EECE 4353 Image Processing

Lecture Notes on Mathematical Morphology: Binary Images

Richard Alan Peters II

Department of Electrical Engineering and Computer Science

Fall Semester 2016

What is Mathematical Morphology?

It is:

- nonlinear,
- built on Minkowski set theory,
- part of the theory of finite lattices,
- for image analysis based on shape,
- extremely useful, yet not often used.

Uses of Mathematical Morphology

- image enhancement
- image segmentation
- image restoration
- edge detection
- texture analysis
- particle analysis
- feature generation
- skeletonization
- shape analysis
- image compression
- component analysis
- curve filling
- general thinning
- feature detection
- noise reduction
- space-time filtering

Notation and Image Definitions

An image is a mapping, **I**, from a set, S_P, of pixel coordinates to a set, G, of values such that for every coordinate vector, $\mathbf{p} = (r,c)$ in S_P, there is a value $I(\mathbf{p})$ drawn from G. S_P is also called the *image plane*.

A *binary image* has only 2 values. That is, $G = \{v_{fg}, v_{bg}\}$, where v_{fg}, is called the *foreground* value and v_{bg} is called the *background* value.

Often, in the literature, the foreground value is $v_{fg} = 0$ and the background is $v_{bg} = -\infty$. (That is to be consistent with the mathematical morphology of intensity images.) Other possibilities are $\{v_{fg}, v_{bg}\} = \{0,\infty\}$, $\{0,1\}$, $\{1,0\}$, $\{0,255\}$, and $\{255,0\}$.

In this lecture $\{v_{fg}, v_{bg}\} = \{255, 0\}$, or $\{v_{fg}, v_{bg}\} = \{0, 255\}$, although the foreground is often displayed in different colors for contrast.

Notation and Image Definitions

> S_p is the set of of all pixel locations in the image, *i.e.*, the pixel grid, or the support of the image.

The foreground of binary image **I** is

$$\mathrm{FG}\{\mathbf{I}\} = \left\{ \mathbf{I}(\mathbf{p}),\ \mathbf{p} = (r,c) \in S_{\mathrm{P}} \,\middle|\, \mathbf{I}(\mathbf{p}) = v_{\mathrm{fg}} \right\},$$

i.e. the set of locations, **p**, where I(**p**) = v_{fg}. Similarly, the background is

$$\mathrm{BG}\{\mathbf{I}\} = \left\{ \mathbf{I}(\mathbf{p}),\ \mathbf{p} = (r,c) \in S_{\mathrm{P}} \,\middle|\, \mathbf{I}(\mathbf{p}) = v_{\mathrm{bg}} \right\}.$$

The two sets form a partition of the image:

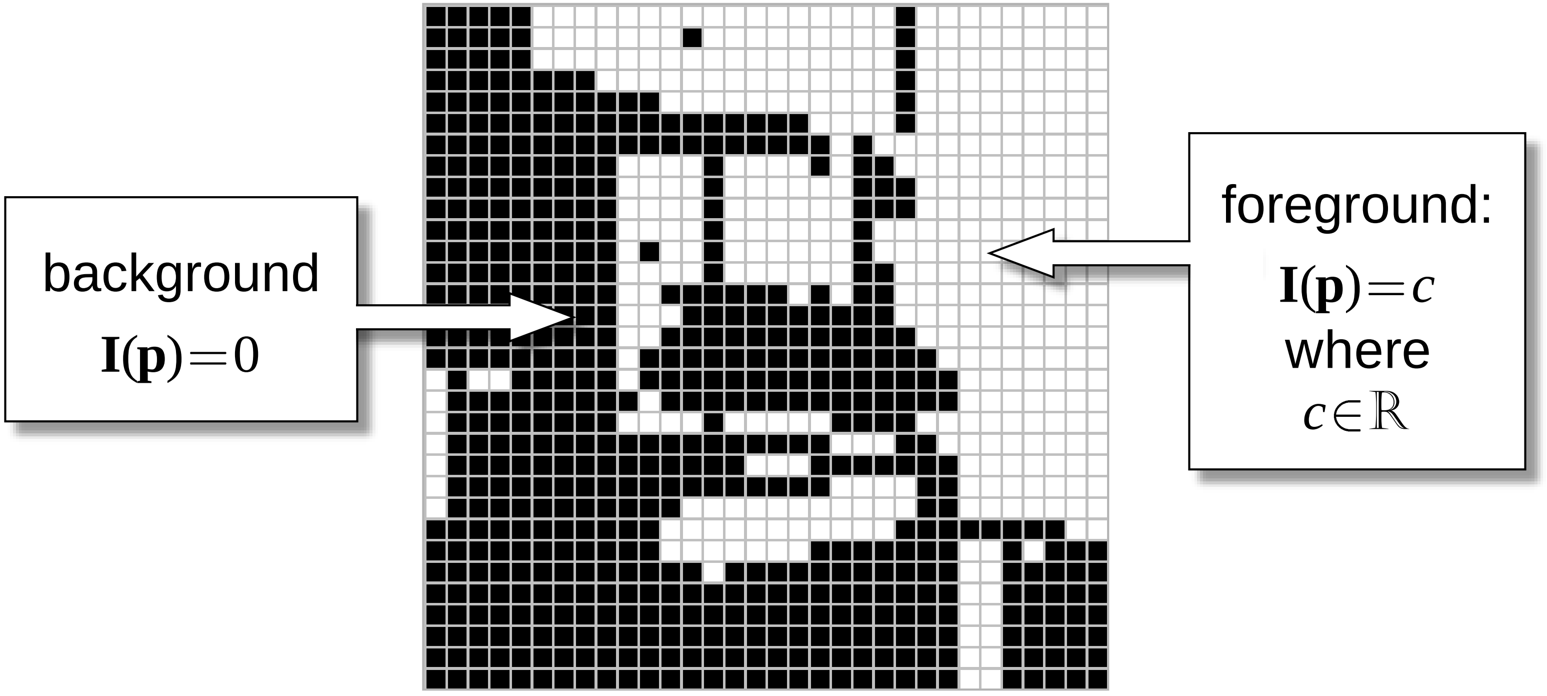
$$\mathrm{FG}\{\mathbf{I}\} \cup \mathrm{BG}\{\mathbf{I}\} = \mathbf{I} \ \text{ and } \ \mathrm{FG}\{\mathbf{I}\} \cup \mathrm{BG}\{\mathbf{I}\} = \varnothing.$$

Moreover,

$$\mathrm{BG}\{\mathbf{I}\} = \{\mathrm{FG}\{\mathbf{I}\}\}^{\mathrm{C}} \ \text{ and } \ \mathrm{FG}\{\mathbf{I}\} = \{\mathrm{BG}\{\mathbf{I}\}\}^{\mathrm{C}}.$$

The background is the complement of the foreground and vice-versa.

A Binary Image



This represents a digital image. Each square is one pixel.

Support of an Image

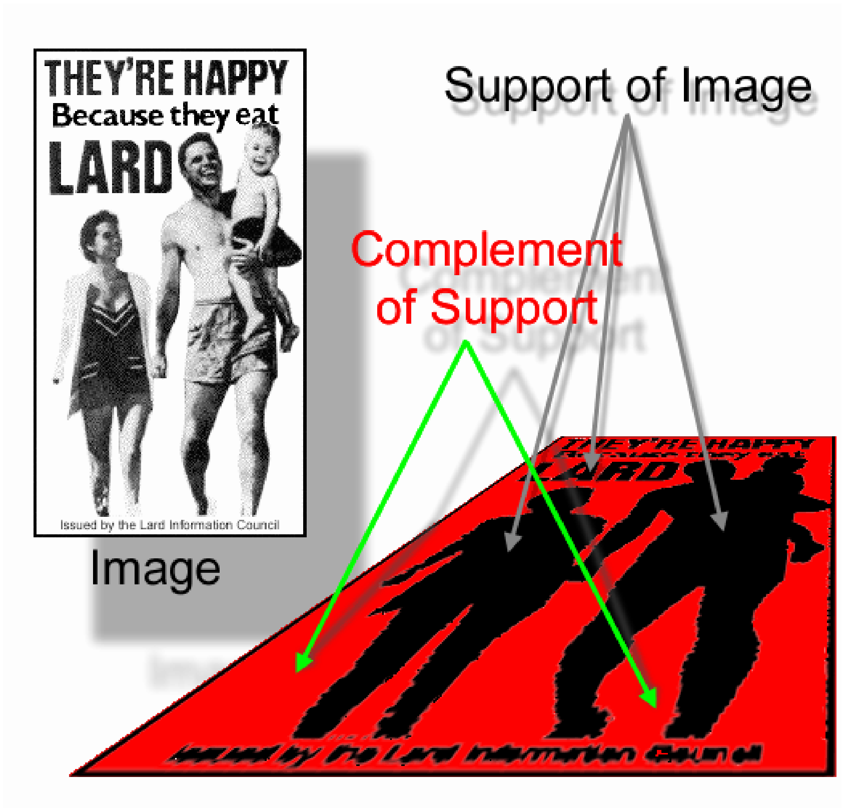
The support of a binary image is often defined as the set of locations of the foreground pixels. *I.e.*,

$$\operatorname{supp}\{\mathbf{I}\} = \left\{\mathbf{p} = (r,c) \in S_{\mathrm{P}} \,\middle|\, \mathbf{I}(\mathbf{p}) = v_{\mathrm{fg}}\right\}.$$

In that case, the complement of the support is the set of background pixel locations within the image plane.

$$\left\{\operatorname{supp}\{\mathbf{I}\}\right\}^{\mathrm{C}} = \left\{\mathbf{p} = (r,c) \in S_{\mathrm{P}} \,\middle|\, \mathbf{I}(\mathbf{p}) = v_{\mathrm{bg}}\right\}.$$

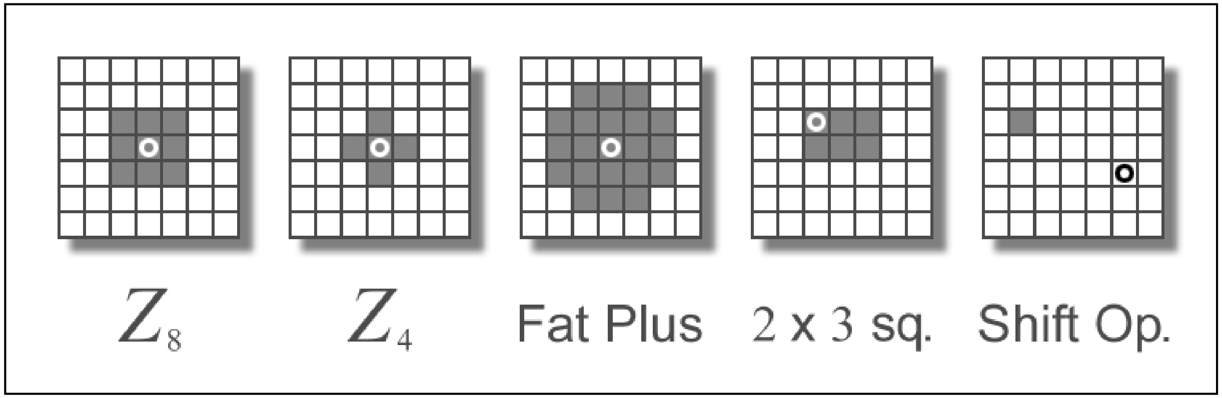
But other times, the support is defined as the set of *all* pixel locations in the image both foreground and background.



Structuring Element (SE)

A structuring element is a small image – used as a moving window – whose support delineates pixel neighborhoods in the image plane.

Example SEs



FG is gray ; BG is white

It can be of any shape, size, or connectivity (more than 1 piece, have holes). In the figure the circles mark the location of the structuring element's origin which can be placed anywhere relative to its support.

Structuring Element

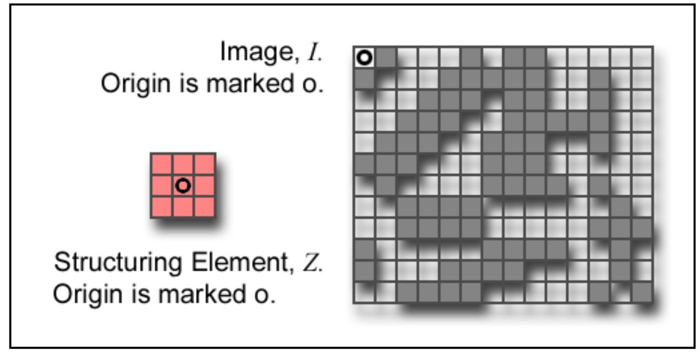
S_p is the set of of all pixel locations in the image.

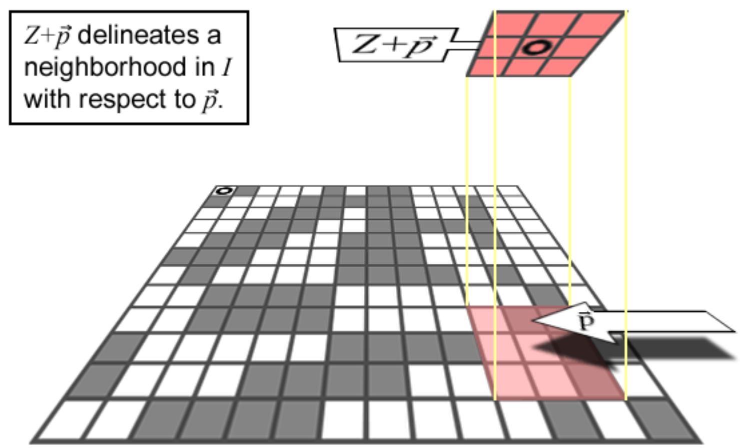
Let **I** be an image and ***Z*** a SE.

Z+**p** means that ***Z*** is moved so that its origin coincides with location **p** in S_p.

Z+**p** is the *translate* of ***Z*** to location **p** in S_p.

The set of locations in the image delineated by **Z**+**p** is called the **Z**-*neighborhood* of **p** in **I** denoted N{**I**,***Z***}(**p**).





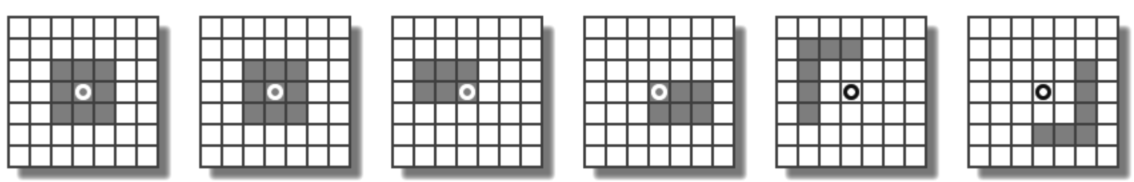
Reflected Structuring Elements

Let **Z** be a SE and let $\mathfrak{S}$ be the square of pixel locations that contains the set $\{(r,c), (-r,-c) \mid (r,c) \in \mathrm{supp}(\mathbf{Z})\}$. Then

$$\breve{\mathbf{Z}}(\rho,\chi) = \mathbf{Z}(-\rho,-\chi) \quad \text{for all} \quad (\rho,\chi) \in \mathfrak{S}.$$

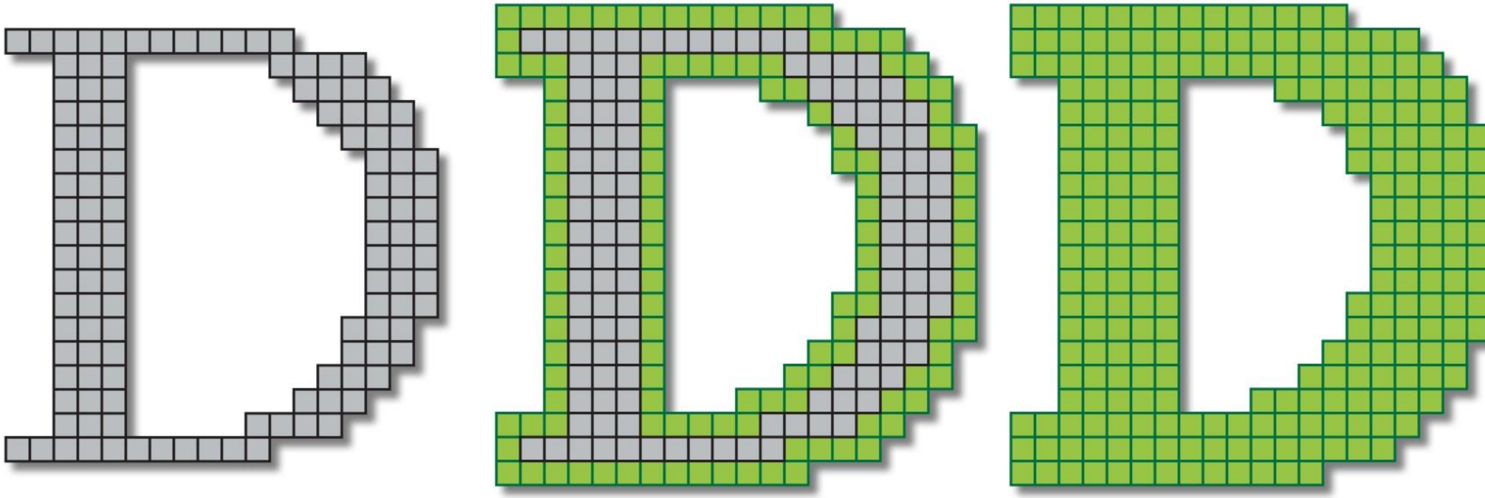
is the reflected structuring element.

$\breve{\mathbf{Z}}$ is **Z** rotated by 180º around its origin.



Z_1 $\quad \check{Z}_1$ $\quad Z_2$ $\quad \check{Z}_2$ $\quad Z_3$ $\quad \check{Z}_3$

Dilation



Dilation of Binary Images

There are a number of equivalent definitions of dilation. Three of them that apply to binary images are:

$$\mathbf{I} \oplus \mathbf{Z} = \left\{ \mathbf{p} \in S_{\mathrm{P}} \,\middle|\, \left[\left(\breve{\mathbf{Z}} + \mathbf{p} \right) \cap \mathrm{fg}\left\{ \mathbf{I} \right\} \right] \neq \varnothing \right\}.$$

The set of all pixel locations, $\mathbf{p}$, in the image plane where the intersection of $\breve{\mathbf{Z}}+\mathbf{p}$ with the foreground of $\mathbf{I}$ is not empty.

$$\mathbf{I} \oplus \mathbf{Z} = \bigcup_{\mathbf{p} \in \mathrm{fg}\{\mathbf{I}\}} \left(\mathbf{Z} + \mathbf{p} \right).$$

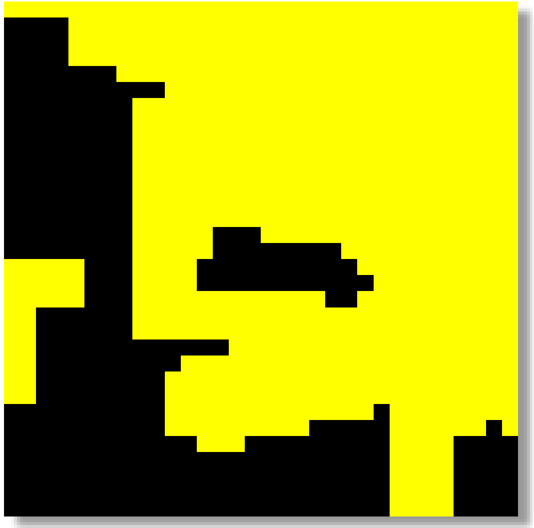
The union of copies of the SE, one translated to each pixel location in the forground of the image.

$$\mathbf{I} \oplus \mathbf{Z} = \bigcup_{\mathbf{p} \in \mathrm{fg}\{\mathbf{Z}\}} \left(\mathbf{I} + \mathbf{p} \right).$$

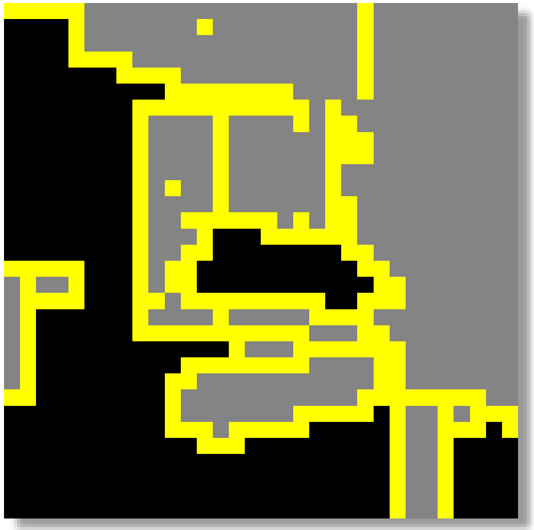
The union of copies of the image, one translated to each pixel location in the foreground of the SE.

Dilation

The locus of pixels $\mathbf{p} \in S_P$ such that $(\breve{\mathbf{Z}}+\mathbf{p}) \cap \mathrm{I} \neq \varnothing$.



dilated image



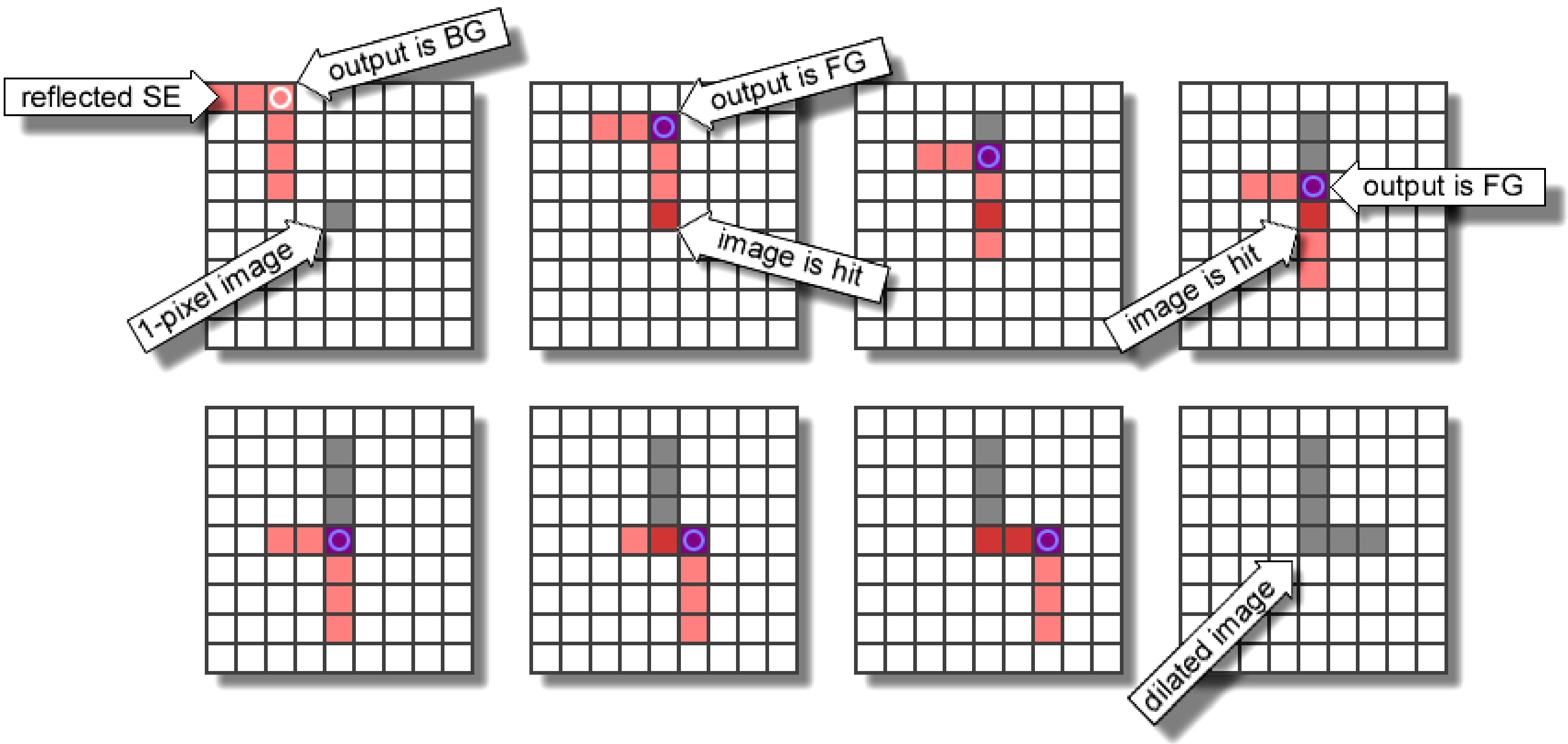
original / dilation

original image

SE = $\mathbf{Z}_8$

This is a piece of a larger image. Boundary effects are not apparent

Dilation using a Reflected SE



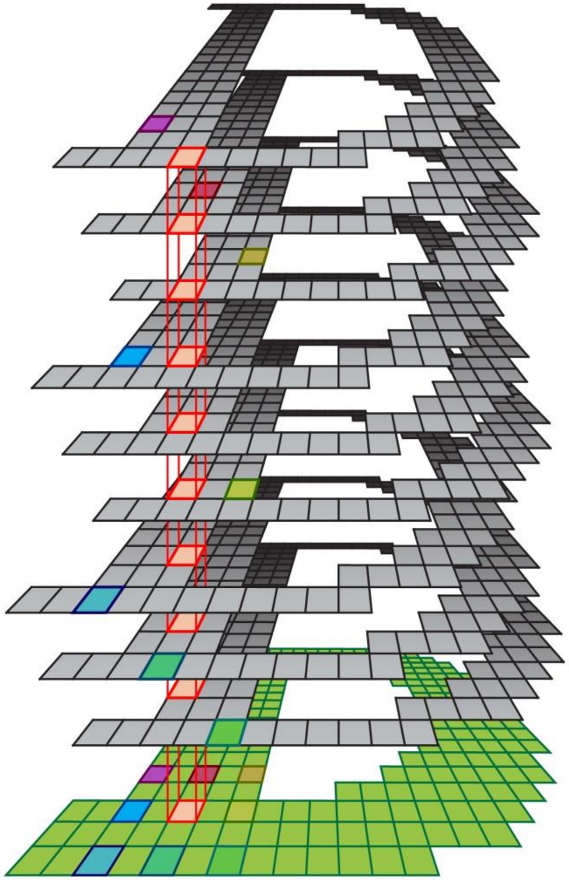
Fast Computation of Dilation

The fastest way to compute *binary* dilation is to use the union-of-translates-of-the-image definition. That is, use

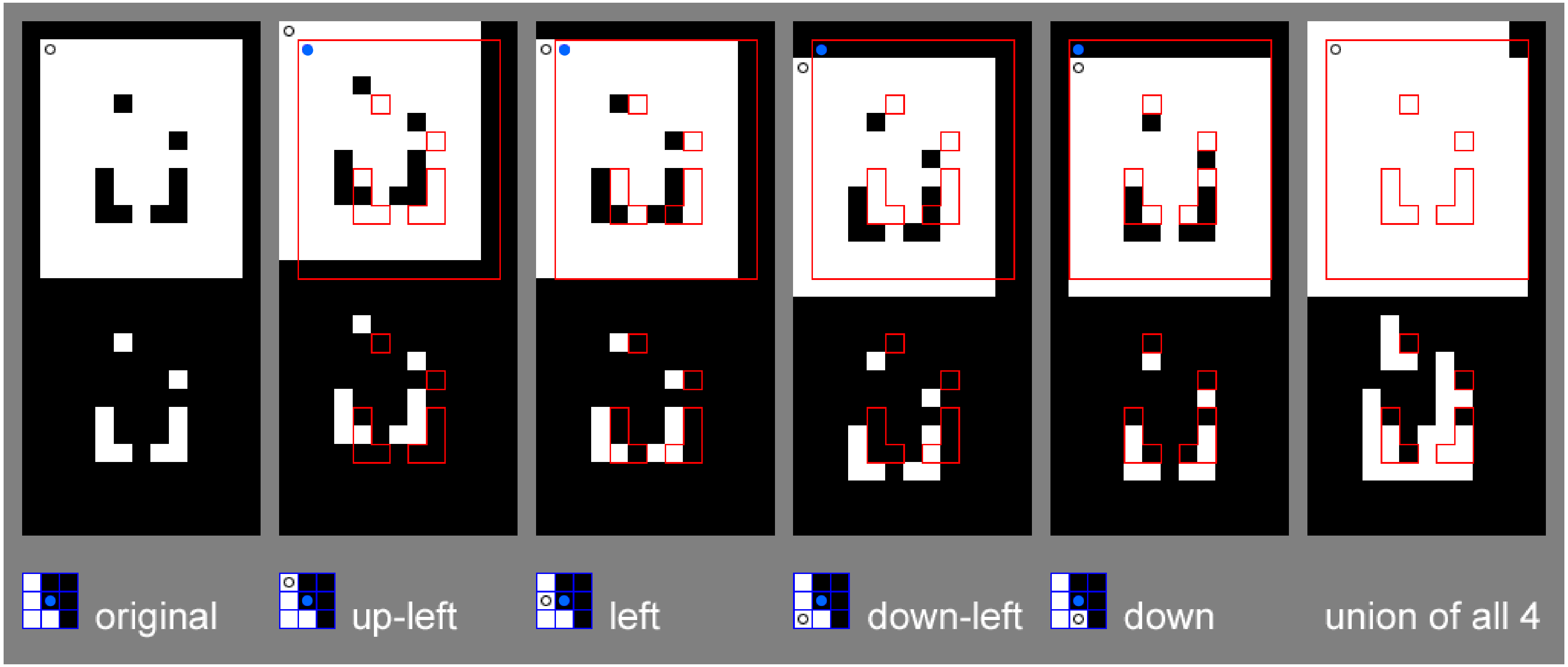
$$\mathbf{J} = \mathbf{I} \oplus \mathbf{Z} = \bigcup_{\mathbf{q} \in \mathbf{Z}} \mathbf{I} + \mathbf{q}.$$

Assume the dimensions of **I** are $R \times C$, the dimensions of **Z** are $N \times M$, and **Z**'s origin is offset from the upper left hand corner (ULHC) by ρ rows and χ columns. Allocate a scratch image, **T**, that is $(R+N-1) \times (C+M-1)$ and initialized to zeros. Then, for each FG pixel loc (v, u) in **Z** (measured from the ULHC of **Z**) perform a logical OR between $\mathbf{I}+(v, u)$ and **T**. Put the results in **T**. When done, copy to **J** the $R \times C$ subarray of **T** starting at (ρ,χ).

Dilation through Image Shifting

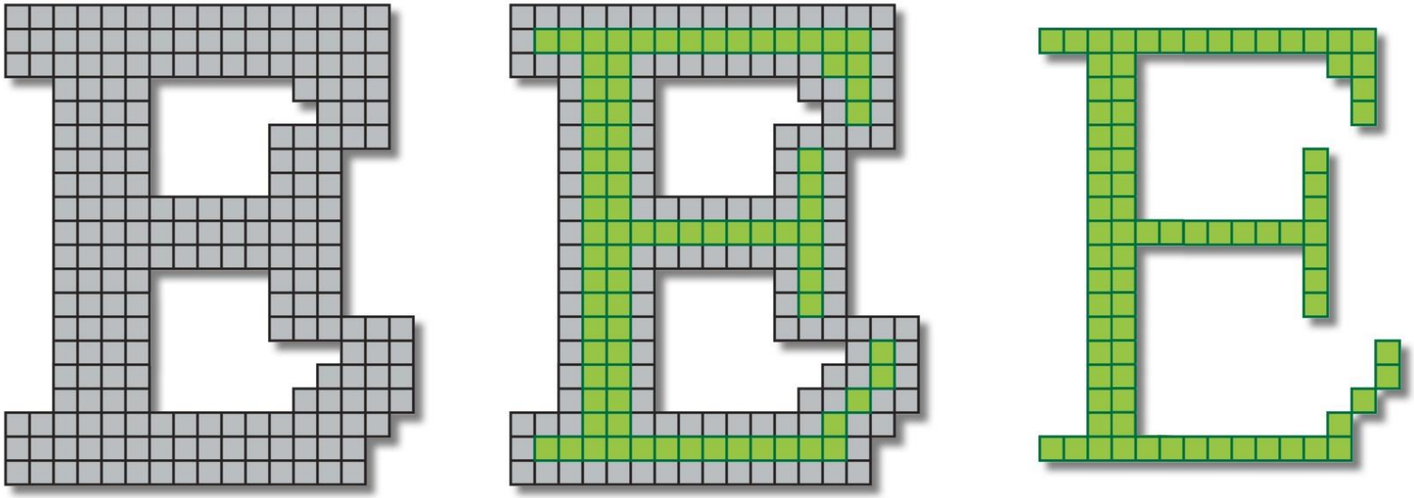


Dilation through Image Shifting



The red outlines indicate the positions of the features in the original images.

Erosion



Erosion of Binary Images

There are a number of equivalent definitions of erosion. Three of them that apply to binary images are:

$$\mathbf{I} \ominus \mathbf{Z} = \left\{ \mathbf{p} \in S_{\mathrm{P}} \,\middle|\, (\mathbf{Z} + \mathbf{p}) \in \mathrm{fg}\{\mathbf{I}\} \right\}.$$

The set of all pixel locations, **p**, in the image plane where **Z**+**p** is contained in fg{**I**}.

$$\mathbf{I} \ominus \mathbf{Z} = \bigcap_{\mathbf{p} \in \mathrm{fg}\{\mathbf{I}\}} \left(\breve{\mathbf{Z}} + \mathbf{p} \right).$$

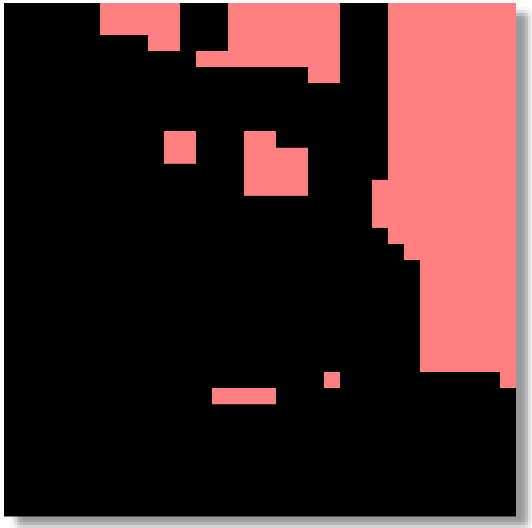
The intersection of copies of the refl. SE, one translated to each pixel location in the foreground of the image.

$$\mathbf{I} \ominus \mathbf{Z} = \bigcap_{\mathbf{p} \in \mathrm{fg}\{\breve{\mathbf{Z}}\}} \left(\mathbf{I} + \mathbf{p} \right).$$

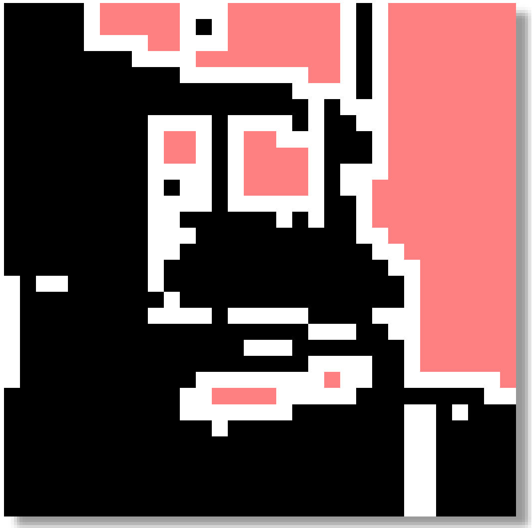
The intersection of copies of the image, one translated to each pixel location in the foreground of the refl. SE.

Erosion

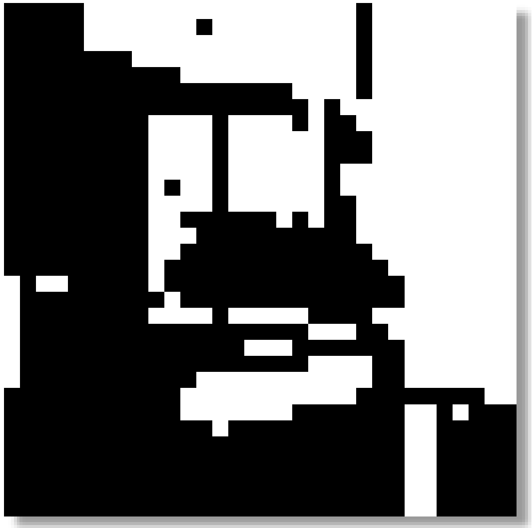
The locus of pixels $\mathbf{p} \in S_P$ such that $\mathbf{Z}+\mathbf{p} \subset \mathrm{I}$.



eroded image



erosion / original



original image

SE = $\mathbf{Z}_8$

This is a piece of a larger image. Boundary effects are not apparent

Fast Computation of Erosion

The fastest way to compute *binary* erosion is to use the intersection-of-translates-of-the-image definition. That is, use

$$\mathbf{J} = \mathbf{I} \ominus \mathbf{Z} = \bigcap_{\mathbf{q} \in \mathrm{fg}\{\breve{\mathbf{Z}}\}} \mathbf{I} + \mathbf{q}.$$

Assume the dimensions of **I** are $R \times C$, the dimensions of **Z** are $N \times M$, and **Z**'s origin is offset from the upper left hand corner (ULHC) by ρ rows and χ columns. Allocate a scratch image, **T**, that is $(R+N-1) \times (C+M-1)$ and initialized to **I**. Rotate[1] **Z** by 180º. Then, for each FG pixel loc (v, u) in $\check{\mathbf{Z}}$ (measured from the ULHC of $\check{\mathbf{Z}}$) perform a logical AND between $\mathbf{I}+(v, u)$ and **T**. Put the results in **T**. When done, copy to **J** the $R \times C$ subarray of **T** starting at $(N-\rho, M-\chi)$.

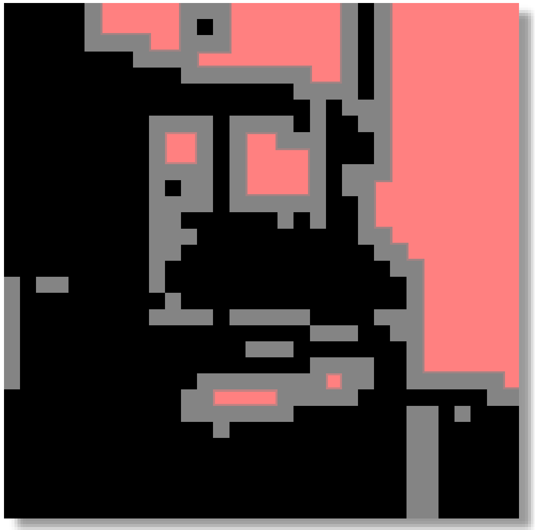
[1]In matlab the fastest way to rotate Z by 180º is

```
» Zrefl = flipud(fliplr(Z));
```

Comparison of Erosion and Dilation

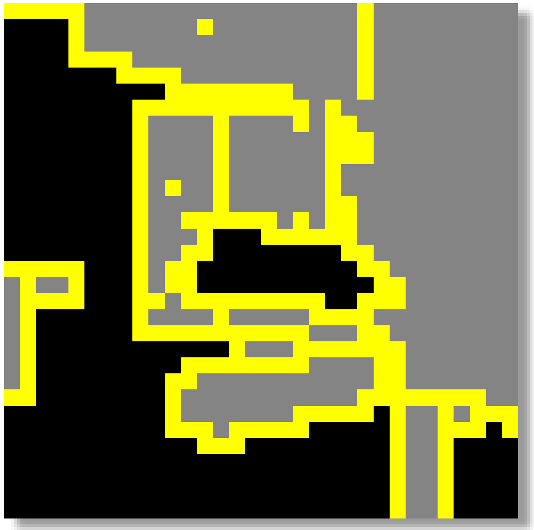
original contains erosion

dilation contains original



erosion / original

erosion / original / dilation



original / dilation

SE = $\mathbf{Z}_8$

This is a piece of a larger image. Boundary effects are not apparent

Erosion from Dilation / Dilation from Erosion

Dilation and erosion are duals of each other with respect to complementation:

$$\mathbf{I}^{C} \oplus \breve{\mathbf{Z}} = \left\{ \mathbf{I} \ominus \mathbf{Z} \right\}^{C} \quad \text{and} \quad \mathbf{I}^{C} \ominus \breve{\mathbf{Z}} = \left\{ \mathbf{I} \oplus \mathbf{Z} \right\}^{C}$$

That is, dilation with the reflected SE of the complement of a binary image is the complement of the erosion. Erosion with the reflected SE of the complement of the image is the complement of the dilation. It follows that,

$$\mathbf{I} \ominus \mathbf{Z} = \left\{ \mathbf{I}^{C} \oplus \breve{\mathbf{Z}} \right\}^{C} \quad \text{and} \quad \mathbf{I} \oplus \mathbf{Z} = \left\{ \mathbf{I}^{C} \ominus \breve{\mathbf{Z}} \right\}^{C}$$

erosion can be performed with dilation and vice versa. That implies that only one or the other must be implemented directly.

Opening and Closing

Opening is erosion by **Z** followed by dilation by **Z**.

$$\mathbf{I} \circ \mathbf{Z} = \left(\mathbf{I} \ominus \mathbf{Z}\right) \oplus \mathbf{Z}$$

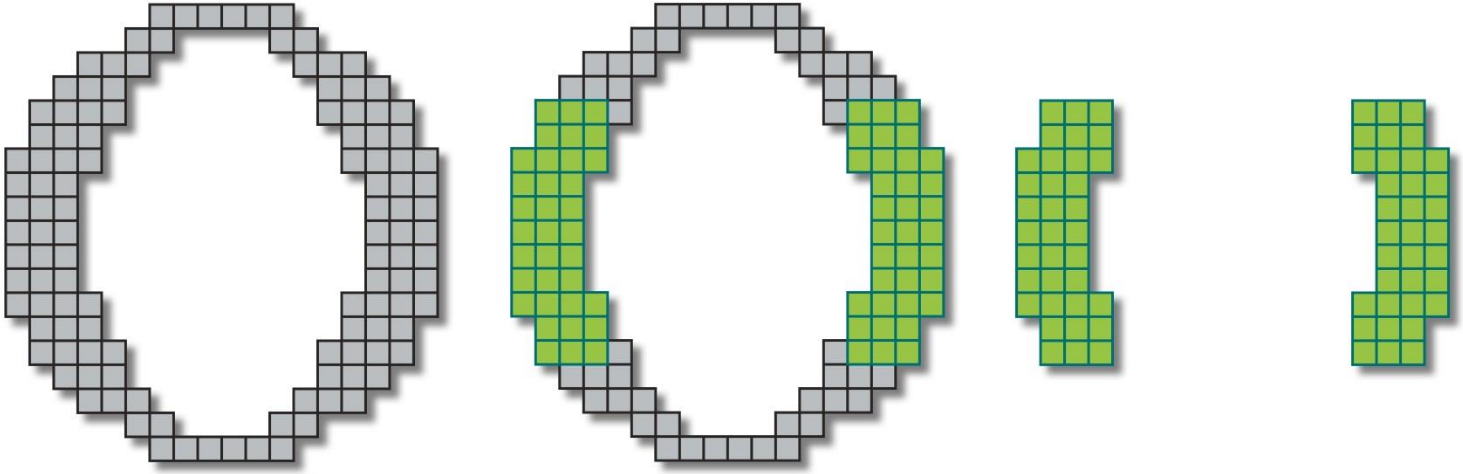
The opening is the best approximation of the image FG that can be made from copies of the SE, given that the opening is contained in the original. $\mathbf{I} \circ \mathbf{Z}$ contains no FG features that are smaller than the SE.

Closing is dilation by **Ž** followed by erosion by **Ž**.

$$\mathbf{I} \bullet \mathbf{Z} = \left(\mathbf{I} \oplus \breve{\mathbf{Z}}\right) \ominus \breve{\mathbf{Z}}$$

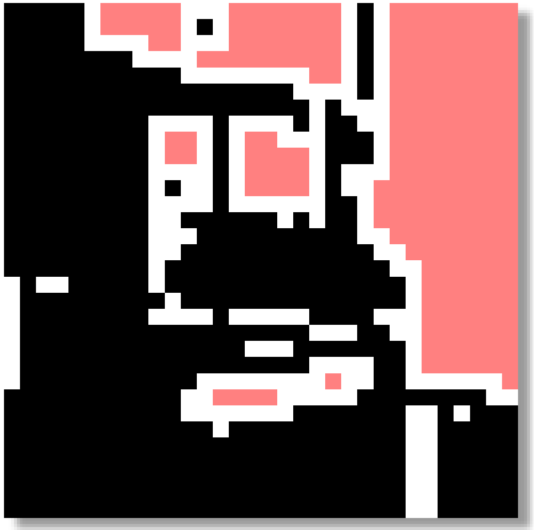
The closing is the best approximation of the image BG that can be made from copies of the SE, given that the closing is contained in the image BG. $\mathbf{I} \bullet \mathbf{Z}$ contains no BG features that are smaller than the SE.

Opening

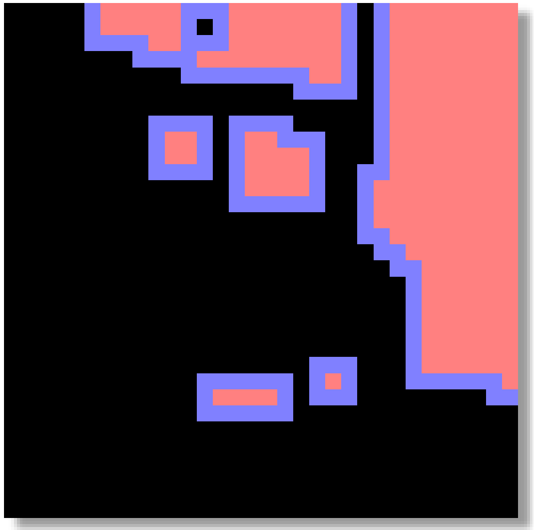


Opening is Erosion Followed by Dilation

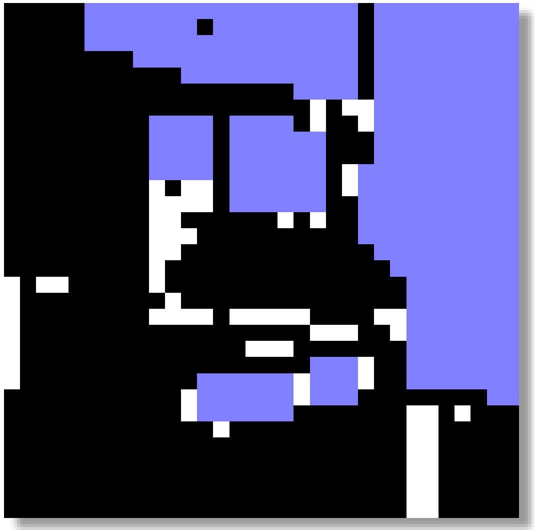
erode the original	dilate the erosion	dilated erosion



erosion / original



erosion / opening

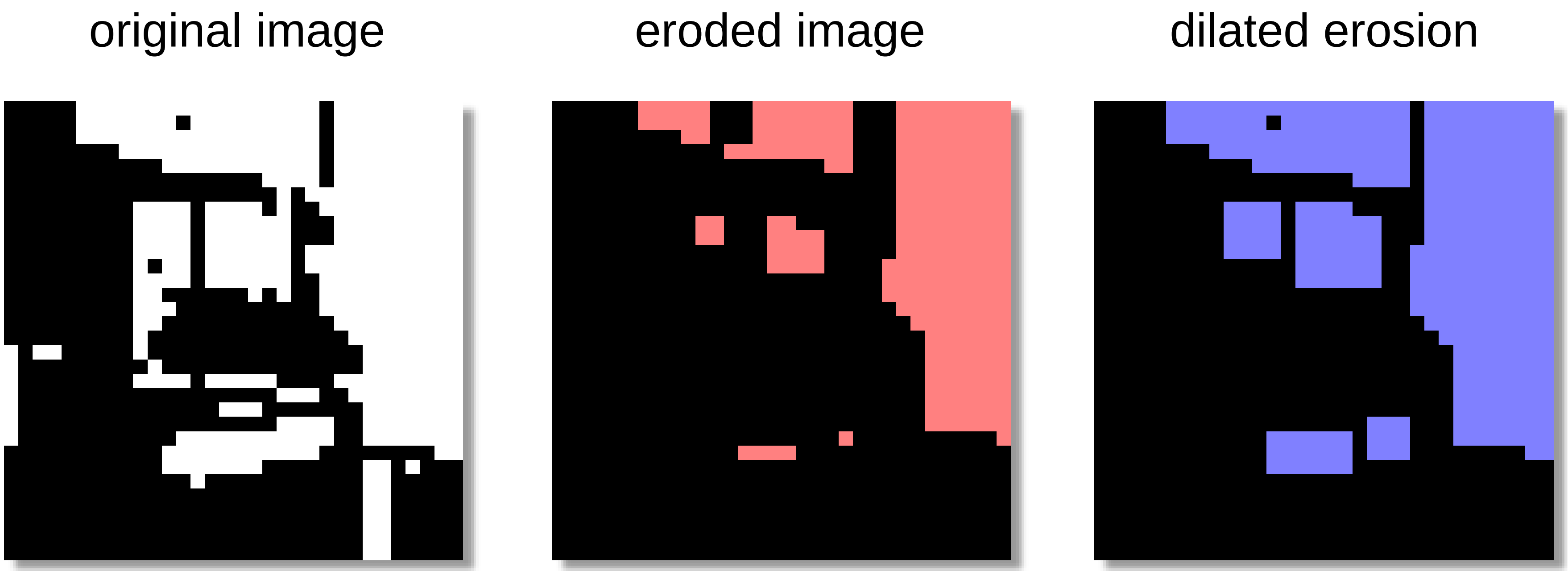


opening / original

SE = $\mathbf{Z}_8$

This is a piece of a larger image. Boundary effects are not apparent

Opening is Erosion Followed by Dilation



original · erosion · opening

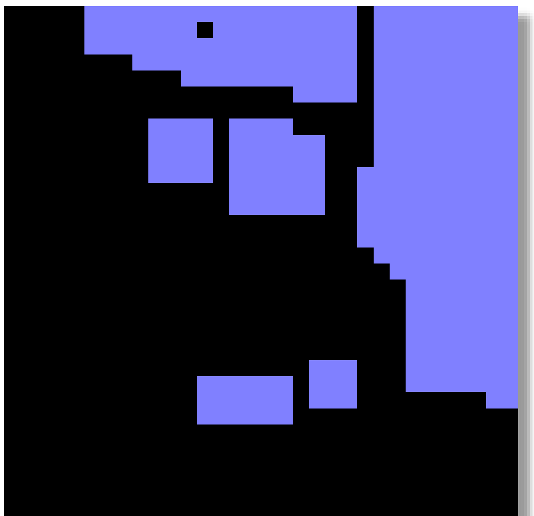
SE = $\mathbf{Z}_8$

This is a piece of a larger image. Boundary effects are not apparent

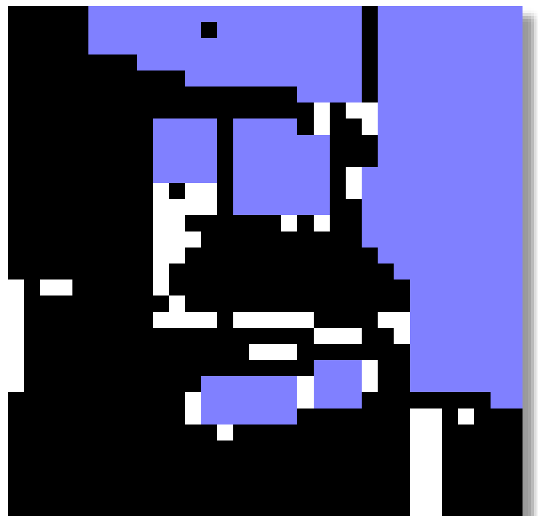
Opening

SE = $\mathbf{Z}_8$

The union of translates of $\mathbf{Z}$ such that $\mathbf{Z}+\mathbf{p}\subset\mathbf{I}$.



open image

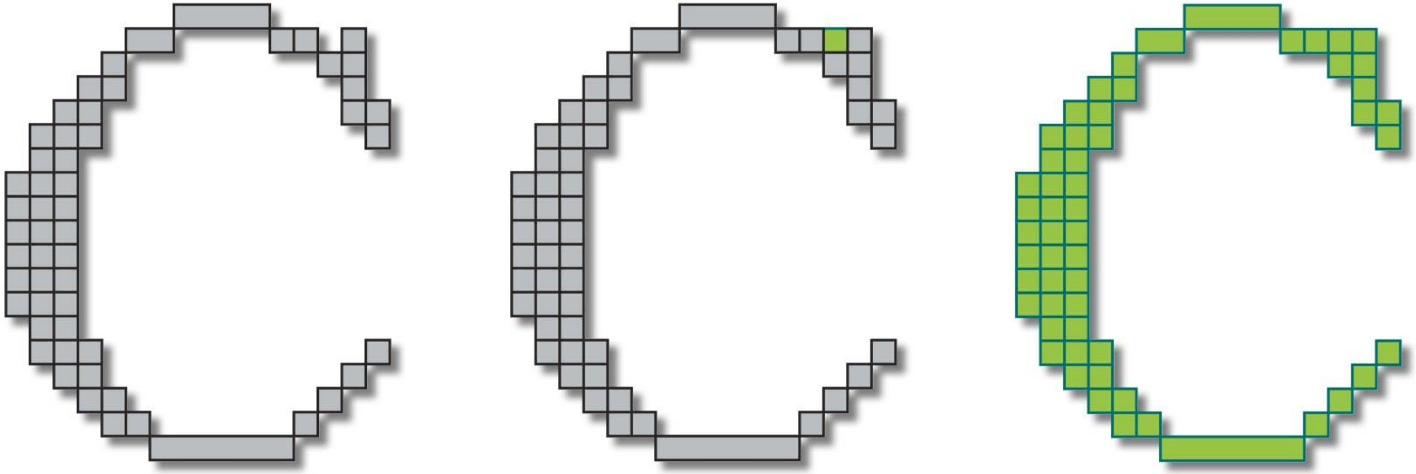


opening / original

original image

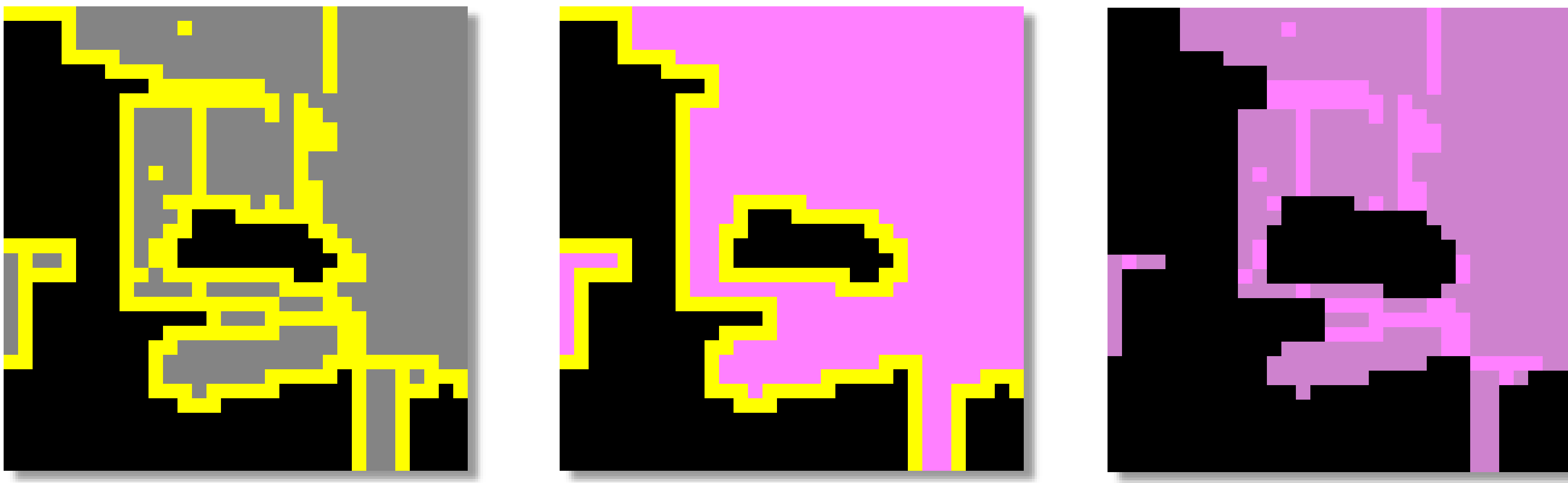
The opening of **I** by **Z** is the best approximation of **I** that can be made by taking the union of translated copies of **Z**, subject to the constraint that the opening be contained by the original image.

Closing



Closing is Dilation Followed by Erosion[1]

original image | erode the dilation | to get the closing



original / dilation | closing / dilation | closing / original

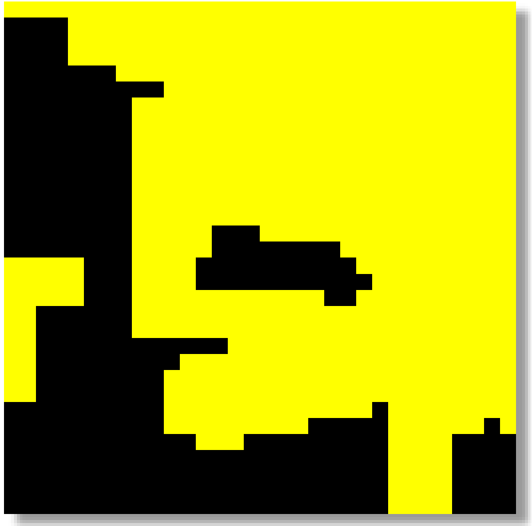
SE = $\mathbf{Z}_8$

[1]using the reflected SE, $\check{\mathbf{Z}}$

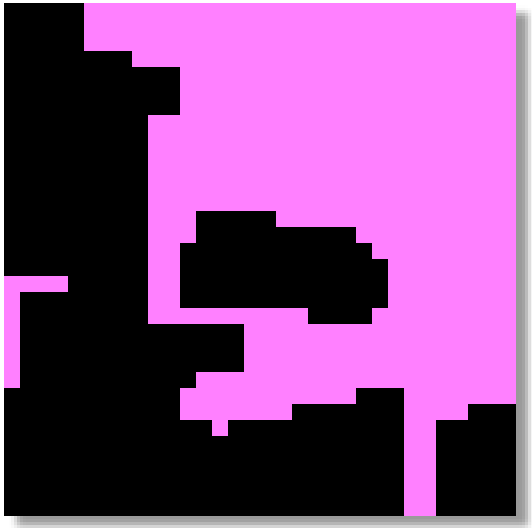
Closing is Dilation Followed by Erosion[1]

original image | dilated image | eroded dilation

original



dilation



closing

SE = $\mathbf{Z}_8$

[1]using the reflected SE, $\check{\mathbf{Z}}$

Duality Relationships

Erosion in terms of dilation: $\mathbf{I} \ominus \mathbf{Z} = \left[\mathbf{I}^C \oplus \breve{\mathbf{Z}} \right]^C$

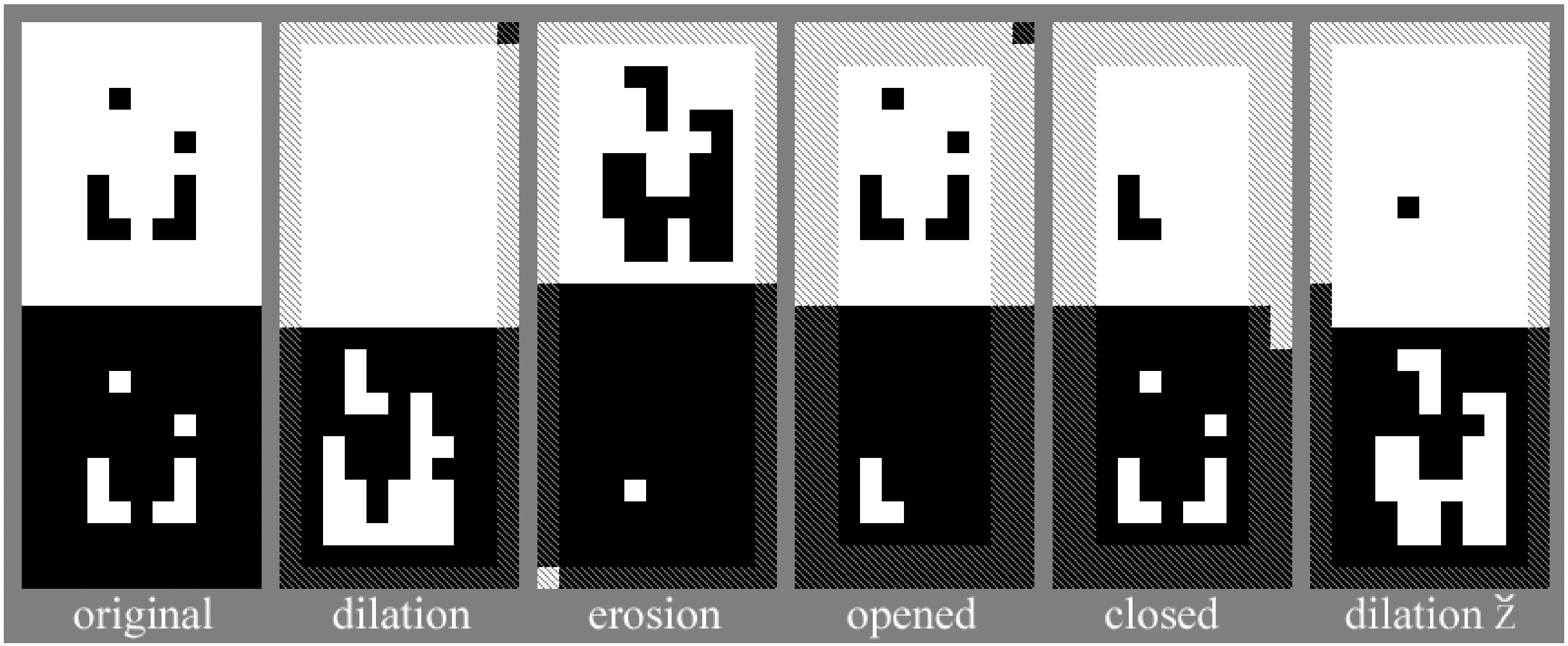
Dilation in terms of erosion: $\mathbf{I} \oplus \mathbf{Z} = \left[\mathbf{I}^C \ominus \breve{\mathbf{Z}} \right]^C$

Opening in terms of closing: $\mathbf{I} \circ \mathbf{Z} = \left[\mathbf{I}^C \bullet \mathbf{Z} \right]^C$

Closing in terms of opening: $\mathbf{I} \bullet \mathbf{Z} = \left[\mathbf{I}^C \circ \mathbf{Z} \right]^C$

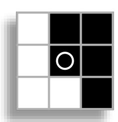
$\mathbf{I}^C$ is the complement of $\mathbf{I}$ and $\check{\mathbf{Z}}$ is the reflected SE.

Binary Ops with Asymmetric SEs



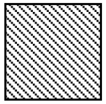
"L" shaped SE

O marks origin



Foreground: white pixels

Background: black pixels



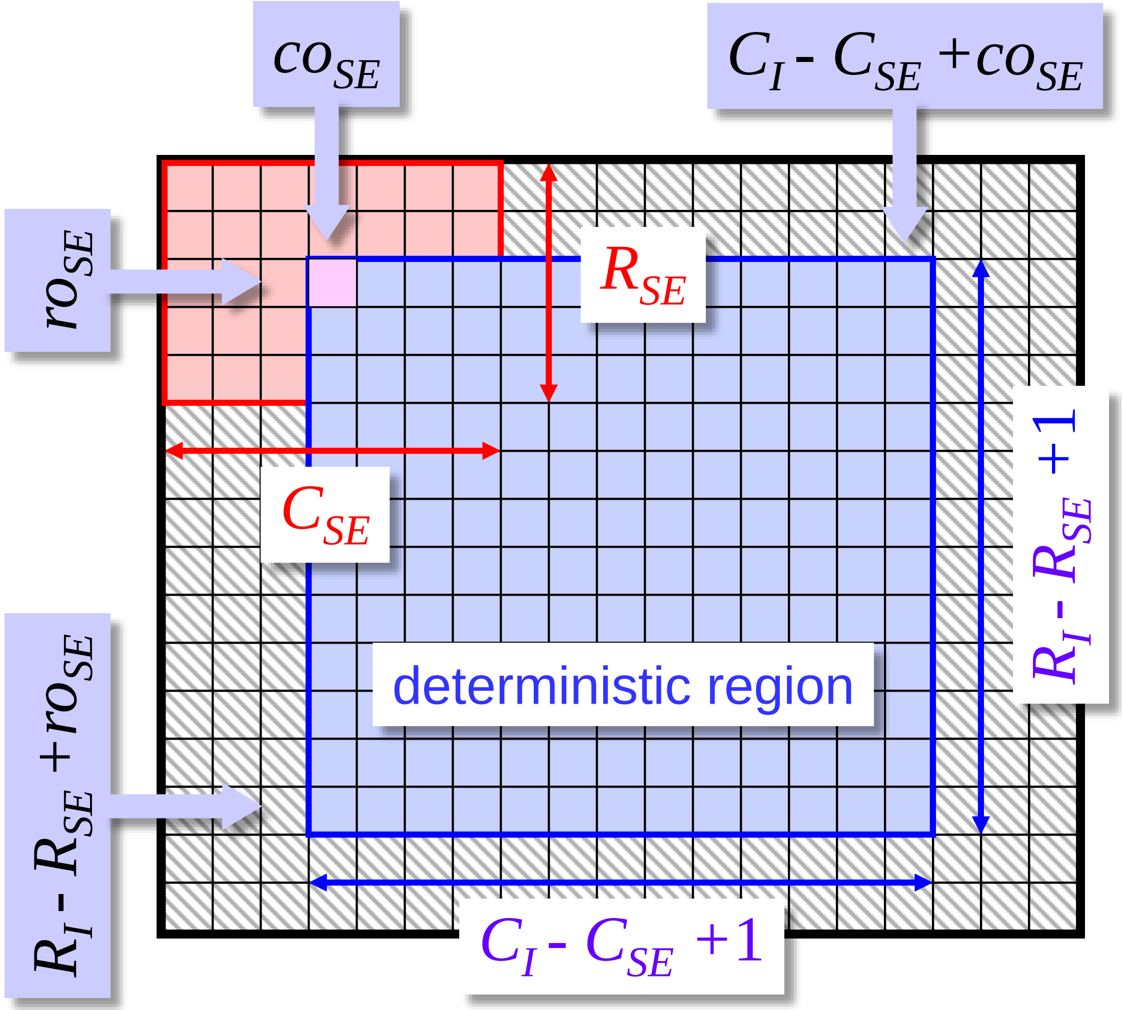
Cross-hatched pixels are indeterminate.

Border Effects

Erosion & Dilation

Since morph. ops. are neighborhood ops., there is a band of pixels around the border of the resultant image where the values are indeterminate.

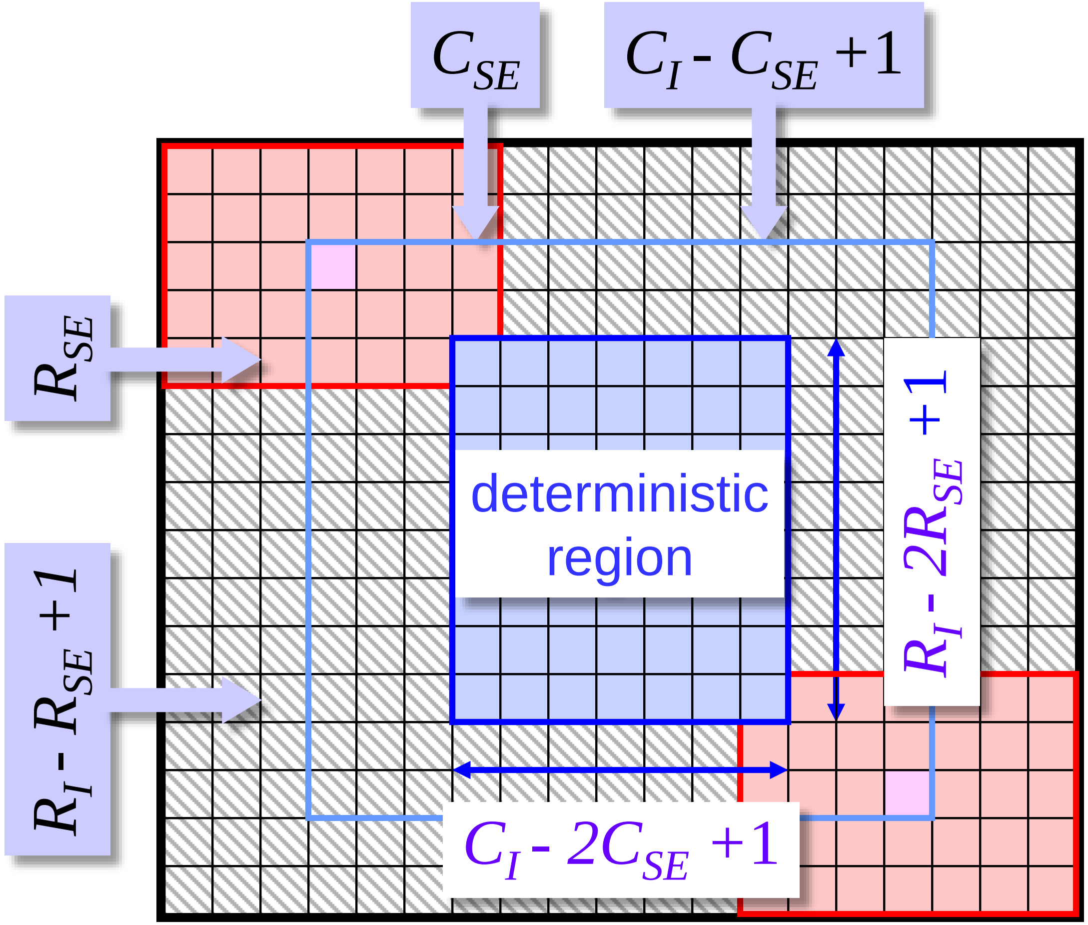
The actual values of pixels in the indet. region depend on the specific algorithm used.



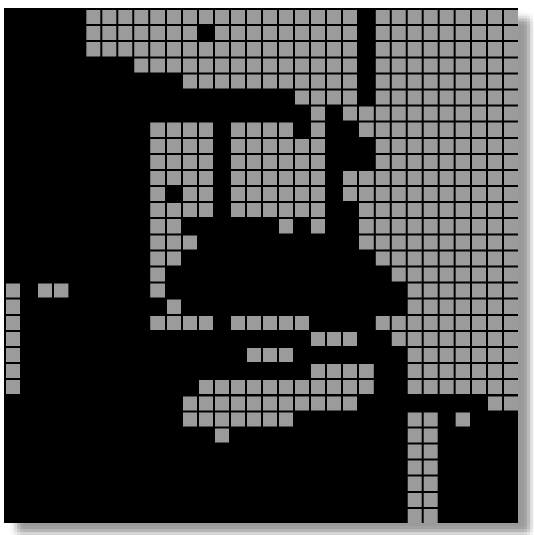
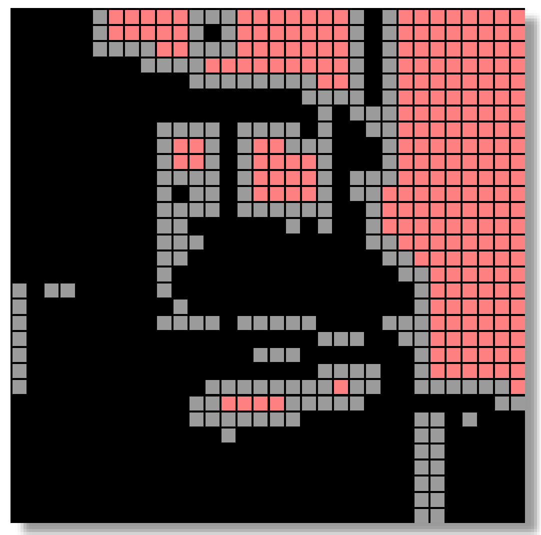
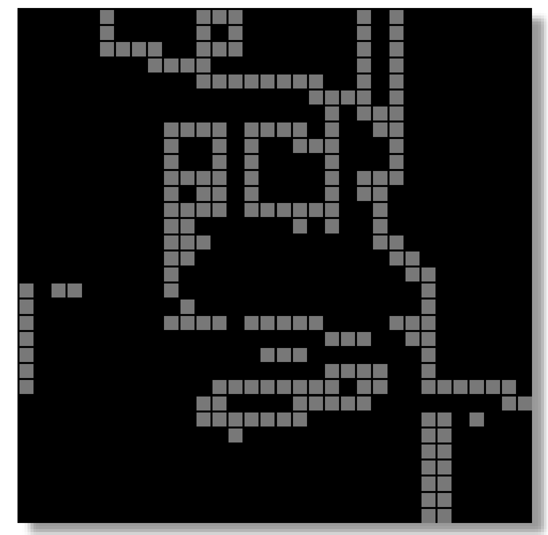
Border Effects

Opening & Closing

Since opening & closing iterate erosion & dilation, the boundaries of the deterministic region are 2x as far from the image border as are those of erosion or dilation.



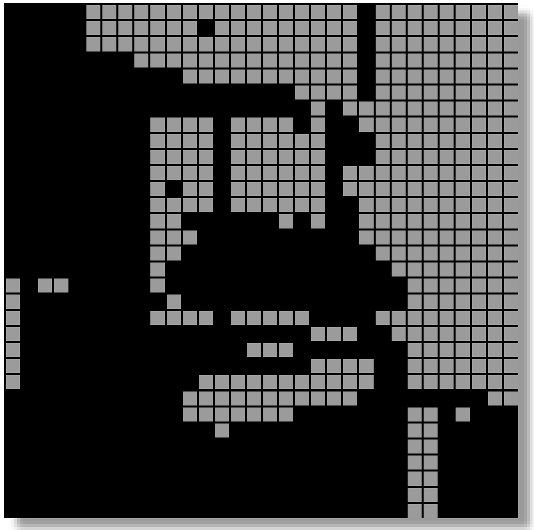
Boundary Extraction

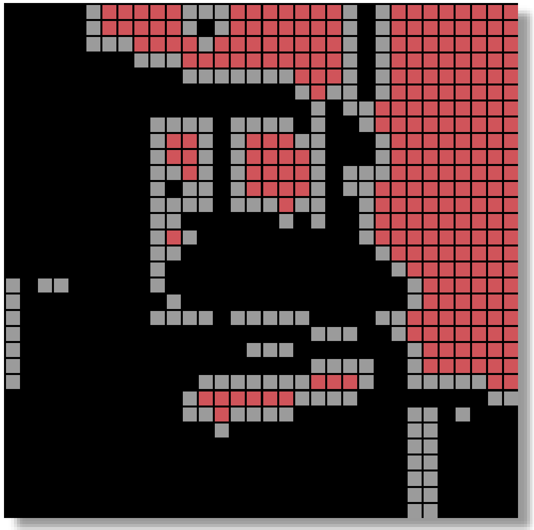
binary image	8-connected SE	4-conn inside bdry
		
original	erosion by square	difference

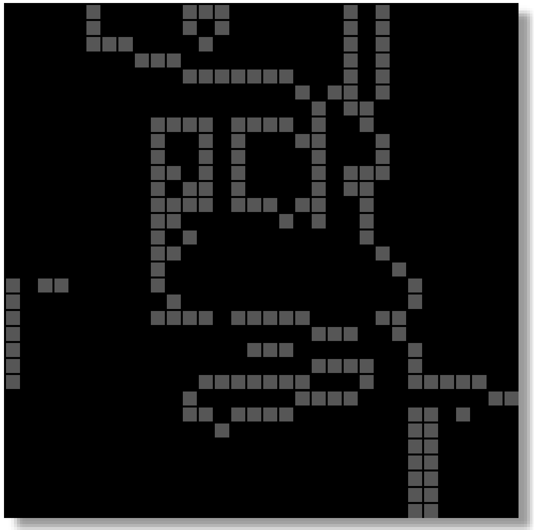
This is a piece of a larger image. Boundary effects are not apparent

Boundary Extraction

binary image	4-connected SE	8-conn inside bdry





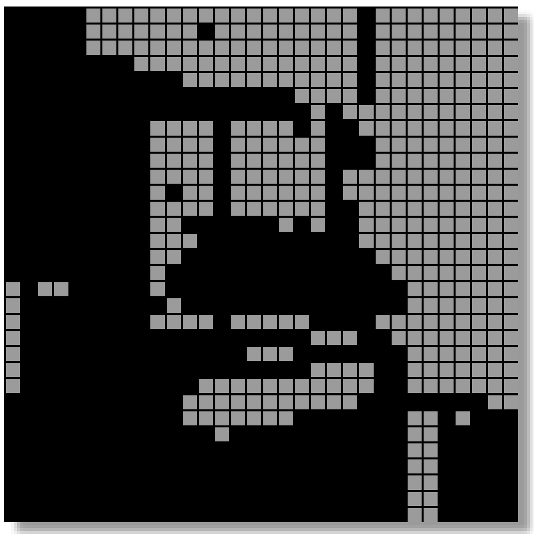


original	erosion by plus	difference

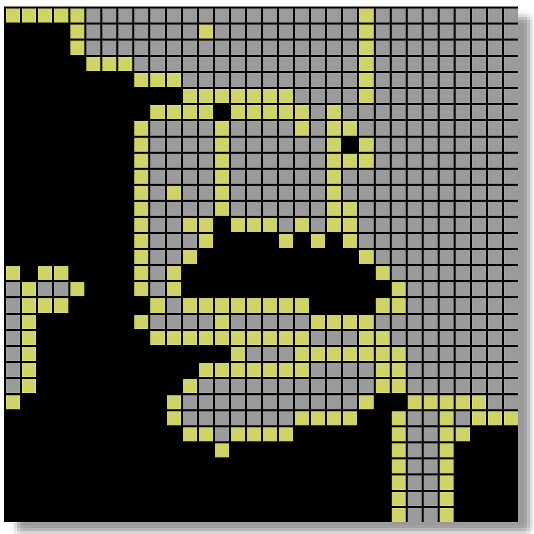
This is a piece of a larger image. Boundary effects are not apparent

Boundary Extraction

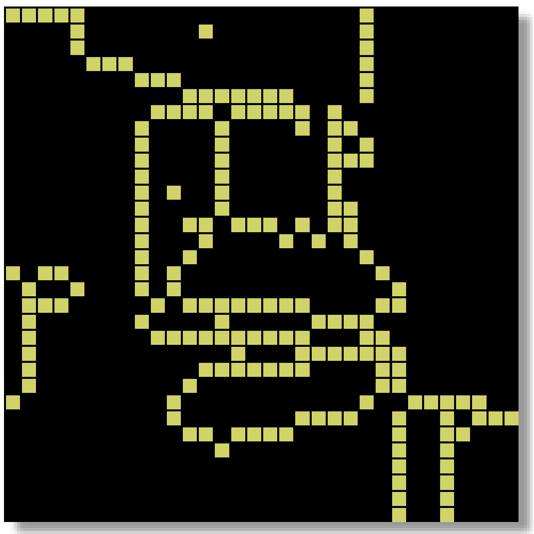
binary image	4-connected SE	8-conn outside bdry



original



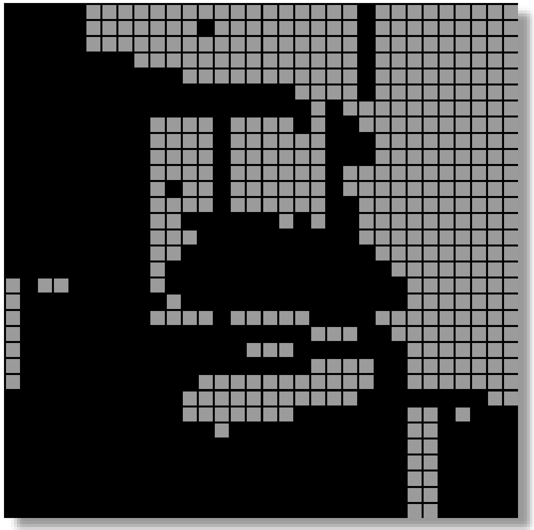
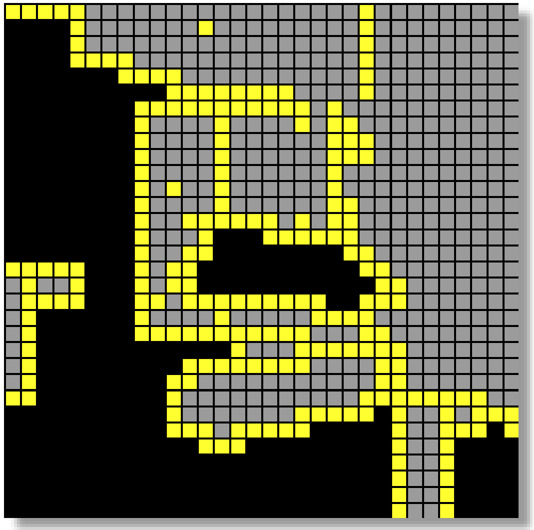
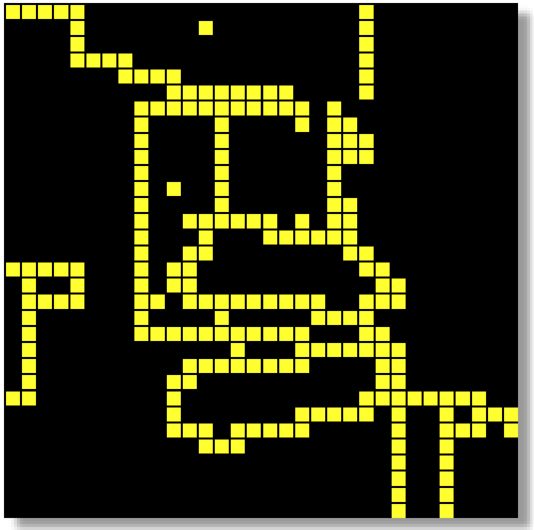
dilation by plus



difference

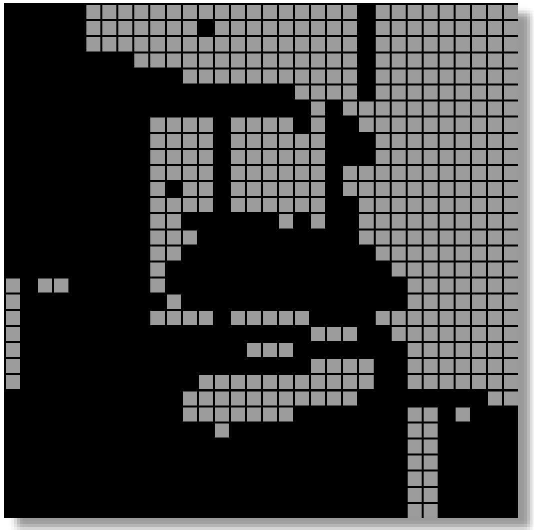
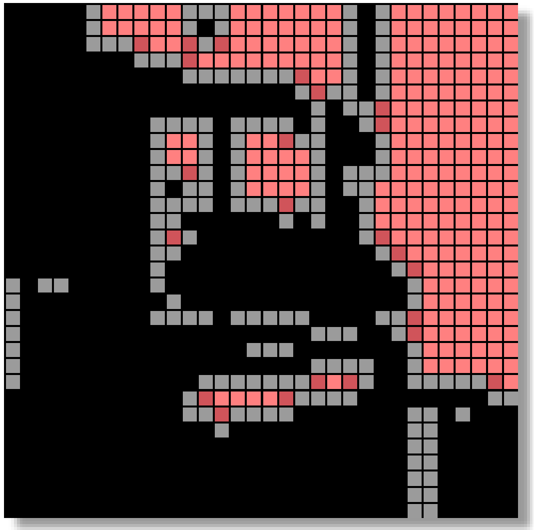
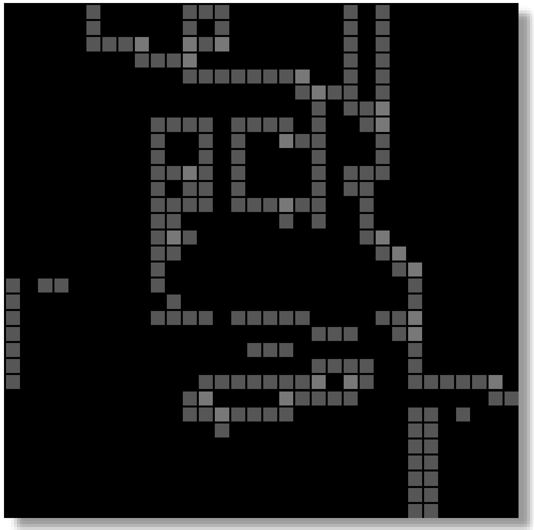
This is a piece of a larger image. Boundary effects are not apparent

Boundary Extraction

binary image	8-connected SE	4-conn outside bdry
		
original	dilation by square	difference

This is a piece of a larger image. Boundary effects are not apparent

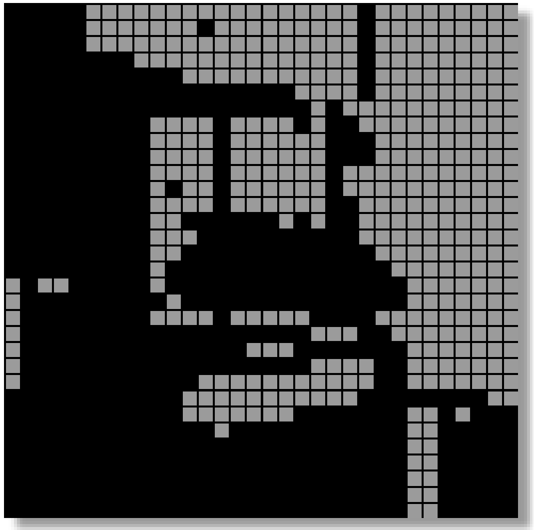
Boundary Extraction

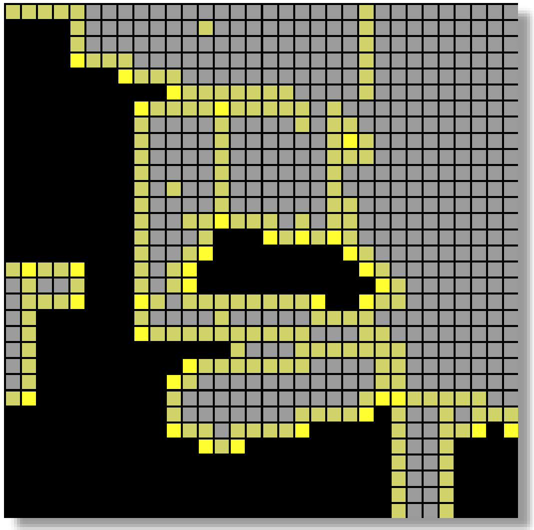
binary image	erosion by square is	8-conn inside bdry is
		
original	in erosion by plus	in 4-conn inside bdry

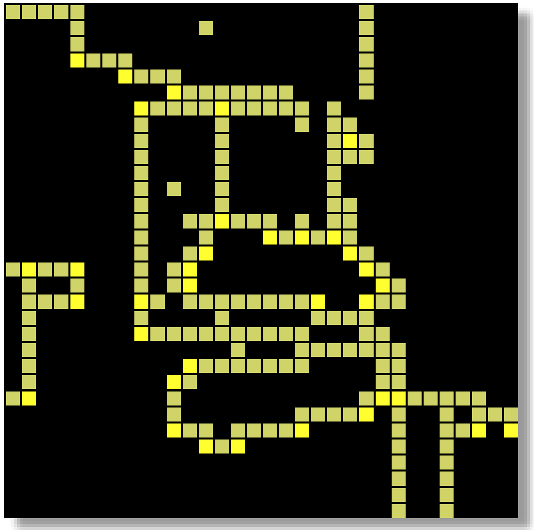
This is a piece of a larger image. Boundary effects are not apparent

Boundary Extraction

binary image | dilation by plus is | 8-conn outside bdry is





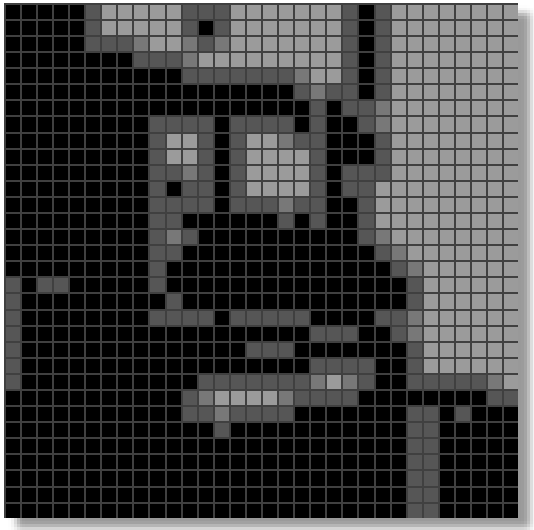


original | in dilation by square | in 4-conn outside bdry

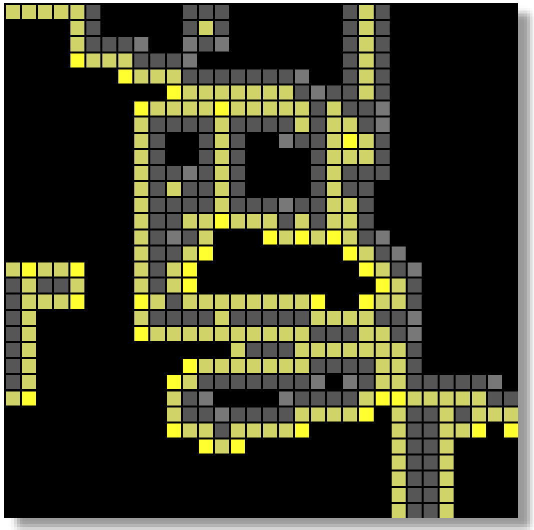
This is a piece of a larger image. Boundary effects are not apparent

Boundary Extraction

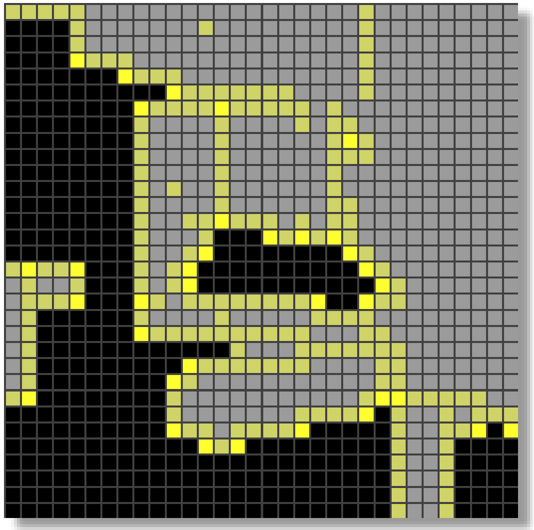
inside boundaries | are disjoint from | outside boundaries



8-bdry/4-bdry/orig



all 4 boundaries

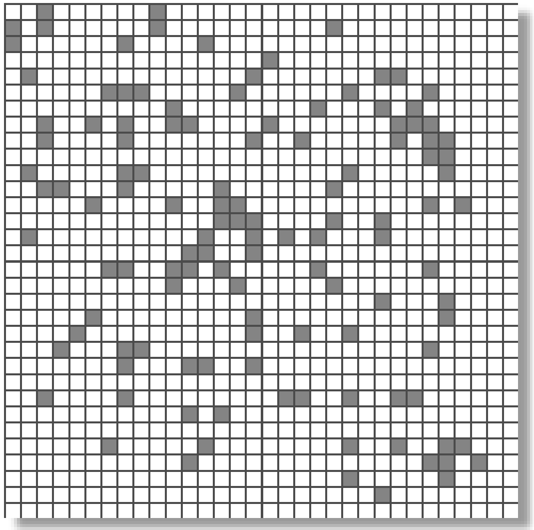


orig/8-bdry/4-bdry

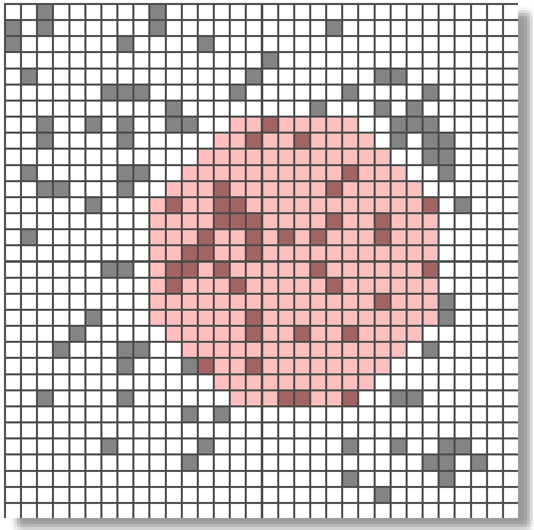
This is a piece of a larger image. Boundary effects are not apparent

Conditional Dilation

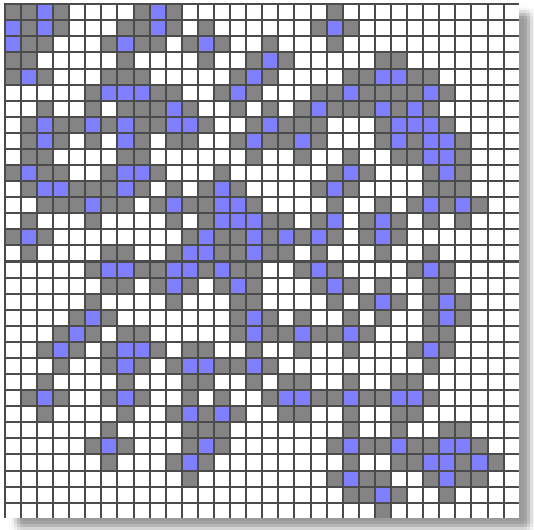
original image



mask over original



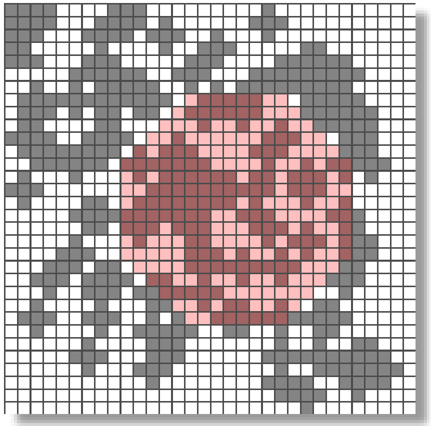
dilated original

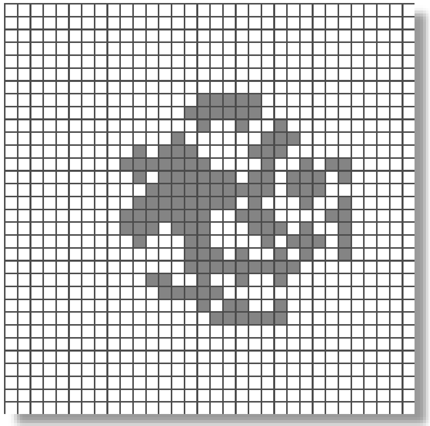


dilation inside a mask

Conditional Dilation

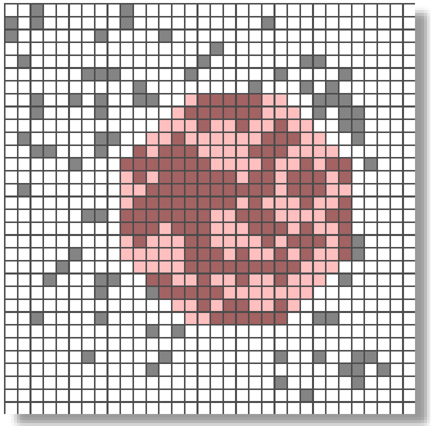
mask over dilated

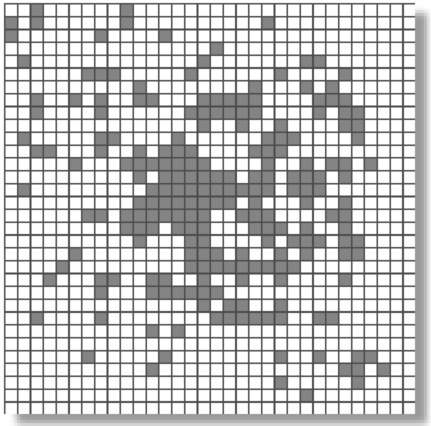




masked dilated

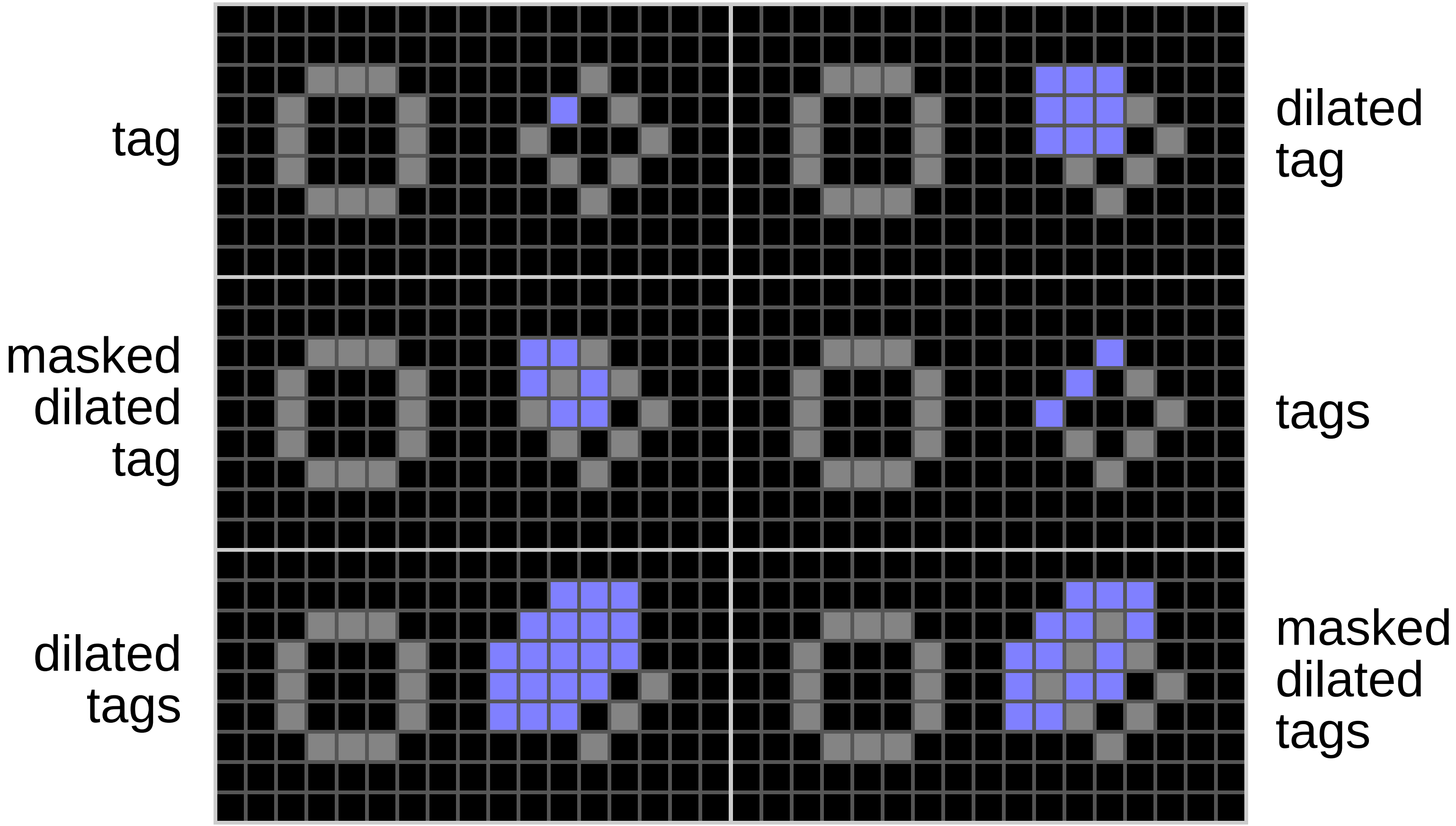
masked dilated union with original



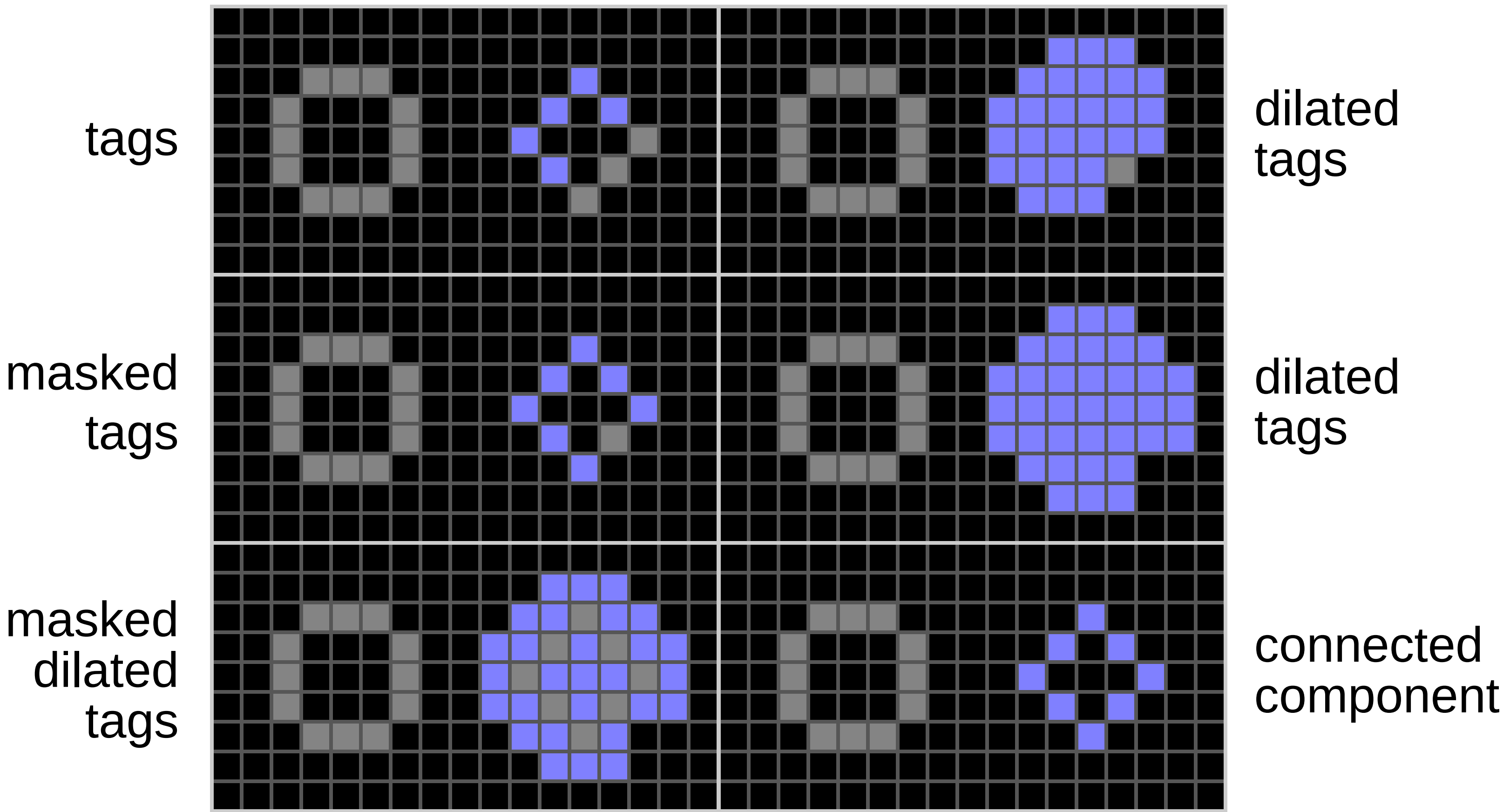


conditionally dilated with respect to mask

Connected Component Extraction



Connected Component Extraction

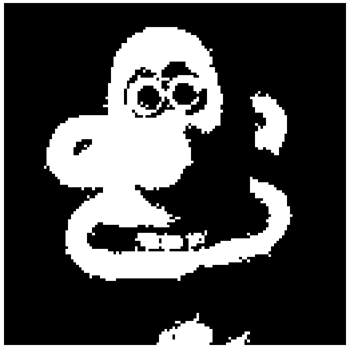


Binary Reconstruction

Used after opening to *grow back* pieces of the original image that are connected to the opening.


original


opened


reconstructed

Removes of small regions that are disjoint from larger objects without distorting the small features of the large objects.

Algorithm for Binary Reconstruction

1. $J = I \circ Z$, where Z is any SE.
2. $T = J$,
3. $J = J \oplus Z_k$, where k=4 or k=8,
4. $J = I \text{ AND } J$, [*Take only those pixels from J that are also in I .*]
5. if $J \neq T$ then go to 2,
6. else stop; [*J is the reconstructed image.*]

This is the same as connected component extraction with the opened image, J, containing the tags. The choice of Z_k determines the connectivity of the result.

Algorithm for Bina

Usually a program for reconstruction will take both J and I as inputs. E.g,

```
K = ReconBin(I,J,Z);
```

Then the algorithm starts at step 2.

1. $J = I \circ Z$, where
2. $T = J$,
3. $J = J \oplus Z_k$, where k=4 or k=8,
4. $J = I \text{ AND } J$, [*Take only those pixels from J that are also in I .*]
5. if $J \neq T$ then go to 2,
6. else stop; [*J is the reconstructed image.*]

This is the same as connected component extraction with the opened image, J, containing the tags. The choice of Z_k determines the connectivity of the result.

Skeletonization

Let Skel($\mathbf{I}$,r) be the set of pixels in $\mathbf{I}$ such that if $\mathbf{p} \subseteq$ Skel($\mathbf{I}$,r) then $D_p(r)$, is a maximal disk of radius r in $\mathbf{I}$. That is, Skel($\mathbf{I}$,r) is the locus of centers of maximal disks of radius r in $\mathbf{I}$. Then

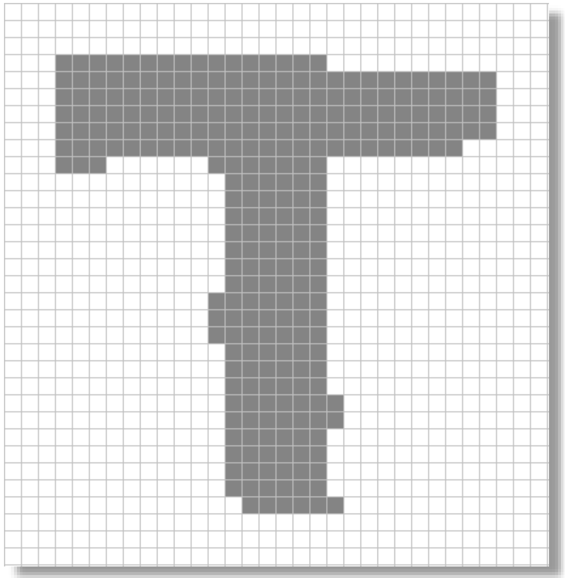
$$S = \bigcup_{r=0}^{\infty} \mathrm{Skel}(\mathbf{I}, r)$$

That is, the skeleton of $\mathbf{I}$ is the union of all the sets of centers of maximal disks.

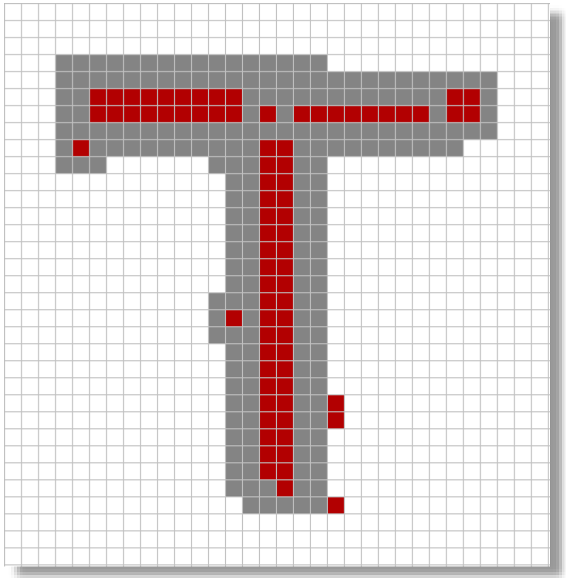
Note that for any actual image $\mathbf{I}$, the union will not be infinite, since $\mathbf{I}$ is bounded (not infinite in extent).

Skeletonization

Original shape

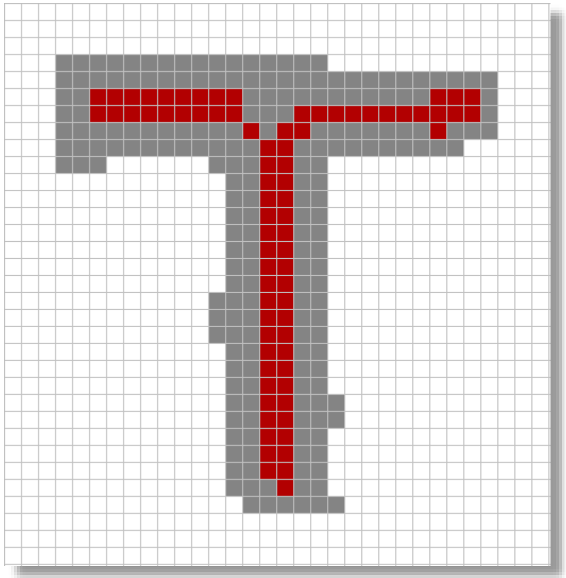


Raw skeleton (red)



is the locus of centers of maximal disks.

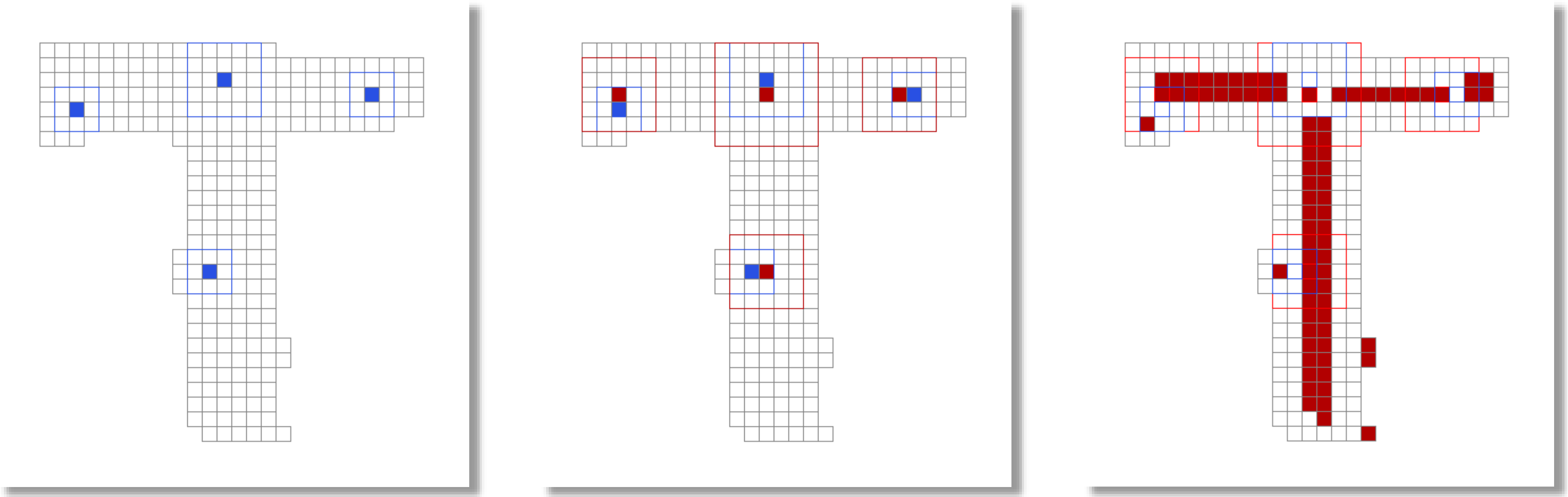
Pruned and connected



skeleton

Skeletonization: Maximal Disks

non maximal “disks” | maximal disks (red) | non max & max disks

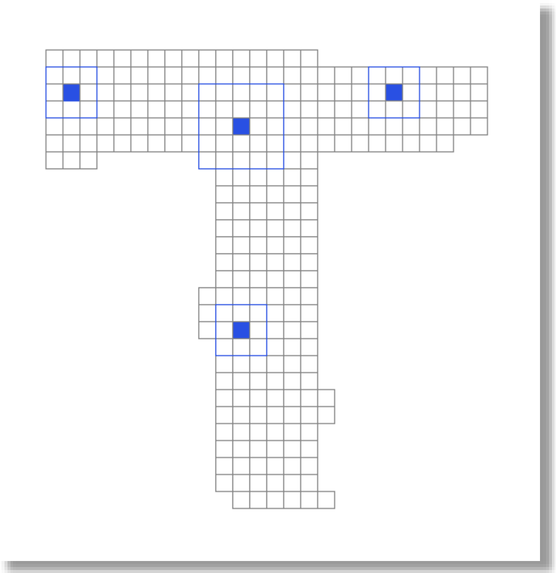


“disks” are squares | non max disks (blue) | over skeleton

The maximal disk at pixel loc **p** is the largest disk in the fg that includes **p**.

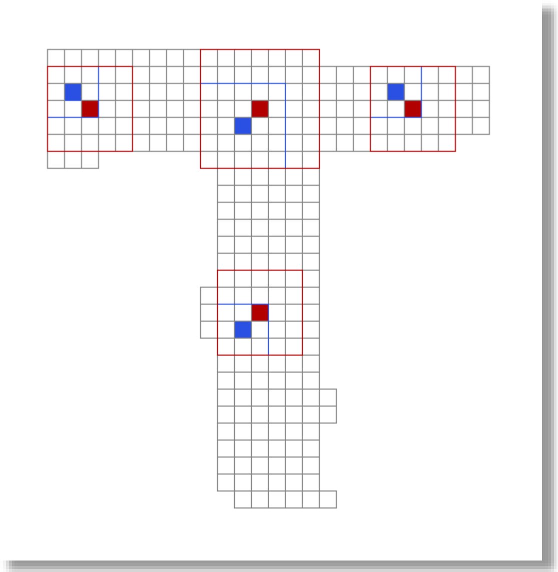
Skeletonization: Maximal Disks

non maximal "disks"



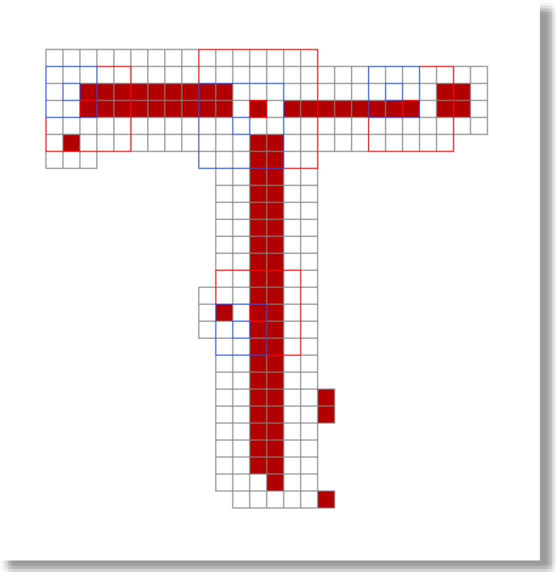
"disks" are squares

maximal disks (red)



non max disks (blue)

non max & max disks

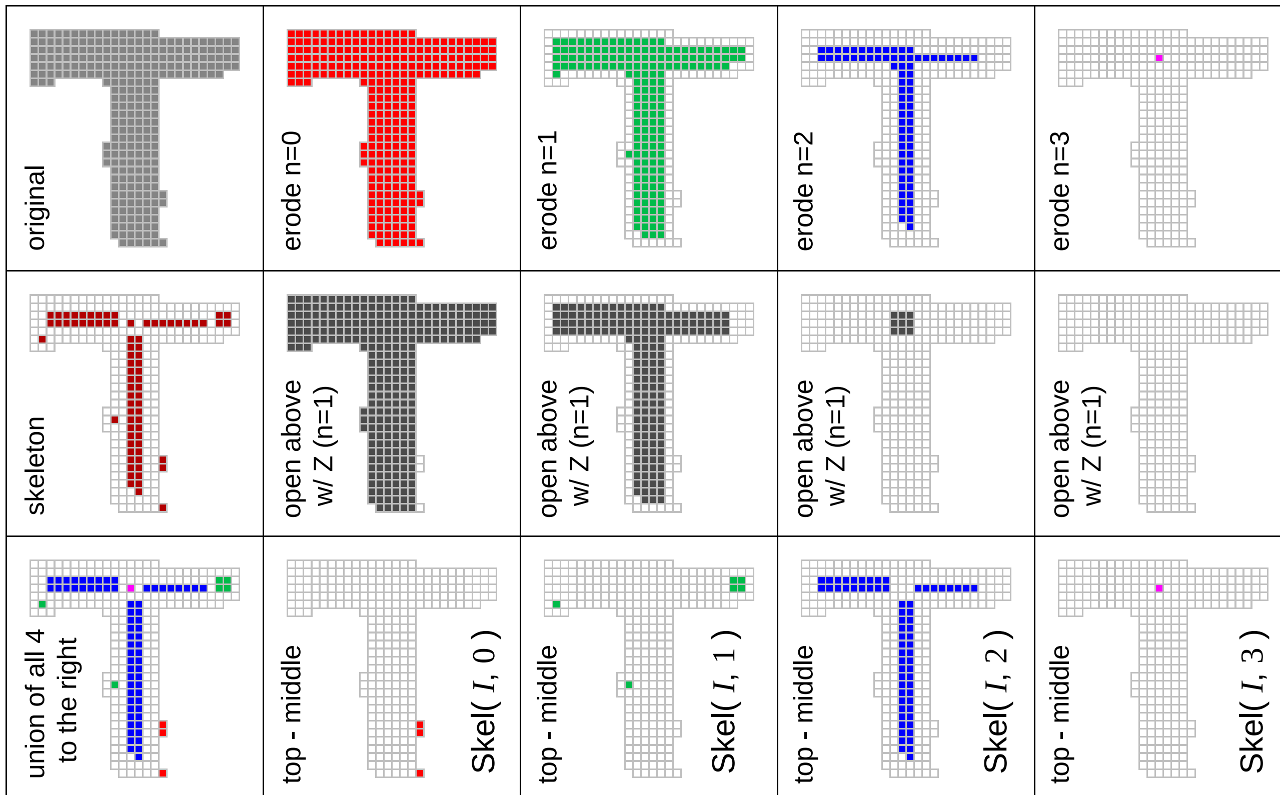


over skeleton

The maximal disk at pixel loc **p** is the largest disk in the fg that includes **p**.

Computation of the Skeleton

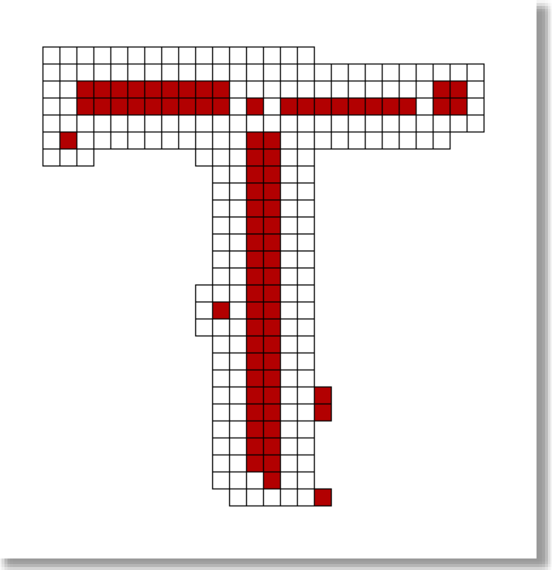
- SE = Z_8 = = Sq(3)
- n = 0: SE = 1 pixel
- n = 1: SE = Sq(3)
- n = 2: SE = Sq(5)
- n = 3: SE = Sq(7)

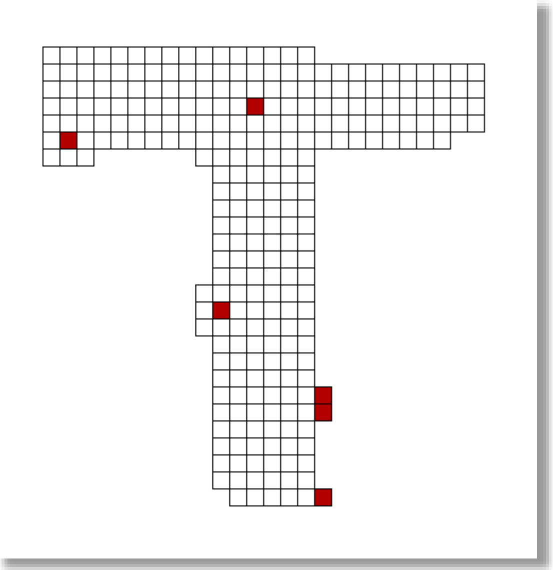


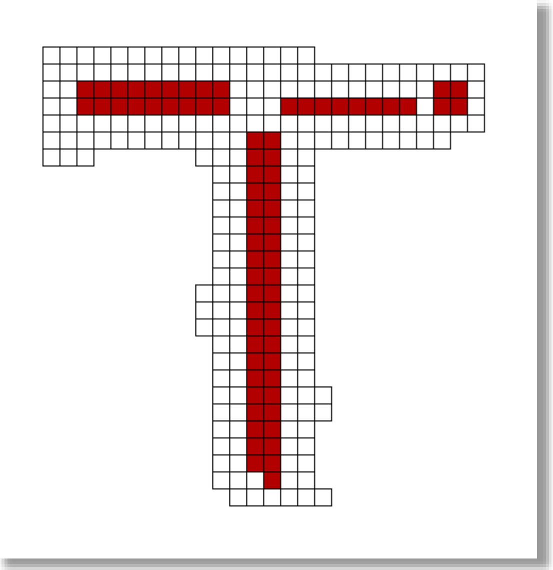
Note that the result is disconnected and has spurious points.

Skeletonization: Delete Spurious Pixels

raw skeleton	def. spurious pixels as	pruned skeleton



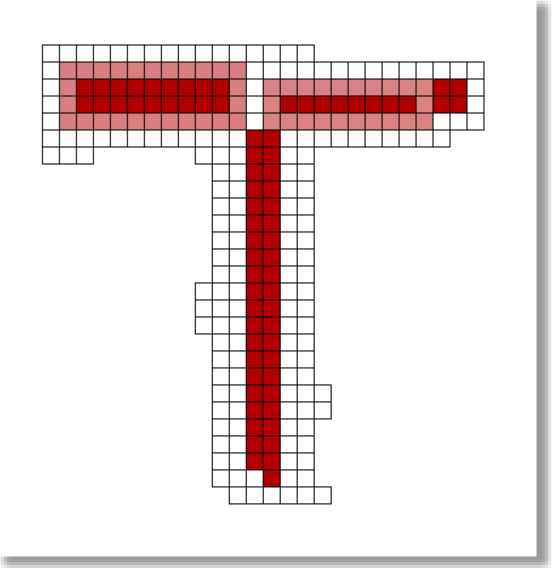




has spurious pixels	conn. comp. of < 3 pix.	raw less spurious

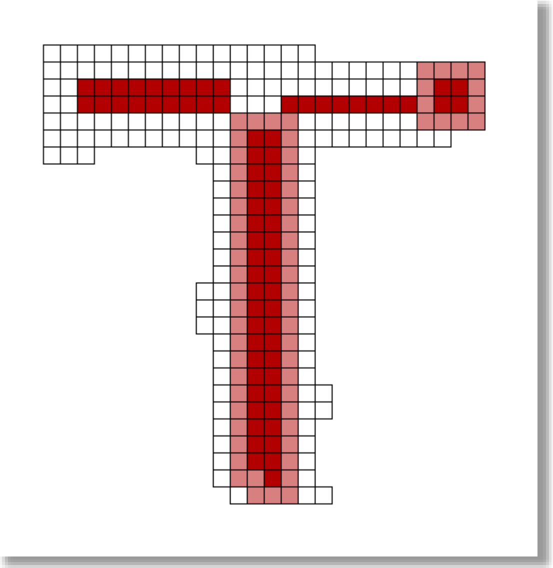
Skeletonization: Reconnect Components

2 components of



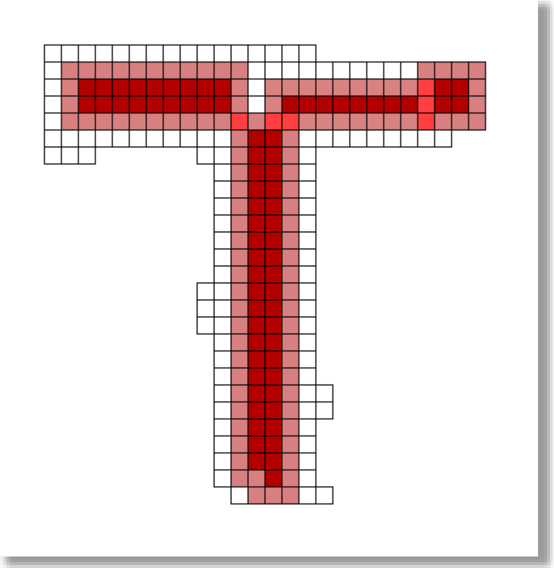
pruned skel. dilated.

2 other components



of pruned skel dilated.

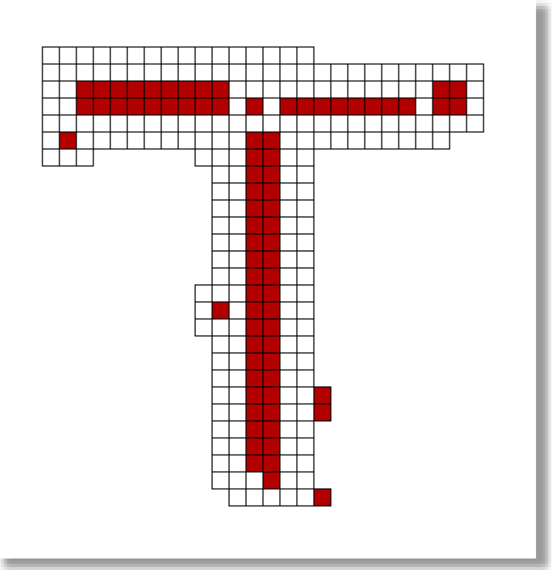
Intersection of



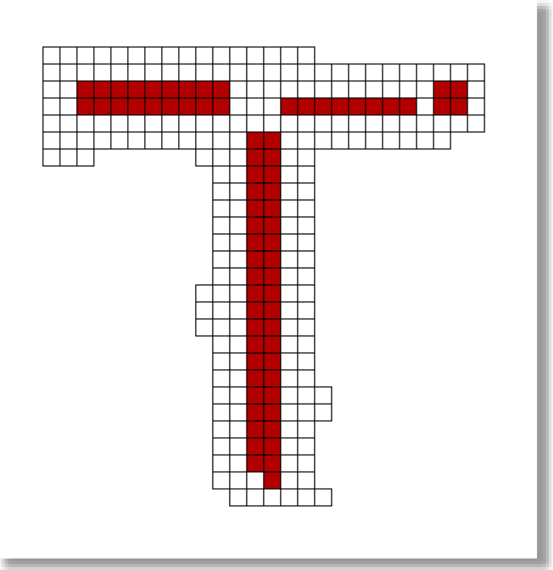
dilated components.

Skeletonization

raw skeleton



pruned skeleton



reconnected skeleton