Chapter 09: Image Morphology

Table of Contents

[Structuring Elements 3](#_Toc138876496)

[Dilation 4](#_Toc138876497)

[Binary Dilation 6](#_Toc138876498)

[Erosion 6](#_Toc138876499)

[Binary Erosion 8](#_Toc138876500)

[Detecting Objects by Size 8](#_Toc138876501)

[Opening and Closing 9](#_Toc138876502)

[Circular Structuring Elements 9](#_Toc138876503)

[Hit or Miss Transform 11](#_Toc138876504)

[Boundary Extraction 13](#_Toc138876505)

[Region Filling 13](#_Toc138876506)

[Extraction of Connected Components 14](#_Toc138876507)

[Recursive Dilation 14](#_Toc138876508)

[Recursive Erosion 14](#_Toc138876509)

[Gray-Scale Images 14](#_Toc138876510)

[Erosion and Dilation 14](#_Toc138876511)

[Opening and Closing 15](#_Toc138876512)

[Morphological Smoothing 16](#_Toc138876513)

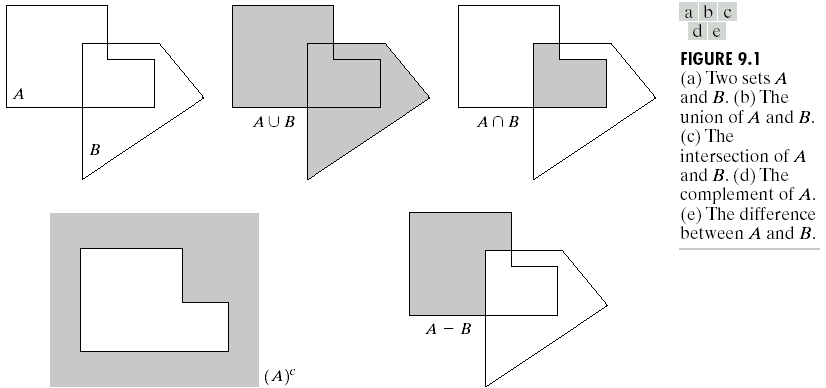
[Morphological Gradient 16](#_Toc138876514)

[Top-Hat and Bottom-Hat Transform 17](#_Toc138876515)

[Granulometry 18](#_Toc138876516)

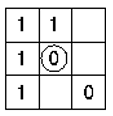
[Textural Segmentation 18](#_Toc138876517)

**Image Morphology** involves using set theory to manipulate specific structures in images. For example, given two structures, we can perform operations like union, intersection, etc.



## Structuring Elements

A **Structuring Element** (SE) is just a kernel, even though we don’t use that term to avoid confusion. It has a central position and based on the overlap between the SE and the structure it is interacting with, along with the values of the SE, we can perform different morphological operations.



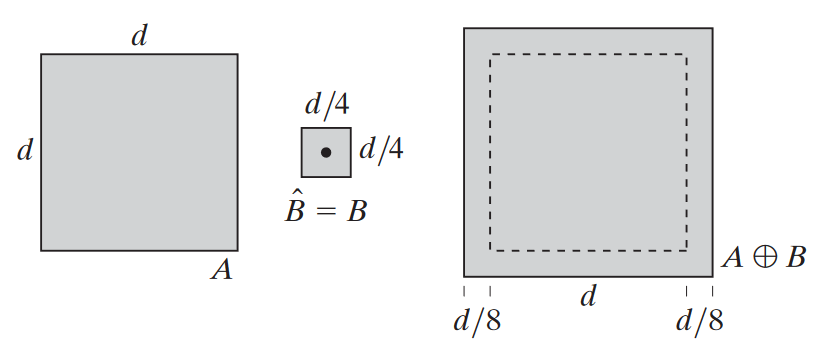
In the above SE, the empty cells denote positions that are **don’t cares**. This allows us to create SEs of any shape, not just square ones. The circle at the center denotes the centering position of the SE, i.e., which pixel of the structure is being modified by the SE.

## Dilation

The **dilation** of a set by another set is formally defined as

In plain English, any values where the SE overlaps with the structure are valid. Technically speaking, is the reflection of , but for the time being we will consider that they are the same.

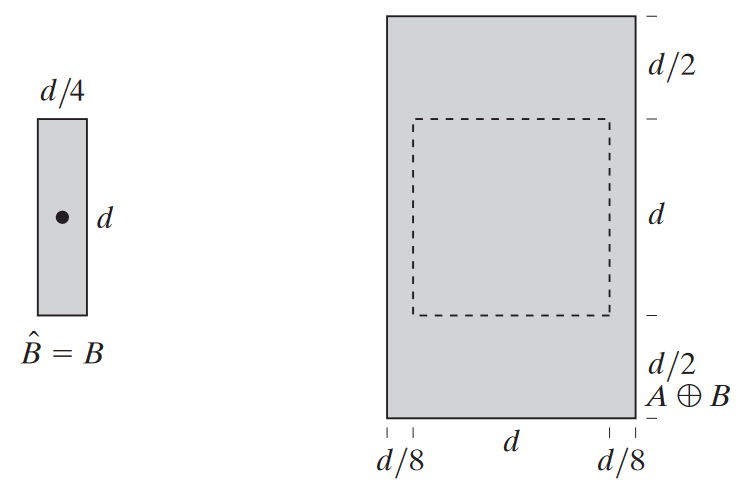
Suppose the SE is a 3x3 matrix. Thus, for there to be any overlap between B and A, the central position of B must be at least at the corner of A. Moving B in this manner creates a boundary around A of 1px.



Since the size of increases, the operation is called a **dilation**.

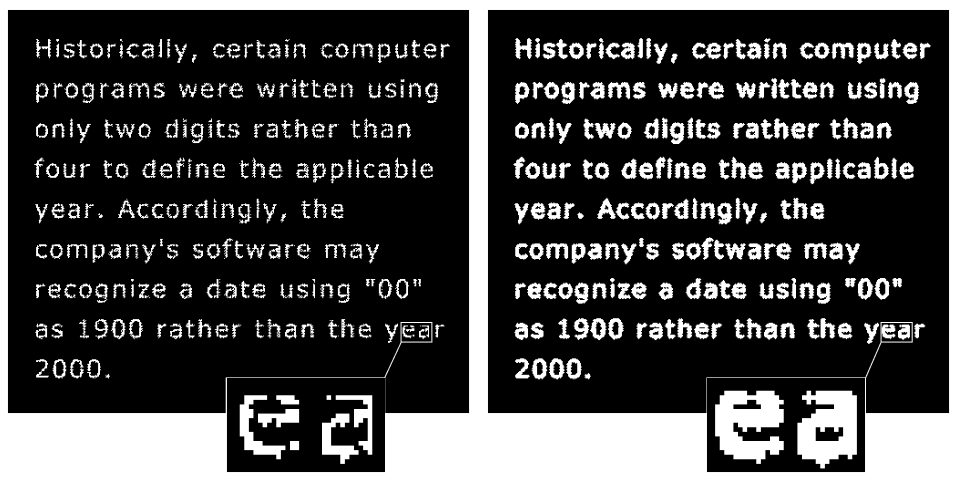
If was a 5x5 matrix, then there would be 2 pixels around the boundary of via the same process. This is the equivalent of performing a dilation with a 3x3 matrix two times.

can be of any shape, and depending on its shape, the dilation of will vary. For example, if we have a very tall SE, will be dilated more in the vertical direction (since will be able to overlap sooner in that direction).



If is a single pixel, then that pixel will grow into the exact shape of .

The dilation operation can be useful in situations such as OCR text recognition. Text in black and white might be broken in places and disconnected during the scanning process. These gaps can be connected by dilating the text. A gap of at most 2px can be fixed using a 3x3 matrix.



### Binary Dilation

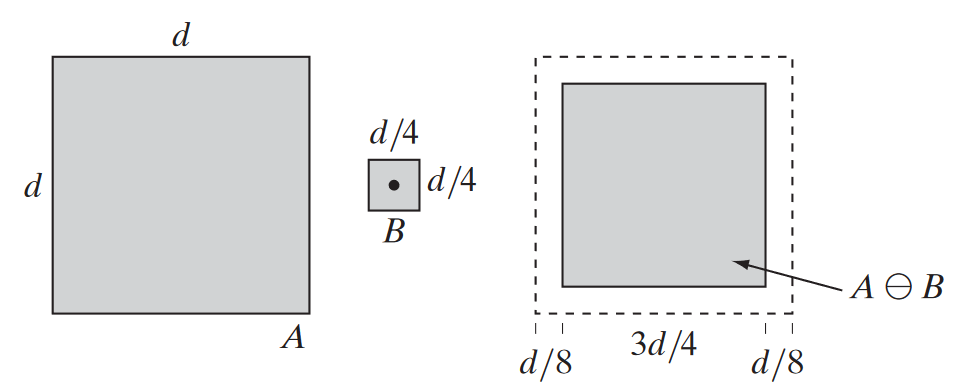
**Binary Dilation**, also called Minkowski addition, for an image by an SE , is defined as:

## Erosion

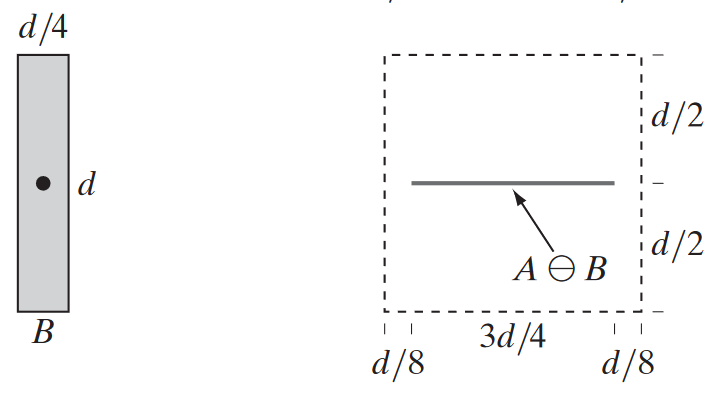
The **Erosion** of a set by another set is formally defined as:

This can also be written as:

This process is practically exactly the same as dilation except that instead of any values where any part of is inside , we need to ensure that the entirety of is inside . If we ensure this and move 1px at a time, assuming is a 3x3 matrix, we will end up shrinking by 1px in every direction.



If was a 5x5 matrix, the shrinking would by of 2px on every side. If was an odd shape, the direction along which is longer will shrink more.



If we have a structure that is smaller then , the structure will be eliminated entirely.

### Binary Erosion

The **binary erosion**, also called the Minkowski Subtraction, of an image by an SE is defined as:

## Detecting Objects by Size

Suppose we have an image with squares of several sizes and we want to take out the squares specifically of the size 15x15. To do this, we first apply erosion using a 15x15 SE. This removes all squares that are smaller and makes the squares of 15x15 size exactly of 1px. If we dilate this again using a 15x15 SE, then we get back the 15x15 squares.

However, what if we have larger squares as well, such as 17x17 squares. These will not be removed by the 15x15 SE erosion. Instead, we can apply erosion with a 17x17 SE (thus removing the 15x15 squares as well), dilate it back using a 17x17 SE (thus getting back just the 17x17 squares) and subtract this from the previous image (thus removing the 17x17 squares).

## Opening and Closing

Basically, dilation expands an image and erosion shrinks it.

In **opening**, we first erode an object with the SE and then dilate it again with the SE . This smooths contours, breaks isthmuses and eliminates protrusions.

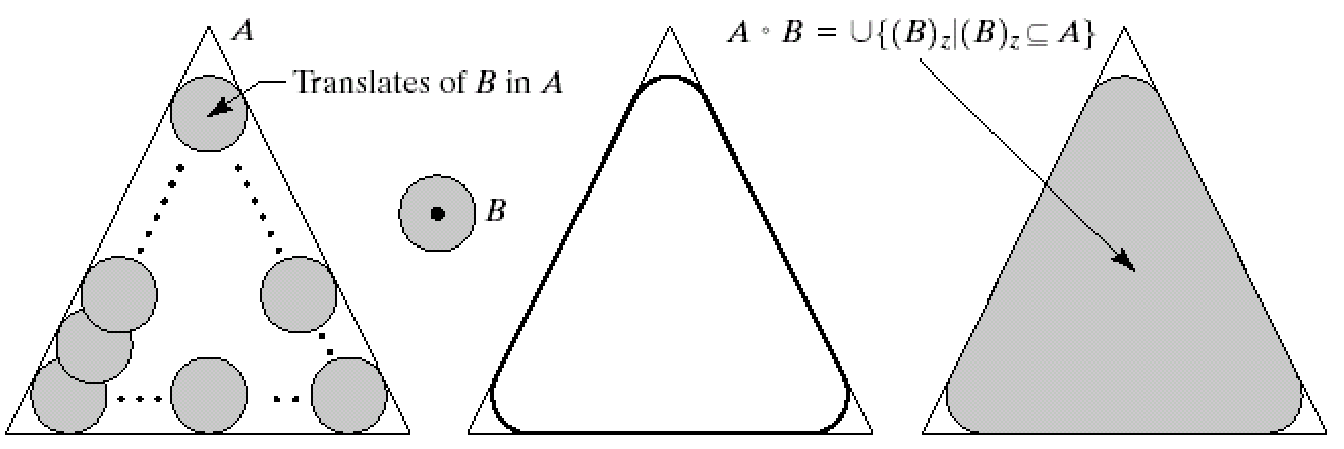
**Binary opening** is defined as

In **closing**, we first dilate an object with the SE and then erode it with the SE . This fuses breaks, holes and gaps.

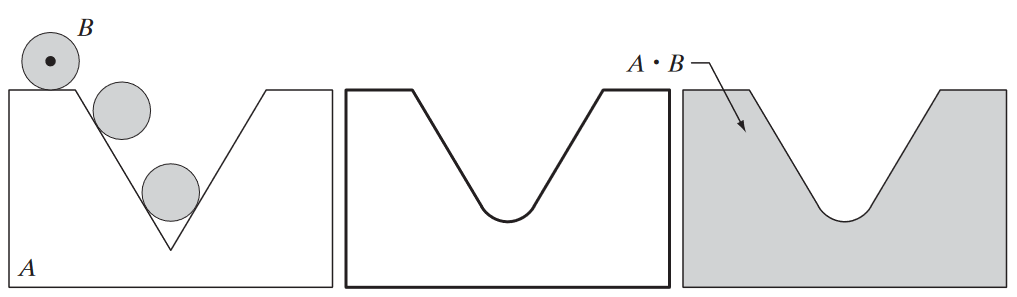
**Binary closing** is defined as

## Circular Structuring Elements

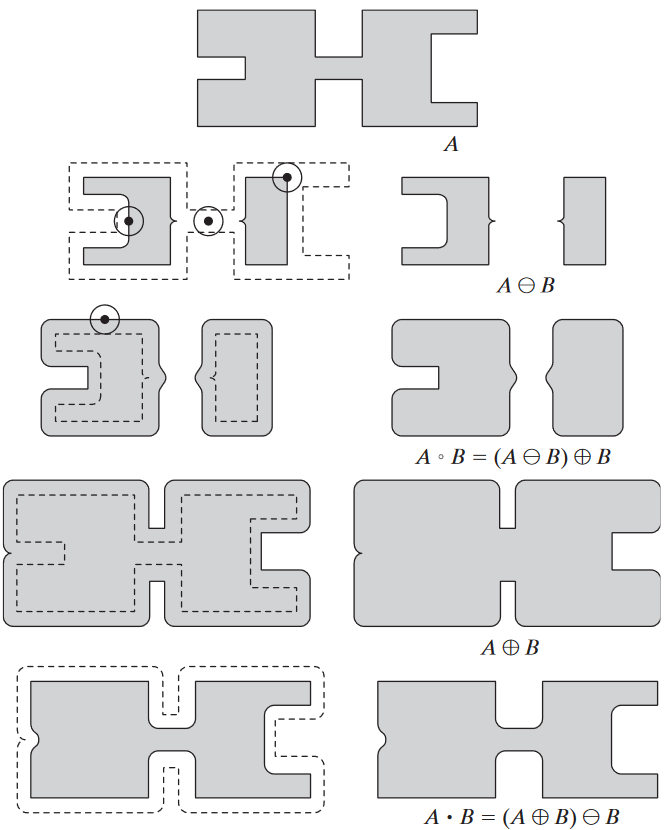
A **circular structuring element** along with the opening or closing operation can be used to achieving a **rounded effect**. Applying the opening operation with a circular SE is comparable to rolling a ball along the inner edges of the shape. This results in the **external corners**, the ones protruding out of the shape, becoming rounded.



Applying closing with a circular SE on the other hand, is the equivalent of rolling a ball on the outer edges of the shape. This results in the **internal corners**, those going into the shape, becoming rounded.

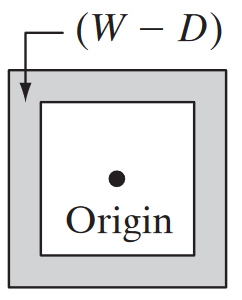


The rounding effects are happening on top of the existing effects of the opening and closing operations. Opening still causes isthmuses to break and closing still causes gaps to be filled. The combined effects can be seen in the images below:

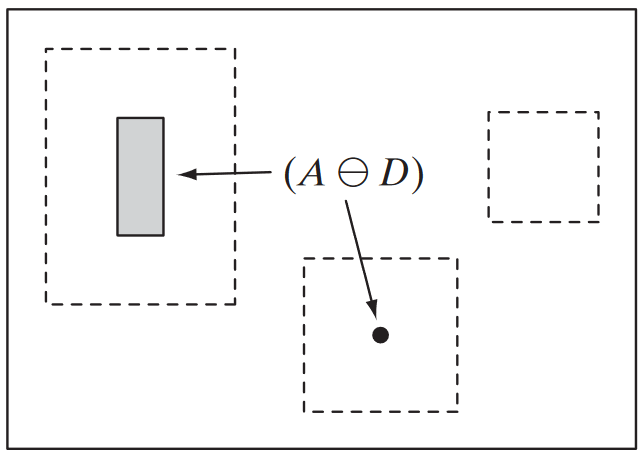


## Hit or Miss Transform

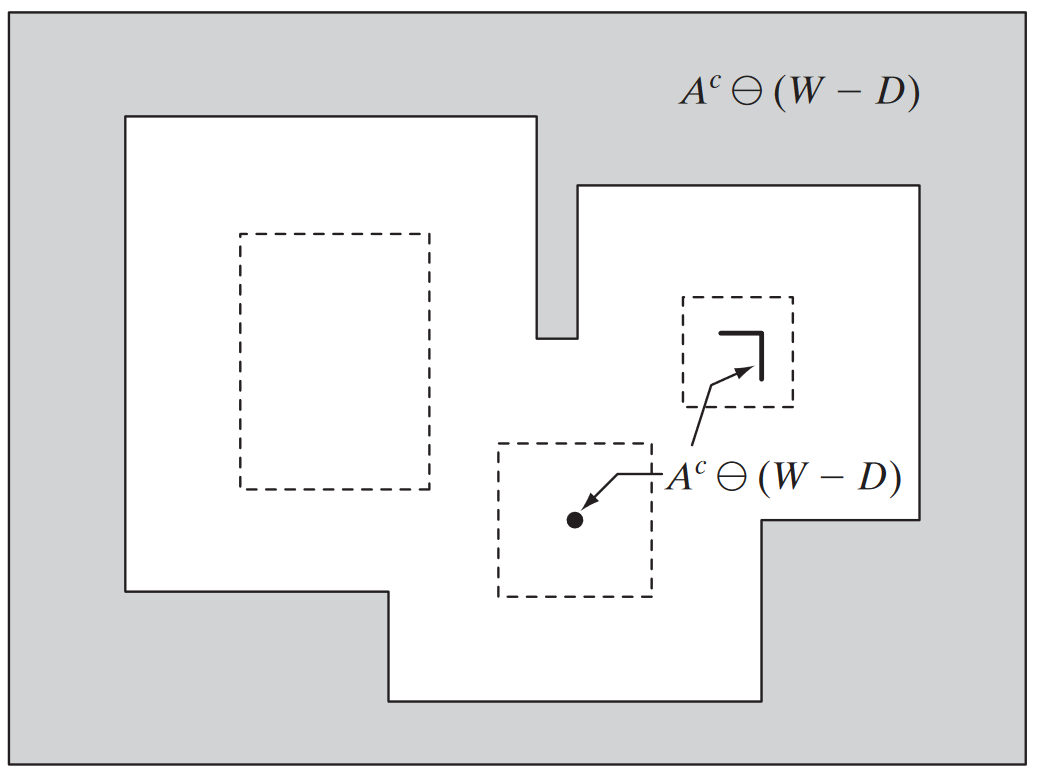
**Hit or Miss Transform** is used for shape detection. Suppose we have a shape that we want to detect inside an object. To do this, we require a combination of two SEs, and . is the exact same shape as and is filled with 1s, while is the shape of filled with 0s with a border of size filled with 1s.



First, we apply erosion using . This creates a single point at the center of .



Next, we take the complement of the image and erode it with .



The intersection of these two images leaves just the point of . We can dilate this back to original shape using .

Formally, this is defined as

or

However, if the object is a combination of 1s and 0s, we can use just .

## Boundary Extraction

Boundary extraction is formally defined as

Thus, we first erode the object with the SE and then subtract the eroded object from the original one. Since erosion shrinks the object, the difference should be at the boundaries.

## Region Filling

Suppose we have a structure with a hole in between that we want to fill. To do this, we pick any point in the hole and apply a dilation operation on it using an SE. This will give us some filling, but it is most likely be of the wrong shape (i.e., not just the hole). To make it the correct shape, we take the intersection of the filling and , which has 1s in the position of the hole. We keep repeating this process until we find that there is no further change, meaning the shape of the hole has been found. Taking the union of this and gives us the image with the hole filled.

## Extraction of Connected Components

Extracting connected components is exactly the same as region filling, except we start with any point on , apply dilation using an SE and take the intersection between the result and itself. In this case, we only have 1s in the shape of , which means we can discover the shape gradually.

## Recursive Dilation

Repeatedly applying dilation on a shape will cause it to expand. The result of applying dilation repeatedly with a small SE is the same as applying dilation once with a large SE.

## Recursive Erosion

Repeatedly applying erosion on a shape will cause it to eventually disappear. The result of applying dilation repeatedly with a small SE is the same as applying erosion once with a large SE.

## Gray-Scale Images

### Erosion and Dilation

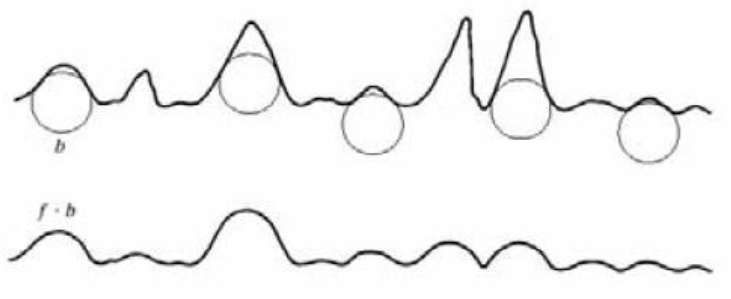
The **dilation** operation on a gray-scale image is the equivalent of applying a **max-filter**. Brighter regions are expanded, while darker ones are reduced or eliminated entirely. **Erosion** on the other hand, is like applying a **min-filter**. Darker regions are expanded while bright details are reduced or eliminated entirely.

### Opening and Closing

Imagine that we have a line of different intensity values.

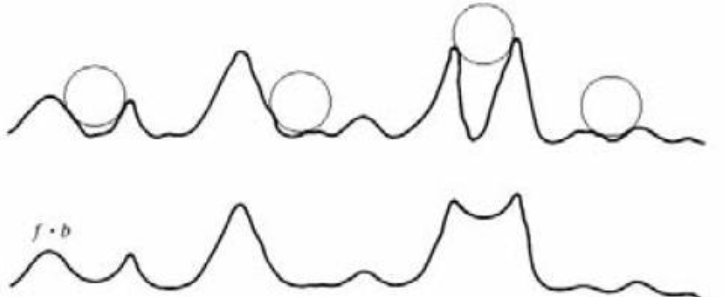


Applying **opening** on this is like rolling a ball beneath the curve.



Thus, sudden sharp increases in intensity are lowered, meaning small light details are removed without changing the overall brightness. The initial **erosion** is what removes the small light details, but this also darkens the image. The following **dilation** is what increases the intensities again.

Applying **closing** on the original line is like rolling a ball above the curve.



Thus, sudden drops in intensity are increased, meaning small dark details are removed without changing the overall brightness. The initial **dilation** removes the dark details, but it also increases the overall brightness. The following **erosion** darkens the image again.

### Morphological Smoothing

The **smoothing** operation involves applying **opening** followed by **closing**. The opening operation removes small white regions, and the closing operation removes small dark regions.

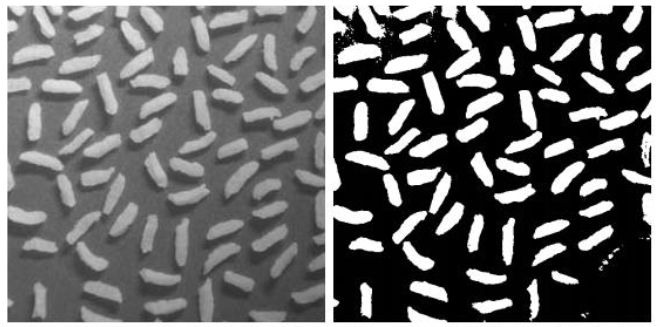
### Morphological Gradient

To obtain the gradient, we first apply **dilation**. This thickens the white regions. Next, we apply **erosion**, which shrinks the white regions. The difference between these two images will give us a resulting image that emphasizes the boundaries. Formally,

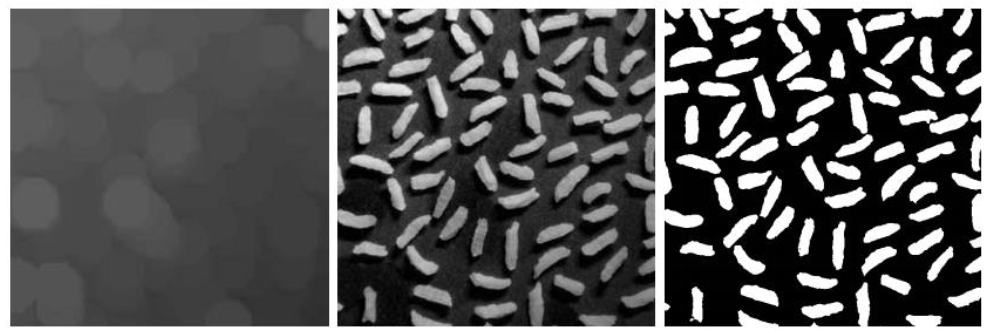
## Top-Hat and Bottom-Hat Transform

**Top-Hat Transform** is used to correct non-uniform background illumination for light objects on a dark background. We apply an opening operation and subtract the result from the original image.

If we apply thresholding on an image with a non-unform background, we will eliminate some of the foreground objects.



Instead, we can apply an opening operation. This will eliminate the bright objects but retain the background. This background has the non-uniform illumination. Subtracting this from the original image will make the background uniform. Thresholding will then work.



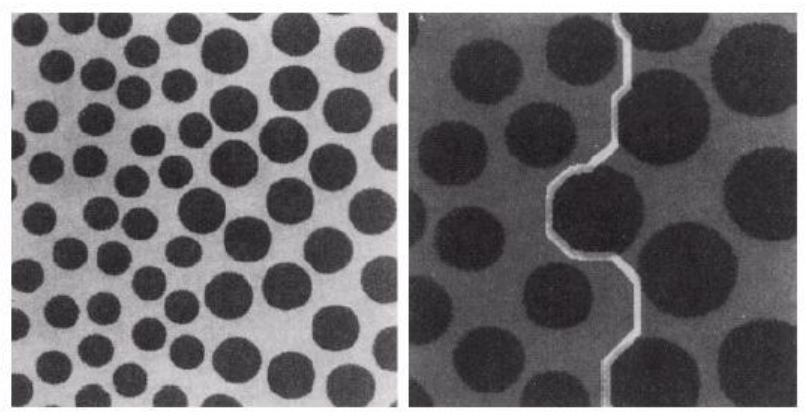
**Bottom-Hat Transform** is the exact same process but for light objects on a non-uniform white background. In this case, we need to subtract the original image from the closing operation’s result to obtain the same effect.

## Granulometry

**Granulometry** is used to detect the sizes of shapes in an image. We apply opening operations with SEs of increasing size and observe the differences in intensity values. At the points where the operations causes some shapes to disappear (meaning the size of the SE corresponds to the size of the shapes), there will be a significant change in intensity.

## Textural Segmentation

**Textural Segmentation** involves separating regions containing shapes of different sizes. Suppose we have an image with small circles on one side and large circles on the other size, and we want to find the boundary between the two sides.



To do this, we can first apply a **closing operation** with an SE that is larger than the smaller circles. This will remove that entire area giving us a **white region**. Next, we can apply the **opening operation** with an SE larger than the largest gap between the larger circles. This will make the other side **completely dark**. Finally, we can apply **morphological gradient** to find the boundary.

