Neighbourhoud operations in image processing and their functionalities

Sina Rismanchian
B.Sc student
Department of Computer Engineering
Sharif University of Technology
Email: rismanchians@ce.sharif.edu

Sobhan Mohammadpour
B.Sc student
Department of Computer Engineering
Sharif University of Technology
Email: jmmccart@uci.edu

ABSTRACT

This article tries to introduce some of neighbourhood operations in image processing and their functionalities. Neighbourhood operations, called filters, are techniques for modifying or enhancing an image. Image processing filters have many types and everyone has it's specific functionality, advantages and disadvantages that we will discuss them in this article.

1 Introduction

Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. It is a type of signal processing in which input is an image and output may be image or characteristics/features associated with that image. Nowadays, image processing is among rapidly growing technologies. It forms core research area within engineering and computer science disciplines too.

Image processing basically includes the following three steps:

Importing the image via image acquisition tools; Analysing and manipulating the image; Output in which result can be altered image or report that is based on image analysis.

There are two types of methods used for image processing namely, analogue and digital image processing. Analogue image processing can be used for the hard copies like printouts and photographs. Image analysts use various fundamentals of interpretation while using these visual techniques. Digital image processing techniques help in manipulation of the digital images by using computers. The three general phases that all types of data have to undergo while using digital technique are pre-processing, enhancement, and display, information extraction [14].

2 Terminology

Keywords and their meanings [1] [2]:

2.1 Image Processing

The field of computer science that develops techniques for enhancing digital images to make them more enjoyable to look at, and easier to analyze by computers as well as humans.

2.2 Morphological Operation

A category of image-processing techniques that operate on the structure of the objects in an image.

2.3 Convolution

A method of calculating the new value of a central pixel in a neighborhood by multiplying each pixel in the neighborhood by a corresponding weight; the new value of the central pixel is the sum of the multiplications.

2.4 Noise

Unwanted changes to the values of the pixels in an image, often introduced by the imaging device during capture. Examples include impulse noise and Gaussian noise.

2.5 Impulse Noise

Also called salt and pepper noise, impulse noise introduces very light (salt) and very dark (pepper) pixels that stand out from their neighbors.

2.6 Gaussian Noise

A form of image noise that adds small positive and negative deviations to the pixels in an image, often caused by the random variations between the elements of a CCD sensor. Plotting the number of occurrences of each deviation on a histogram produces the bell-shaped curve of the normal distribution, which is also called the Gaussian distribution.

2.7 Corrupted Pixel

A pixel value altered by noise.

2.8 Kernel

A rectangular grid of convolution weights.

2.9 Greyscale Image

An image composed of pixels that present shades of grey.

2.10 Photo Restoration

The application of a series of image-processing routines to enhance a damaged photograph.

2.11 Edge

Edges mark the boundaries between the objects in a scene. A large change in pixel brightness over a small number of pixels often indicates the presence of an edge.

2.12 Image smoothing

Image smoothing refers to any image-to-image transformation designed to smoothen or flatten an image by reducing the rapid pixel-to-pixel variation in greylevels.

2.13 Linear filters

Linear filters are also know as convolution filters as they can be represented using a matrix multiplication.

2.14 Nonlinear filters

Thresholding and image equalisation are examples of nonlinear operations, as is the median filter (Discussed later at this article).

2.15 Ramp Edge

A region of pixels that separates a region of light pixels from a region of dark pixels. The pixels in the region change gradually from light to dark.

2.16 Sharpening

An area process that emphasizes the detail in an image.

3 Neighbourhood operations

In many different kinds of digital image processing, the basic operation is as follows: at each pixel in a digital image we place a neighborhood around that point, analyze the values of all the pixels in the neighborhood according to some algorithm, and then replace the original pixel's value with one based on the analysis performed on the pixels in the neighborhood. The neighborhood then moves successively over every pixel in the image, repeating the process [3].

3.1 Median filtering

The median filter is a nonlinear digital filtering technique, often used to remove noise from an image or signal. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise (but see discussion below), also having applications in signal processing.

3.1.1 Simple median filtering

3.1.1.1 Functionality

The simple median filter replaces the center pixel of the window (Eg. 3X3.5X5 etc.) considered, by median of the window. If the center pixel is either 0 (Pepper) or 255 (Salt), it is replaced by median of the window which will be other than 0 or 255. The major drawback of standard median filter is that even if the pixel under consideration is uncorrupted (other than 0 or 255), it is replaced by the median of the window. This will deteriorate the overall visual quality of the image. In addition, the simple median filter fails to preserve the edges. This class of filter belongs to the class of edge preserving smoothing filters which are non-linear filters. This means that for two images A(x) and B(x):

$$median[A(x)+B(x)] \neq median[A(x)] + median[B(x)]$$

It works as follows. Window = 23, 32, 41, 255, 45, 52, 23, 32, 41. The window sorted in ascending order = 23, 23, 32, 32, 41, 41, 45, 52, 255. Median is the mid value after sorting i.e., 41. Hence the uncorrupted pixel is replaced by median of the window [4].

$$\begin{array}{ccc} \text{Sample window} & & \text{Output} \\ \begin{bmatrix} 23 & 32 & 41 \\ 255 & (45) & 52 \\ 23 & 32 & 41 \end{bmatrix} & \rightarrow & \begin{bmatrix} 23 & 32 & 41 \\ 255 & (41) & 52 \\ 23 & 32 & 41 \end{bmatrix} \end{array}$$

If the pixel under consideration is corrupted as shown below, the impulse noise will be removed following the same method [4].

$$\begin{array}{ccc} \text{Sample window} & & \text{Output} \\ \begin{bmatrix} 23 & 32 & 41 \\ 255 & (255) & 52 \\ 23 & 32 & 41 \end{bmatrix} & \rightarrow & \begin{bmatrix} 23 & 32 & 41 \\ 255 & (41) & 52 \\ 23 & 32 & 41 \end{bmatrix} \end{array}$$

These filters smooth the data while keeping the small and sharp details. The median is just the middle value of all the values of the pixels in the neighborhood. Note that this is not the same as the average (or mean); instead, the median has half the values in the neighborhood larger and half smaller. The median is a stronger "central indicator" than the average. In particular, the median is hardly affected by a small number of discrepant values among the pixels in the neighborhood. Consequently, median filtering is very effective at removing various kinds of noise. Fig. 1 illustrates an example of simple median filtering (SMF) [3].

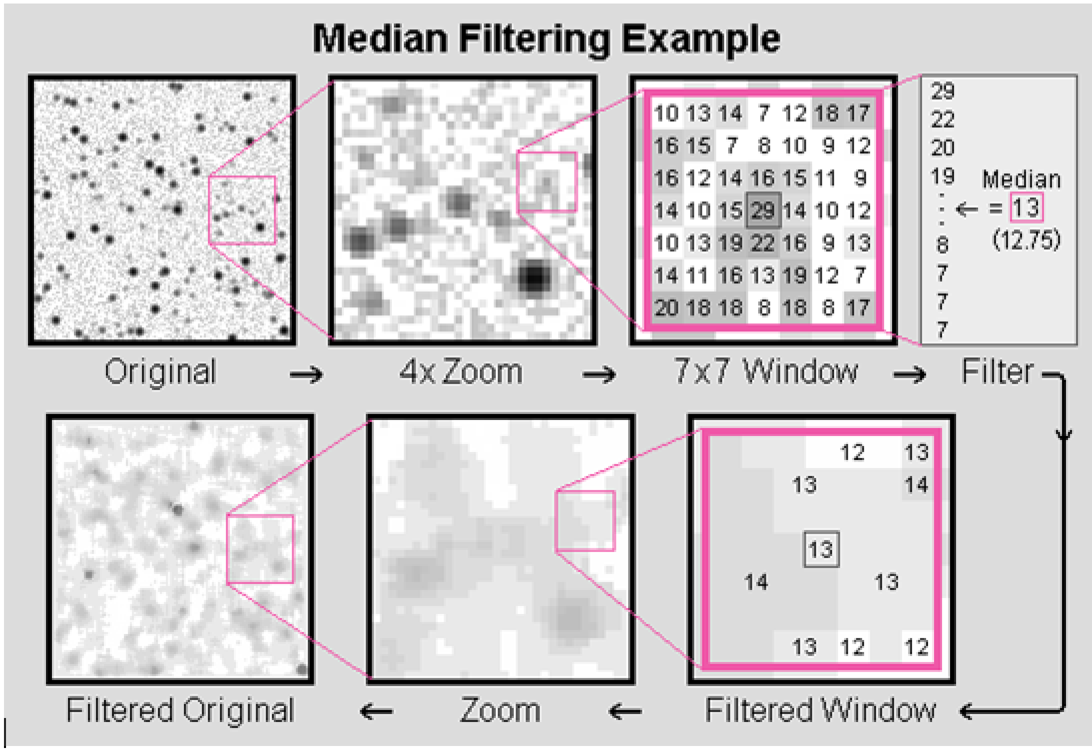


Fig. 1. SMF example

Furthermore, in Fig. 2, On the left is an image containing a significant amount of salt and pepper noise. On the right is the same image after processing with a simple median filter [2].

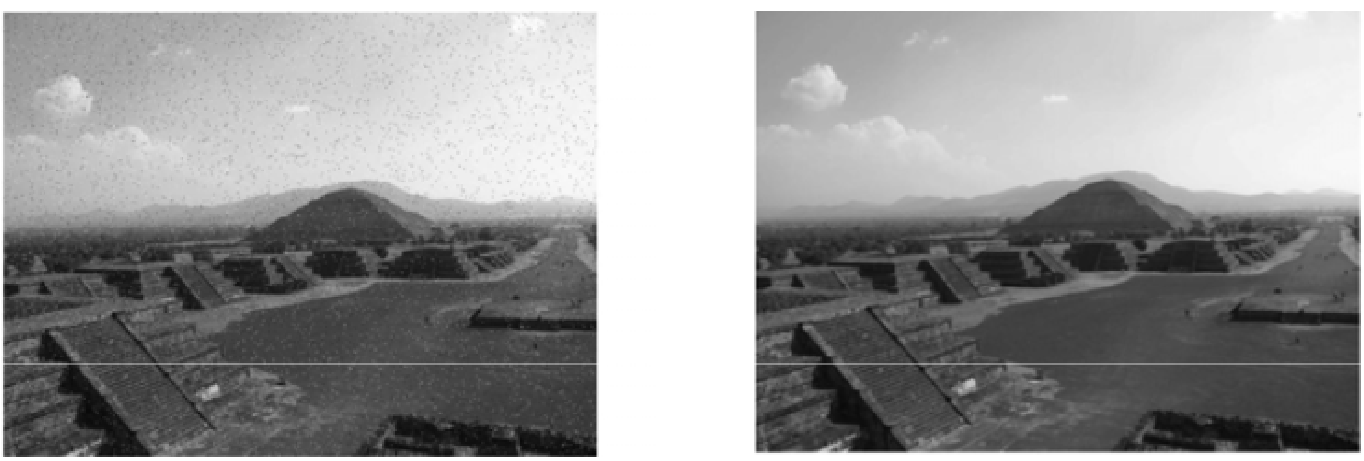


Fig. 2. SMF example

3.1.1.2 The disadvantage of the simple median filter

Although median filter is a useful non-linear image smoothing and enhancement technique. It also has some disadvantages. The median filter removes both the noise and the fine detail since it can't tell the difference between the two. Anything relatively small in size compared to the size of the neighborhood will have minimal affect on the value of the median, and will be filtered out. In other words, the median filter can't distinguish fine detail from noise [3].

3.1.2 Adaptive median filtering

Therefore the adaptive median filtering has been applied widely as an advanced method compared with standard median filtering. The Adaptive Median Filter performs spatial processing to determine which pixels in an image have been affected by impulse noise. The Adaptive Median Filter classifies pixels as noise by comparing each pixel in the image to its surrounding neighbor pixels. The size of the neighborhood is adjustable, as well as the threshold for the comparison. A pixel that is different from a majority of its neighbors, as well as being not structurally aligned with those pixels to which it is similar, is labeled as impulse noise. These noise pixels are then replaced by the median pixel value of the pixels in the neighborhood that have passed the noise labeling test.

There are three purposes for adaptive median filtering:

1. Remove impulse noise
2. Smoothing of other noise
3. Reduce distortion, like excessive thinning or thickening of object boundaries

3.1.2.1 Functionality

1. Adaptive median filter changes size of Sxy (the size of the neighborhood) during operation.
2. **Notation**
   Zmin = minimum gray level value in Sxy
   Zmax = maximum gray level value in Sxy
   Zmed = median of gray levels in Sxy
   Zxy = gray level at coordinates (x, y)
   Smax = maximum allowed size of Sxy

3. **Algorithm**
   Level A: A1 = Zmed - Zmin
   A2 = Zmed - Zmax
   if A1 ¿ 0 AND A2 ¡ 0, go to level B
   else increase the window size
   if window size ¡ Smax, repeat level A
   else output Zxy
   Level B: B1 = Zxy - Zmin
   B2 = Zxy - Zmax
   if B1 ¿ 0 AND B2 ¡ 0, output Zxy
   else output Zmed
4. **Explanation**
   Level A: IF Zmin ¡ Zmed ¡ Zmax, then
   •Zmed is not an impulse
   (1) go to level B to test if Zxy is an impulse ...
   ELSE
   •Zmed is an impulse
   (1) the size of the window is increased and
   (2) level A is repeated until ...
   (a) Zmed is not an impulse and go to level B or
   (b) Smax reached: output is Zxy
   Level B: IF Zmin ¡ Zxy ¡ Zmax, then
   •Zxy is not an impulse
   (1) output is Zxy (distortion reduced)
   ELSE
   •either Zxy = Zmin or Zxy = Zmax
   (2) output is Zmed (standard median filter)
   •Zmed is not an impulse (from level A)

3.1.2.2 Advantages

The standard median filter does not perform well when impulse noise is Greater than 0.2, while the adaptive median filter can better handle these noises also b. The adaptive median filter preserves detail and smooth non-impulsive noise, while the standard median filter does not [3].

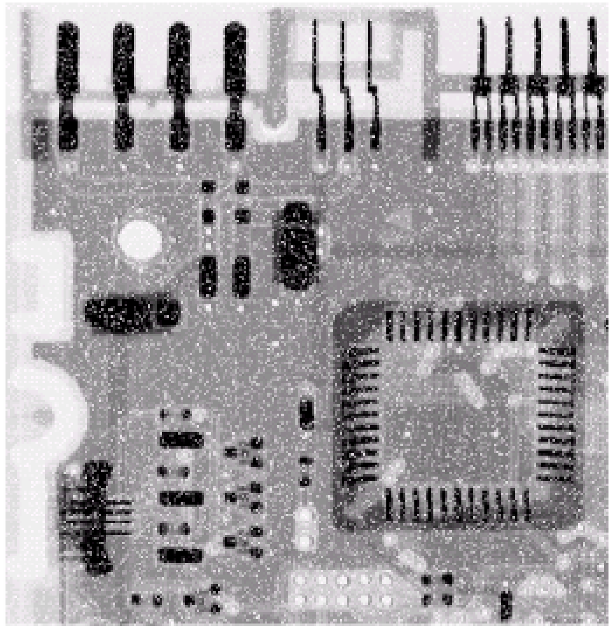
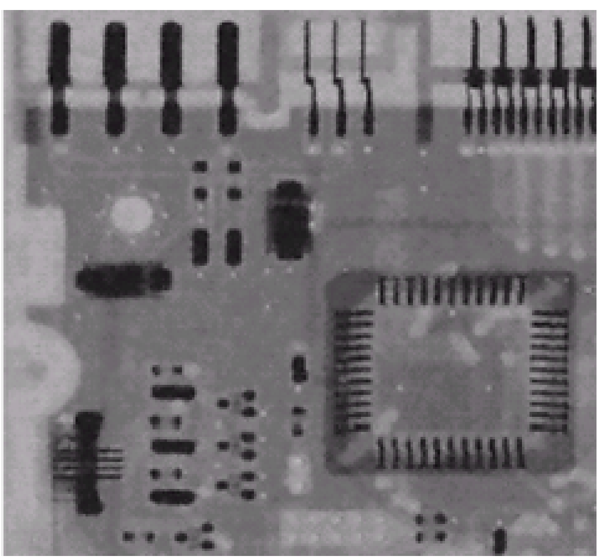
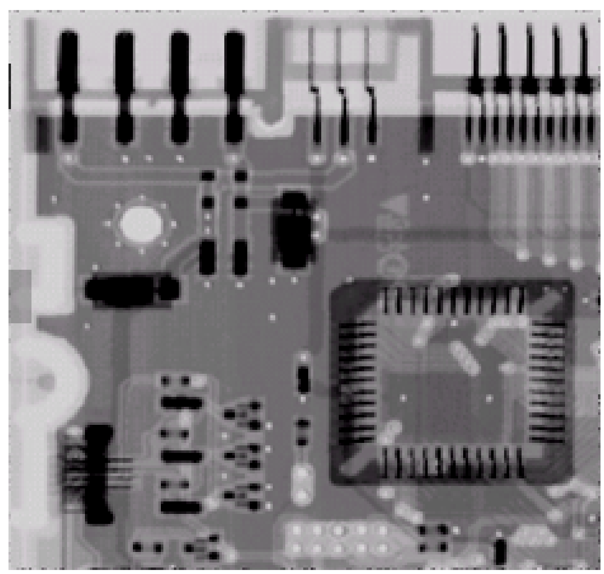


Fig. 3. Image corrupted by impulse noise with a probability of 0.1



(a) Result of standard median filtering



(b) textResult of adaptive median filtering

3.2 Mean filtering

Smoothing may be accomplished by applying an averaging mask that computes a weighted sum of the pixel greylevels in a neighbourhood and replaces the centre pixel with that greylevel. The image is blurred and its brightness retained as the mask coefficients are all-positive and sum to one. The mean filter is one of the most basic smoothing filters.

3.2.1 Functionality

Mean filtering is usually thought of as a convolution operation as the mask is successively moved across the image until every pixel has been covered. Like other convolutions it is based around a kernel, which represents the shape and size of the neighbourhood to be sampled when calculating the mean. Larger kernels are used when more severe smoothing is required.

$$\begin{array}{cc} 3\times 3 \text{ kernel} & n\times n \text{ kernel} \\ \frac{1}{9}\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} & \frac{1}{n^2}\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix}\end{array}$$

Fig. 4. Effect of mean filtering: (a) original crowd image; (b) Blurred image with n=7 and (c) 11.

Fig. 4 (a) shows a mean mask for a 3×3 window, while a more general $n\times n$ mask is shown in Fig. 1 (b). Variations on the mean filter include threshold averaging, wherein smoothing is applied subject to the condition that the center pixel greylevel is changed only if the difference between its original value and the average value is greater than a preset threshold. This causes the noise to be smoothed with a less blurring in image detail.

It must be noted here that the smoothing operation is equivalent to low-pass filtering as it eliminates edges and regions of sudden graylevel change by replacing the center pixel greylevel by the neighbourhood average. It effectively eliminates pixel greylevels that are unrepresentative of their surroundings. Noise, due to its spatial decorrelatedness, generally has a higher spatial frequency spectrum than the normal image components. Hence, a simple low-pass filter can be very effective in noise cleaning.

Smoothing filters thus find extensive use in blurring and noise removal. Blurring is usually a preprocessing step bridging gaps in lines or curves, helping remove small unwanted detail before the extraction of relevant larger objects. Figs. 4 (b) and (c) show the effect of averaging for various window sizes n (7 and 11).

The mean filter is an important image processing tool and finds use in Gaussian noise reduction, blurring before thresholding to eliminate small detail, bridging gaps in broken characters for improved machine perception in OCRs, cosmetic processing of human faces images to reduce fine skin lines and blemishes, etc. The mean is also used as a derived or texture feature in image segmentation process [5].

3.2.2 Disadvantages

There are some potential problems [2]:

1. A single pixel with a very unrepresentative value can significantly affect the average value of all the pixels in its neighbourhood.
2. When the filter neighbourhood straddles an edge, the filter will interpolate new values for pixels on the edge and so will blur that edge. This may be a problem if sharp edges are required in the output.

3.3 Comparison between the median filter and the mean filter

Sometimes we are confused by median filter and average filter, thus lets do some comparison between them. The median filter is a non-linear tool, while the average filter is a linear one. In smooth, uniform areas of the image, the median and the average will differ by very little. The median filter removes noise, while the average filter just spreads it around evenly. The

performance of median filter is particularly better for removing impulse noise than average filter.
As Fig. 5 shown below are the original image and the same image after it has been corrupted by impulse noise at 10%. This means that 10% of its pixels were replaced by full white pixels. Also shown are the median filtering results using 3x3 and 5x5 windows; three iterations of 3x3 median filter applied to the noisy image; and finally for comparison, the result when applying a 5x5 mean filter to the noisy image [3].

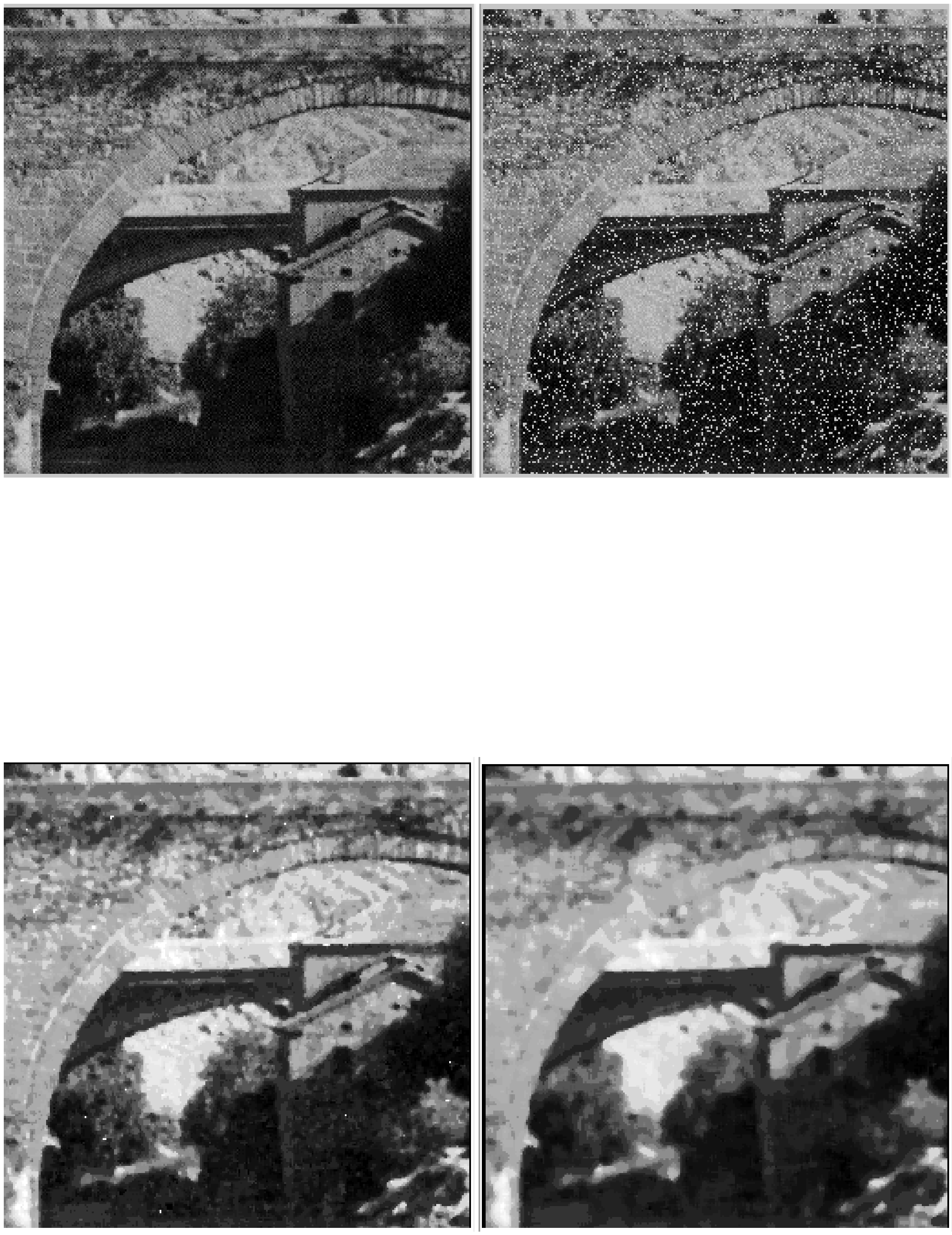


Fig. 5. $3x3$ Median Filtered and $5x5$ Median Filtered

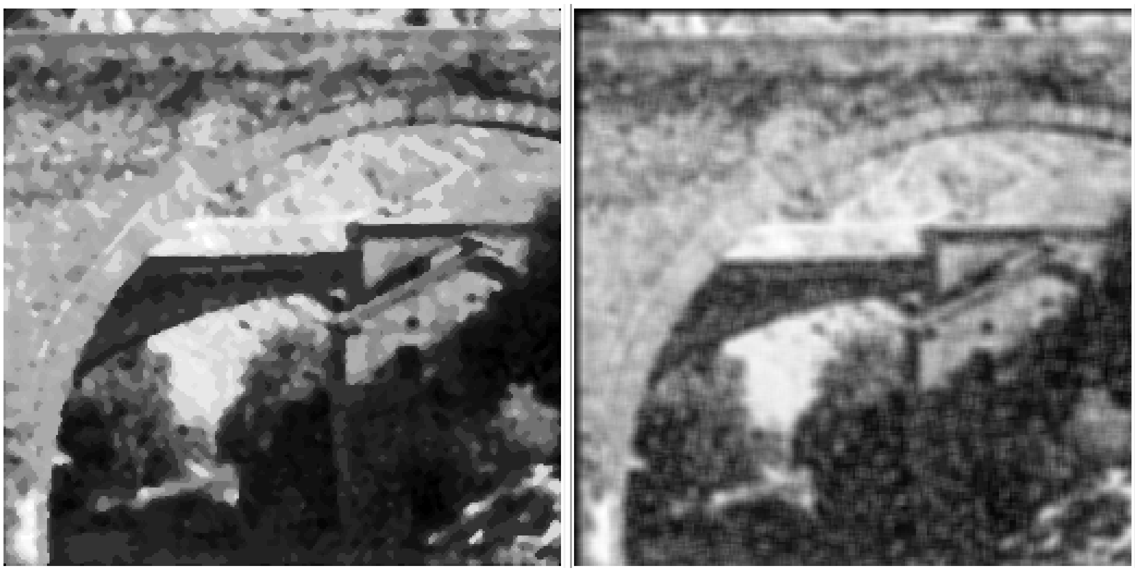
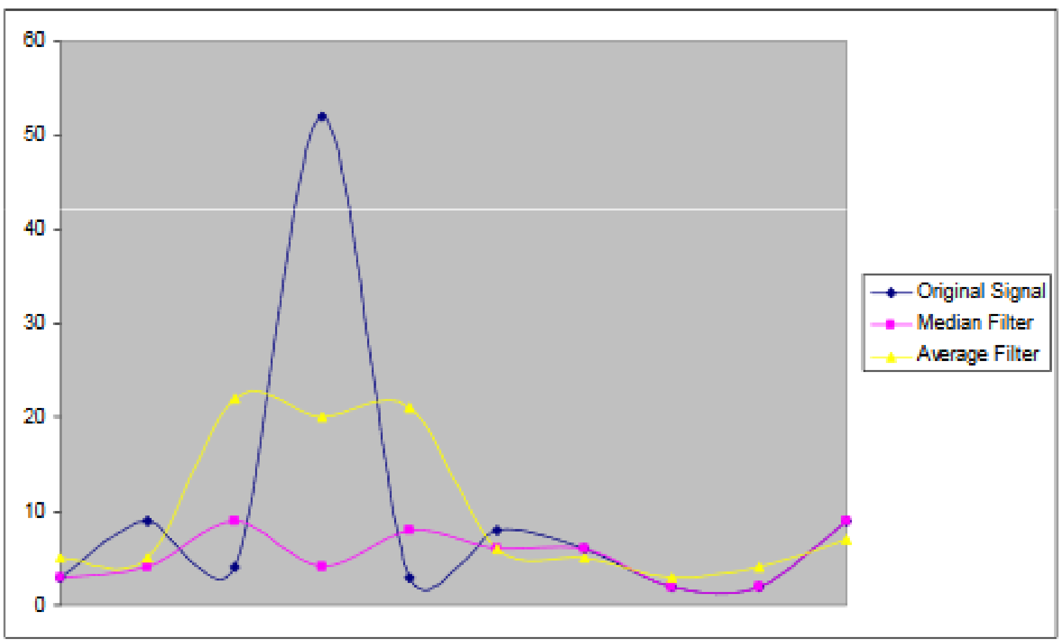


Fig. 6. $3x3$ Mean Filtered and $5x5$ Mean Filtered

The graph below shows the 1D signals from the median and average filter examples [2].



3.4 Dilation

Dilation is one of the basic operators in mathematical morphology. It is applied to binary image but can also be applied to grayscale image. Dilation causes the objects to grow in size. The effect of this operation will gradually increase the boundaries of foreground pixels, thus areas grow in size and holes in that region become smaller [6].

3.4.1 Functionality

The process of the structuring element B on the image A and moving it across the image in a way like convolution is defined as dilation operation. The two main inputs for the dilation operator are the image which is to be dilated and a set of coordinate points known as a structuring element which define also as a kernel. The exact effect of the dilation on the input image is determined by this structuring element. The following steps are the mathematical definition of dilation for binary images:

1. Suppose that X is the set of Euclidean coordinates corresponding to the input binary image, and that is the K set of coordinates for the structuring element.
2. Let K_x denote the translation of K so that its origin is at x.
3. Then the dilation of X by K is simply the set of all points x such that the intersection of k_x with X is non-empty.

It dilation is defined as set operation. A is dilated by B, written as $A \oplus B$, is defined as below:

$$A \oplus B = \{z|(\hat{B})_z \cap A \neq \Phi)\}$$

Among them, Φ is for the empty set, B is for the structure element, and $\hat{B}$ is for the reflection of collection B. An example is shown in Fig. 7, Note that with a dilation operation, all the 'black' pixels in the original image will be retai ned, any boundaries will be expanded, and small holes will be filled [7].

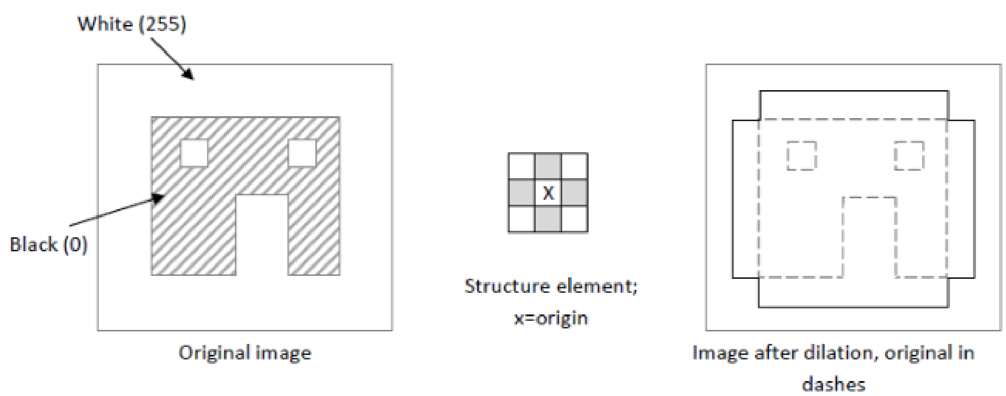


Fig. 7. Representation of the dilation process

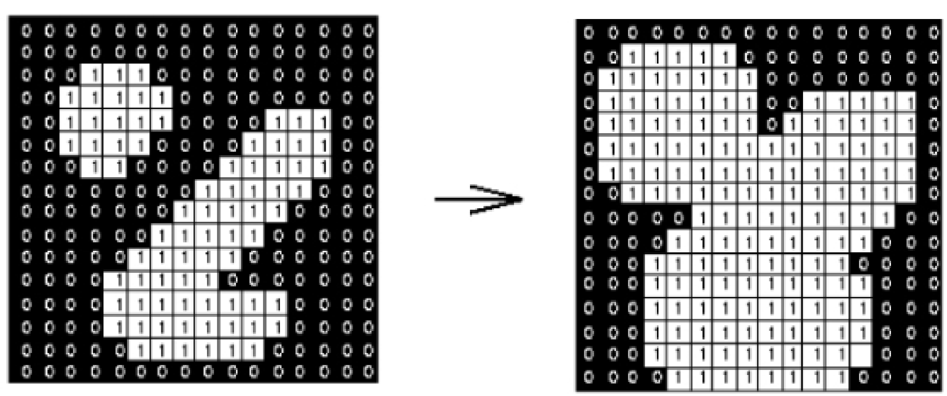
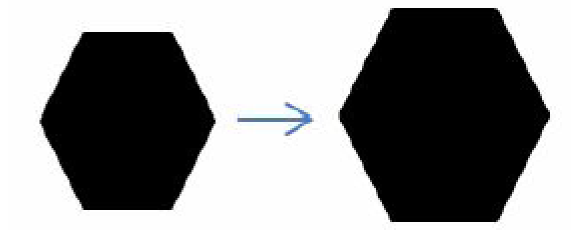


Fig. 8. Effect of dilation using a 3×3 square structuring element on binary image

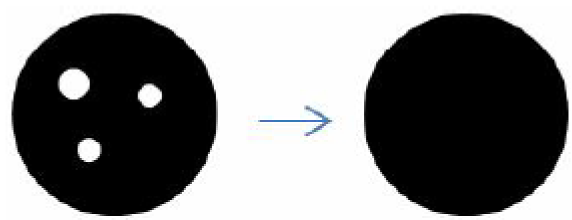
3.4.2 Applications

Dilation can be used for [8]:

1. Expanding shapes



2. Filling holes, gaps and gulfs



3.5 Erosion

Erosion is also one of the basic operators in mathematical morphology. Erosion causes the objects to shrink or become thin in size. Erosion basically erodes away the boundaries of the foreground which results in areas of those pixels shrink in size and holes of those areas become larger. So, after dilution and filling the holes of object in som e images the boundaries get mixed up so to somewhat separate the boundaries erosion is applied so as to make the boundaries of the objects thinner for better output [6].

3.5.1 Functionality

The erosion process is as same as dilation, but the pixels are converted to 'white', not 'black'. The two main inputs for the erosion operator are the image which is to be eroded and a set of coordinate points known as a structuring element which define also as a kernel. The exact effect of the erosion on the input im age is determined by this structuring element. The following steps are the mathematical definition of erosion for binary images:

1. Suppose that X is the set of Euclidean coordinates corresponding to the input binary image, and that is the K set of coordinates for the structuring element.
2. Let K_x denote the translation of K so that its origin is at x.
3. Then the erosion of X by K is simply the set of all points x such that the k_x is a subset of X.

A is eroded by B, recorded as $A\Theta B$, and defined as:

$$A \ominus B = \{z|(B)_z \cap A^c \neq \Phi\}$$

Among them, Φ is for the empty set, B is for the structure element, and A^c is for the supplement of collection A . In Fig. 9, the only remaining pixels are those that coincide to the origin of the structuring element where the entire structuring element was contained in the existing object. Because the structuring element is 3 pixels wide, the 2 - pixel - wide right leg of the image [7].

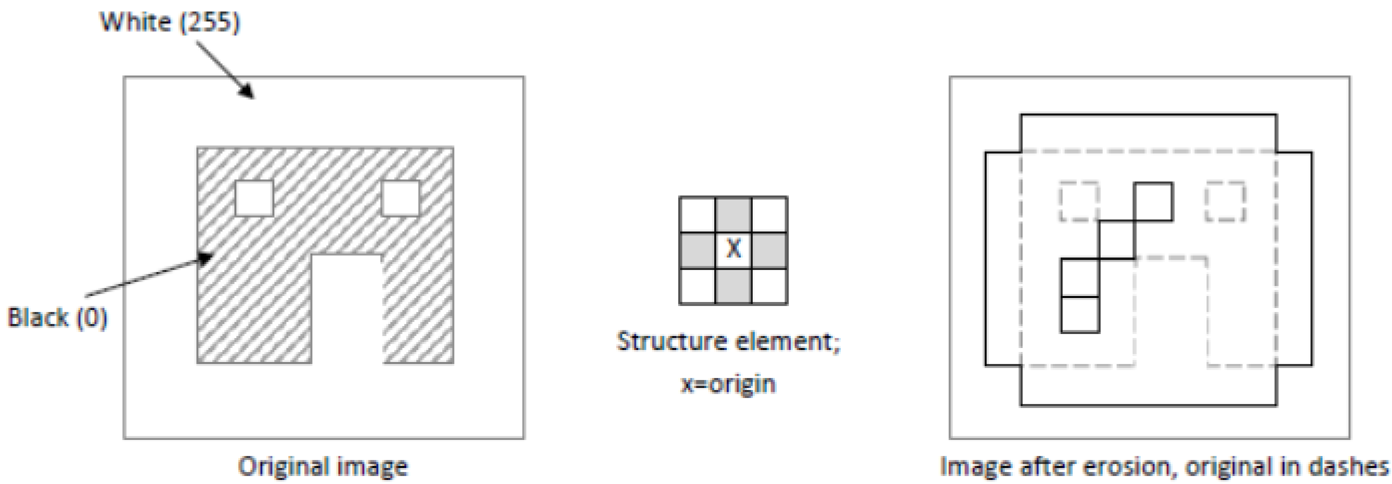


Fig. 9. Representation of erosion process

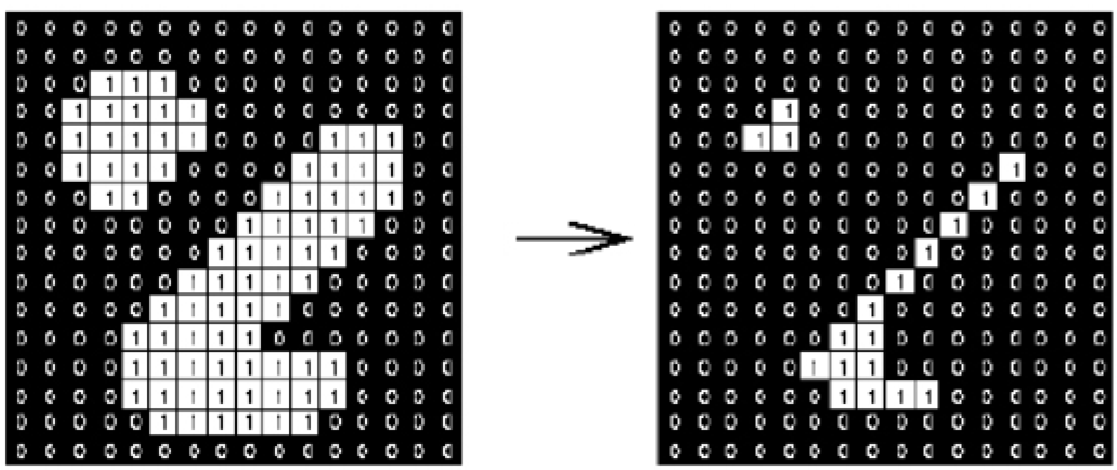
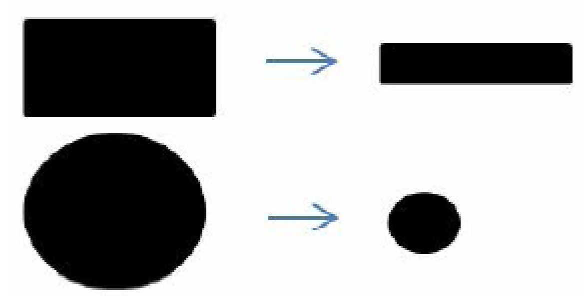


Fig. 10. Effect of erosion using a 33 square structuring element on binary image

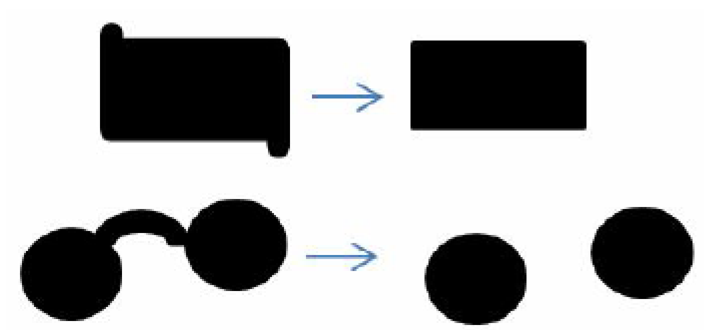
3.5.2 Applications

Erosion can be used for [8]:

Shrinking shapes



Removing bridges, branches and small protrusions



3.6 Opening

The combination of the two main operations, dilation and erosion, can produce more complex sequences. Opening and closing are the most useful of these for morphological filtering. An opening operation is defined as erosion followed by a dilation using the same structuring element for both operations.

3.6.1 Functionality

The basic two inputs for opening operator are an image to be opened, and a structuring element. Graylevel opening consists simply of graylevel erosion followed by graylevel dilation. See Fig. 11 for the illustration of the opening process [7]. Using the structure element B to do the open operation on the set A , expressed as $A \circ B$, definite as

$$f \circ b = (f \ominus b) \oplus b$$

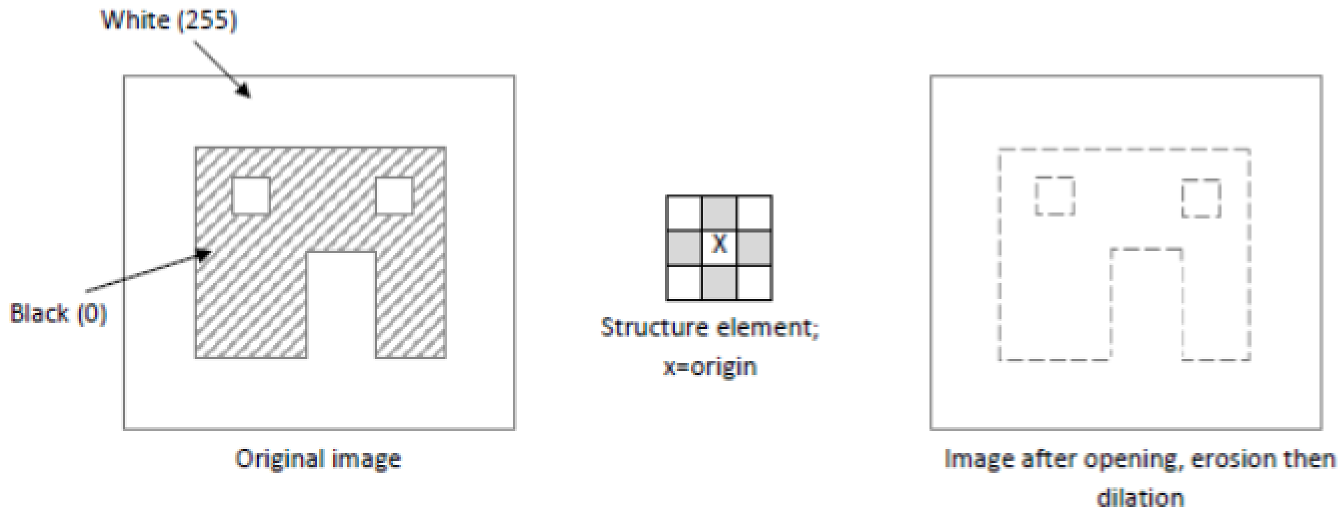


Fig. 11. Representation of the opening process

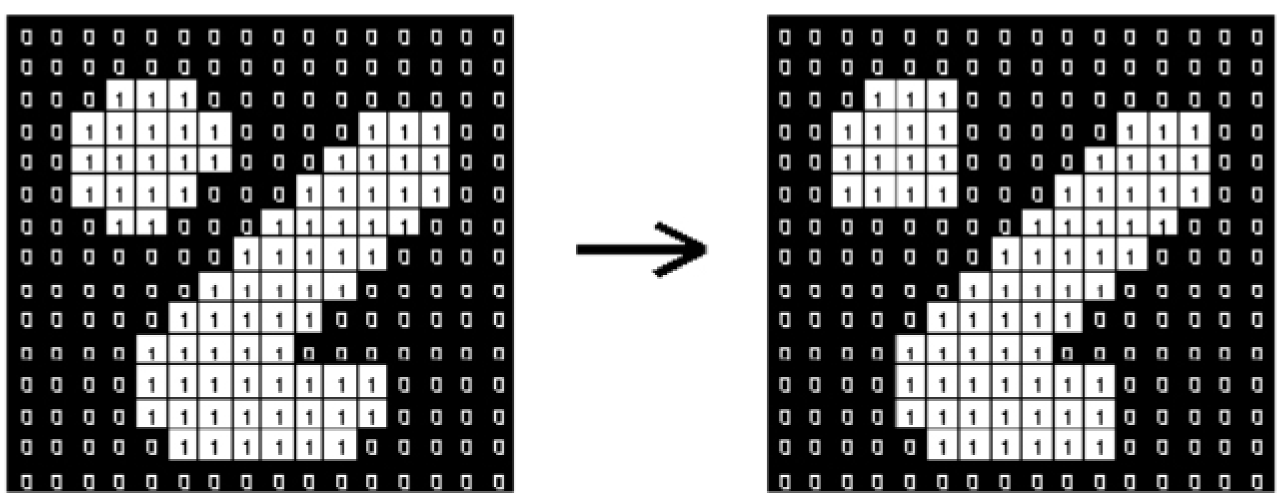


Fig. 12. Effect of opening using a 33 square structuring element on binary image

Opening can remove small bright spots (i.e. salt) and connect small dark cracks [9].

3.6.2 Applications

Use of opening is smoothing the edges, breaking the narrow joints or separates the objects and thinning the protrusions that are present in an image [10], as shown below:

Fig. 13. (a) Original image, (b) Image after opening

3.7 Closing

Closing is opening performed in reverse. Simply it is defined as dilation then erosion using the same structuring element for both operations. The main inputs of the closing operator are an image to be closed and a structuring element. The process of a graylevel dilation followed by graylevel erosion is defined as Graylevel closing [7].

3.7.1 Functionality

Fig. 14 shows the effects of filling in holes and closing gaps which define as closing operation. Like binary image , at opening operation , Use b to erode f plainly first, then on the results obtained , us e b to do dilate operation. Also, using b to do closing operation on f , expressed as $f \bullet b$, definite as

$$f \bullet b = (f \oplus b) \ominus b$$

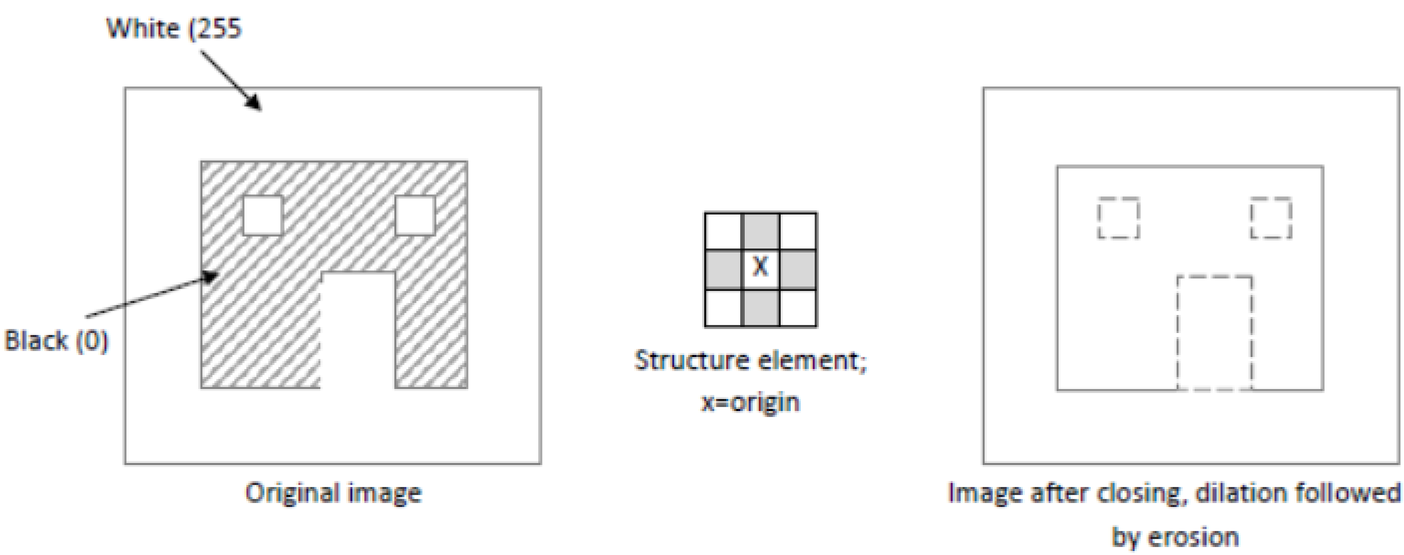


Fig. 14. Representation of the closing process

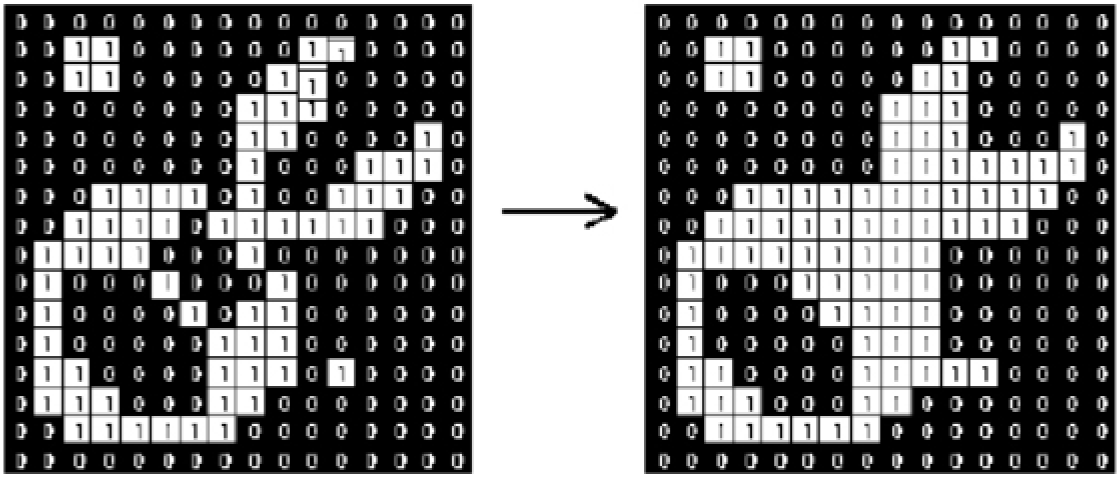


Fig. 15.   Effect of closing using a 3×3 square structuring element on binary image

Closing can remove small dark spots (i.e. pepper) and connect small bright cracks [9].

3.7.2 Applications

Closing is useful for smoothing sections of contours, eliminates small holes and fills gaps in contours [10], As shown below:

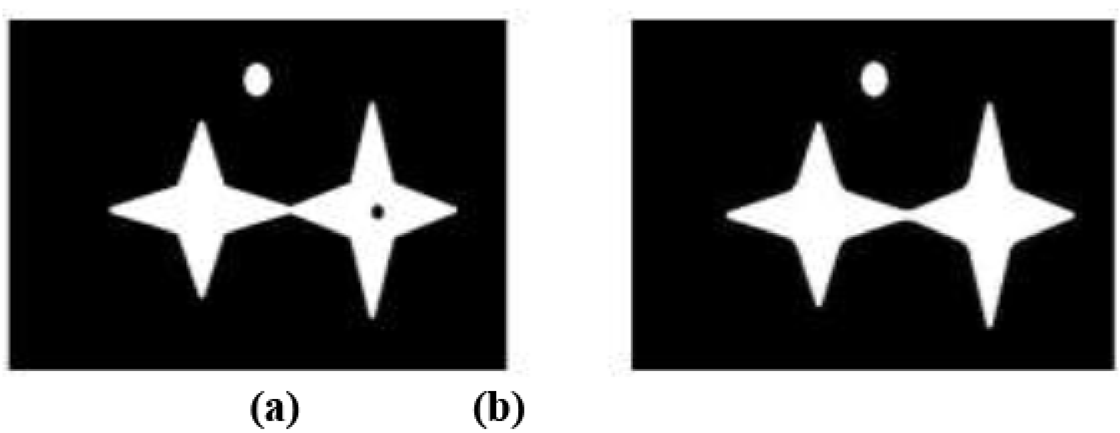


Fig. 16.   (a) Original Image (b) Image after closing

3.8 White and Black top-hat filter

In mathematical morphology and digital image processing, top-hat transform is an operation that extracts small elements and details from given images. There exist two types of top-hat transform: The white top-hat transform is defined as the difference between the input image and its opening by some structuring element; the black top-hat transform is defined dually as the difference between the closing and the input image. Top-hat transforms are used for various image processing tasks, such as feature extraction, background equalization, image enhancement, and others. [12].

3.8.1 Functionality

Let

$$f : E \to R$$

be a grayscale image, mapping points from an Euclidean space or discrete grid E (such as R^2 or Z^2) into the real line. Let b(x) be a grayscale structuring element. Then, the white top-hat transform of f is given by:

$$T_w(f) = f - f \circ b$$

The black top-hat transform of f is given by:

$$T_b(f) = f \bullet b - f$$

3.8.2 Applications

The white top-hat transform returns an image, containing those ”objects” or ”elements” of an input image that: Are ”smaller” than the structuring element (i.e., places where the structuring element does not fit in), and are brighter than their surroundings. The black top-hat returns an image, containing the ”objects” or ”elements” that: Are ”smaller” than the structuring element, and are darker than their surroundings. The size, or width, of the elements that are extracted by the top-hat transforms can be controlled by the choice of the structuring element b. The bigger the latter, the larger the elements extracted. Both top-hat transforms are images that contain only non-negative values at all pixels [12].

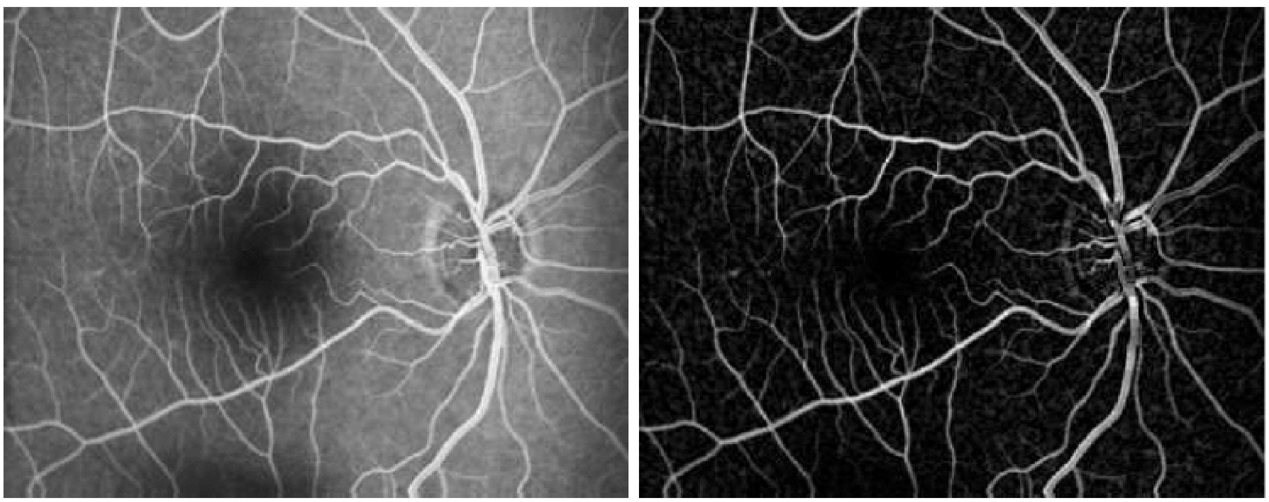


Fig. 17. Left: Image of the background of an eye, Right:Vessel extraction with a white top hat [13].

Fig. 18. Left: Original image, Right:Image after black top hat filter [9].

3.9 Bottom hat filter

One principal application of these transforms is in removing objects from an image by using an SE in the opening and closing that does not fit the objects to be removed. The difference then yields an image with only the removed objects. The top-hat is used for light objects on a dark background and the bottom-hat for dark objects on a light background. An important use of top-hat transformation is in correcting the effects of non-uniform illumination. Bottom-hat morphological operator subtracts input image from result of morphological closing on the input image. Applied to binary image, the filter allows getting all object parts, which were added by closing filter, but were not removed after that due to formed connections/fillings [12].

Fig. 19. Before and after Bottom-Hat Transformation

3.10 Edge detection operators

Edge detection is a crucial step in object recognition. It is a process of finding sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. In short, the goal of edge detection is to produce a line drawing of the input image. The extracted features are then used by computer vision algorithms, e.g. recognition and tracking.
A classical method of edge detection involves the use of operators, a two dimensional filter. An edge in an image occurs when the gradient is greatest. The operator works by identifying these large gradients to find the edges. There are vast amount of operators designed to detect certain types of edges. The operators can be configured to search for vertical, horizontal, or diagonal edges. One major problem with edge detection is when noise is present in images. It is not enough to simply reduce the noise, because the image will be either distorted or blurred. Fortunately, the operator can average enough data to discount localized noisy pixels. However, the change in intensity is not always a step change. The intensity change can also be gradual where the operator then has to be modified for proper edge detection. Consequently, there are problems of missing true edges, false edge detection, and high computational time. In next two sections, Sobel and Canny detection methods are investigated [15].

3.10.1 Sobel operator

The Sobel operator is a discrete differential operator. The operator utilizes two 3x3 kernels: one estimates the gradient in the x-direction, while the other one estimates the gradient in the y-direction.

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Fig. 20. Sobel Operator uses 3×3 Kernel Masks

The image is convolved with both kernels to approximate the derivatives in horizontal and vertical change. At each given point, magnitude of the gradient can be approximated with:

$$G = \sqrt{G_x^2 + G_y^2}$$

However, it is faster to compute the gradient magnitude with:

$$G = |G_x| + |G_y|$$

Due to Sobel operators smoothing effect (Gaussian smoothing), it is less sensitive to noise present in images. On the other hand, smoothing affects the accuracy of edge detection. In other words, the Sobel method does not produce image with high accuracy for edge detection, but its quality is adequate enough to be used in numerous applications [15].

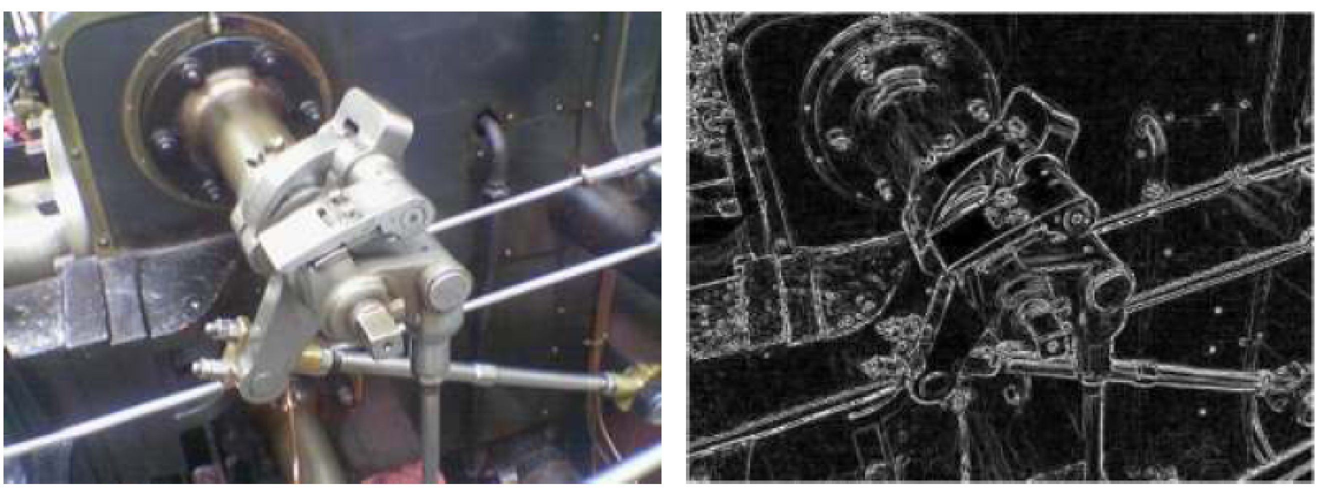


Fig. 21. Original image and the resulting image after applying sobel operator.

3.10.2 Canny Edge Detector

The Canny operator is widely known as the optimal detector, developed by John F. Canny in 1986. There are multiple steps to implement the Canny operator. First, a Gaussian filters is used to smooth the image to remove noise in an image.

Second, compute the gradient magnitude similar to Sobel operator. Third, non-maximum suppresion is applied in which the algorithm removes pixels that are not part of an edge. The final step involves the use of hysteresis thresholding along edges. Hystersis uses two thresholds, upper and lower. If a pixel gradient is higher than the upper threshold, then the pixel will be marked as an edge. If a pixel gradient is below the lower threshold, then the pixel will be discarded. Finally, if the pixel gradient is between the two thresholds, then only the pixel that is connected above the upper threshold is marked as an edge [15].

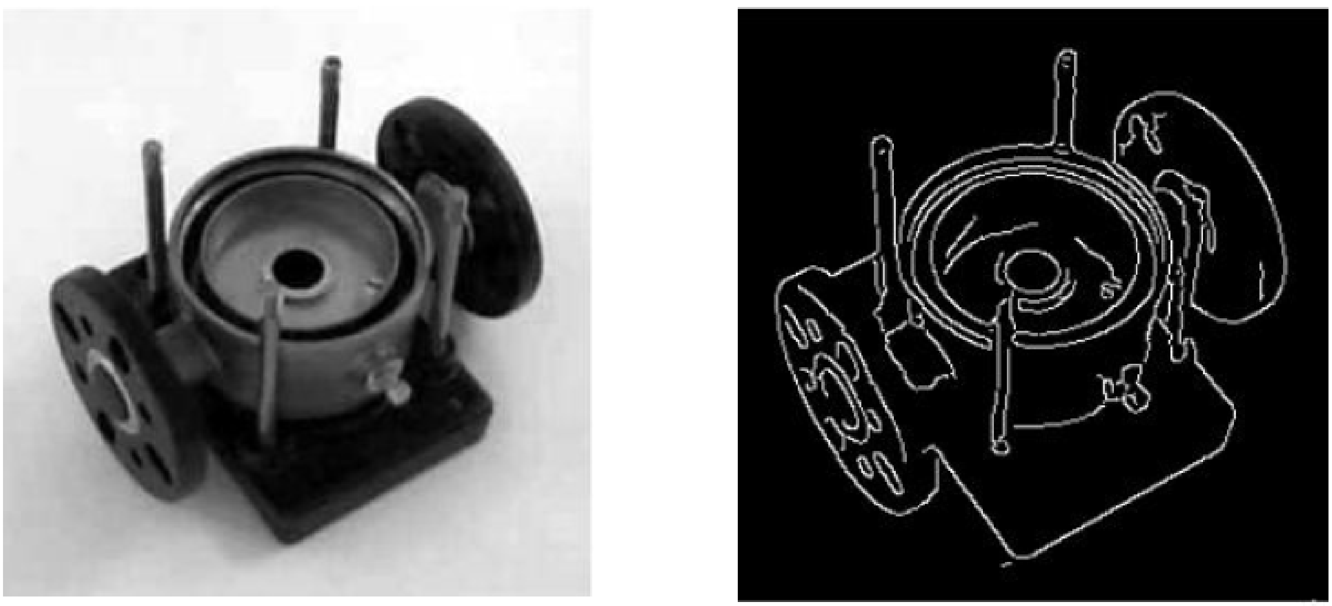


Fig. 22. Original image and the resulting image after applying canny operator.

References

[1] Jeffrey Morgan. A Glossary of Image Processing Terms

[2] https://www.cs.auckland.ac.nz/courses/compsci373s1c/PatricesLectures/Image

[3] PENG Lei. Adaptive Median Filtering.

[4] E.Jebamalar Leavline, D.Asir Antony Gnana Singh Salt and Pepper Noise Detection and Removal in Gray Scale Images: An Experimental Analysis. Bharathidasan Institute of Technology, Anna University Chennai, Regional Centre Tiruchirappalli- 620 024.

[5] S. Rakshit, A. Ghosh, B. Uma Shankar. Machine Intelligence Unit, Indian Statistical Institute, 203 B.T. Road, Kolkata - 700108, India. Fast mean filtering technique (FMFT)

[6] Image Segmentation using Morphological Operations. Diya Chudasama, Tanvi Patel, Shubham Joshi, Ghanshyam I. Prajapati. International Journal of Computer Applications (0975 8887) Volume 117 - No. 18, May 2015.

[7] IMAGE RESTORATION BASED ON MORPHOLOGICAL OPERATIONS. A.M.Raid, W.M.Khedr, M.A.El-dosuky and Mona Aoud. International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol. 4, No.3, June 2014.

[8] Extraction of superficial and volumetric features in three dimensional digital images using morphological methods. Towheed, sultana

[9] http://scikit-image.org/docs/dev/auto_examples/xx_applications/plot_morphology.html

[10] Applications of Morphological Operators using Image Morphological Algorithms. Sakshi Arora, Rahul Pandey. SSRG International Journal of Electronics and Communication Engineering (SSRG-IJECE) Volume 3 Issue 8 August 2016

[11] Image Sharpening By Gaussian And Butterworth High Pass Filter. AYUSH DOGRA and PARVINDER BHALLA. Biomedical & Pharmacology Journal, Vol. 7(2), 707-713 (2014)

[12] STUDY AND IMPLEMENTATION OF VARIOUS MORPHOLOGY BASED IMAGE CONTRAST ENHANCEMENT TECHNIQUES. Suman Thapar, Shevani Garg. International Journal of Computing & Business Research ISSN (Online): 2229-6166

[13] Image Processing and Computer Vision. Joachim Weickert. lecture 11, 2004.

[14] https://sisu.ut.ee/imageprocessing/book/1

[15] Sobel Operator and Canny Edge Detector. Daniel Kim. University of Washington, ECE 480 Fall 2013.