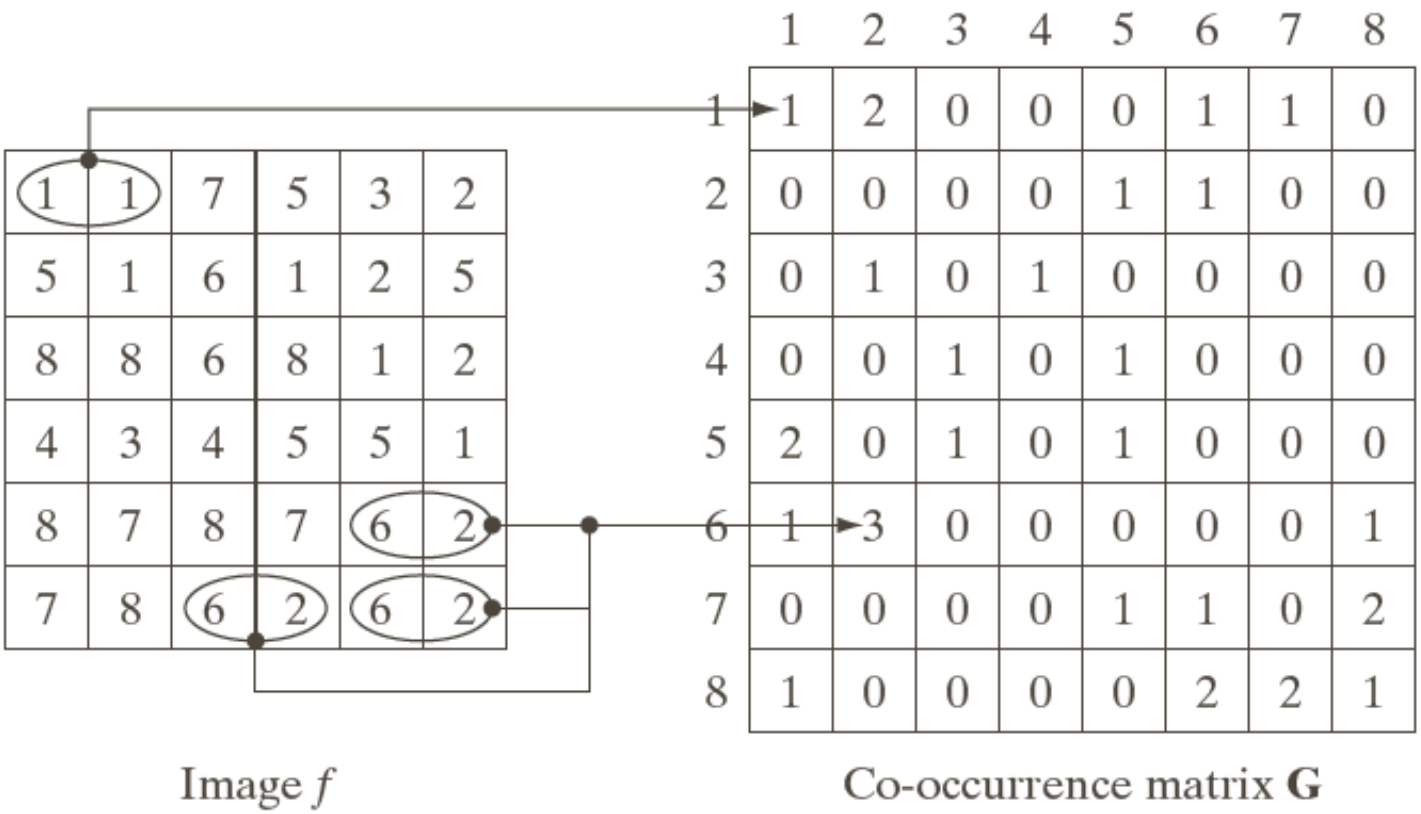
Texture	Mean	Standard deviation	R (normalized)	Third moment	Uniformity	Entropy
Smooth	82.64	11.79	0.002	−0.105	0.026	5.434
Coarse	143.56	74.63	0.079	−0.151	0.005	7.783
Regular	99.72	33.73	0.017	0.750	0.013	6.674

Co-occurrence Matrices

- Let Q be an operator that defines the position of two pixels relative to each other.
- Consider an image f, with L possible gray levels.
- Let $\mathbf{G}$ be a matrix whose element g_{ij} is the number of times that pixel pairs with intensities z_i and z_j occur in f in the position specified by Q.
- $\mathbf{G}$ is called co-occurrence matrix.

Co-occurrence Matrices (cont.)

- Quantization of intensities due to computational load. Many matrices per region. Here L=8 gray levels.



Co-occurrence Matrices (cont.)

$$p_{ij} = \frac{g_{ij}}{\sum_{i,j} g_{ij}}$$

$$P(i) = \sum_{j=1}^{K} p_{ij}$$

$$m_r = \sum_{i=1}^{K} iP(i)$$

$$\sigma_r^2 = \sum_{i=1}^{K} (i - m_r)^2 P(i)$$

Descriptor	Explanation	Formula
Maximum probability	Measures the strongest response of **G**. The range of values is [0, 1].	$\max_{i,j}(p_{ij})$
Correlation	A measure of how correlated a pixel is to its neighbor over the entire image. Range of values is 1 to −1, corresponding to perfect positive and perfect negative correlations. This measure is not defined if either standard deviation is zero.	$\sum_{i=1}^{K}\sum_{j=1}^{K}\frac{(i - m_r)(j - m_c)p_{ij}}{\sigma_r\sigma_c}$ $\sigma_r \neq 0; \sigma_c \neq 0$
Contrast	A measure of intensity contrast between a pixel and its neighbor over the entire image. The range of values is 0 (when **G** is constant) to $(K - 1)^2$.	$\sum_{i=1}^{K}\sum_{j=1}^{K}(i - j)^2 p_{ij}$
Uniformity (also called Energy)	A measure of uniformity in the range [0, 1]. Uniformity is 1 for a constant image.	$\sum_{i=1}^{K}\sum_{j=1}^{K}p_{ij}^2$
Homogeneity	Measures the spatial closeness of the distribution of elements in **G** to the diagonal. The range of values is [0, 1], with the maximum being achieved when **G** is a diagonal matrix.	$\sum_{i=1}^{K}\sum_{i=1}^{K}\frac{p_{ij}}{1 + \lvert i - j\rvert}$
Entropy	Measures the randomness of the elements of **G**. The entropy is 0 when all p_{ij}'s are 0 and is maximum when all p_{ij}'s are equal. The maximum value is $2 \log_2 K$. (See Eq. (11.3-9) regarding entropy).	$-\sum_{i=1}^{K}\sum_{i=1}^{K}p_{ij}\log_2 p_{ij}$

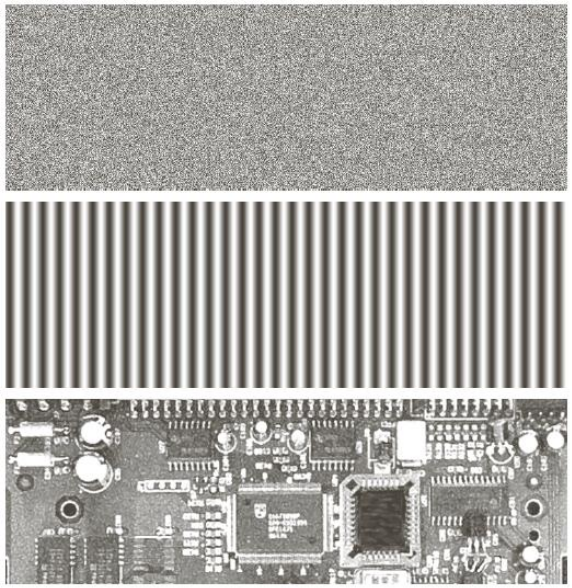
TABLE 11.3 Descriptors used for characterizing co-occurrence matrices of size $K \times K$. The term p_{ij} is the ijth term of **G** divided by the sum of the elements of **G**.

Co-occurrence Matrices (cont.)

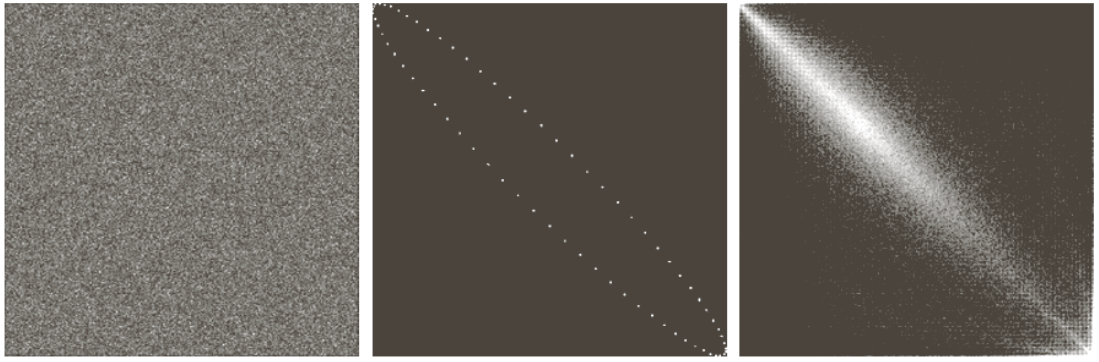
Random noise

Periodic texture (sine)

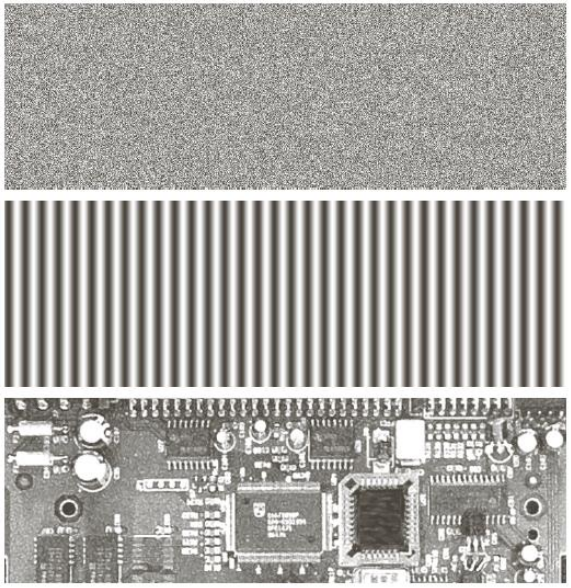
Mixed texture

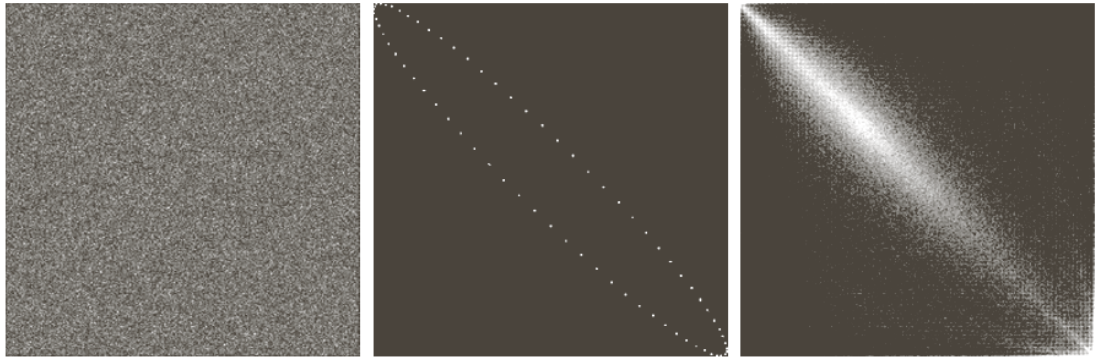


256x256 co-occurence matrices “one position immediately to the right”.



Co-occurrence Matrices (cont.)





Normalized Co-occurrence Matrix	Descriptor					
	Max Probability	Correlation	Contrast	Uniformity	Homogeneity	Entropy
G_1/n_1	0.00006	−0.0005	10838	0.00002	0.0366	15.75
G_2/n_2	0.01500	0.9650	570	0.01230	0.0824	6.43
G_3/n_3	0.06860	0.8798	1356	0.00480	0.2048	13.58

Co-occurrence Matrices (cont.)

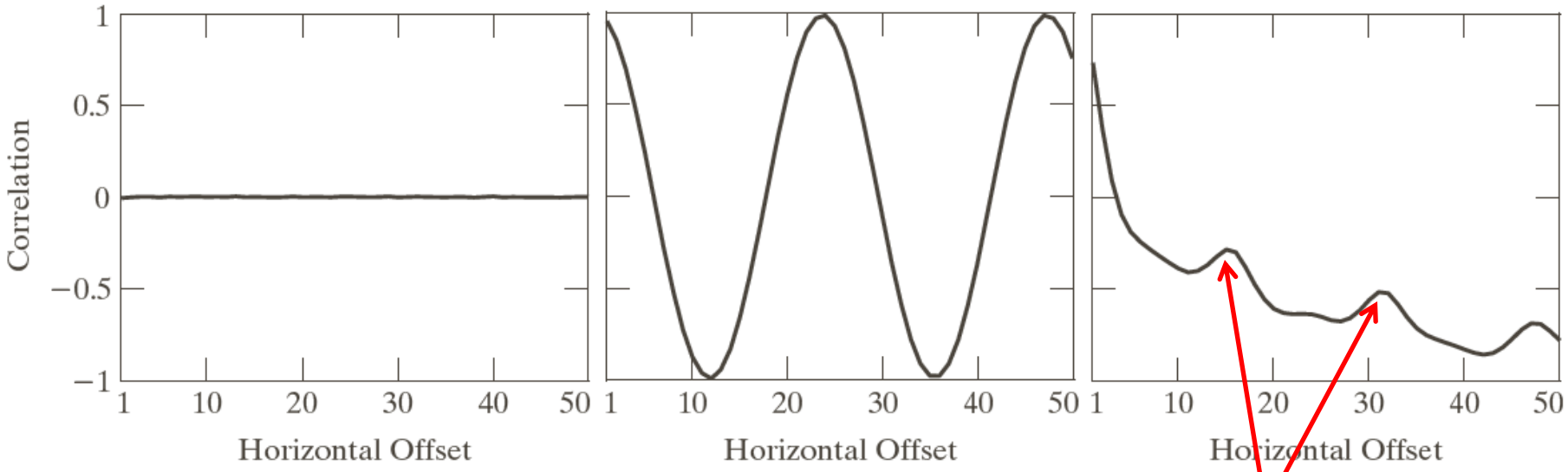
- Is there an image portion containing a certain texture (a repetitive pattern)?
- Sequences of co-occurrence matrices are employed.
  - the correlation descriptor for varying horizontal offset of adjacent pixels may be calculated.
- In the next experiment, this is performed for co-occurence matrices computed for horizontal offsets from 1 to 50 pixels.

Co-occurrence Matrices (cont.)

Noise image

Sine image

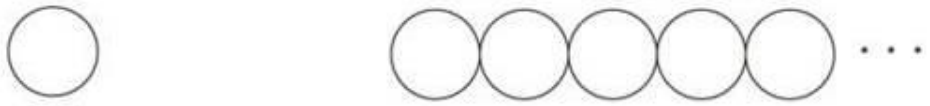
Circuit image



Disadvantage: co-occurrence matrices are not efficient for coarse texture description

Structural approaches

- Rule that indicates that the symbol S may be rewritten as aS: $S \rightarrow aS$
- Three repetitions of the rule yield the string $aaaS$.
- If a represents a circle and the meaning "circles to the right" is assigned to a string of the form aaa...then the rule allows the generation of the pattern:



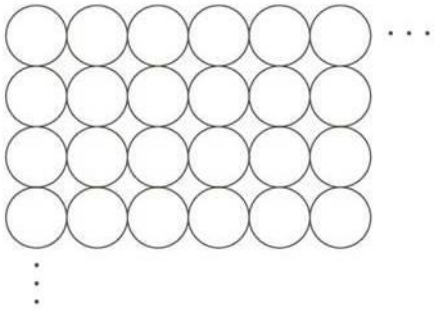
a $aaa...$

Structural approaches (cont.)

- We add new rules, for example:

$$S \to bA,\ A \to cA,\ A \to c,\ A \to bS,\ S \to a$$

- b means “circle down”.
- c means “circle to the left”.
- $aaabccbaa$ corresponds to a 3x3 matrix of circles.
- Larger texture patterns may be generated.
- We will treat the topic in detail later.



Spectral Approaches

- The Fourier transform (FT) is useful for description of the directionality of periodic or almost periodic structures.
  - Peaks in the FT give the principal direction of patterns.
  - The location of peaks gives the fundamental period of patterns.
  - The FT is symmetric around the origin and only half of the frequency plane needs to be considered.

Spectral Approaches (cont.)

- The spectrum is expressed in polar coordinates $S(r,\theta)$ for simplification
  - We define $S_r(\theta)$ which is a 1D function for a given r and $S_\theta(r)$ which is a 1D function for a given θ.
  - Analyzing $S_\theta(r)$ for a fixed value of θ yields the behavior of the spectrum along a radial direction from the origin.
  - Analyzing $S_r(\theta)$ for a fixed value of r yields the behavior of the spectrum along a circle of radius r which is centered at the origin.

Spectral Approaches (cont.)

- A more global description is obtained by integrating these functions:

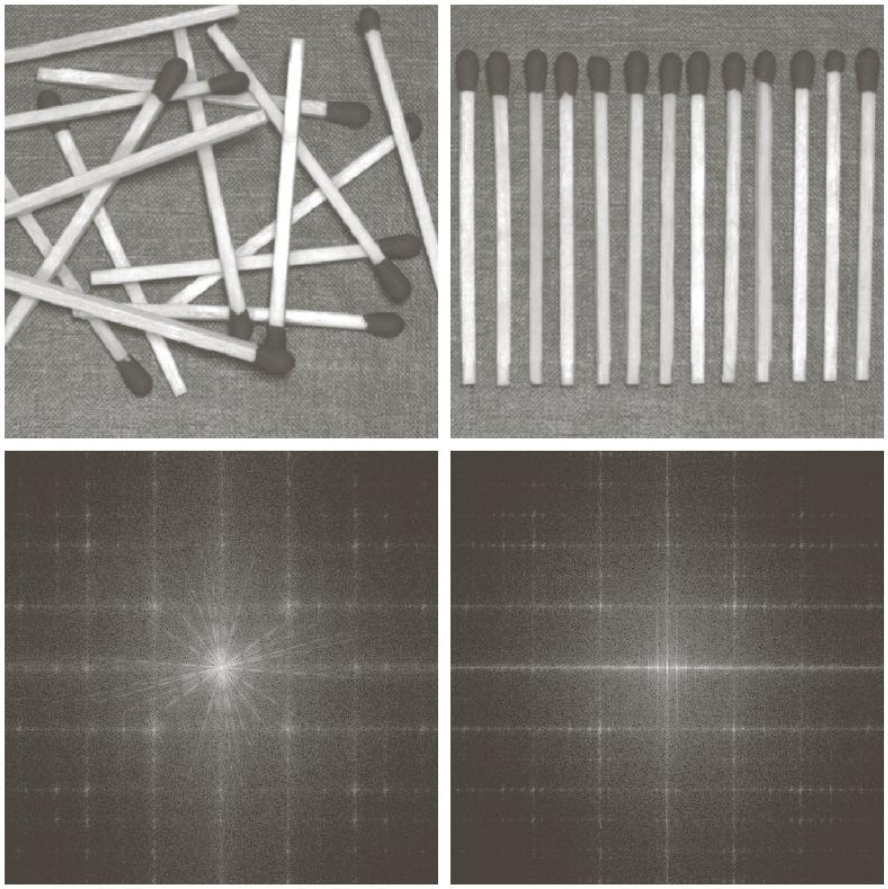
$$S(r) = \sum_{\theta=0}^{\pi} S_\theta(r) \qquad S(\theta) = \sum_{r=1}^{R_0} S_r(\theta)$$

  - with R_0 being a the radius of a circle centered at the origin.

- The result constitute a par of values $[S(r), S(\theta)]$ for each pair of coordinates (r,θ).

Spectral Approaches (cont.)

Images and FT



The periodic bursts extending in 2D in both spectra are due to the periodic texture of coarse background material.

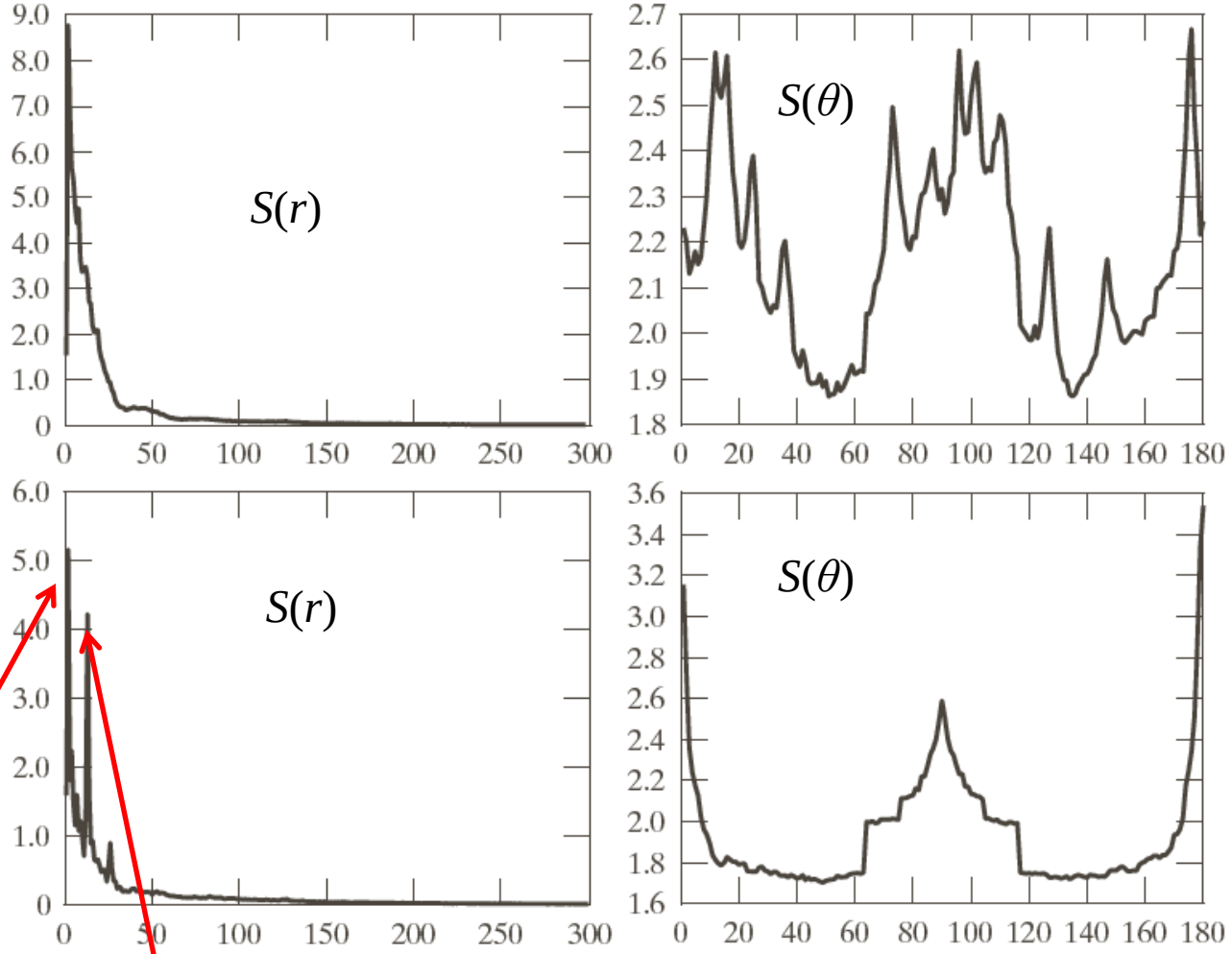
The other dominant components in the left are due to the random orientation of object edges.

In the right figure, the energy not associated with the background is along the horizontal axis.

Spectral Approaches (cont.)

Random

Ordered



Periodic horizontal repetition:
Matches and background.

θ= 0, 90 and 180, indicate that there is information at these orientations.

Moment invariants

- $2D$ moment of order $(p+q)$ of a MxN image is defined as:

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x,y),$$

$$p = 0,1,2,\ldots \text{ and } \; q = 0,1,2,\ldots$$

- It may be used on gray scale images as well as on shapes. For a shape boundary we may consider that $f(x,y)=1$ on the boundary and zero otherwise.

Moment invariants (cont.)

- The corresponding central moment of order $(p+q)$ is defined as:

$$\mu_{pq} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(x-\bar{x})^p (y-\bar{y})^q f(x,y),\ \bar{x} = \frac{m_{10}}{m_{00}},\ \ \bar{y} = \frac{m_{01}}{m_{00}}$$

$$p = 0,1,2,\ldots \text{ and } \ q = 0,1,2,\ldots$$

- The normalized central moments are defined as:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}},\quad \gamma = \frac{p+q}{2}+1$$

Moment invariants (cont.)

- A set of seven invariant moments can be derived from the second and third moments. Invariance is obtained with respect to:
  - translation
  - rotation
  - scale change
  - mirroring

Moment invariants (cont.)

$$\varphi_1 = \eta_{20} + \eta_{02}$$

$$\varphi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\varphi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

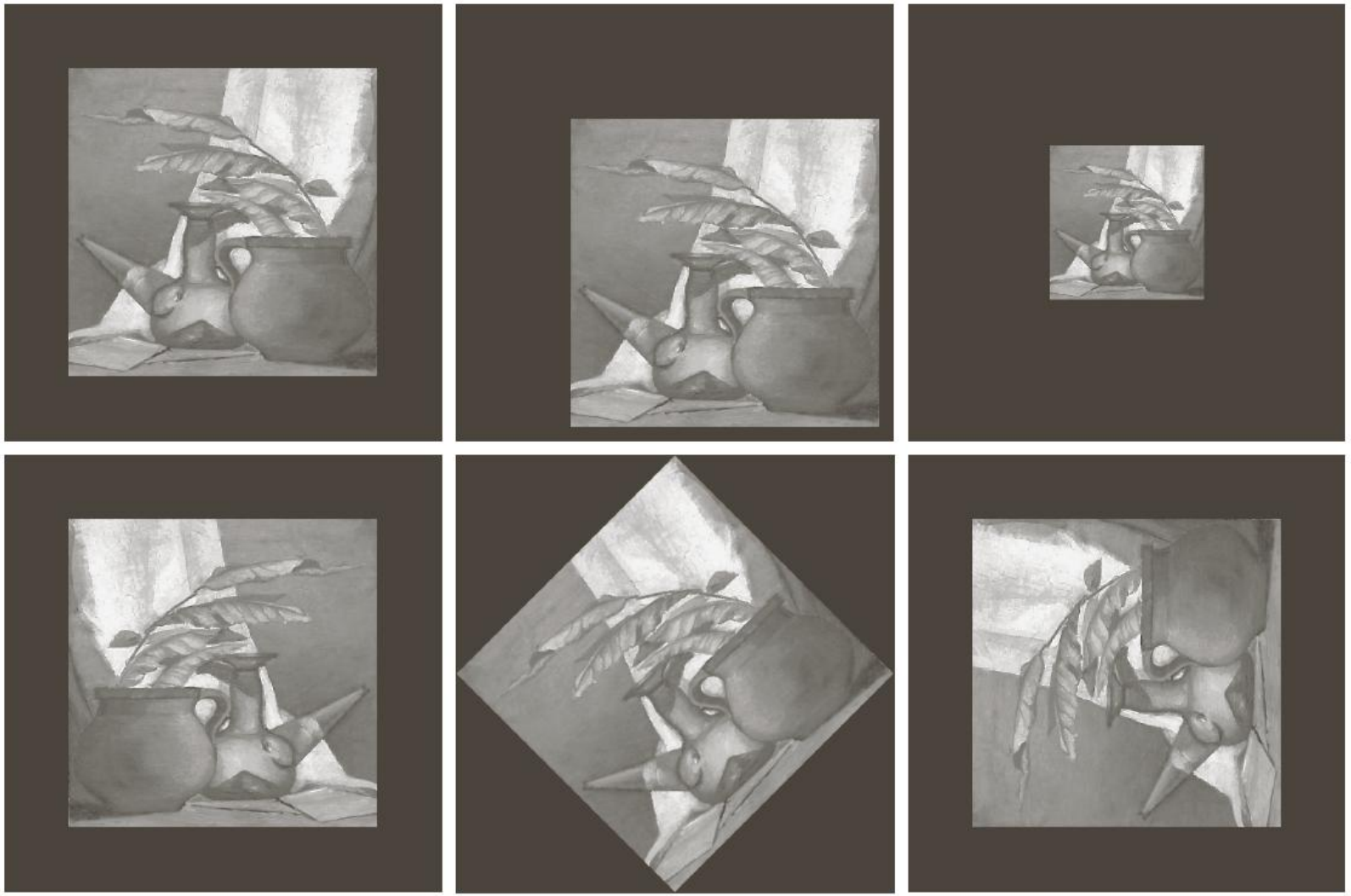
$$\varphi_4 = (\eta_{30} + \eta_{12})^2 + 3(\eta_{21} + \eta_{03})^2$$

$$\varphi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{21})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right]$$
$$+ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$

Moment invariants (cont.)

$$\varphi_6 = (\eta_{20} - 3\eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$
$$+ 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$
$$\varphi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right]$$
$$+ (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$

Moment invariants (cont.)



Moment invariants (cont.)

Moment Invariant	Original Image	Translated	Half Size	Mirrored	Rotated 45°	Rotated 90°
ϕ_1	2.8662	2.8662	2.8664	2.8662	2.8661	2.8662
ϕ_2	7.1265	7.1265	7.1257	7.1265	7.1266	7.1265
ϕ_3	10.4109	10.4109	10.4047	10.4109	10.4115	10.4109
ϕ_4	10.3742	10.3742	10.3719	10.3742	10.3742	10.3742
ϕ_5	21.3674	21.3674	21.3924	21.3674	21.3663	21.3674
ϕ_6	13.9417	13.9417	13.9383	13.9417	13.9417	13.9417
ϕ_7	−20.7809	−20.7809	−20.7724	20.7809	−20.7813	−20.7809

- $\text{Sgn}(\varphi_i)\log_{10}(|\varphi_i|)$ is shown to reduce the dynamic range.
- The values of the moments are very close.
- The sign of φ_7 is different for the mirrored image.

Principal Component Analysis (PCA)

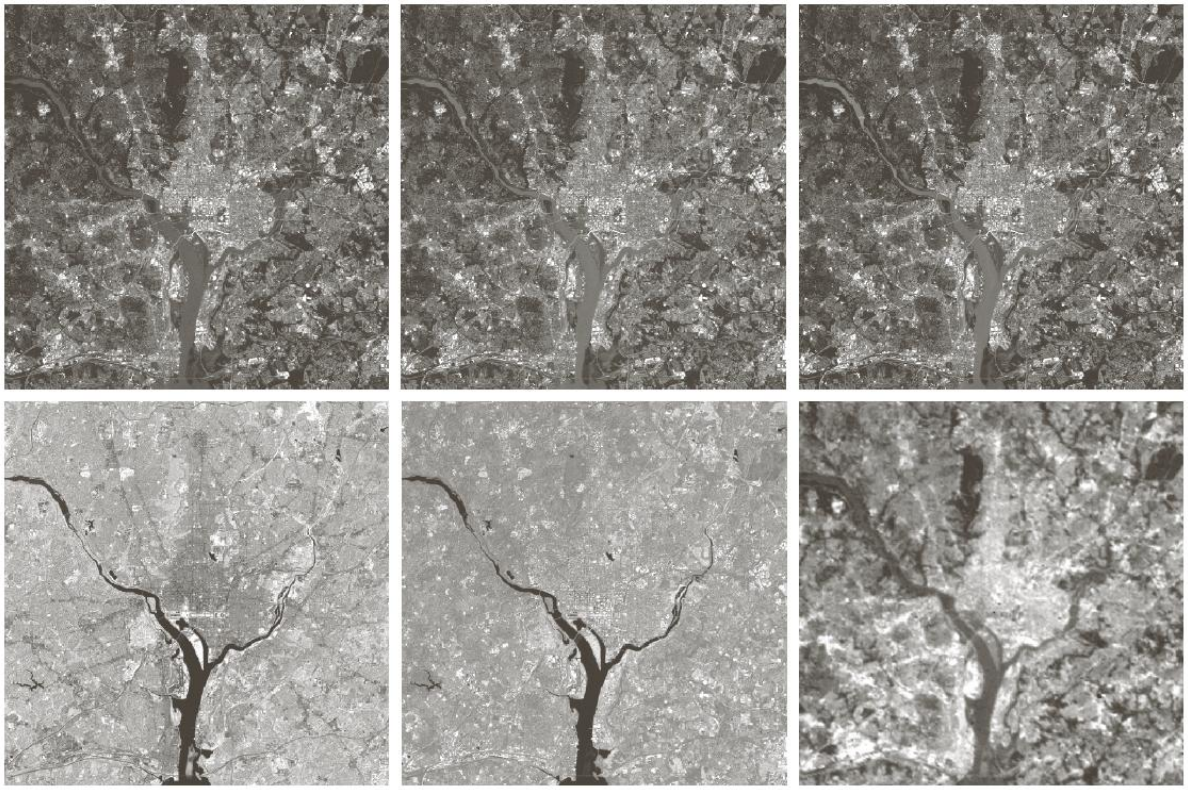
- We want to construct a low-dimensional linear subspace that best explains the variation in the components of a multidimensional image:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

- $n=3$ for RGB images or $n=6$ for LANSAT images.

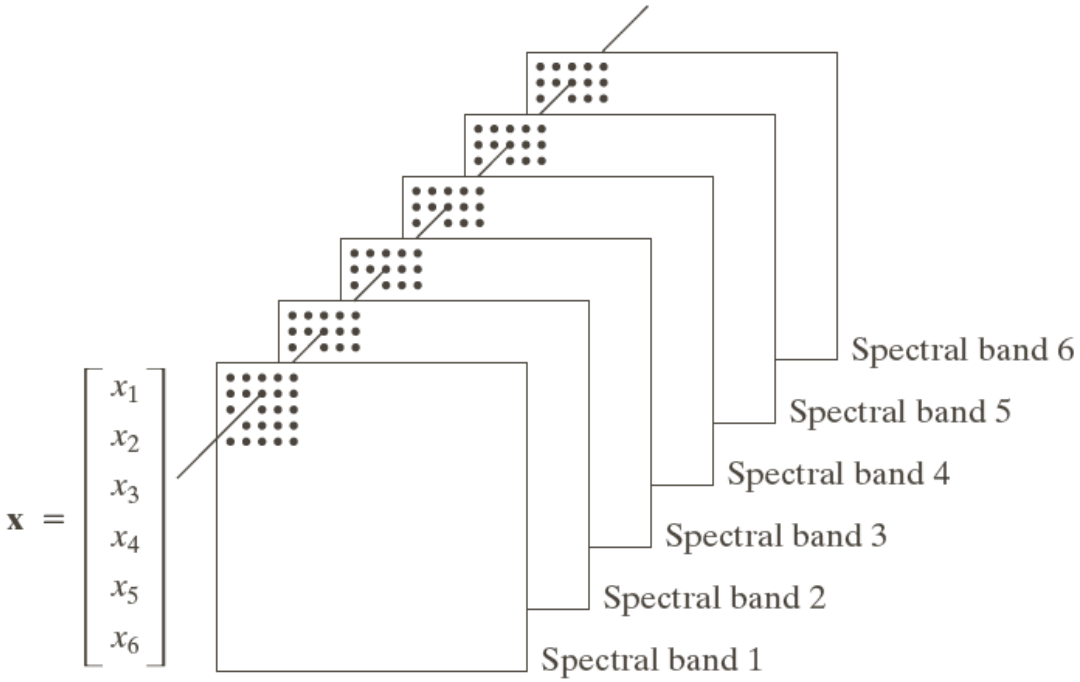
Principal Component Analysis (cont.)

- Multispectral images in the visible blue, visible green, visible red, near infrared, middle infrared and thermal infrared bands.



Principal Component Analysis (cont.)

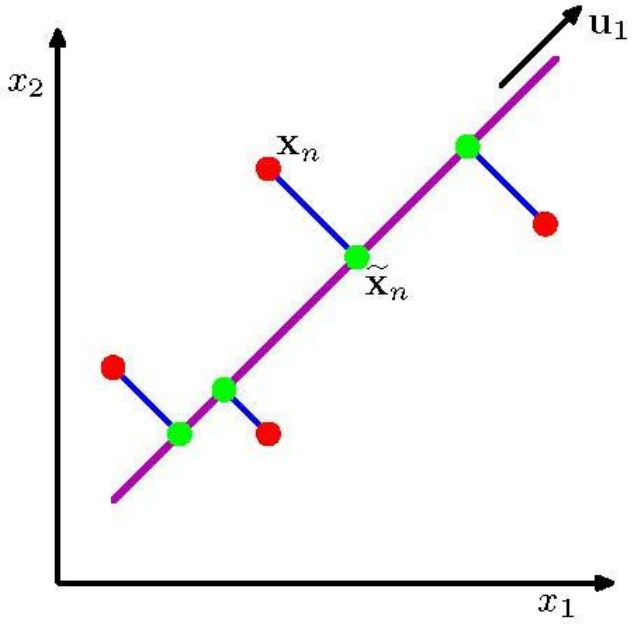
- One n-dimensional vector for each pixel is created.



Principal Component Analysis (cont.)

- Goal: find a low dimensional representation for the vectors that
  - minimizes the projection error (the distance between the initial vectors and their projections)

  or equivalently
  - maximizes the variance of the projected data.



Principal Component Analysis (cont.)

- Given: N data points $\mathbf{x}_1, \ldots, \mathbf{x}_N$ in R^d
- We want to find a new set of features that are linear combinations of the original ones:

$$w(\mathbf{x}_i) = \mathbf{u}^T(\mathbf{x}_i - \mathbf{m}_\mathbf{x}), \qquad \mathbf{m}_\mathbf{x} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$$

- What unit vector $\mathbf{u}$ in R^d captures the most variance of the data?

Principal Component Analysis (cont.)

- The variance of the projected data:

$$\mathrm{var}\left(w(\mathbf{x}_i)\right) = \frac{1}{N}\sum_{i=1}^{N} w(\mathbf{x}_i)\, w^T(\mathbf{x}_i) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{u}^T(\mathbf{x}_i - \mathbf{m}_\mathbf{x})\left(\mathbf{u}^T(\mathbf{x}_i - \mathbf{m}_\mathbf{x})\right)^T$$

Projection of data point

$$= \frac{1}{N}\sum_{i=1}^{N} \mathbf{u}^T(\mathbf{x}_i - \mathbf{m}_\mathbf{x})(\mathbf{x}_i - \mathbf{m}_\mathbf{x})^T\mathbf{u} = \mathbf{u}^T\,\frac{1}{N}\left[\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{m}_\mathbf{x})(\mathbf{x}_i - \mathbf{m}_\mathbf{x})^T\right]\mathbf{u}$$

Covariance matrix of data

$$= \mathbf{u}^T\boldsymbol{\Sigma}\mathbf{u}$$

Principal Component Analysis (cont.)

- We now estimate vector $\mathbf{u}$ maximizing the variance:

$$\mathbf{u}^T \boldsymbol{\Sigma} \mathbf{u}$$

subject to: $\mathbf{u}^T \mathbf{u} = \| \mathbf{u} \|^2 = 1$

because any multiple of $\mathbf{u}$ maximizes the objective function.

- The Lagrangian is $J(\mathbf{u}; \lambda) = \mathbf{u}^T \boldsymbol{\Sigma} \mathbf{u} + \lambda (1 - \mathbf{u}^T \mathbf{u})$

leading to the solution: $\boldsymbol{\Sigma} \mathbf{u} = \lambda \mathbf{u}$

which is an eigenvector of $\boldsymbol{\Sigma}$. The one maximizing J corresponds to the largest eigenvalue of $\boldsymbol{\Sigma}$.

Principal Component Analysis (cont.)

- The direction that captures the maximum variance of the data is the eigenvector corresponding to the largest eigenvalue of the data covariance matrix.

- The top *k* orthogonal directions that capture the most variance of the data are the *k* eigenvectors corresponding to the *k* largest eigenvalues.

Principal Component Analysis (cont.)

- Because $\boldsymbol{\Sigma}$ is real, symmetric and positive definite its eigenvalues are non negative and its eigenvectors are orthogonal.
- Let $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_d$ be the eigenvalues of $\boldsymbol{\Sigma}$ in descending order and

$$\mathbf{A} = \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_d^T \end{bmatrix}$$

be a matrix whose rows are the eigenvectors of $\boldsymbol{\Sigma}$.

Principal Component Analysis (cont.)

- The transformation that maps $\mathbf{x}$ to the new k-dimensional ($k<d$) subspace (using the first k eigenvectors) is

$$\mathbf{y} = \mathbf{A}_k(\mathbf{x} - \mathbf{m}_\mathbf{x}) = \begin{bmatrix} \mathbf{v}_1^T \\ \vdots \\ \mathbf{v}_k^T \end{bmatrix}(\mathbf{x} - \mathbf{m}_\mathbf{x})$$

- Properties of the new subspace:

$$\mathbf{m}_\mathbf{y} = \mathbf{0}$$

$$\boldsymbol{\Sigma}_\mathbf{y} = \mathbf{A}_k \boldsymbol{\Sigma} \mathbf{A}_k^T = \begin{bmatrix} \lambda_1 & & 0 & 0 \\ & \lambda_1 & & 0 \\ 0 & & \ddots & \\ 0 & 0 & & \lambda_k \end{bmatrix}$$

The components of $\mathbf{y}$ are uncorrelated

Principal Component Analysis (cont.)

- Because the rows of $\mathbf{A}$ are orthonormal $\mathbf{A}^{-1} = \mathbf{A}^{T}$

- Therefore, we can reconstruct the original data $\mathbf{x}$ from their k-dimensional projections $\mathbf{y}$:

$$\hat{\mathbf{x}} = \mathbf{A}_k^T \mathbf{y} + \mathbf{m}_{\mathbf{x}}$$

- The mean square reconstruction error is:

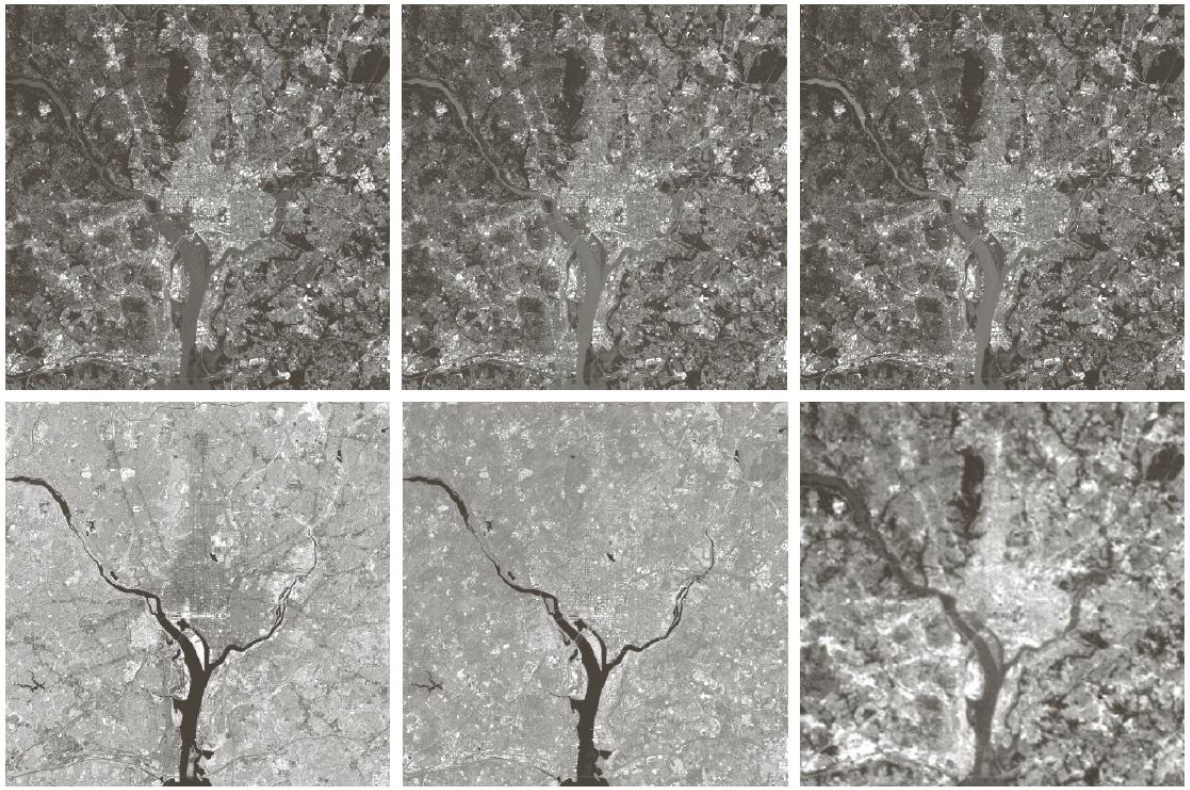
$$e = \sum_{j=1}^{d} \lambda_j - \sum_{j=1}^{k} \lambda_j = \sum_{j=k+1}^{d} \lambda_j$$

which is zero if we make use of all of the eigenvectors.

Principal Component Analysis (cont.)

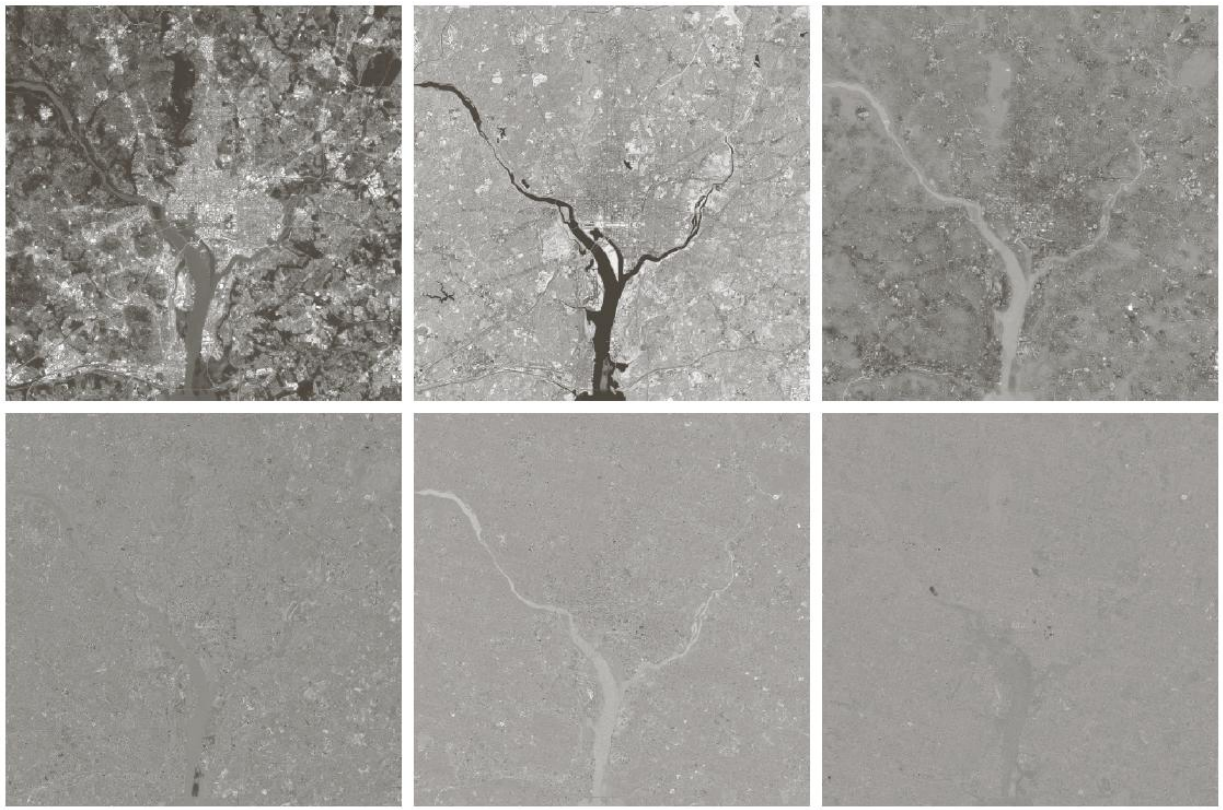
- In our example, the eigenvalues of the covariance matrix of the 6-dimensional pixels are:

λ_1 = 10344
λ_2 = 2966
λ_3 = 1401
λ_4 = 203
λ_5 = 94
λ_6 = 31



Principal Component Analysis (cont.)

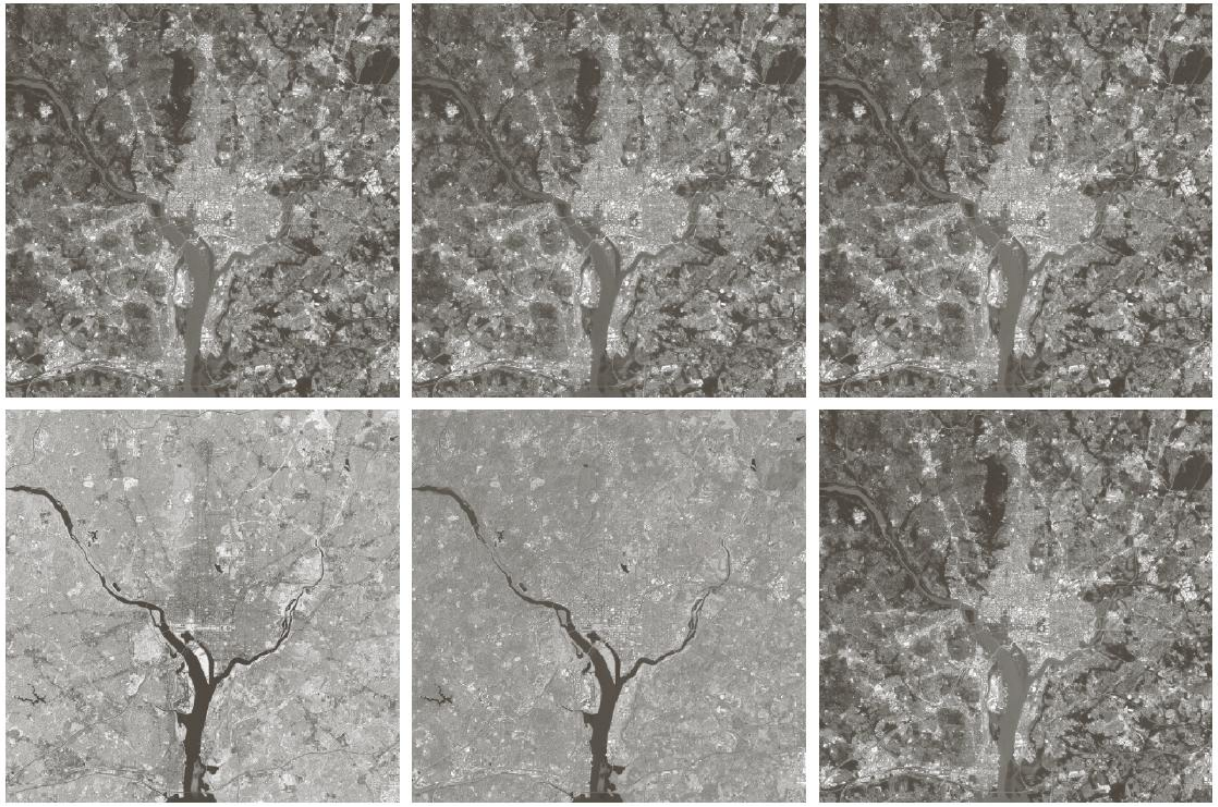
- The six principal component images $\mathbf{y} = \mathbf{A}_6(\mathbf{x} - \mathbf{m}_\mathbf{x})$
- The first two images account for 89% of the total variance.



Principal Component Analysis (cont.)

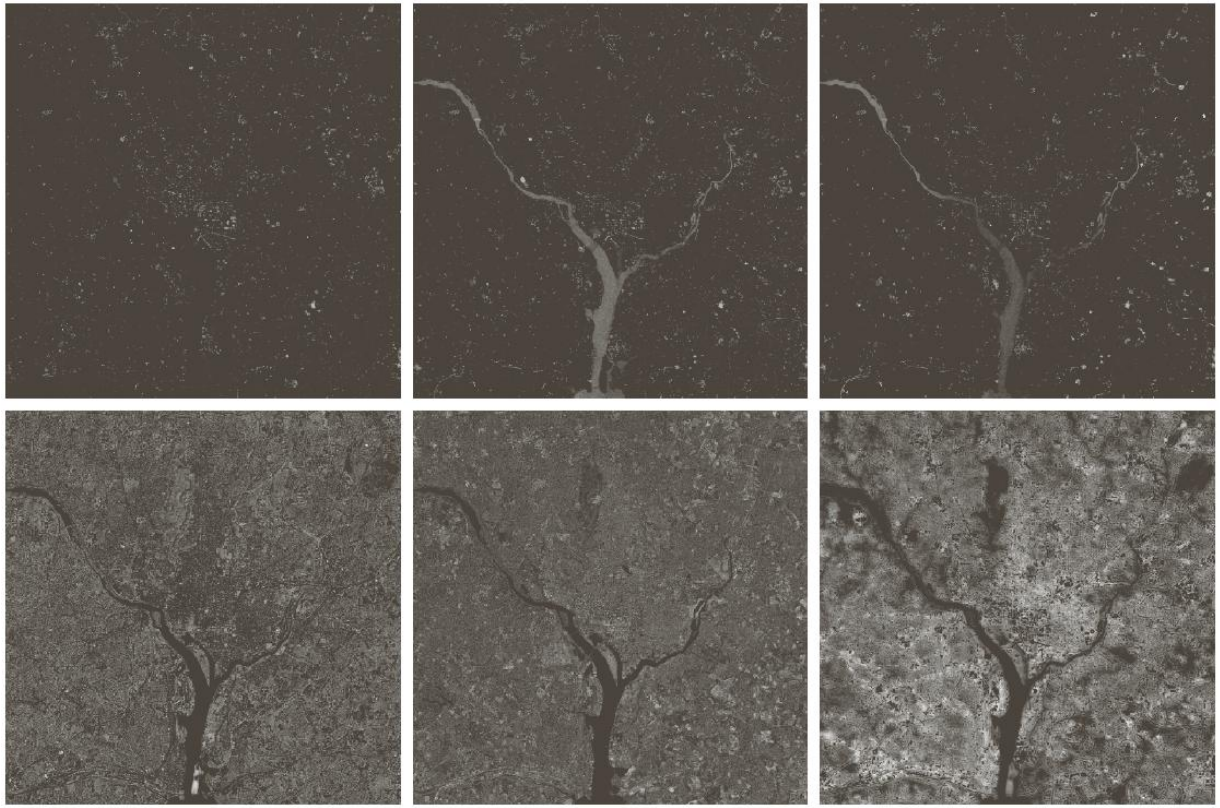
- Reconstructed images using the first two principal components

$$\hat{\mathbf{x}} = \mathbf{A}_2^T \mathbf{y} + \mathbf{m}_\mathbf{x}$$



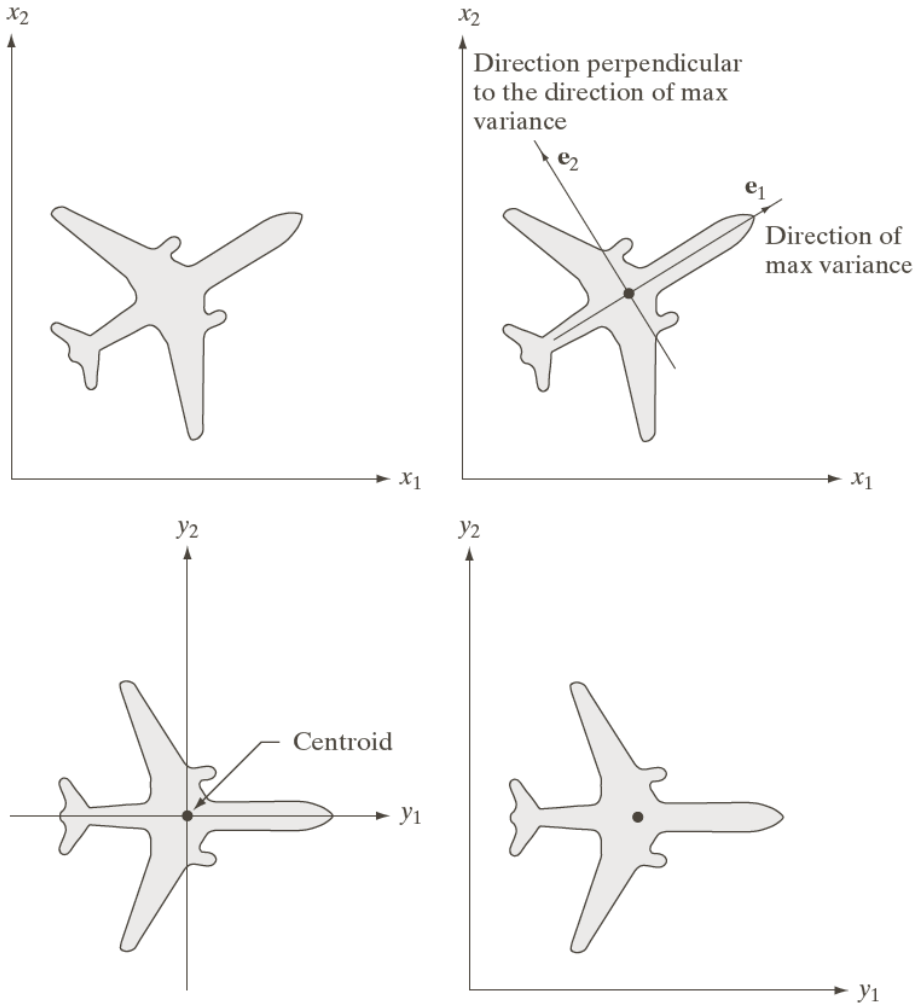
Principal Component Analysis (cont.)

- Reconstruction error (enhanced for better visualization).



Principal Component Analysis (cont.)

- PCA is a convenient way to normalize 2D boundaries with respect to rotation and translation.
- Shifting by $(y_{1\min}, y_{2\min})$ makes the coordinate non negatives.
- Further division by λ_1 and λ_2 normalizes the scale.



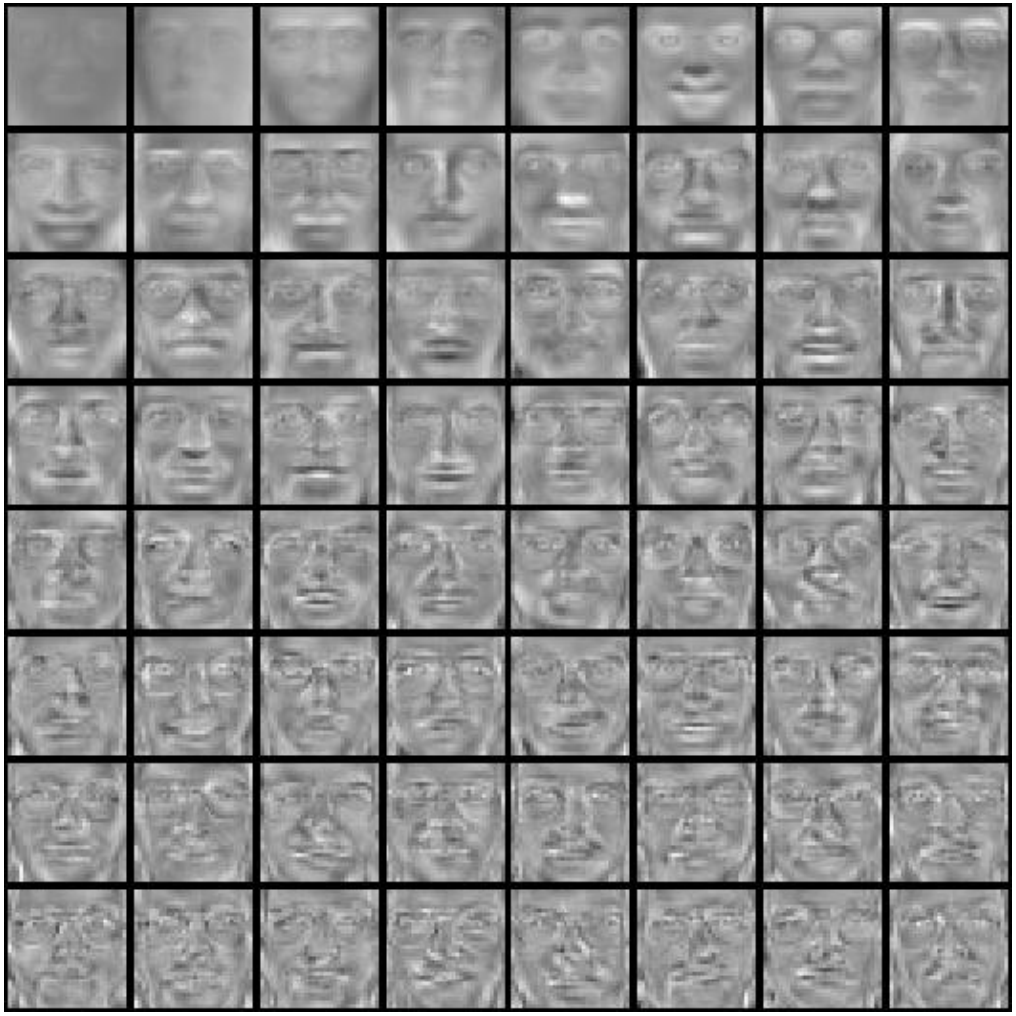
Principal Component Analysis (cont.)

- Application: eigenfaces for face recognition

Principal Component Analysis (cont.)

- Eigenfaces

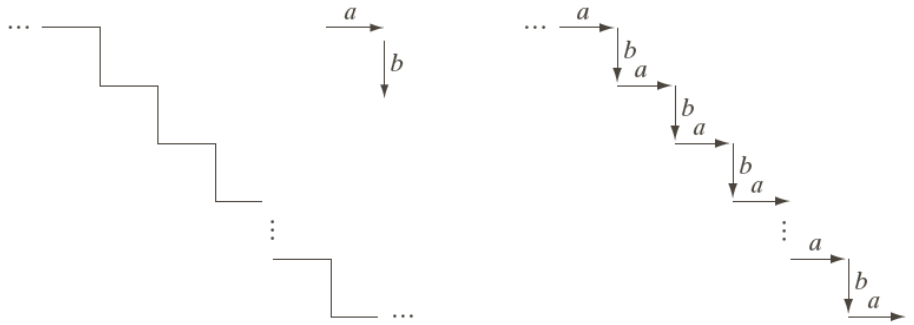
Mean image



Relational descriptors

- Rewriting rules that capture the basic repetitive pattern.
- It applies to both boundaries and regions.
- Example: the staircase structure has been extracted from an image and we want to describe it.
- We employ two primitive elements and a set of rules.

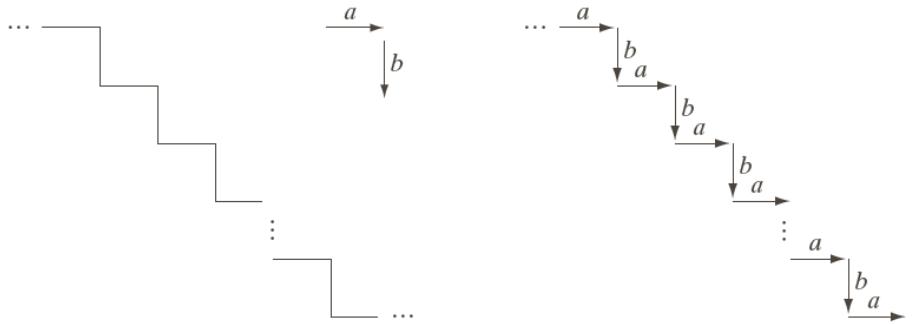
(1) $S \rightarrow aA$,
(2) $A \rightarrow bS$, and
(3) $A \rightarrow b$,



Relational descriptors (cont.)

- S and A are variables and a and b are constants.
- Rule 1 indicates that the *starting symbol* S can be replaced by a and a variable A.
- Rules 2 and 3 indicate that variable A in turn can be replaced by b and S or b alone.
  - Replacing A by bS leads to the first rule and the procedure may be repeated.
  - Replacing A by b terminates the process as there ar no more variables to be processed.

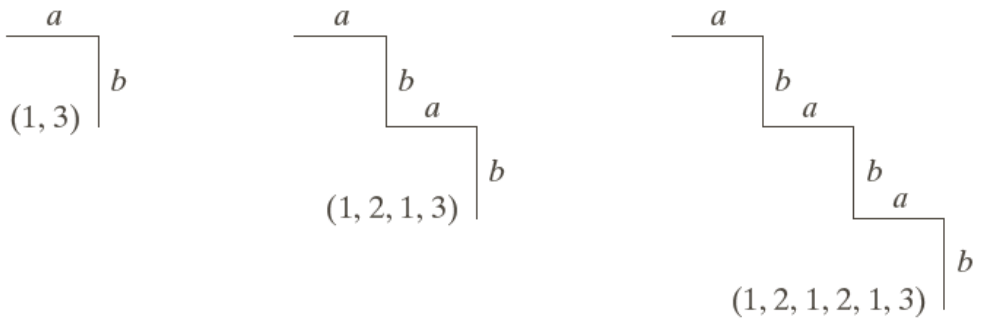
(1) $S \rightarrow aA$,
(2) $A \rightarrow bS$, and
(3) $A \rightarrow b$,



Relational descriptors (cont.)

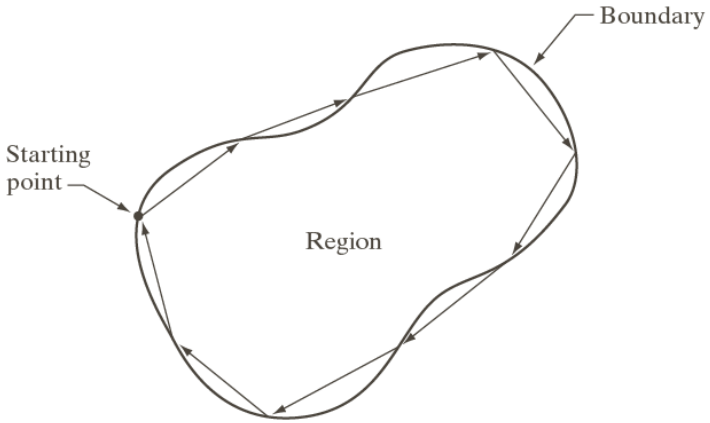
- Example derivations of the rules.
- The relationship between a and b is preserved as the rules force an a to be followed by a b.

(1) $S \rightarrow aA$,
(2) $A \rightarrow bS$, and
(3) $A \rightarrow b$,



Relational descriptors (cont.)

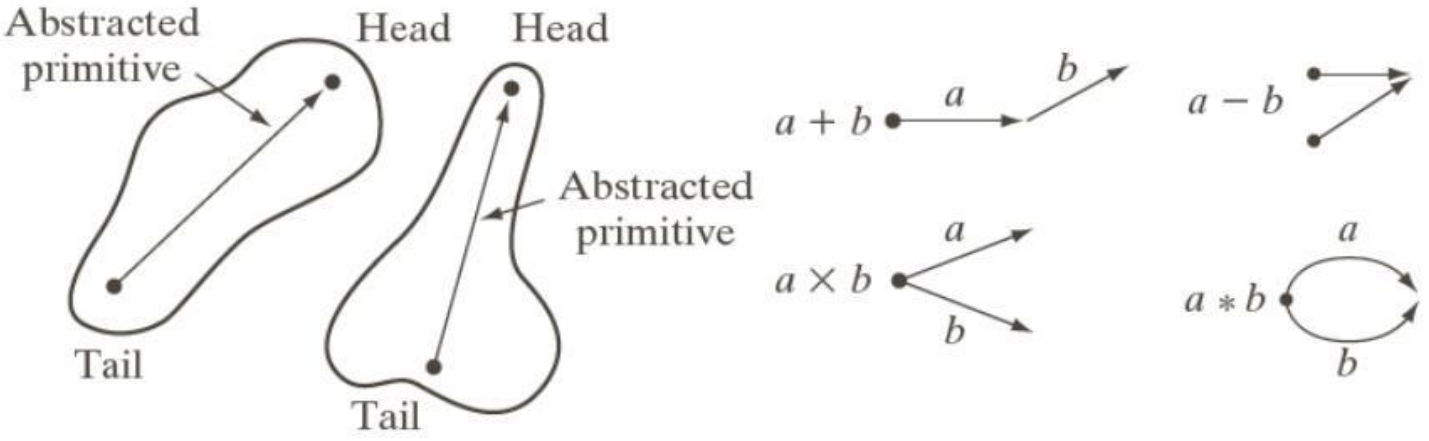
- These strings are 1D structures.
- Applications of the rules to images requires an establishment of methods for reducing 2D positional relations to 1D relations.
- An approach is to follow the contour of an object and code the result with head-to-tail segments of specified direction and length.



Relational descriptors (cont.)

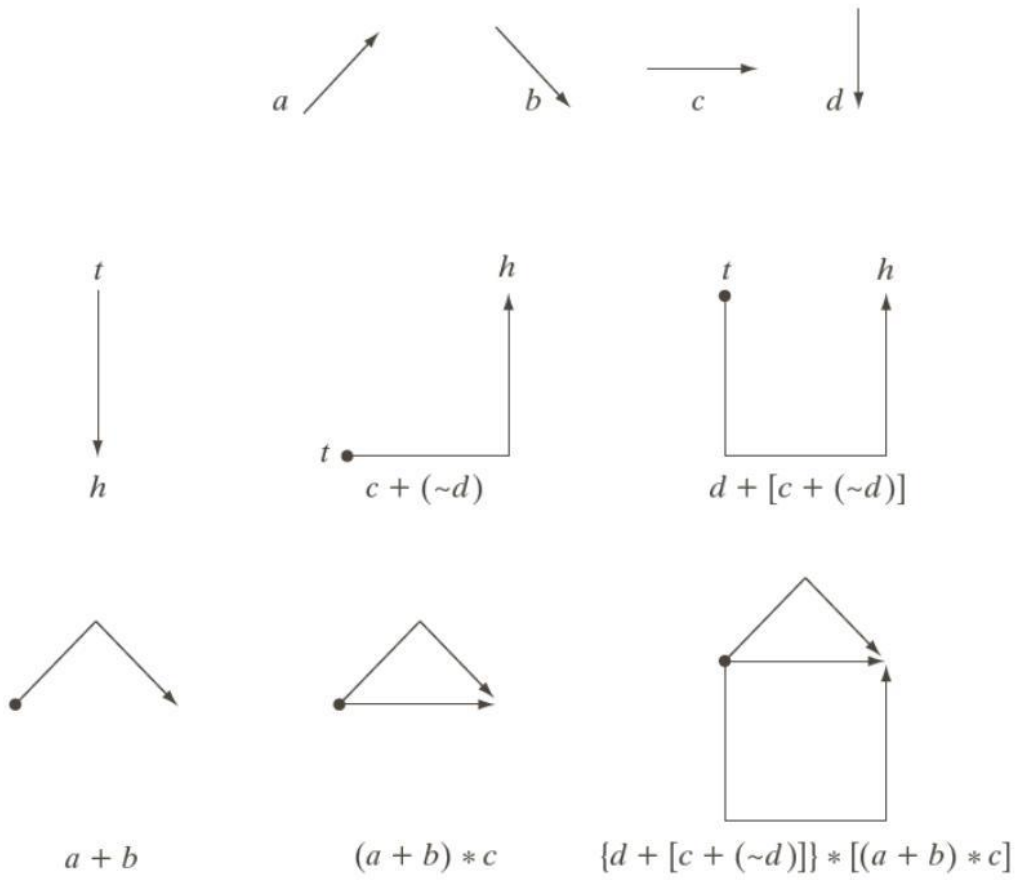
- Another approach is to employ directed line segments with other ways, besides head-to-tail connections.

Primitives and operations among them



Relational descriptors (cont.)

- Step by step generation of a specific shape.



Relational descriptors (cont.)

- Tree structures better represent non contiguous textured regions.
- Important information in a tree
  - A set of words describing the node (e.g. image region).
  - Relation between a node and its neighbors (e.g. “inside of”).