Morphological operations

Mathematical morphology and binary operations

The basic operations are:
•Erode/Dilate
•Open/Close
•Hit-Miss

A variety of algorithms can be based on the above basic operations. Here, we give a few examples:

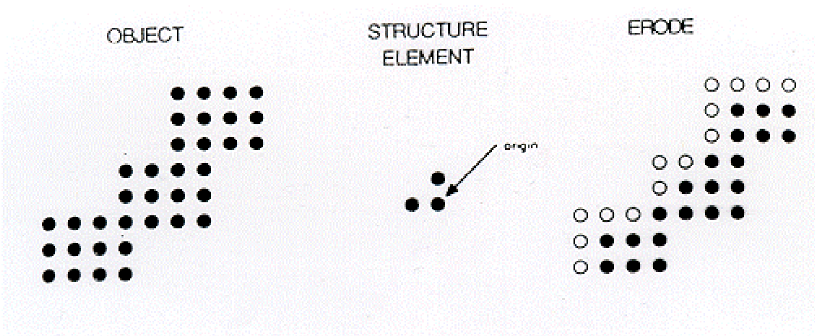
•Skeleton
•Distance transforms
•Shape features

Mathematical morphology is mainly based on set theory. Intuitively, the whole area is about adding or removing pixels from a binary image according to certain rules depending on neighborhood patterns

Erosion

Let B_X denote the translation of B so that its origin is located at x. Then the erosion of X by B is defined as the set of all points x such that B_X is included in X, that is:

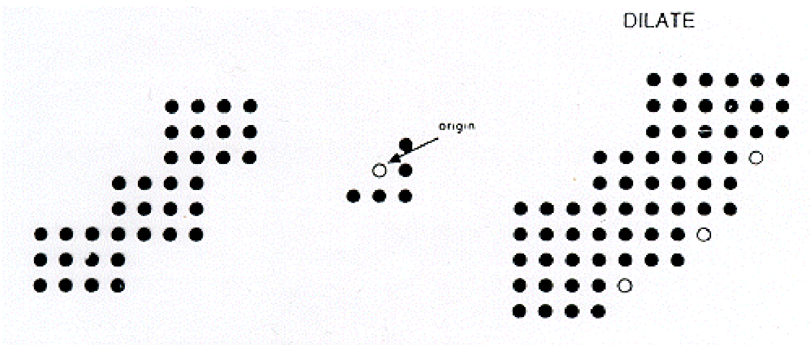
$$X \ominus B = \{x | B_x \subset X\}$$



Dilation

Similarly, the dilation of X by B is defined as the set of all points x such that B_X hits X, i.e. they have a non-empty intersection:

$$X \oplus B = \{x \mid B_x \cap X \neq \emptyset\}$$

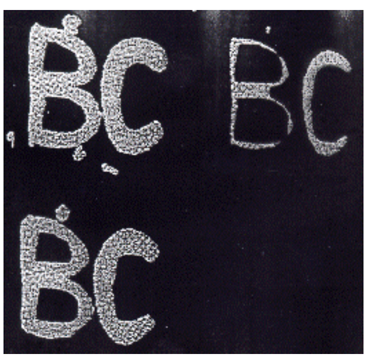


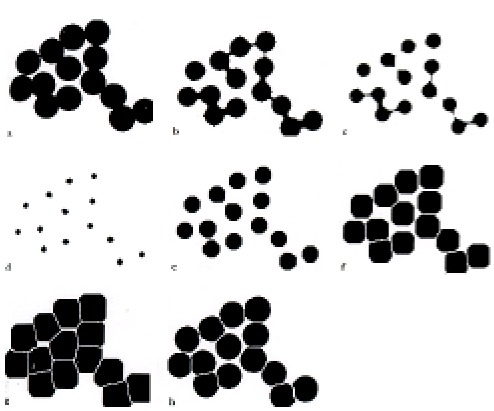
Open

Open is defined as an erosion followed by a dilation:

$$X_B = (X \ominus B) \oplus B$$

Examples of the open operation



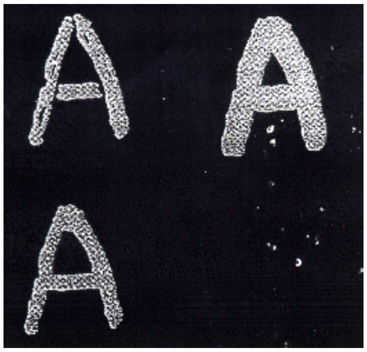


Close

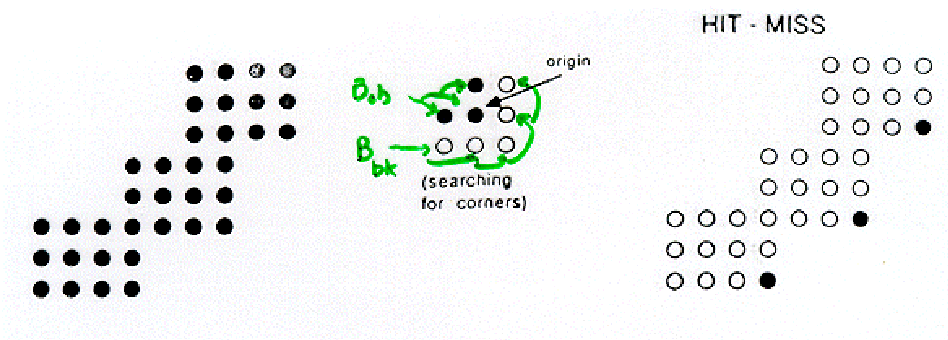
Close is defined by a dilation followed by an erosion :

$$X^B = (X \oplus B) \ominus B$$

Example of a close operation



Hit-Miss



The hit-miss operation means searching for a match or a specific configuration:

B_{ob} and B_{bk} are the sets formed from pixels in B that should belong to the object and the background, respectively:

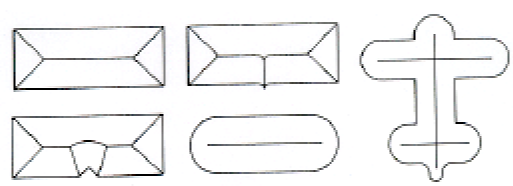
$$X \otimes B = (X \ominus B_{ob}) / (X \oplus B_{bk})$$

where $/$ means set difference

Skeletonization

Definition:	The set of points whose distance from the nearest boundary is locally maximum
Shape recovery:	Take the union of circular discs centered on the skeleton and having radii equal to the associated contour distance
Application:	Compact shape representation. Works best for long and thin features
Alternative:	Medial axis transform. Those points being equidistant from at least two boundary points (Prairie fire)

Skeleton examples



Skeletonization algorithm

		0			1	1			1		0	
Phase1:West	0	1	1	0	1	1	0	1	1	0	1	1
		1	1		0				1		0	
		0			0			0			0	
Phase2:North	1	1	0	0	1	1		1		0	1	0
	1	1			1	1	1	1	1		1	
	1	1			0		1				0	
Phase3:East	1	1	0	1	1	0	1	1	0	1	1	0
		0		1	1		1				0	
		1	1	1	1		1	1	1		1	
Phase4:South	0	1	1	1	1	0		1		0	1	0
		0			0			0			0	

The algorithm runs in four phases: west-north-east-south. Each of the four phases is associated with a set of hit-miss operators.
If the number of operators in the set is 3 (the three leftmost) the object will shrink to a skeleton. If the number is 4, it will shrink to a point.

The algorithm is as follows:
1. Run phase 1 (west). For each pixel position in the input image, test the west hit-miss operators for a hit.
2. If any of the operators results in a hit, set the corresponding position in the output image to zero. Otherwise, set to one.
3. Proceed in the same manner with phase 2, 3 and 4. If there still remain pixel positions that can be set to zero by a hit, go to 1.
4. The skeleton (or point) is extracted

Distance metrics

a) Euclidian $d_e(x,y)=\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$
b) City block $d_{cb}(x,y)=|x_1-x_2|+|y_1-y_2|$
c) Chessboard $d_{ch}(x,y)=\max\{|x_1-x_2|,|y_1-y_2|\}$

Below you will see the distances from a point using the above metrics.

		3	3	3		
	3	2	2	2	3	
3	2	1	1	1	2	3
3	2	1	$\boxed{0}$	1	2	3
3	2	1	1	1	2	3
	3	2	2	2	3	
		3	3	3		

a)

			3			
		3	2	3		
	3	2	1	2	3	
3	2	1	$\boxed{0}$	1	2	3
	3	2	1	2	3	
		3	2	3		
			3			

b)

3	3	3	3	3	3	3
3	2	2	2	2	2	3
3	2	1	1	1	2	3
3	2	1	$\boxed{0}$	1	2	3
3	2	1	1	1	2	3
3	2	2	2	2	2	3
3	3	3	3	3	3	3

c)

Distance transform

The distance transform uses a binary image as input and produces for each pixel position a value that is the distance from that pixel to the nearest background pixel. Examples of application areas are:
– Fast morphological operations
– Skeletonization
– Computation of shape features (e.g. form factor)

Distance transform algorithm using chessboard metrics algorithm

Phase 1: apply this filter from the upper left corner of the image:

If A>0: A=min(B+1, C+1)

	C
B	A

A	C
B	

Phase 2: apply this filter from the lower right corner of the image:

If A>0: A=min(A,B+1, C+1)

Distance map result

	1	**1**	**1**		**1**	**1**	**1**		
	1	**2**	**2**		**1**	**2**	**2**		
	1	**2**	**3**						
	1	**2**	**3**		**1**	**1**	**1**	**1**	**1**
	1	**2**	**3**		**1**	**2**	**2**	**2**	**2**
					1	**2**	**3**	**3**	**3**
		1	**1**		**1**	**2**	**3**	**4**	**4**
	1	**2**			**1**	**2**	**3**	**4**	**5**

Phase 1

	1	**1**	**1**		**1**	**1**	**1**		
	1	**2**	**1**		**1**	**1**	**1**		
	1	**2**	**1**						
	1	**2**	**1**		**1**	**1**	**1**	**1**	**1**
	1	**1**	**1**		**1**	**2**	**2**	**2**	**1**
					1	**2**	**3**	**2**	**1**
		1	**1**		**1**	**2**	**2**	**2**	**1**
	1	**1**			**1**	**1**	**1**	**1**	**1**

Phase 2

Shape features

Shape factor: $S = \frac{p^2}{4p A}$

$A = Area$

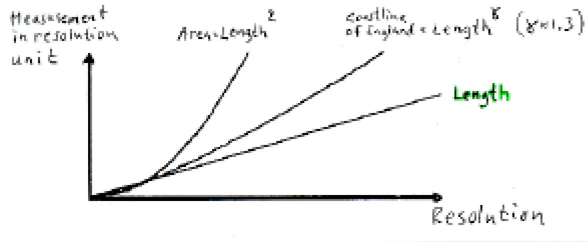
$p = perimeter$

$S = 1$ for a circle

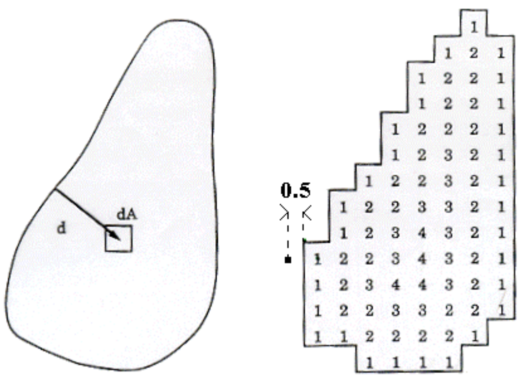
Great Britain

Area can be determined, what about perimeter?

Higher resolution: Perimeter $\rightarrow \infty$



Shape features



$$\bar{d} = \frac{1}{A}\iint_A d(x,y)\,dA \text{ (for a circle, } \bar{d} = r/3\text{)}$$

$$FORM = \frac{1}{9p}\cdot\frac{A}{\bar{d}^2} \text{ (for a circle, } FORM = 1\text{)}$$

$$\bar{d} = \frac{\sum_A \left[d(x,y) - \frac{1}{2} \right]}{A} = \frac{\sum_A d(x,y)}{A} - \frac{1}{2}$$

where A is the number of pixels in the object

$$\bar{d} = \frac{30\cdot 1 + 31\cdot 2 + 12\cdot 3 + 4\cdot 4}{77} - 0.5 \approx 1.37$$

$$FORM \approx \frac{1}{9p}\cdot\frac{77}{(1.37)^2} \approx 1.45$$