Digital Image Processing
Module 5: Part 2

Image Description and Representation

Image Representation and Description?

Objective:
To represent and describe information embedded in an image in other forms that are more suitable than the image itself.

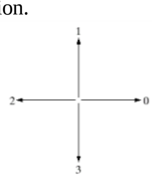
Benefits:
- Easier to understand
- Require fewer memory, faster to be processed
- More "ready to be used"

What kind of information we can use?
- Boundary, shape
- Region
- Texture
- Relation between regions

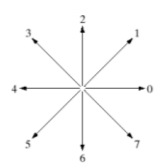
Shape Representation by Using Chain Codes

Why we focus on a boundary?
The boundary is a good representation of an object shape and also requires a few memory.

Chain codes: represent an object boundary by a connected sequence of straight line segments of specified length and direction.



4-directional chain code



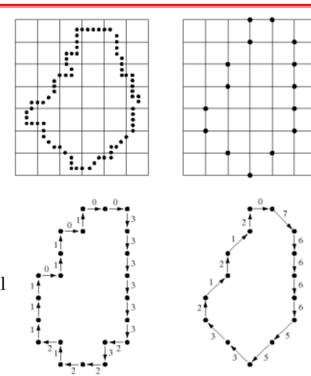
8-directional chain code

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Examples of Chain Codes

Object boundary (resampling)

Boundary vertices



4-directional chain code

8-directional chain code

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

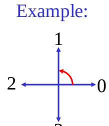
The First Difference of a Chain Codes

Problem of a chain code:
a chain code sequence depends on a starting point.

Solution: treat a chain code as a circular sequence and redefine the starting point so that the resulting sequence of numbers forms an integer of minimum magnitude.

The first difference of a chain code: counting the number of direction change (in counterclockwise) between 2 adjacent elements of the code.

Example:



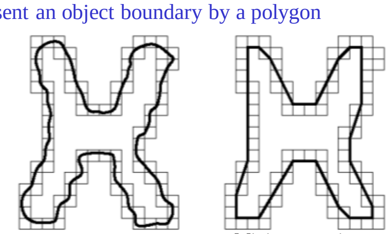
Chain code :	The first difference
0 → 1	1
0 → 2	2
0 → 3	3
2 → 3	1
2 → 0	2
2 → 1	3

Example:
- a chain code: 10103322
- The first difference = 3133030
- Treating a chain code as a circular sequence, we get the first difference = 33133030

The first difference is rotational invariant.

Polygon Approximation

Represent an object boundary by a polygon



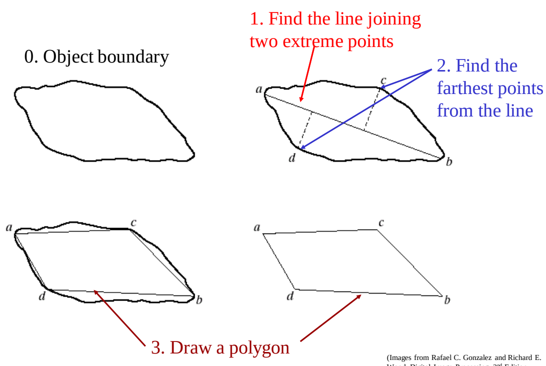
Object boundary

Minimum perimeter polygon

Minimum perimeter polygon consists of line segments that minimize distances between boundary pixels.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

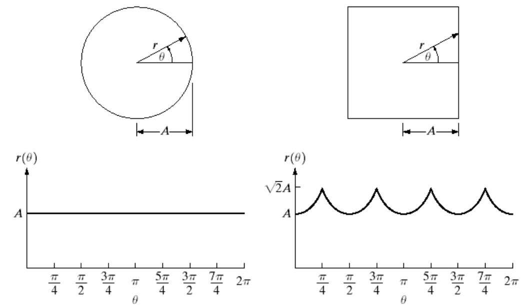
Polygon Approximation:Splitting Techniques



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.)

Distance-Versus-Angle Signatures

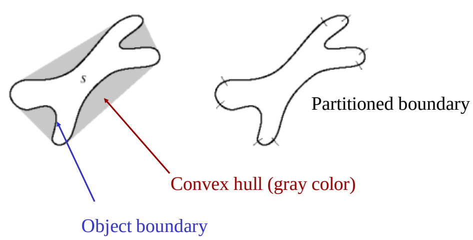
Represent an 2-D object boundary in term of a 1-D function of radial distance with respect to θ.



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.)

Boundary Segments

Concept: Partitioning an object boundary by using vertices of a convex hull.



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.)

Convex Hull Algorithm

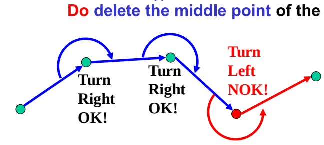
Input : A set of points on a cornea boundary

Output: A set of points on a boundary of a convex hull of a cornea

1. Sort the points by x-coordinate to get a sequence $p_1, p_2, \ldots, p_n$

For the upper side of a convex hull

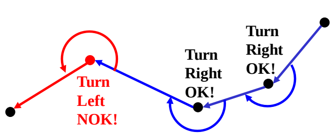
2. Put the points p_1 and p_2 in a list L_{upper} with p_1 as the first point
3. For i = 3 to n
4. Do append p_i to L_{upper}
5. While L_{upper} contains more than 2 points and the last 3 points in L_{upper} do not make a right turn
6. Do delete the middle point of the last 3 points from L_{upper}



Convex Hull Algorithm (cont.)

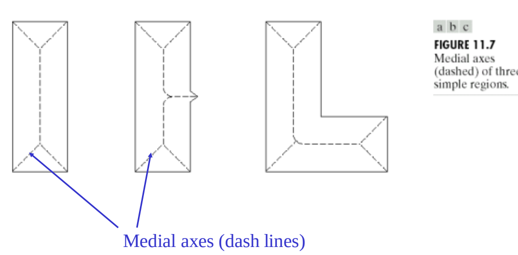
For the lower side of a convex hull

7. Put the points p_n and p_{n-1} in a list L_{lower} with p_n as the first point
8. For i = n-2 down to 1
9. Do append p_i to L_{lower}
10. While L_{lower} contains more than 2 points and the last 3 points in L_{lower} do not make a right turn
11. Do delete the middle point of the last 3 points from L_{lower}
12. Remove the first and the last points from L_{lower}
13. Append L_{lower} to L_{upper} resulting in the list L
14. Return L



Skeletons

Obtained from thinning or skeletonizing processes



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.)

Thinning Algorithm

Concept:
1. Do not remove end points
2. Do not break connectivity
3. Do not cause excessive erosion

Apply only to contour pixels: pixels "1" having at least one of its 8 neighbor pixels valued "0"

Notation:

Let

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

= Neighborhood arrangement for the thinning algorithm

Let $N(p_1) = p_2 + p_3 + \ldots + p_8 + p_9$

$T(p_1)$ = the number of transition 0-1 in the ordered sequence $p_2, p_3, \ldots, p_8, p_9, p_2$.

Example

0	0	1
1	p_1	0
1	0	1

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

Thinning Algorithm (cont.)

Step 1. Mark pixels for deletion if the following conditions are true.
a) $2 \le N(p_1) \le 6$
b) $T(p_1) = 1$
c) $p_2 \cdot p_4 \cdot p_6 = 0$
d) $p_4 \cdot p_6 \cdot p_8 = 0$

(Apply to all border pixels)

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Step 2. Delete marked pixels and go to Step 3.

Step 3. Mark pixels for deletion if the following conditions are true.
a) $2 \le N(p_1) \le 6$
b) $T(p_1) = 1$
c) $p_2 \cdot p_4 \cdot p_8 = 0$
d) $p_2 \cdot p_6 \cdot p_8 = 0$

(Apply to all border pixels)

Step 4. Delete marked pixels and repeat Step 1 until no change occurs.

Example: Skeletons Obtained from the Thinning Alg.

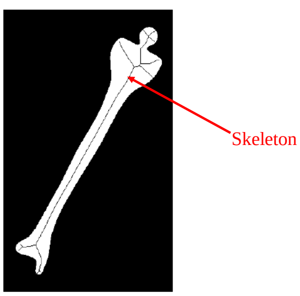


FIGURE 11.10 Human leg bone and skeleton of the region shown superimposed.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Boundary Descriptors

1. Simple boundary descriptors:
we can use
- Length of the boundary
- The size of smallest circle or box that can totally enclosing the object

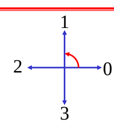
2. Shape number

3. Fourier descriptor

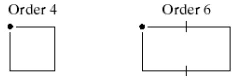
4. Statistical moments

Shape Number

Shape number of the boundary definition:
the first difference of smallest magnitude
The order n of the shape number:
the number of digits in the sequence



Order 4 Order 6

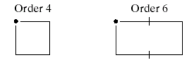


	Order 4	Order 6
Chain code:	0 3 2 1	0 0 3 2 2 1
Difference:	3 3 3 3	3 0 3 3 0 3
Shape no.:	3 3 3 3	0 3 3 0 3 3

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Shape Number (cont.)

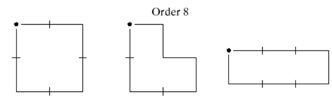
Order 4 Order 6



	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

Shape numbers of order 4, 6 and 8

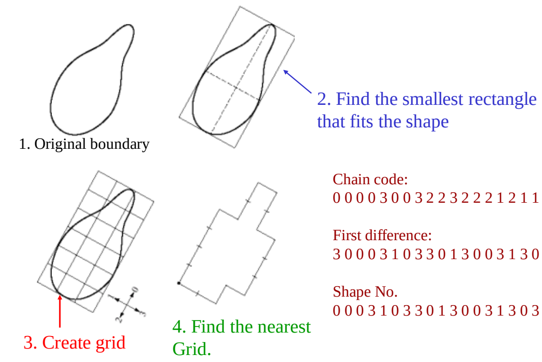
Order 8



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

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Example: Shape Number



Chain code:
0 0 0 0 3 0 0 3 2 2 3 2 2 2 1 2 1 1

First 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

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Fourier Descriptor

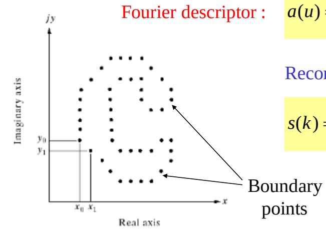
Fourier descriptor: view a coordinate (x,y) as a complex number (x = real part and y = imaginary part) then apply the Fourier transform to a sequence of boundary points.

Let s(k) be a coordinate of a boundary point k : $s(k) = x(k) + jy(k)$

Fourier descriptor : $a(u) = \frac{1}{K}\sum_{k=0}^{K-1} s(k)e^{-2\pi uk/K}$

Reconstruction formula

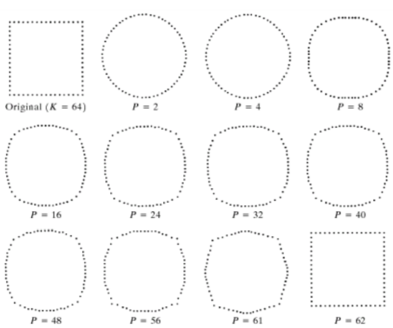
$$s(k) = \frac{1}{K}\sum_{k=0}^{K-1} a(u)e^{2\pi uk/K}$$



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Example: Fourier Descriptor

Examples of reconstruction from Fourier descriptors



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

P is the number of Fourier coefficients used to reconstruct the boundary

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Fourier Descriptor Properties

Some properties of Fourier descriptors

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	$s_p(k) = s(k - k_0)$	$a_p(u) = a(u)e^{-j2\pi k_0 u/K}$

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Statistical Moments

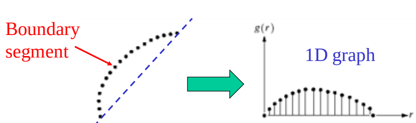
Definition: the nth moment

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

where

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

Example of moment:
The first moment = mean
The second moment = variance



1. Convert a boundary segment into 1D graph
2. View a 1D graph as a PDF function
3. Compute the nth order moment of the graph

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.