1.1 Opening and Closing Filters and Noise

Pepper noise consists of collections of small dark objects. Pepper noise is removed by the open filter. The open filter is $A \circ B = \cup\{B + p \mid B + p \subseteq A\}$ where B is a shape primitive. Only the parts of the image are passed that are part of some translate of the shape primitive that fits within image A [Dougherty, 1992, pp. 21,22]. One often uses a symmetric structuring element so that results do not depend on the orientation of the picture. If an image is made up entirely of translates of the shape primitive (the image is a union of these translates), then it is fully passed by the opening operator. The trick is to find a structuring element which passes the image but eliminates the pepper noise. If N is a noise image, pepper noise, and S is the image then $S \circ B \subseteq (S \cup N) \circ B \subseteq S \cup N$. The fitted image lies between the opened image S (signal) and the noisy image.

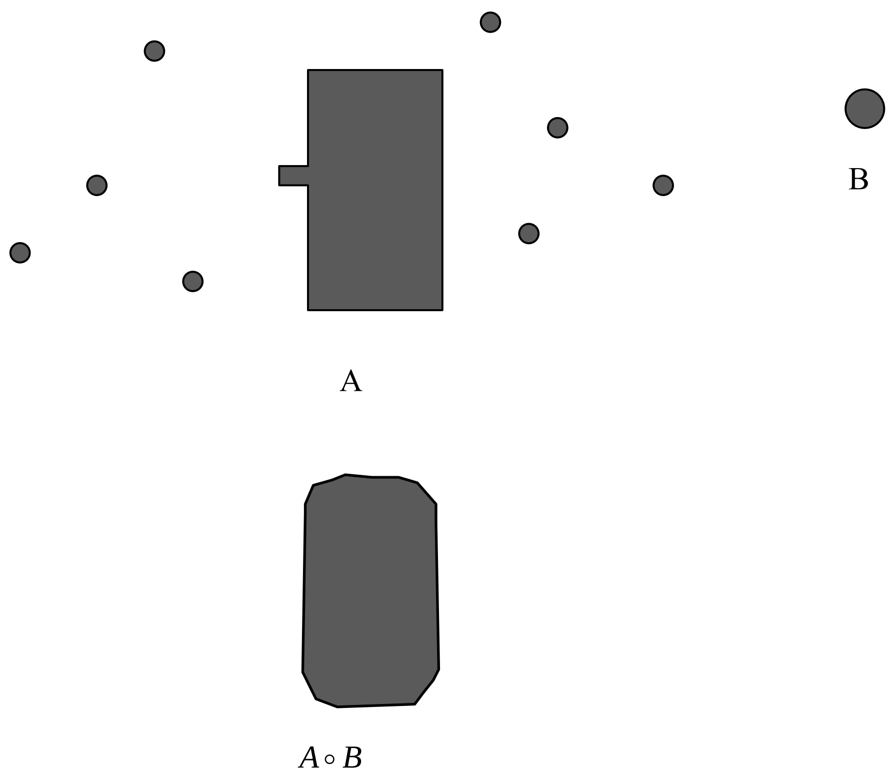


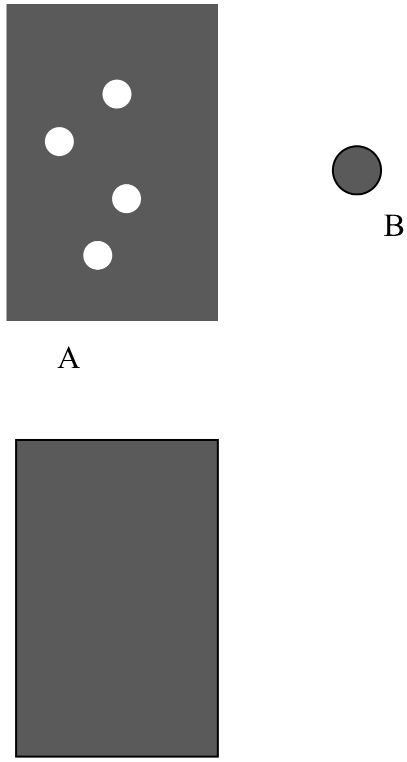
Figure 1. Filtering Pepper Noise

One often starts with small structuring elements B and increases the size systematically. For example

$$S \circ nB = (S \Theta nB) \oplus nB$$

$$nB = \underbrace{B \oplus B \oplus \quad \oplus B}_{n \text{ times}}$$

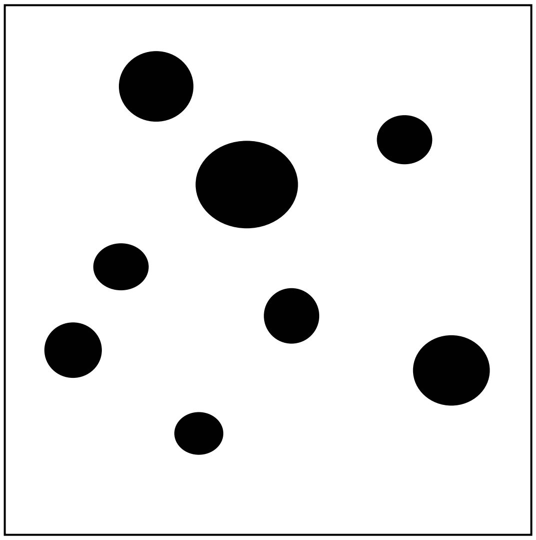
Salt noise consists of small white objects. The close operator will remove salt noise in a manner similar to the way the open operator removes pepper noise.



$$A \bullet B = (A \oplus B) \Theta B$$

Figure 2 . Filtering Salt Noise

The following example shows an operation of removing pepper noise from an image with the open operator [Dougherty, 92, pp. 23].



Image

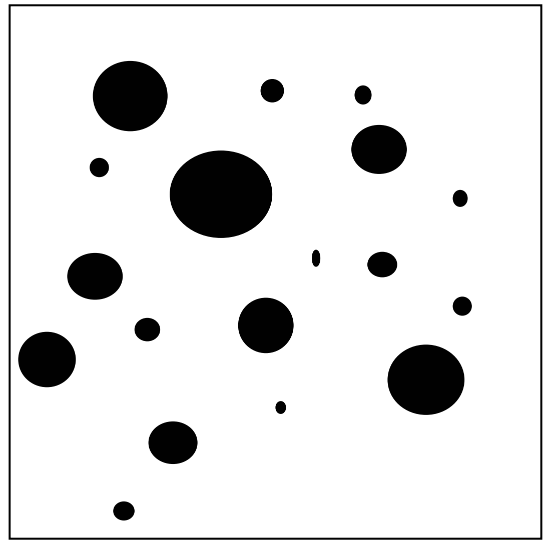
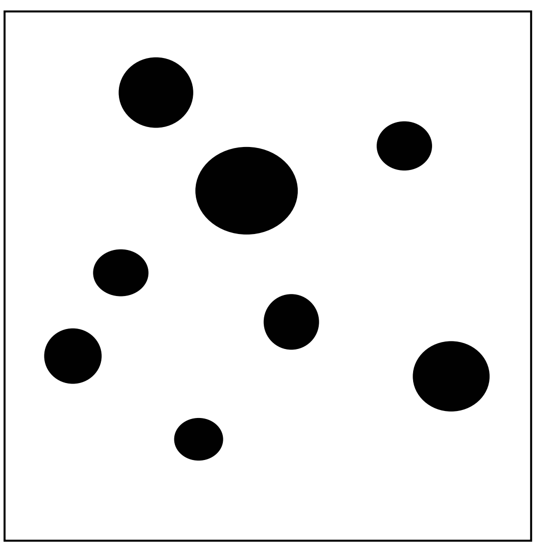
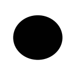


Image with Noise



Open Filtered Image



Structuring Element

Figure 3. Example Filtered Image

1.2 Filtering with a Set of Structuring Elements

An opening passes only the parts of an image that conform to the shape of the structuring element. Suppose one desires to pass parts of an image that conforms to one of a number of structuring elements. Use a filter which uses a number of openings, one for each shape primitive [Dougherty, 92, pp. 26,28]. The filter output is the union of the individual openings. Let β be a base class of structuring elements and $T(A) = \cup\{A \circ B \mid B \in \beta\}$. This opening filter will remove pepper noise. It is desired to remove noise from corrupted images. The invariant class where T(A)=A does not change the image is the set of images formed as the unions of translations of the base class β. Consider the following example with structuring elements that pass line segments and corners.

0	0	0	0	1	0	1	0	0	0	0	1
1	1	1	0	1	0	0	1	0	0	1	0
0	0	0	0	1	0	0	0	1	1	0	0
	hor line			ver line			diag a line			diag b line	

0	1	0	0	1	0	0	0	0	0	0	0
0	1	1	1	1	0	1	1	0	0	1	1
0	0	0	0	0	0	0	1	0	0	1	0
	corner a			corner b			corner c			corner d	

Figure 4. Example Set of Structuring Elements

The set β consists of this set of structuring elements.
The following example shows the application of these structuring elements.

0	0	0	1	0	0	0	0
0	0	0	1	1	0	0	0
0	0	1	1	1	0	0	0
0	0	0	1	0	1	1	1
0	1	1	0	0	0	1	0
0	0	0	0	0	0	0	1
0	0	1	0	0	1	0	1
0	0	0	0	0	0	0	0

img

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0
0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

open with

0	0	0
1	①	1
0	0	0

0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

open with

1	0	0
0	①	0
0	0	1

0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0
0	0	1	1	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

open with

0	1	0
1	1	0
0	0	0

0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	1	1	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

open with

0	1	0
1	1	1
0	1	0

Figure 5. Example Filtered Image

Closing using a set of structuring elements operates in a similar fashion and are good for filtering subtractive (salt) noise corrupting the foreground.

$$T(A) = \cap\{A \bullet B | B \in B\}$$

1.3 Boundary detection

For a set A the quantity $(A \oplus B) - A$ is the external boundary while $A - (A \Theta B)$ is the internal boundary [Dougherty, 92, pp. 32]. The morphological gradient is $(A \oplus B) - (A \Theta B)$ and straddles the boundary. The quantity A-B is set difference.

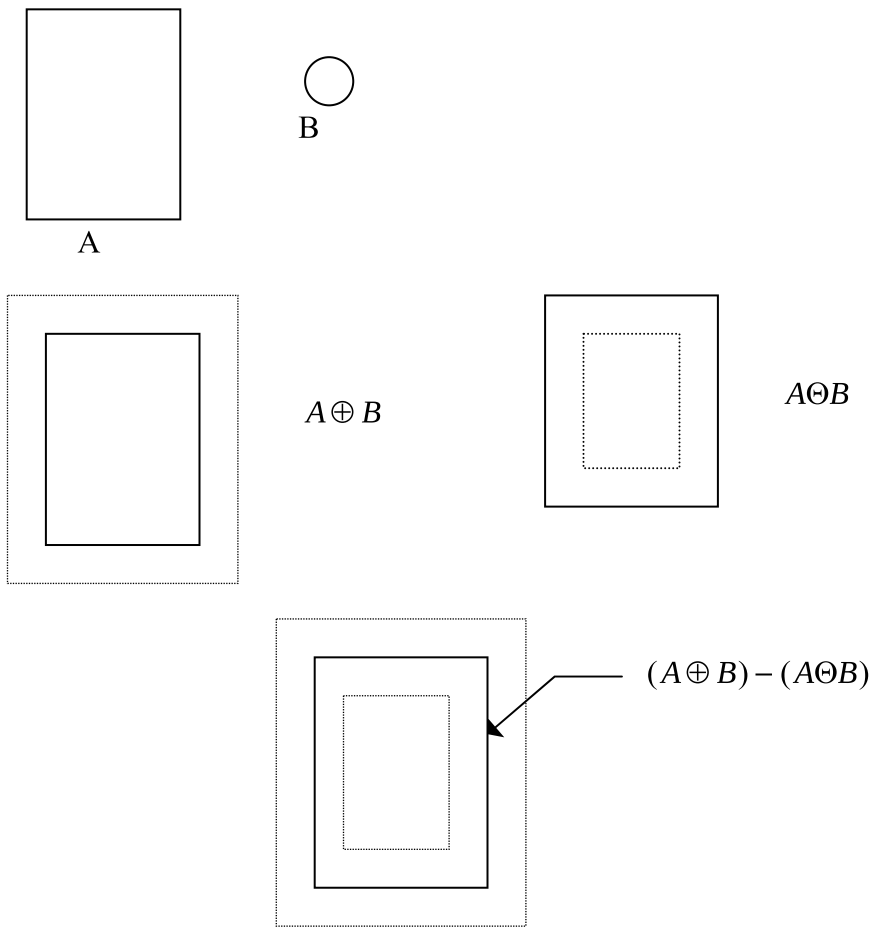
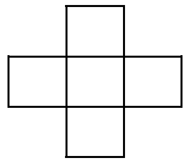
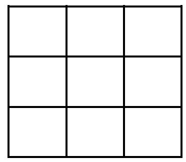


Figure 6. Morphological Gradient

Thicker boundaries are obtained from larger structuring elements. The type of disc *B* that one considers will affect the result. A 4-connected disc will give 4-connected boundaries while an 8-connected disc will give 8-connected boundaries.



4 neighbor disc



8 neighbor disc

Figure 7. Four and Eight Neighborhoods

1.4 Conditional Dilation and Region Filling

An object dilated by a structuring element containing the origin is expanded. If this is done repeatedly the object grows without limit. One can restrict the growth by intersecting the dilation with another set. This is called conditional dilation [Dougherty, 92, pp. 33, 39; Gonzalez and Woods, 1992, pp. 532]. Let $A \subseteq C$ and B be a structuring element. The conditional dilation of A by B relative to C is $A(+)B = \cup\{(B + a) \cap C \mid a \in A\}$ or $A(+)B = (A \oplus B) \cap C$. An application is to find a component of a binary image. One may write conditional dilation with respect to C as $(A(+)B)_C = (A \oplus B) \cap C$ if C is not clearly understood or stated.

Consider filling a region surrounded by a curve. Let *C* be the curve and *p* a point inside the curve. One could also consider this as C being the boundary of a region and p a point in the interior of the region. Then

$S_1 = \{p\}$ and

$S_k = (S_{k-1} \oplus B) \cap A = S_{k-1}(+)B$ with respect to A where A=C^c the complement of C. One terminates the process when $S_k = S_{k+1}$.

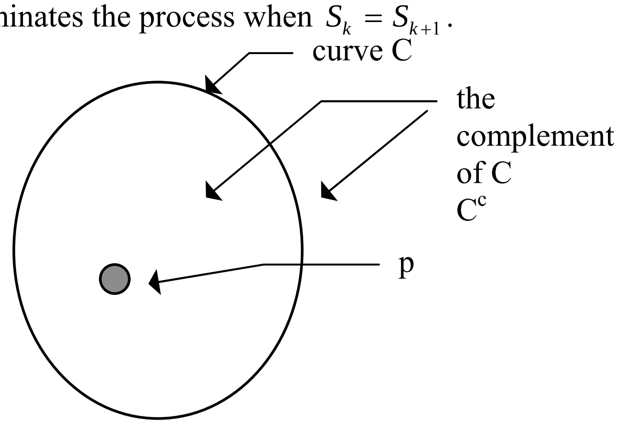


Figure 8. Curve Surrounding a Pixel

The size of ball B relative to the thickness of the curve may affect the results. If the ball is large then the region, then the operation may jump the curve.

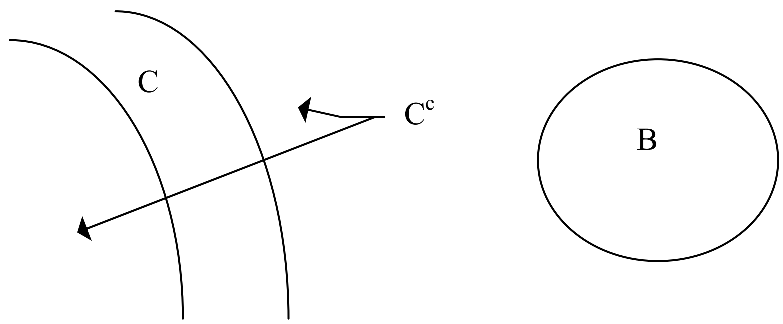


Figure 9. Thickness of the Curve and Ball B

Consider now a variant of conditional dilation for the case of a broken curve [Dougherty, 92, pp. 42].. Let *C* be a curve. Let $S_1 = \{p\}$ be inside the curve. Let B be the structuring element and

$S_k = (S_{k-1}(+)B)_D = \cup\{B+a \mid a \in S_{k-1}, (B+a) \cap C = \varphi\}$

This operator does not allow the expansion to touch the curve *C*. This will back the region away from the curve.

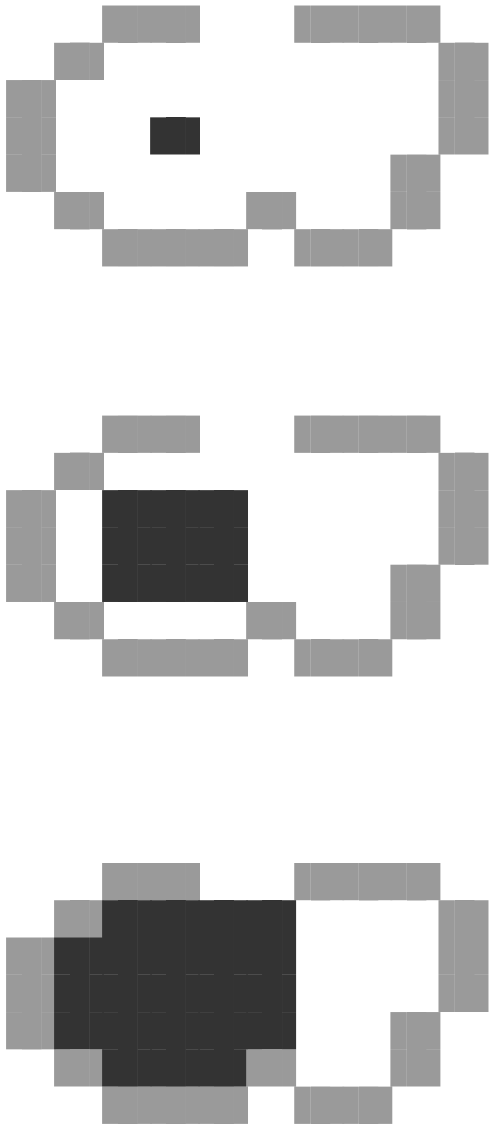


Figure 10. Example of Conditional Dilation

1.5 Geodesic Dilation

Conditional dilation and geodesic dilation are the same operation with slightly different nomenclature. Let T be a set and S a set within set A which is a component or connected subset of T. Let B be a structuring element. Then $(S(+)B)_T$ may jump across components of T if B is large. One desires to fill in each component of T. Let B be the unit ball (a 3 by 3 cell with the origin at the center). Geodesic dilation is $\delta_T^n(S,B) = \underbrace{((((S(+)B)_T(+)B)_T(+)\ldots(+)B)_T}_{\text{n times}}$ which is done by performing $(S(+)B)_T$ n times [Dougherty, 92, pp. 39] with respect to T. This is called n size geodetic dilation of S relative to T. It is also written $\delta_T^n(S,B)=(S(+)B)_n$ when the set T is understood. This form of dilation keeps the dilation from jumping gaps. In the limit $\delta_T^\infty(S,B)=\lim_{n\to\infty}\delta_T^n(S,B)=[(S\oplus B)\bigcap T]^\infty$ [Najman and Schmitt, 1996]. Note that this operation is not the same as $(S(+)nB)_T$ because large structuring elements may jump gaps.

Consider the following figure where T consists of the two components A_1 and A_2, i.e. $T = A_1 \cup A_2$ where A_1 and A_2 are connected sets.

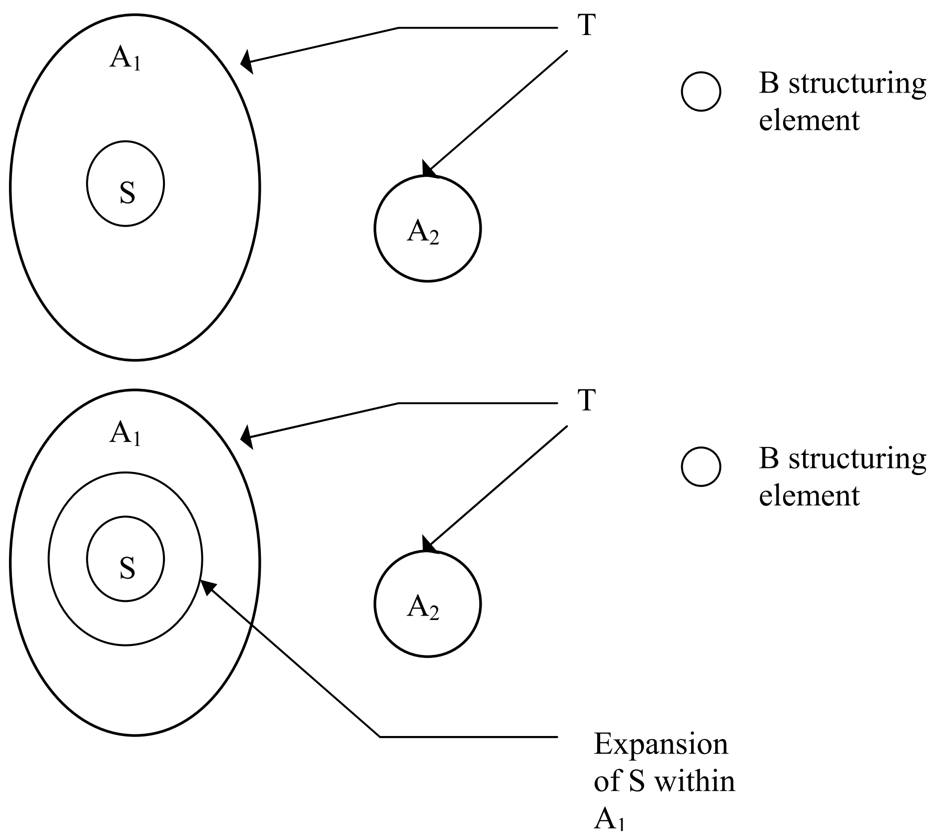


Figure 11. Geodesic Dilation

One can use geodetic dilation to compute geodesic distances. Recall that geodetic distance is $d_A(p_1,p_2)=n$ where A is a connected set. This is the shortest distance between

p_1 and p_2 with paths in set A. Let B be a 3x3 structuring element with the origin at the center. Let $S=\{p_1\}$. Then form $S_n = (S(+)nB)_A = (S(+)B)_n$ as geodetic dilation. When S_n contains p_2 for the first time then $d_A(p_1,p_2)=n$.

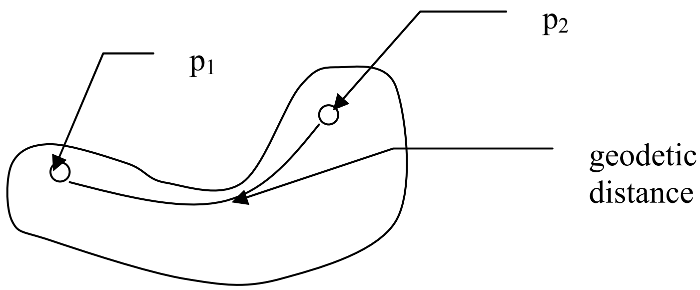


Figure 12. Geodesic Distance

This operation gives a form of reconstruction. Starting with marker points or regions one can reconstruct the object with successive dilations. These operations can also be used to fill holes into objects, and eleminate objects touching the borders of the image [Vision Design, May 99]. These type operations would need a specialized function to perform a set intersection on the final output.

The formulas for reconstruction of a set are given below [Vincent, 1993]. Let B be the unit ball. Let $S \subseteq A$ where A is a connected set

$$\delta_A^1(S) = \{(S \oplus B) \bigcap A\}$$

$$\delta_A^n(S) = \underbrace{\delta_A^1(S) \circ \delta_A^1(S) \circ \ldots \circ \delta_A^1(S)}_{\text{n times}}$$

The reconstruction of A from S is given by the formula

$$\rho_A(S) = \bigcup_{n \geq 1} \delta_A^n(S) .$$

The set S is a marker set and the set A is referred to as the mask set.

1.6 Representation by Basis Sets

Let T image operator that transforms one image into another which is translation invariant. That is, T(A+p)=T(A)+p. Suppose it is also monotonically increasing. That is $A_1 \subset A_2 \Rightarrow T(A_1) \subset T(A_2)$.The operations of erosion, dilation, opening, and closing satisfy these properties. Translation invariance means the filter acts the same across the image. The increasing property means that if one object is a subset of another then the same is true for the filtered image.

With every translation-invariant, increasing mapping *T*, there is a set of images called the kernel, Ker[T] where $Ker[T] = \{A \mid o \in T(A)\}$. The transformed image contains the origin. Then $T(A) = \cup\{A\Theta B \mid B \in Ker[T]\}$ [Dougherty, 92, pp. 13, 57]. It is the union of erosions by all elements in its kernel. In addition, if B_1, B_2 are in Ker[T] and $B_1 \subseteq B_2$ then $A\Theta B_1 \subseteq A\Theta B_2$ and B_2 is unnecessary.

The set of structuring sets $\beta \subseteq Ker[T]$ is a basis set for T if

a) $B \in \beta \Rightarrow B \not\subset B_1$.
It not a proper subset. The set is not redundant.

b) $B \in \beta \Rightarrow \exists\, B_1 \in Ker[T] \ni B_1 \subseteq B$. The set is sufficient for representation.

The set $\beta \subseteq Ker[T]$ is a basis set for T or a Bas[T]. And

$$T(A) = \cup\{A\Theta B \mid B \in Bas[T]\}$$

This is the Matheron representation. Digital operators have a Matheron representation.

Consider the following example. Here T is the opening, $T(A) = A \circ B$,by B a structuring set. Bas[T] is all translates of B that contain the origin. In order to construct translation invariant increasing filters look for basis sets of structuring elements Bas[T] and apply the formula. To construct translation-invariant, increasing filters look for bases sets of structuring elements, Bas[T] to accomplish goals and apply the formula $\cup\{A\Theta B \mid B \in Bas[T]\}$. The set Bas[T] is the set of all translates of the origin. The Matherson representation of the filter is $T(A) = A \circ B = \cup\{A\Theta B_i \mid B_i \in Bas[T]\}$.

example

B

0	1	0
1	①	1
0	1	0

	1	
1	①	1
	1	

B_1

	1	
①	1	1
	1	

B_2

	①	
1	1	1
	1	

B_3

	1	
1	1	①
	1	

B_4

	1	
1	1	1
	①	

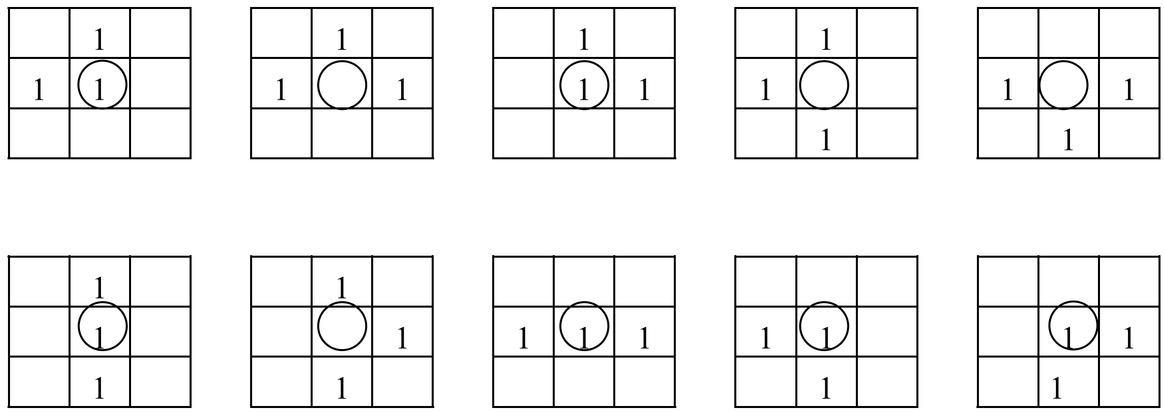
B_5

5 basis filters

Figure 13. Basis Sets

$$A \circ B = \cup\{(A \Theta B_i)|i = 1,\ldots,5\}$$

Consider the median filter over the 4-neighborhood. The basis sets



basis filters

1.7 Hit-or-Miss Transform

The hit-or-miss transform acts upon the inside and outside of a set. It is used for shape detection. One uses two structuring elements in this operation called B_1, B_2 to form a structuring pair $B = (B_1, B_2)$. The set B_1 is the hit structuring element while B_2 is the miss structuring element [Dougherty, 92, pp. 63,64; Gonzalez and Woods, 1992, pp. 528]. The transform is

$A \circledast B = (A \Theta B_1) \cap (A^c \Theta B_2)$

In addition, $A \circledast B = (A \Theta B_1) - (A \oplus \hat{B}_2)$. This is set difference. A point p is in the output iff B_1 translated to *p* fits inside *A* and B_2 translated to *p* fits outside of *A* or inside of A^c. In addition, B_1, B_2 are disjoint. The sets A and A^c are involved in determining the output. This operation is used for object recognition. For example, to recognize and mark a rectangle. One could use the following structuring elements to pass A.

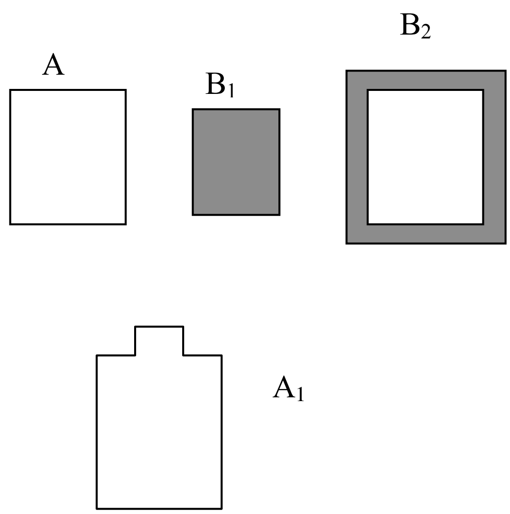


Figure 14. Hit-or -Miss Transform

The set A_1 is not passed. The next example further shows this.

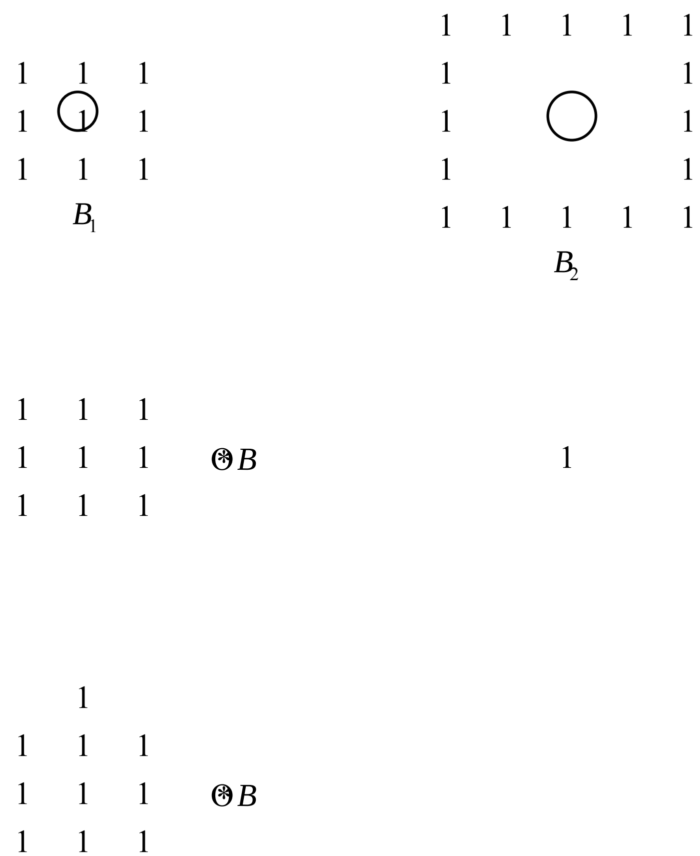


Figure 15. Example of Hit-or-Miss Transform

The first filter passes since and has a 1 in the output while the second does not pass the transformation.

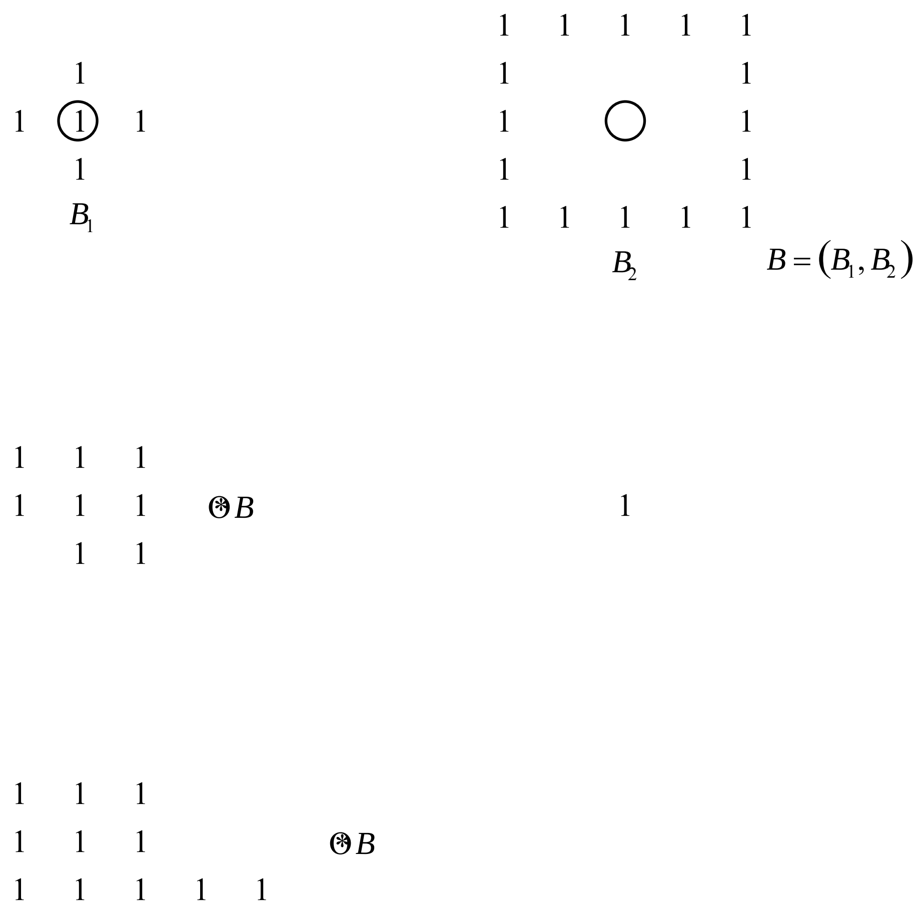


Figure 16. Example of Hit-or-Miss Transform

The first one passes but the second one does not pass the transformation.

The following is a variation of the definition [Gonzalez and Woods, 1992, pp. 530]. Let B be a structuring element and W a local window that encloses set B. If B is a structuring element then the hit or miss transform becomes
$A \circledast B = (A \Theta B) \cap (A^c \Theta (W - B))$.

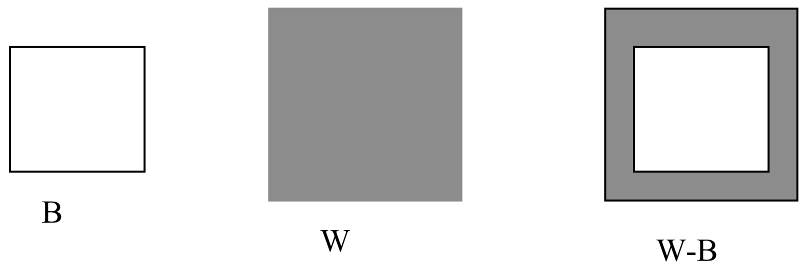


Figure 17. Alternate Definition of Hit-or-Miss Transform

1.8 Convex Hull

Let A be a set. Then the convex hull of A is the smallest convex set which contains *A* [Gonzalez and Woods, 1992, pp. 535]. The convex hull can be generated with suitable selected structuring elements.

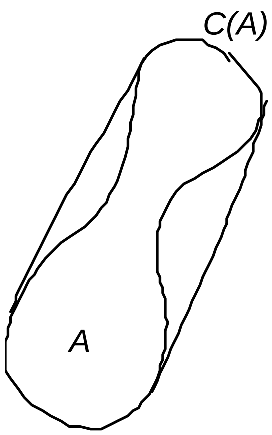


Figure 18. Convex Hull of a Set

Let $\beta = \{B^1, B^2, \cdots, B^n\}$ be a set of structuring elements.

Let

$H_k^i = (H_{k-1}^i \circledast B^i) \cup A$, i=1,2,3,...,n, k=1,2,3, ...

Let $H_0^1 = A$ where $\circledast$ is the hit or miss transform. Then D^i is set equal to H_k^i where $H_k^i = H_{k-1}^i$ in the iteration process. And then the convex hull is

$C(A) = \bigcup_{\ell=1}^{4} D^{\ell}$

The following example demonstrates the idea.

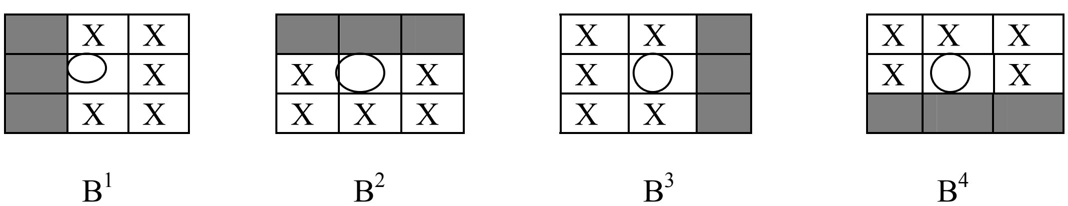
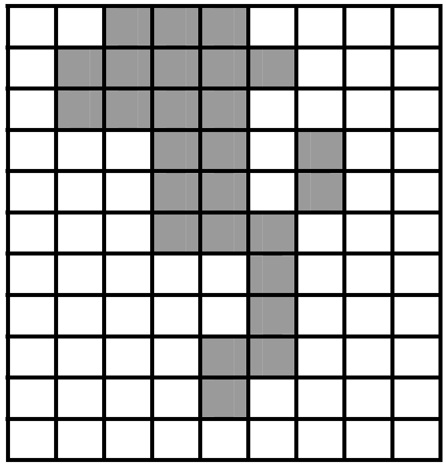
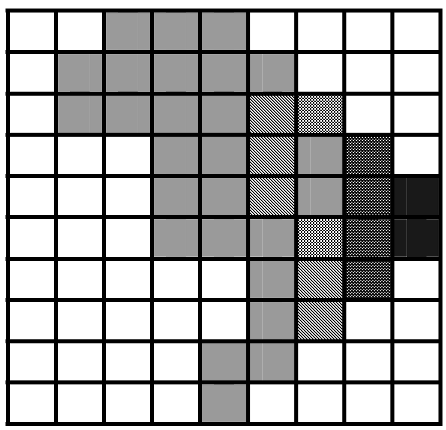


Figure 19. Structuring Elements for Convex Hull

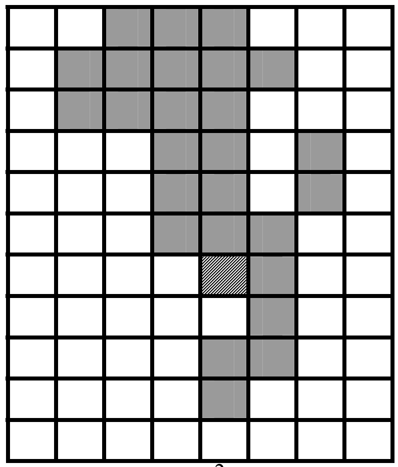
The dark cells are the ones. The X cells are not used in the calculation. The blank cells are zeros. The origin is at the center.



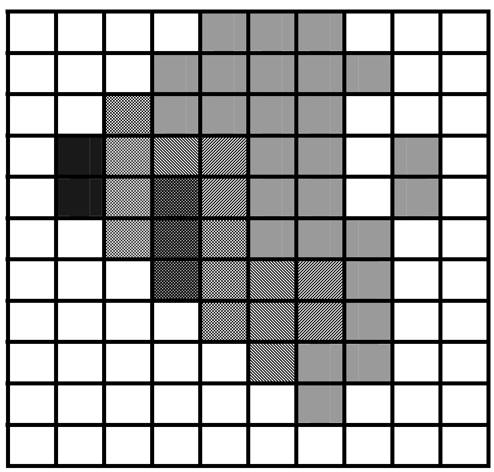
Original Set



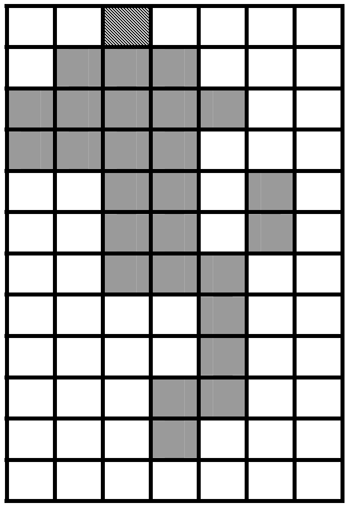
Result after application of the B^1 structuring element. The different shadings indicate the different steps in the process.



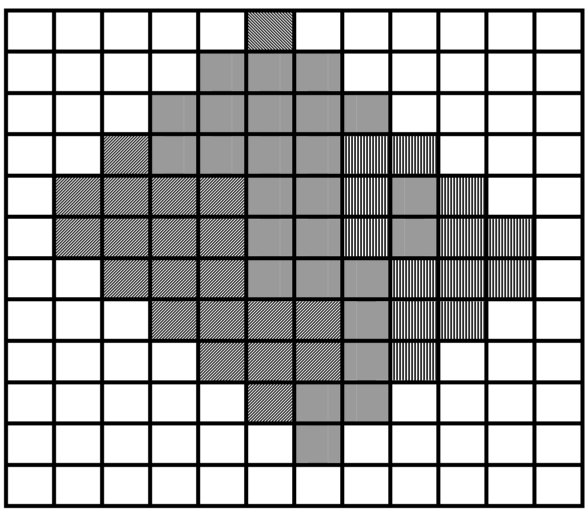
Result after application of the B^2 structuring element.



Result after application of the B^3 structuring element.



Result after application of the B^4 structuring element.



Result after application of all the structuring elements.

Figure 20. Example Convex Hull Calculation

The following examples shows how each structuring operation is formed. The X's are not used in the calculation.

1	X	X
1	**0**	X
1	X	X

B_1^1

0	X	X
0	**1**	X
0	X	X

B_2^1

1	0	1
1	0	0
1	0	0

A

0	1	0
0	1	1
0	1	1

A^c

1

$A \Theta B_1^1$

1

$A^c \Theta B_2^1$

Figure 21. Example Structuring Elements