

Noise Removal with Morphological Operations Opening and Closing Using Erosion and Dilation

ALKA BISHNOI¹

¹ Mtech. Ece, Suraj College Of Engg. & Technology, Distt. Mahendergarh, University-Mdu Rohtak, State-Haryana.

Abstract: The mathematical operations are proposed in this paper. By using two mathematical operations erosion and dilation we can add and remove pixels. We can remove the noise or interference in power system. Opening and closing operations also discussed with erosion and dilation. These four morphological operations are also helpful in developing a morphological filter.

I. Introduction

Erosion and dilation

Two fundamental mathematical morphological operations are: dilation and erosion. In fact, many of the morphological algorithms are based on these two primitive operations. In gray scale images, we develop algorithms for boundary extraction via a morphological gradient operation, and for region partitioning based on texture content. We operation are also useful in smoothing and sharpening, which often are useful as per- or post processing steps.

The morphological operators may be classified as

- (1) Binary morphological operators
- (2) Grayscale morphological operators

Binary morphological operations deal with set of points of the domain D and the gray scale morphological operators deal with functions.

Opening and Closing

The combination of erosion followed by dilation is called an opening, referring to the ability of this combination to open up gaps between just-touching features. It is one of the most commonly used sequences for removing pixel noise from binary images. Performing the same operations in the opposite order (dilation followed by erosion) produces a different result. This sequence is called a closing because it can close breaks in features. There are several parameters that can be used to adjust erosion and dilation operations, particularly the neighbor pattern or rules for adding or removing pixels and the number of iterations, as discussed below. In most opening or closing operations, these are kept the same for both the erosion and the dilation.

1.1 Erosion

Erosion removes pixels from features in an image or, equivalently, turns pixels OFF that were originally ON. The purpose is to remove pixels that should not be there. The simplest example is pixels that have been selected by thresholding because they fell into the brightness range of interest, but that do not lie within large regions with that brightness. Instead, they may have that brightness value either accidentally, because of finite noise in the image, or because they happen to straddle a boundary between a lighter and darker region and thus have an averaged brightness that happens to lie within the range selected by thresholding. These pixels cannot be distinguished by simple thresholding because their brightness value is the same as that of the desired regions. It may be possible to remove them by using Boolean logic, for instance using the gray level as one criterion and the gradient as a second one, and requiring that the pixels to be kept have the desired gray level and a low gradient.

The simplest kind of erosion is to remove (set to OFF) any pixel touching another pixel that is part of the background (is already OFF). This removes a layer of pixels from around the periphery of all features and regions, which will cause some shrinking of dimensions and may create other problems if it causes a feature to break up into parts. We will deal with these difficulties below. Erosion can entirely remove extraneous pixels representing point noise or line defects (e.g., scratches) because these defects are frequently only 1 or 2 pixels wide.

1.2 Dilation

Instead of removing pixels from features, a complementary operation known as dilation (or sometimes dilatation) can be used to add pixels. The classical dilation rule, analogous to that for erosion, is to add (set to ON) any background pixel that touches another pixel that is already part of a foreground region (is already ON). This will add a layer of pixels around the periphery of all features and regions, which will cause some increase in dimensions and may cause features to merge. It also fills in small holes within features.

Because erosion and dilation cause a reduction or increase in the size of regions, respectively, they are sometimes known as etching and plating or shrinking and growing. There are a variety of rules for deciding which pixels to add or remove and for forming combinations of erosion and dilation.

In the rather simple example described above and illustrated in Figure, erosion to remove the extraneous lines of pixels between light and dark phases causes a shrinking of the features. Following the erosion with dilation will more or less restore the pixels around the feature periphery, so that the dimensions are (approximately) restored. However, isolated pixels and lines that have been completely removed do not cause any new pixels to be added. They have been permanently erased from the image.

II. Mathematical Representation of Erosion and Dilation

Let $f(x)$ denote a one-dimensional original signal whose definition domain is $\{0,1,\dots,N\}$. Let $g(x)$ denote structuring element whose definition domain is $\{0,1,\dots,M\}$. It is further assumed that $M < N$ and the origin of $g(x)$ is defined in 0. The formula of dilation operation is:

$$(f \oplus g)(n) = \max_{\substack{m=0,1,\dots,M-1 \\ n=0,1,\dots,N+M-2}} \{f(n-m) + g(m)\}$$

The formula of erosion operation is:

$$(f \ominus g)(n) = \min_{\substack{m=0,1,\dots,M-1 \\ n=0,1,\dots,N-M}} \{f(n+m) - g(m)\}$$

The calculations of dilation and erosion are very simple, which contain only addition and subtraction, not involving multiplication and division. So the processing speed corresponding to real-time signal is fast and the delay is small. Compared with traditional digital filtering algorithm, their calculations are greatly simple.

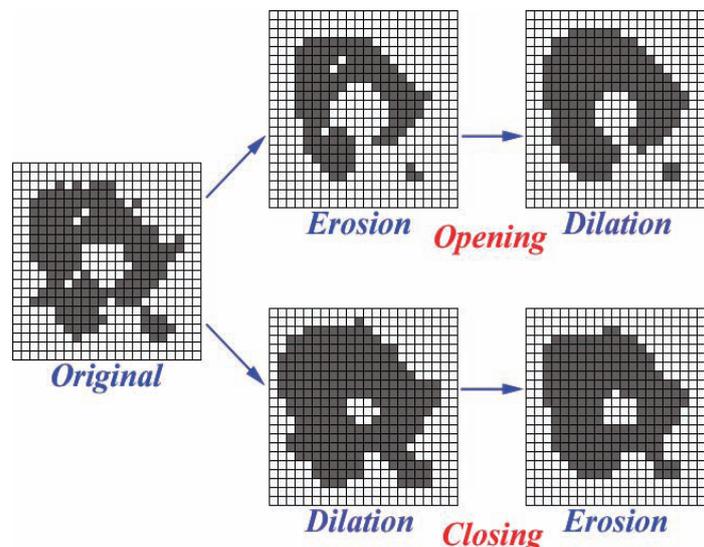


Fig 1.1 Combining erosion and dilation to produce an opening or a closing.

The result is different depending on the order of application of the two operations. Since the original image is ambiguous, it is necessary to use a priori knowledge to select the proper combination.

2.1 Opening

Openings can be used in some cases to separate touching features. In the example shown in Figure 2.1, the features are all similar in size. This fact makes it possible to continue the erosion until all features have separated but none have been completely erased. After the separation is complete, dilation grows the features back toward their original size. They would merge again unless logic is used to prevent it. A rule that prevents turning a pixel ON if its neighbors belong to different features maintains the separation shown in the figure. This requires performing feature identification for the pixels, so the logic discussed above is required at each step of the dilation. An additional rule prevents turning on any pixel that was not on in the original image, so that the features are restricted to their original sizes. If the features had different original sizes, the separation lines would not be positioned correctly at the junctions, and some features might disappear completely before others separated.

The formula for Opening operation is,

$$(f \circ g)(x) = ((f \ominus g) \oplus g)(x)$$

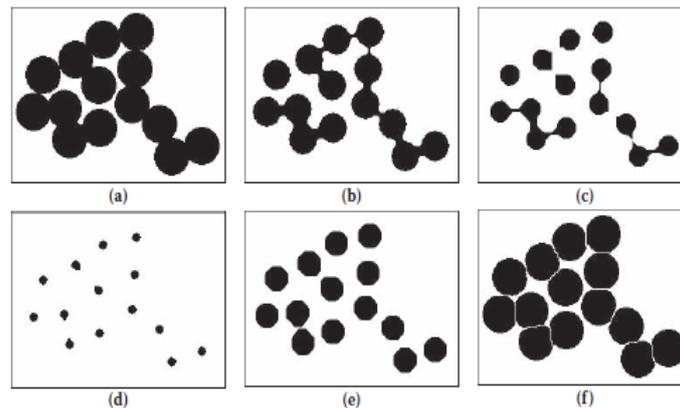


Fig 2.1 Separation of touching features by erosion/dilation: (a) original test image; (b) after two cycles of erosion; (c) after four cycles; (d) after seven cycles (features are now all fully separated); (e) four cycles of dilation applied to image d (features will merge on next cycle); (f) additional cycles of non-merging dilation restricted to the original pixel locations, which restores the feature boundaries.

2.2 Closing

As shown in Figure 1.1, the closing sequence is performed in the other order, a dilation followed by erosion, and the result is not the same. Instead of removing isolated pixels that are ON, the result is to fill in places where isolated pixels are OFF, missing pixels within features, or narrow gaps between portions of a feature. Figure 2.2 shows an example of a closing used to connect the parts of the cracked fibers shown in cross section. The cracks are all narrow, so dilation causes the pixels from either side to spread across the gap. The increase in fiber diameter is then corrected by erosion, but the cracks do not reappear.

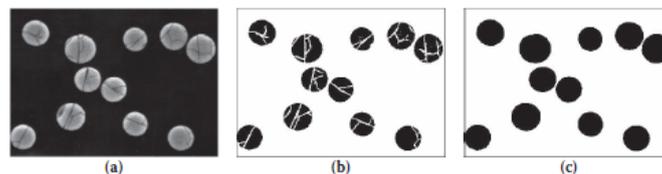


Fig 2.2 Joining parts of features with a closing: (a) original image, cross section of cracked glass fibers; (b) brightness thresholding, showing divisions within the fibers; (c) after application of a closing.

The classical erosion and dilation operations illustrated above turn a pixel ON or OFF if it touches any pixel in the opposite state. Usually, touching in this context includes any of the adjacent 8 pixels, although some systems deal only with the 4 edge-sharing neighbors. These operations would be much simpler and more isotropic on a hexagonal pixel array, because the pixel neighbor distances are all the same, but practical considerations lead to the general use of a grid of square pixels.

The formula for Closing operation is ,

$$(f \bullet g)(x) = ((f \oplus g) \ominus g)(x)$$

III. Conclusion

By using morphological operations erosion and dilation in opening and closing operations are very helpful in filters. They can add and remove pixels. Morphological filters developed on the basis of combination of closing and opening operations are discussed. This is a future aspect.

REFERENCES

- [1.] Rafael C. Gonzalez and Richard E. Woods “Digital Image Processing” , Third Edition
- [2.] WANG Jing, LIU Pan, LIU Di-chen “The design and analysis of improved adaptive generalized morphological filter(2008),(School of Electrical Engineering of Wuhan University, Wuhan 430072, China)
- [3.] Rafael C. Gonzalez and Richard E. Woods “Digital Image Processing using Matlab” , Third Edition
- [4.] MATLAB 7.0.4 (help)
- [5.] John C.Russ “The image Processing Handbook” Fifth edition
- [6.] Image Processing Toolbox For Use with MATLAB User’s Guide ,Version 2 ,The mathswok.inc
- [7.] <http://www.mathworks.com/access/helpdesk/help/toolbox/images/index.html>