STUDY AND IMPLEMENTATION OF MORPHOLOGY FOR IMAGE SEGMENTATION

A thesis report

Submitted in the partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Electronic Instrumentation & Control Engineering

Submitted by

Saurabh Pandey
Roll No-800851026

Under the supervision of
Mr. M.D. Singh
Assistant professor, EIED

DEPARTMENT OF ELECTRICAL AND INSTRUMENTATION ENGINEERING
THAPAR UNIVERSITY
PATIALA –147004
JULY - 2010

DEDICATED

TO

MY PARENTS

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, **"Study And Implementation of Morphology for Image Segmentation"** in partial fulfillment of the requirements for the award of degree of Master of Engineering in Electronic Instrumentation and Control Engineering submitted in Electrical and Instrumentation Engineering, Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of **Mr. M.D. Singh** (Assistant Professor) and refers other researcher's works which are duly listed in the reference section.

The matter embodied in this report has not been submitted anywhere for the award of any degree.

Date: 15/07/10

Saurabh Pandey
Roll No - 800851026

It is certified that the above statement made by the student is correct to the best of our knowledge and belief.

Mr. M.D. Singh
Assistant Professor, EIED
(Supervisor)
Thapar University, Patiala

15/7/10
Dr. Smarajit Ghosh
Professor & Head, EIED
Thapar University, Patiala

Dr. R. K. Sharma
Dean of Academic Affairs
Thapar University, Patiala

ACKNOWLEDGEMENT

The real spirit of achieving a goal is through the way of excellence and austere discipline. I would have never succeeded in completing my task without the cooperation, encouragement and help provided to me by various personalities.

With deep sense of gratitude I express my sincere thanks to my esteemed and worthy supervisor, **Mr. M.D. Singh,** Assistant Professor, Department of Electrical and Instrumentation Engineering, Thapar University, Patiala for his valuable guidance in carrying out this work under his effective supervision, encouragement, enlightenment and cooperation. Most of the novel ideas and solutions found in this thesis are the result of our numerous stimulating discussions. His feedback and editorial comments were also invaluable for writing of this thesis.

I shall be failing in my duties if I do not express my deep sense of gratitude towards **Dr. Smarajit Ghosh,** Professor & Head of the Department of Electrical & Instrumentation Engineering, Thapar University, Patiala who has been a constant source of inspiration for me throughout this work.

I am also thankful to all the staff members of the Department for their full cooperation and help.

This acknowledgement would be incomplete if I do not mention the emotional support and blessings provided by my friends. I had a pleasant enjoyable and fruitful company with them.

My greatest thanks are to all who wished me success especially my parents, my brother and sisters, my best friends whose support and care makes me stay on earth.

Place: Thapar University, Patiala
Date: 15/07/10

Saurabh Pandey
800851026

ABSTRACT

The field of computer vision is concerned with extracting features and information from images in order to make analysis of images easier so that more and more information can be extracted. The task of image segmentation is a first step in many computer vision methods and serves to simplify the problem by grouping the pixels having similar attributes in the image. It is hard to clearly define image segmentation because there are many levels of detail in an image and therefore many possible ways of meaningfully grouping pixels i.e., various methods of image segmentation exist utilizing different image characteristics, e.g. shape, texture, motion, contrast, gray level etc. This thesis, presents a way to approach image segmentation of ultrasound images as well segmentation followed by counting of objects in images by using mathematical morphology, explain an efficient implementation for this approach, and show segmentation results using mathematical morphology.

ORGANIZATION OF THESIS

The whole of the work is divided into six chapters; the brief discussion is as follows.

1. The first chapter gives the introduction about the motivation and objective of this work, followed by background which describes previous work done on this subject in a very brief manner.
2. The second chapter is the review of literature i.e. a summary about the developments in the subject so far.
3. The third chapter gives a detailed description of morphology. It covers the description and function of each part which comes in this project.
4. The forth chapter discusses the methodology used and testing part of the thesis.
5. The fifth chapter shows the result of the proposed study and implementation of morphology for image segmentation.
6. Finally, sixth chapter concludes thesis with future scope.

TABLE OF CONTENTS

LIST OF FIGURE

LIST OF TABLES

LIST OF ABBREVIATIONS

2-D	Two-dimensional
3-D	Three-dimensional
CT	Computed Tomography
DEMP	Derivative of the Extended Morphological Profiles
GIF	Graphics Interchange Format
JPEG	Joint Photographic Experts Group
LOG	Laplacian of Gaussian
MATLAB	Matrix Laboratory
MM	Mathematical Morphology
MMC	Morphological Multiscale Characteristic
NA	Not Applicable
PNG	Portable Network Graphics
TIFF	Tagged Image File Format

CHAPTER-1
INTRODUCTION

Segmentation is defined as the separation of portions having similar characteristics such as gray level, color, texture, brightness, and contrast. This is a fundamental and necessary concept which plays an important role in digital image processing and biomedical image processing areas. There are many technique of segmenting an image. The techniques available for segmentation of images is application specific, depend upon imaging modality and type of image part to be studied i.e. it is not possible to segment all images by a particular method. In other words there is not any universal technique. Every method has its own merits and demerits depending upon application used.

The most commonly used segmentation techniques can be classified into two broad categories: first is Region based segmentation techniques that find the regions satisfying a given homogeneity criterion, and secondly Edge-based segmentation techniques that look for edges between regions with different characteristics.

Region based techniques are based on the principle of homogeneity - pixels with similar characteristics are grouped with each other to form a homogenous region. Criteria for homogeneity are most of the time gray level of pixels and thresholding is often used. The limitation of region based segmentation is that in this technique some seeding points are required to initialize the process, the segmentation results are dependent on the choice of seeds and there are chances of under segmentation and over segmentation of regions in the image. And also there is not any unique thresholding method so these affect the output segmentation depending upon different methods of thresholding.

Watershed segmentation is a region-based technique which utilizes image morphology. It requires selection of at least one marker ("seed" point) interior to each object of the image, including the background as a separate object. The markers are chosen by an operator or are provided by an automatic procedure. Once the objects are marked, they can be grown using a morphological watershed transformation. The disadvantage of the watershed method is that the different types of images need different thresholds. If the thresholds are not set correctly, then the objects are under-segmented or over-segmented. Additionally, slight changes in the threshold can significantly alter the segmentation results.

Edge based segmentation is the most common method based on detection of edges i.e. boundaries which separate distinct regions. Edge detection technique is based on marking of discontinuities in gray level, color etc., and often these edges represent boundaries between objects. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. This technique divides an image on the basis of boundaries. The limitations of edge based method include performance degradation by the presence of noise. If there is some type of noise in image, this technique gives corresponding edges for that noise which results in extra segmentation. Also fake and weak edges presented in the detected edge image may have a negative influence on segmentation and results in oversegmentation and these techniques are required to be used in conjunction with region-based technique for complete segmentation.

As discussed so far there are so many disadvantages of available methods. So we required a method which segment images having noises, not detect weak and fake edges and a method where selecting of threshold does not affect the segmenting output.

1.1 Objective

This thesis work deals with two objectives. One is to segment ultrasonic images and other is to segmentation followed counting of objects in images using concept of morphology.

1.2 Basic of Digital Image Processing

Digital image processing which a subfield of digital signal processing is the use of computer algorithms to perform image processing on digital images. Digital image processing has many advantages over analog image processing; it allows a much wider range of algorithms to be applied to the input data, and can avoid problems such as the build-up of noise and signal distortion during processing. Since images are defined over two dimensions digital image processing can be modelled in the form of Multidimensional Systems.

Images are produced by a variety of physical devices, including still and video cameras, x-ray devices, electron microscopes, radar, and ultrasound, and used for a variety of purposes, including entertainment, medical, business (e.g. documents), industrial, military, civil (e.g. traffic), security, and scientific. The goal in each case is for an observer, human or machine, to extract useful information about the scene being imaged.

An image is defined as a two-dimensional function, f(x, y), where x and y are spatial coordinates, and the amplitude of f at any pair of coordinates (x, y) is known as the intensity or gray level of the image at that particular point. The image is known as digital image when x, y, and the amplitude values of f are finite and discrete values. Processing of digital images by means of a digital computer is done in the field of Digital Image Processing. A digital image constitutes a finite number of elements, each of which has a particular location and a particular value. These elements are called picture elements or image elements or pels or pixels. Pixel is the term most widely used to denote the elements of a digital image [18].

There are three types of computerized processes in digital image processing system.

- Low-level processes
- Mid-level processes and
- High-level processes

Low-level processes consist of operations such as image preprocessing (to reduce noise), contrast enhancement, and image sharpening. In a low-level process both its inputs and outputs are images.

Mid-level processing on images consists of operations such as segmentation, description of those objects to reduce them to a form suitable for computer processing, and recognition of individual objects. In a mid-level process inputs generally are images, but its outputs are attributes extracted from those images (e.g., edges, contours, and the identity of individual objects).

Higher-level processing involves "making sense" of an ensemble of recognized objects, as in image analysis, performing the cognitive functions normally associated with vision.

The processes of acquiring an image of the area containing the text, preprocessing that image, extracting (segmenting) the individual characters, describing the characters in a form suitable for computer processing, and recognizing those individual characters are in the scope of what is called digital image processing [18].

1.3 Applications of Image Processing

There are a large number of applications of image processing in diverse spectrum of human activities-from remotely sensed scene interpretation to biomedical image interpretation.

1.3.1 Automatic Visual Inspection System

Automated visual inspection systems are essential to improve the productivity and the quality of the product in manufacturing and allied industries.

- **Automatic inspection of incandescent lamp filaments:** An interesting application of automatic visual inspection involves inspection of the bulb manufacturing process. Often the filaments of the bulbs get fused after short duration due to erroneous geometry of the filament, e.g., nonuniformity in the pitch of the wiring in the lamp. Manual inspection is not efficient to detect such aberrations. In an automated vision-based inspection system, a binary image slice of the filament is generated, from which the silhouette of the filament is produced. This silhouette is analyzed to identify the non-uniformities in the pitch of the filament geometry inside the bulb. Such a system has been designed and installed by the General Electric Corporation.
- **Faulty component identification:** Automated visual inspection may also be used to identify faulty components in an electronic or electromechanical system. The faulty components usually generate more thermal energy. The infra-red (IR) images can be generated from the distribution of thermal energies in the assembly. By analyzing these IR images, we can identify the faulty components in the assembly.
- **Automatic surface inspection systems:** Detection of flaws on the surfaces is important requirement in many metal industries. For example, in the hot or cold rolling mills in a steel plant, it is required to detect any aberration on the rolled metal surface. This can be accomplished by using

image processing techniques like edge detection, texture identification, fractal analysis, and so on.

1.3.2 Remotely Sensed Scene Interpretation

Information regarding the natural resources, such as agricultural, hydrological, and mineral, forest, geological resources, etc., can be extracted based on remotely sensed image analysis. For remotely sensed scene analysis, images of the earth's surface are captured by sensors in remote sensing satellites or by a multi-spectral scanner housed in an aircraft and then transmitted to the Earth Station for further processing.

Techniques of interpreting the regions and objects in satellite images are used in city planning, resource mobilization, flood control, agricultural production monitoring, etc.

1.3.3 Biomedical Imaging Techniques

Some of the biomedical imaging applications are presented below.

- ***Lung disease identification:*** In chest X-rays, the structures containing air appear as dark, while the solid tissues appear lighter. Bones are more radio opaque than soft tissue. The anatomical structures clearly visible on a normal chest X-ray film are the ribs, the thoracic spine, the heart, and the diaphragm separating the chest cavity from the abdominal cavity. These regions in the chest radiographs are examined for abnormality by analyzing the corresponding segments.
- ***Heart disease identification:*** Quantitative measurements such as heart size and shape are important diagnostic features to classify heart diseases. Image analysis techniques may be employed to radiographic images for improved diagnosis of heart diseases.
- ***Digital mammograms:*** Digital mammograms are very useful in detecting features (such as micro-calcification) in order to diagnose breast tumor. Image processing techniques such as contrast enhancement, segmentation, feature extraction, shape analysis, etc. are used to analyze mammograms. The regularity of the shape of the tumor determines whether the tumor is benign or malignant.

1.3.4 Content-Based Image Retrieval

Retrieval of a query image from a large image archive is an important application in image processing. A number of good search engines are available today for retrieving the text in machine readable form, but there are not many fast tools to retrieve intensity and color images.

The traditional approaches to searching and indexing images are slow and expensive. Thus there is urgent need for development of algorithms for retrieving the image using the embedded content in them. The features of a digital image (such as shape, texture, color, topology of the objects, etc.) can be used as index keys for search and retrieval of pictorial information from large image database. Retrieval of images based on such image contents is popularly called the content-based image retrieval. [19].

1.4 Basics of Image Segmentation

Segmentation which is a separation of structures having similar attributes from the background and from each other is an essential analysis function in image processing for which numerous algorithms have been developed. Typically, segmentation of an object is achieved either by identifying all pixels or voxels that belong to the object or by locating those that form its boundary. The former is based primarily on the intensity of pixels, but other attributes, such as texture, that can be associated with each pixel, can also be used for segmentation. Techniques that locate boundary pixels use the image gradient, which has high values at the edges of objects.

Since classification of pixels is required in image segmentation, it is often treated as a pattern recognition problem and addressed with related techniques.

The principle objective of the segmentation process is to divide an image into portions that are homogeneous with respect to one or more characteristics or features. After segmentation process has occurred the segmented portions have similar attributes. Segmentation is an important tool in image processing and it has been useful in many applications. In digital image processing, segmentation is important for feature extraction, image measurements, and image display. Image segmentation has applications separate from computer vision; it is frequently used to aid in isolating or removing specific portions of an image.

A wide variety of segmentation techniques has been proposed. However, there is no one standard segmentation technique that can produce satisfactory results for all imaging applications. The definition of the goal of segmentation varies according to the goal of the study and the type of the image data.

Segmentation techniques can be divided into classes in many ways, depending on classification scheme:

- Manual, semiautomatic, and automatic.
- Pixel-based (local methods) and region-based (global methods).
- Manual delineation, low-level segmentation (thresholding, region growing, etc), and model-based segmentation (multispectral or feature map techniques, dynamic programming, contour following, etc.)
- Classical (thresholding, edge-based, and region-based techniques).[20]

1.4.1 Classification of Segmentation

The most commonly used segmentation techniques can be classified into two broad categories:

- Region segmentation techniques that find the regions satisfying a given homogeneity criterion, and
- Edge-based segmentation techniques that look for edges between regions with different characteristics.

Thresholding is a common region segmentation method. In this technique a threshold is selected and an image is divided into groups of pixels having values less than the threshold and groups of pixels with values greater or equal to the threshold. There are several thresholding methods: global methods based on gray-level histograms, global methods based on local properties, local threshold selection, and dynamic thresholding. Clustering algorithms achieve region segmentation by partitioning the image into sets or clusters of pixels that have strong similarity in the feature space. The basic operation is to examine each pixel and assign it to the cluster that best represents the value of its characteristic vector of features of interest. Region growing is another class of region segmentation algorithms that assign adjacent pixels or regions to the same segment if their image values are close enough, according to some preselected criterion of closeness.

The objective of edge-based segmentation algorithms is to find object boundaries and segment the portion enclosed by that boundaries in the image. These algorithms usually operate on edge magnitude and/or phase images produced by an edge operator suited according to the characteristics of the image. For example, most gradient operators such as Prewitt, Kirsch, or Roberts's operators are based on the existence of an ideal step edge. Other edge-based segmentation techniques are graph searching and contour following.

Traditionally, most image segmentation techniques use one type of images (MR, CT, PET, SPECT, ultrasound, etc.). However, the performance of these techniques can be improved by combining images from several sources (multispectral segmentation) or integrating images over time (dynamic or temporal segmentation).

1.4.1.1 Region Based Segmentation Techniques

1.4.1.1.1Thresholding

Several thresholding techniques have been developed. Some of them are based on the image histogram; others are based on local properties, such as local mean value and standard deviation, or the local gradient. Global thresholding is the most intuitive approach. In global thresholding only one threshold is selected for the entire image, based on the image histogram. If the threshold depends on local properties of some image regions, for example local average gray value, thresholding is called local. Thresholding is called dynamic or adaptive if the local thresholds are selected independently for each pixel (or groups of pixels).

- **Global Thresholding**

Global thresholding is based on the assumption that the image has a bimodal histogram and, therefore, the object can be extracted from the background by a simple operation that compares image values with a threshold value T. Suppose that we have an image f(x, y) with the histogram shown on Fig. The object and background pixels have gray levels grouped into two dominant modes. One obvious way to extract the object from the background is to select a threshold T that separates these modes.

The thresholded image g(x, y) is defined as:

$$g(x, y) = \begin{cases} 1 & if(x,y) > T \\ 0 & if(x,y) \leq T \end{cases}$$

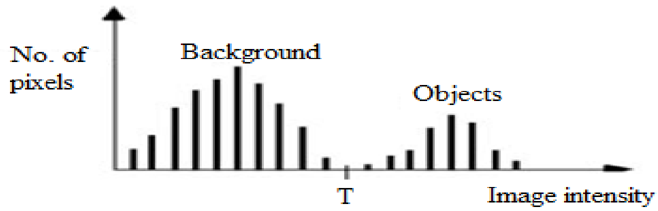


Fig-1.1 An example of bimodal histogram with selected threshold T

The result of thresholding is a binary image, where pixels with intensity value of 1 correspond to objects, while pixels with value 0 correspond to the background.

Fig1.2 shows the result of segmentation by thresholding. The original image (Fig1.2A) contains white cells on a black background. Pixel intensities vary between 0 and 255. The threshold T=127 was selected as the minimum between two modes on a histogram (Fig1.2B), and the result of segmentation is shown in (Fig1.2C), where pixels with intensity values higher than 127 are shown in white. In the last step (Fig1.2D) the edges of the cells were obtained by a Laplacian (second-order derivative), which was applied to the thresholded image on (Fig1.2C). [20]

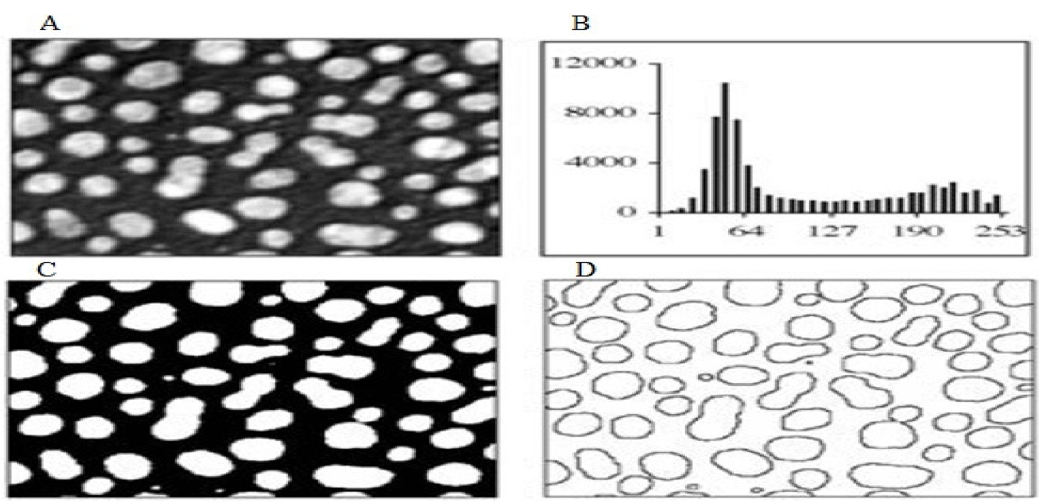


Fig-1.2 An example of global thresholding; (A) Original image, (B) histogram of image A, (C) result of thresholding with T=127, (D) outlines of the white cells after applying a Laplacian to the image shown in C.[20]

There are many other ways to select a global threshold. One of them is based on a classification model that minimizes the probability of error. For example, if an image with a bimodal histogram (e.g., object and background) is given, error can be calculated as the total number of background pixels misclassified as object and object pixels misclassified as background.

In many applications appropriate segmentation is obtained when the area or perimeter of the objects is minimally sensitive to small variations of the selected threshold level. Fig 1.3A shows the intensity profile of an object that is brighter than background, and three threshold levels for segmentation: T1, T2, and T3. A small variation ΔT in the lowest threshold level will cause a significant change in the area or perimeter of the segmented object. The same is true for the highest threshold level. However, a change of ΔT in the middle level will have minimal effect on the area or perimeter of the object. The object area A (T) and perimeter P (T) are functions of the threshold T that often exhibit the trend shown in Fig1.3B. Therefore, the threshold level that minimizes either dA(T)/dT or dP(T)/dT is often a good choice, especially in the absence of operator guidance and when prior information on object locations is not available.

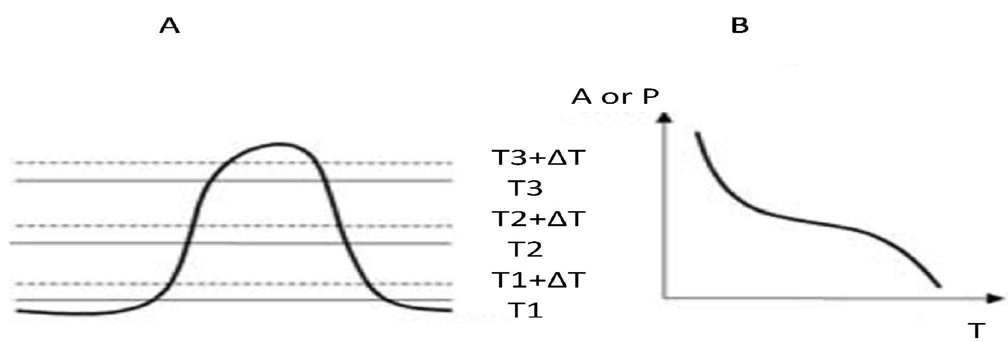


Fig-1.3 An example of sensitivity of the threshold level selection (A) Cross-sectional intensity profile of a light object on a dark background with three thresholding levels T1, T2, and T3, and three other levels generated by adding a small value ΔT; (B) a hypothetical plot of the area A or perimeter P versus thresholding level T [20].

A related technique that evaluates multiple thresholds is based on an estimate of the gradient magnitude around the segmented object edge. The average gradient magnitude is given by:

$$G = \lim_{\Delta T \to 0} \frac{\Delta T \times P(T)}{\Delta A} = \frac{P(T)}{H(T)}$$

where H (T) is the histogram function. The threshold that maximizes the average boundary gradient is selected.

When an image contains more than two types of regions, it may still be possible to segment it by applying several individual thresholds, or by using a multi-thresholding technique. With the increasing number of regions, the histogram modes are more difficult to distinguish, and threshold selection becomes more difficult.

Global thresholding is computationally simple and fast. It works well on images that contain objects with uniform intensity values on a contrasting background. However, it fails (i) if there is a low contrast between the object and the background, (ii) if the image is noisy, or (iii) If the background intensity varies significantly across the image.

- **Local (Adaptive) Thresholding**

In many applications, a global threshold cannot be found from a histogram or a single threshold cannot give good segmentation results over an entire image. For example, when the background is not constant and the contrast of objects varies across the image, thresholding may work well in one part of the image, but may produce unsatisfactory results in other areas. If the background variations can be described by some known function of position in the image, one could attempt to correct it by using gray level correction techniques, after which a single threshold should work for the entire image. Another solution is to apply local (adaptive) thresholding.

Local thresholds can be determined by (i) splitting an image into sub-images and calculating thresholds for each sub-image, or by (ii) Examining the image intensities in the neighbour-hood of each pixel.

In the former method, an image is first divided into rectangular overlapping sub-images and the histograms are calculated for each sub-image. The sub-images used should be large enough to include both object and background pixels. If a sub-image has a bimodal histogram, then the minimum between the histogram peaks should determine a local threshold. If a histogram is unimodal, the threshold can be assigned by interpolation from the local thresholds found for nearby sub-images. In the final step, a second interpolation is necessary to find the correct thresholds at each pixel.

In the latter method, a threshold can be selected using the mean value of the local intensity distribution. Sometimes other statistics can be used, such as mean plus standard deviation, mean of the maximum and minimum values, or statistics based on local intensity gradient magnitude. In general, local thresholding is computationally more expensive than global thresholding. It is very useful for segmenting objects from a varying background, and also for extraction of regions that are very small and sparse.

Many images may contain low-contrast, fuzzy contours. The histogram modes corresponding to the different types of regions in an image may often overlap and, therefore, segmentation by thresholding becomes difficult. Image preprocessing techniques can sometimes help to improve the shape of the image histogram, for example by making it more strongly bimodal. One of the techniques is image smoothing by using the mean (average) or median filter. The mean filter replaces the value of each pixel by the average of all pixel values in a local neighbour-hood (usually an N by N window, where N=3, 5, 7, etc.). In the median filter, the value of each pixel is replaced by the median value calculated in a local neighbour-hood. Median smoothing, unlike the mean filter, does not blur the edges of regions larger than the window used while smoothing out small textural variations

A common smoothing filter is the Gaussian filter, where for each pixel [i, j], the convolution mask coefficients g [i, j] are based on a Gaussian function:

$$g\,[i, j] = \exp\left[\frac{-(i^2+j^2)}{2\sigma^2}\right]$$

where σ is the spread parameter (standard deviation) that defines the degree of Gaussian smoothing: Larger σ implies a wider Gaussian filter and a greater amount of smoothing. The Gaussian filter can be approximated in digital images by an N by N convolution mask. [20]

1.4.1.1.2 Region Growing

As far thresholding methods focus on the difference of pixel intensities whereas the region growing method takes groups of pixels with similar intensities. Region growing, which is also named as region merging starts with a pixel or a group of pixels (called seeds) that belong to the structure of interest being segmented. Generally an operator chooses the seeds, or it can also provided by an automatic seed finding procedure. In the next step neighbouring pixels are examined one at a time and added to the growing region, if they are sufficiently similar based on a uniformity test, (also called a homogeneity criterion). The procedure continues until no more pixels can be added. The object is then represented by all pixels that have been accepted during the growing procedure. One example of the uniformity test is comparing the difference between the pixel intensity value and the mean intensity value over a region. If the difference is less than a predefined value, for example, two standard deviations of the intensity across the region, the pixel is included in the region; otherwise, it is defined as an edge pixel. The results of region growing depend strongly on the selection of the homogeneity criterion. If it is not properly chosen, the regions leak out into adjoining areas or merge with regions that do not belong to the object of interest. Another problem of region growing is that different starting points may not grow into identical regions.

The advantage of region growing is that it is capable of correctly segmenting regions that have the same properties and are spatially separated. Another advantage is that it generates connected regions.

1.4.1.1.3 Watershed Algorithm

Watershed segmentation is a region-based technique which utilizes image morphology. It requires selection of at least one marker ("seed" point) interior to each object of the image, including the background as a separate object. The markers are chosen by an operator or are provided by an automatic procedure that takes into account the application specific knowledge of the objects. Once the objects are marked, they can be grown using a morphological watershed transformation. To understand the watershed, one can think of an image as a surface where the bright pixels represent mountaintops and the dark pixels represents valleys. The surface is punctured in some of the valleys, and then slowly submerged into a water bath. The water will pour in each puncture and start to fill the valleys. However, the water from different punctures is not allowed to mix, and therefore the dams need to be built at the points of first contact. These dams are the boundaries of the water basins, and also the boundaries of image objects.

An application of watershed segmentation to extract lymph nodes on CT images is shown in Fig1.4. In this implementation a Sobel edge operator is used in place of the morphological gradient to extract edge strength.

The original lymph node image is shown in Fig1.4A. In the first step, the operator positions a cursor inside the node (Fig1.4B). All pixels within a radius of two pixels of the mark are used as seed points for the lymph node. To mark the exterior of lymph node, the operator drags the cursor outside of the node to define a circular region, which completely encloses the node (Fig1.4C). All pixels outside this circle mark the background.

In the next step, an edge image is created using the Sobel edge operator (Fig1.4D). The edge image has high values for then pixels with strong edges. With the seed point marking the node interior, the circle marking the background (Fig1.4C), and the edge image generated by the Sobel operator (Fig1.4D), the segmentation proceeds directly with the watershed operation (Fig1.4E). The watershed operation operates on an edge image to separate the lymph node from the surrounding tissue. By using a technique called simulated immersion, the watershed considers whether a drop of water at each point in the edge image would flow to the interior seed point or the exterior marker. Points that drain into the interior belong to the lymph node, whereas points that drain to the exterior belong to the surrounding tissue.

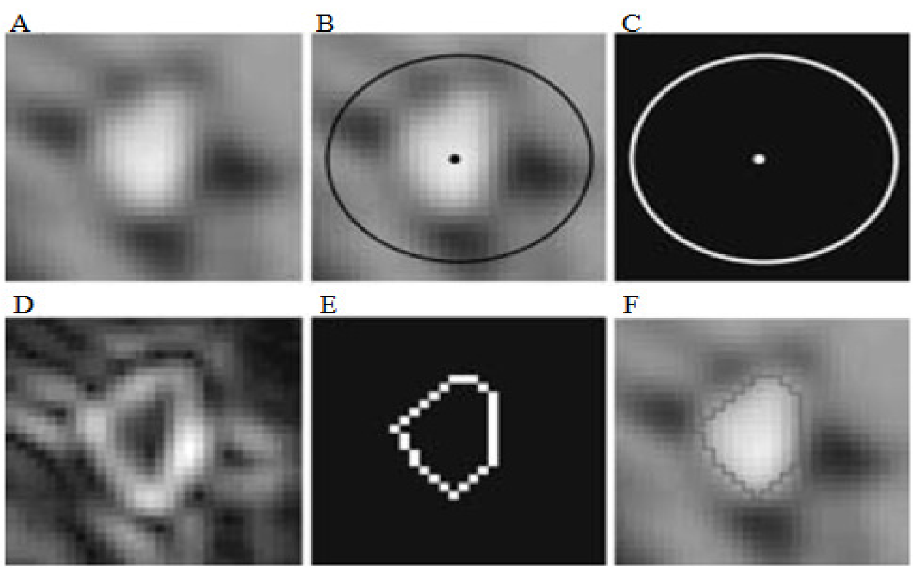


Fig1.4 Image segmentation using Sobel/watershed algorithm(A) Original image of a lymph node (B) operator's marks: a point inside the node, and a circle enclosing the area well outside the node; (C) binary image generated from B; (D) result of a Sobel

edge detection operation performed on the original image A; (E) result of the watershed algorithm performed on image D using markers from image C; (F) edges of the lymph node superimposed on the original image [20].

1.4.1.2 Edge-Based Segmentation Techniques

Edges characterize boundaries and are therefore a problem of fundamental importance in image processing. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. Edge detecting an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image.

There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories:

- Gradient methods which detects the edges by looking for the maximum and minimum in the first derivative of the image.
- Laplacian methods which search for zero crossings in the second derivative of the image to find edges.

An edge or boundary on an image is defined by the local pixel intensity gradient. A gradient is an approximation of the first order derivative of the image function.

For a given image f (x, y), we can calculate the magnitude of the gradient as

$$|G| = \sqrt{G_x^2 + G_y^2} = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

and the direction of gradient as

$$D = tan^{-1}\left(\frac{G_y}{G_x}\right)$$

where G_x and G_y are gradients in directions x and y, respectively. Since the discrete nature of digital image does not allow the direct application of continuous differentiation, calculation of the gradient is done by differencing.

Both magnitude and direction of the gradient can be displayed as images. The magnitude image will have gray levels that are proportional to the magnitude of the local intensity changes, while the direction image will have gray levels representing the direction of maximum local gradient in the original image.

Most gradient operators in digital images involve calculation of convolutions, e.g., weighted summations of the pixel intensities in local neighborhoods. The weights can be listed as a numerical array in a form corresponding to the local image neighbourhood (also known as a mask, window or kernel). For example, in case of a Sobel edge operator, there are two masks:

$$\begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix} \qquad\qquad \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix}$$

The first mask is used to compute G_x while the second is used to compute G_y. The gradient magnitude image is generated by combining G_x and G_y and $|G|$ is obtained by using respective equation. The results of edge detection depend on the gradient mask. Some of the other edge operators are Roberts, Prewitt, Robinson, and Kirsch.

Many edge detection methods use a gradient operator, followed by a threshold operation on the gradient, in order to decide whether an edge has been found. As a result, the output is a binary image indicating where the edges are.

The edge-based techniques are computationally fast and do not require a priori information about image content. The common problem of edge-based segmentation is that often the edges do not enclose the object completely. To form closed boundaries surrounding regions, a postprocessing step of linking or grouping edges that correspond to a single boundary is required. The simplest approach to edge linking involves examining pixels in a small neighbourhood of the edge pixel and linking pixels with similar edge magnitude and/or edge direction. In general, edge linking is computationally expensive and not very reliable.

The Laplacian methods which approximates second-order derivative can also be used to detect edges in which the Laplacian operator ∇^2 of a function f (x, y) is defined as

$$\nabla^2 f\,(x,\,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$$

The Laplacian is approximated in digital images by an N by N convolution mask.

Here are three examples of Laplacian masks that represent different approximations of the Laplacian operator:-

$$\begin{matrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{matrix} \qquad\qquad \begin{matrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{matrix} \qquad\qquad \begin{matrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{matrix}$$

The image edges can be found by locating pixels where the Laplacian makes a transition through zero (zero crossings). All edge detection methods that are based on a gradient or Laplacian are very sensitive to noise. In some applications, noise effects can be reduced by smoothing the image before applying an edge operation. Generally a Gaussian filter is used in order to smooth the image before processing. (This operation is called Laplacian of Gaussian, LoG). The advantage of LoG operator compared to a Laplacian is that the edges of the image are smoother and better outlined.

CHAPTER-2
LITERATURE SURVEY

Kelly, P.A., Chen, G. [1], (1993) described some iterative segmentation algorithms that combine statistical constraints represented in Markov random field models with deterministic constraints imposed by morphological operations. The goal of this whole study is to produce segmentations that have high probability according to the Markov model and are smooth in the sense of being morphologically open and/or closed. They first present several algorithms for binary images, including one that produces a segmentation in which the set of one's is both open and closed. The latter algorithm is then extended to the case of multiregional images to produce a segmentation in which each region is open and closed.

Deng Shiwei and Yuan Baozong [2], (1993) introduced a range image segmentation method using simple morphological operators. They described how to obtain peak and valley image using edge-strength operator, peak and valley detection operators. This procedure makes the depth discontinuities contained in the peaks and the orientation discontinuities in the valleys. The depth discontinuities are extracted and the segmentation is accomplished by controlled region growing. The experimental results prove this range image segmentation technique quick and robust.

Philippe Salembier and Montse Pardas [3], (1994) proposed an algorithm which first takes into account the global information and produces a coarse segmentation, that is, with a small number of regions. Then, the segmentation quality is improved by introducing regions corresponding to more local information. The algorithm, considering sequences as being functions on a 3-D space, directly segments 3-D regions. A 3-D approach is used to get a segmentation that is stable in time and to directly solve the region correspondence problem. According to this study each segmentation stage relies on four basic steps: simplification, marker extraction, decision, and quality estimation. The simplification removes information from the sequence to make it easier to segment. The marker extraction identifies the presence of homogeneous 3-D regions. The goal of the decision is to precisely locate the contours of regions detected by the marker extraction. This decision is performed by a modified watershed algorithm. Finally, the quality estimation concentrates on the coding residue, all the information about the 3-D regions that have not been properly segmented and therefore coded. The procedure allows the introduction of the texture and contour coding schemes within the segmentation algorithm. The coding residue is transmitted to the next segmentation stage to improve the segmentation and coding quality. This segmentation based coding approach is robust and simple. It makes no assumption about the input sequences. It produces several coded results from the simplest to the most complex one while preserving the contour information. It is therefore well suited for progressive coding and transmission applications.

Zhao Yuqian et.al [4], (2005) did an important work for object recognition of the human organs which is an important pre-processing step in medical image segmentation and 3D reconstruction. The proposed algorithm used to detect the edges of lungs CT images with salt-and-pepper noise which is more efficient for denoising and the edge detection than the usually used template-based edge detection algorithm such as LOG operator and Sobel edge detector and general morphological edge detection algorithm such as morphological gradient operation and dilation residue edge detector.

Ying-Tung Hsiao et.al [5], (2005) proposed an image segmentation algorithm by integrating mathematical morphological edge detector with region growing technique. The images are first enhanced by morphological closing operations, and then detect the edge of the image by morphological dilation residue edge detector. Moreover, this algorithm deploy growing seeds into the edge image that obtained by the edge detection procedure. By cross comparing the growing result and the detected edges, the partition lines of the image are generated. The proposed image segmentation approach is capable of producing promising segmentation results.

Jin zou et.al [6], (2006) proposed a simple, practical, effective color image edge detection algorithm, in order to find edges of the images having low contrast, which is based on diminishing width of fuzzy edge and morphological preprocessing, in contrast to traditional algorithms that are based on image enhancement technology. They give up the usual way to detect fuzzy edges through enhancing contrast among different regions, but to detect edge through shortening width of fuzzy edge.

Yuqian Zhao et.al [7], (2006) proposed a novel algorithm to detect complex edge features using multistructure mathematical morphological approach of eight different directions which are obtained by using morphological gradient algorithm that are not possible with conventional mathematical edge detection methods because they are only sensitive to image edge which has the same direction of structure element. The final edge result was got by using synthetic weighted method. The proposed algorithm is more efficient than the usually used single and symmetrical structuring element morphological edge detection operator and differential edge detection operators such as canny operator, LOG operator, Sobel operator and Prewitt operator. The detected edge is more pinpointed, integral and continual, and the edge information is more abundant. Moreover, the novel proposed algorithm can filter the noise more successfully than other operators.

Yuqiu Sun et.al [8], (2006) discussed an algorithm to improve the performance of an automatic target recognition and detection. The method applied in this paper is based on the features of mid-wave and long-wave infrared images that are decomposed into morphology pyramid respectively based on the multiscale mathematical morphology filters. Some features such as local maximum gray level and average gradient strength of every image are extracted at each level of morphology pyramid. Fuse dual band infrared images based on fusion rule put forward, and then reconstruct original image and detect target using contrast threshold segmentation. The experiment results show that dual band infrared images target detection algorithm based on Multiscale morphology algorithm is better than use mid-wave or long-wave infrared images detect targets alone. The performance of target detection can be improved by using mathematical morphology dualband fusion, instead of either mid-wave or long-wave infrared band alone.

Dawei Qi et.al [9], (2007) presented an algorithm of image edge detection based on mathematical morphology of omnidirectional multi-scale element. Mathematical morphology of omnidirectional multi-scale element is defined in order to suppress noise and adapt to different edge in the image. An approach of image edge detection based on morphology of omnidirectional multi-scale element is constructed by power adding combination of morphological operation. The results demonstrate that the method performs better not only in edge detection but also in noise-suppression than classical edge detection operator. The method possesses better real time feature, anti-interruption, and precise orientation feature.

Zhenhua Li et.al [10], (2007) proposed an edge detection algorithm, in order to eliminate the noise in the images which is based on multiscale morphologic edge tracking approach in which proper threshold parameters are selected. It is based on three steps; first, the morphology gradient is calculated under each scale. Second, after tracing the edge images acquired by sequential scale structuring elements, noises have been removed and useful edge informations have been reserved. And then, different weights are imposed to construct the outcome edge image.

Peijun Li and Hongtao Hu [11], (2007) proposed a method for high resolution image segmentation based on the extended morphological profiles, in which First, fundamental morphological vector operations (erosion and dilation) are defined by the extension, taking into account the spatial and spectral information in simultaneous fashion. Theoretical definitions of extended morphological operations are used in the formal definition of the concept of extended morphological profiles, which is constructed based on the repeated use of openings and closings by reconstruction with a structuring element (SE) of increasing size. Then, the morphological multiscale characteristic (MMC) of each pixel is gained through the derivative of the extended morphological profiles (DEMP). A modified method was proposed to obtain the right morphological characteristics of the pixel, which will be used for the final segmentation results. Finally, a simple region merging method based on the distance between two centroids of the neighbouring regions was adopted to further improve the segmentation result. The proposed approach is applied to high resolution Quick Bird multispectral images from urban, agricultural and forest areas for evaluation and comparison with existing methods, in terms of qualitative visual inspection and quantitative criteria. The proposed method demonstrated better performance than the classical morphological segmentation approaches.

Xueshun Wang et.al [12], (2008) proposed a new mathematical morphological double gradient algorithm in which structuring elements are chosen appropriately, in order to suppress noise and be adapted to different edges of images to detect the edge features of decayed wood images. It works better for images having noise as compared to other conventional algorithms, and does not only increase the magnitude of calculation, but also smooth out some slight edges.

Yanlei Xu et.al [13], (2008) discussed an algorithm based on multi-scale and multi-structuring order morphological transformation for gray-scale image edge detection. To noisy gray image, this method use order open and order close operation to filter the noise and use multi-scale to control the effect of edge detection, at the same time, choose different shape and size structure element not only to filter noise but can exactly achieve the edge. The results of this method proposed are proved good edge detection method, which is superior to the tradition edge detection operations and tradition order morphology edge detection operations in restraining noise and holding details.

Dawei Qi et.al [14], (2008) proposed an algorithm in which improved multi-scale and structuring element in mathematical morphology is used to detect log CT image with defect and the omnidirectional structure element provides a new method in log defect recognition by using X-ray computed tomography for log non-destructive testing. Mathematical morphology is used to detect the edge of the log image after acquiring the log image. X-ray computed tomography (CT) was applied to the detection of internal defects in the logs for the purpose of obtaining prior inform action that can be used to arrive at better log sawing decision. Compared with other non-destructive testing computed tomography has advantages such as higher penetrability, higher resolution, fast testing speed and visible testing result. Image edge detection is the main and important study direction of image management and computer vision. The algorithm adopted one improved edge detection method - omnidirectional and multi-scale structure element of mathematical morphology to settle the conflict between noise and extraction of detecting edge. The method possesses better real time feature, anti-interruption, and precise orientation feature.

Yanlei Xu et.al [15], (2008) presented a new method of image edge detection based on multiscale and multi-structuring element order morphology, after studying the edge detection methods of the noisy images and concentrating deeply on the theory of order morphology. The proposed method obtained clear and exact edge of the noisy images. To noisy gray image, this method use order open and order close operation to filter the noise and use multiscale to control the effect of edge detection, at the same time, choose different shape and size structure element not only to filter noise but can exactly achieve the edge.

Wenju Li and Ling Li [16], (2009) discussed a novel approach for vehicle-logo location based on edge detection and morphological filter in order to avoid problems such as car stealing, traffic jam, traffic accidents by using automatic vehicle identification technology in which three steps are defined to calculate the accurate location of vehicle-logo: firstly, the approximate location of vehicle-logo region is determined by the prior knowledge about the position of the vehicle logo; secondly, a method for texture measure is defined in order to recognize the texture of the vehicle-logo background; then vertical and horizontal edge detection are implemented for the vehicle logo background with the horizontal texture and vertical texture respectively to suppress the texture of background which is equivalent to noise ; finally, position for the vehicle-logo is located accurately by using mathematical morphology filters. The proposed algorithm is very effective for not only vehicle logos with horizontal and vertical textures but also vehicle logos with small meshy texture.

Liu Yucheng and Liu Yubin [17], (2009) discussed a new image segmentation algorithm, based on fusion technology which enhances the quality of image segmentation, based on fuzzy mathematical morphology, aiming at the noise and exiguity catchment basin of image that result in over-segmentation phenomenon when adopt traditional watershed algorithm to segment an image. The method can be applied to many practical areas, such as military objective detective, agriculture pest image recognition, resource protecting of ocean, irrigation, and environment and so on. So it has important practical engineering application value and theoretic significance. This method adopted firstly opening-closing algorithm based on fuzzy mathematical morphology to smooth the image and to compute its gradient operators based on basic morphology. Furthermore, the method segmented the gradient image based on fuzzy mathematical morphology to get the results. This algorithm not only can eliminate over-segmentation phenomenon that resulted from traditional mathematical morphology segmentation algorithm, realize goal full segmentation from background, but also can save image detail perfectly.

There are many available techniques discussed so far for detecting image edge features and segmenting different types of images. Some of them are suppressing noise; some are using a thresholding technique to segment them; some are using region growing method to segment images. Every technique has its own merits and demerits such as if a technique is using a thresholding method then there are many methods of thresholding images, so by each method results of segmentation is different. Region growing method of segmentation is perfect for images having fewer noises but these methods take a lot of time for segmenting complex images depending upon number of iterations. Edge based techniques are also not as efficient as they detect noises as edges and in some cases also detect fake edges in images. So there is not any perfect method that segment complex images as well noisy images with greater efficiency.

2.1 Problem Formulation and Proposed Solution

Problem formulation is divided in two parts: firstly segmenting ultrasonic images by utilizing morphology and secondly segmenting images having similar objects followed by counting those objects by using morphology. After studying so many available methods of image segmentation, a method is proposed for segmenting ultrasound images and segmentation followed by counting the objects in the images in order to separate them which utilize mathematical morphology. The proposed algorithm is compared with other image segmentation techniques like edge based techniques for example CANNY, LOG, PREWITT, SOBEL, ROBERTS methods of segmenting images and region growing techniques. After experiments it is found that the proposed method is efficient, reliable, consumes less time and gives result without noise as well as works on complex images.

CHAPTER-3

MORPHOLOGY

3.1 Mathematical Morphology

For the purposes of image analysis and pattern recognition there is always a need to transform an image into another better represented form. During the past five decades image-processing techniques have been developed tremendously and mathematical morphology in particular has been continuously developing because it is receiving a great deal of attention because it provides a quantitative description of geometric structure and shape and also a mathematical description of algebra, topology, probability, and integral geometry. Mathematical morphology is extremely useful in many image processing and analysis applications.

Mathematical morphology denotes a branch of biology that deals with the forms and structures of animals and plants. It analyzes the shapes and forms of objects. In computer vision, it is used as a tool to extract image components that are useful in the representation and description of object shape. It is mathematical in the sense that the analysis is based on set theory, topology, lattice algebra, function, and so on. [18]

3.2 Basic Concepts from Set Theory

Let Z be a set of integers. The sampling process used to produce the digital image may be viewed as a partitioning in the $x\,y$ -plane in to a grid, with coordinates of the centre of each grid being a pair of elements from the Cartesian product Z^2. In the terminology of set theory the function $f\,(x, y)$ is said to be a digital image if $(x$, y) are integers from Z^2 and f is a mapping that assigns an intensity value in the set of real numbers, R, to each distinct pair of coordinates (x ,y). If the elements of R also are integers, a digital image then becomes a two-dimensional function, whose coordinates and the amplitude values are integers.

Let A be a set in Z^2, the elements of which are pixel coordinates (x, y). If $w = (x, y)$ is an element of A, then it is written

$$w \in A$$

Similarly, if w is not an element of A, it is written

$$w \notin A$$

A set of *B* pixels satisfying a particular condition is written as

$$B = \{w \mid \text{condition}\}$$

For example, the set of all pixel coordinates that do not belong to A, denoted by A^c, is given by **complement** of A

$$A^c = \{w \mid w / A\}$$

The **union** of two sets, denoted by

$$C = A \cup B$$

is the set of all elements that belong to either A or B, or both.

Similarly, the **intersection** of two sets A and B is the set of elements that belong to both sets, denoted by

$$C = A \cap B$$

The **difference** of sets A and B, denoted by A – B, is the set of all elements that belong to A but not to B:

$$A - B = \{w \mid w \in A, w / B\}$$

Fig3.1 illustrating these set operations.

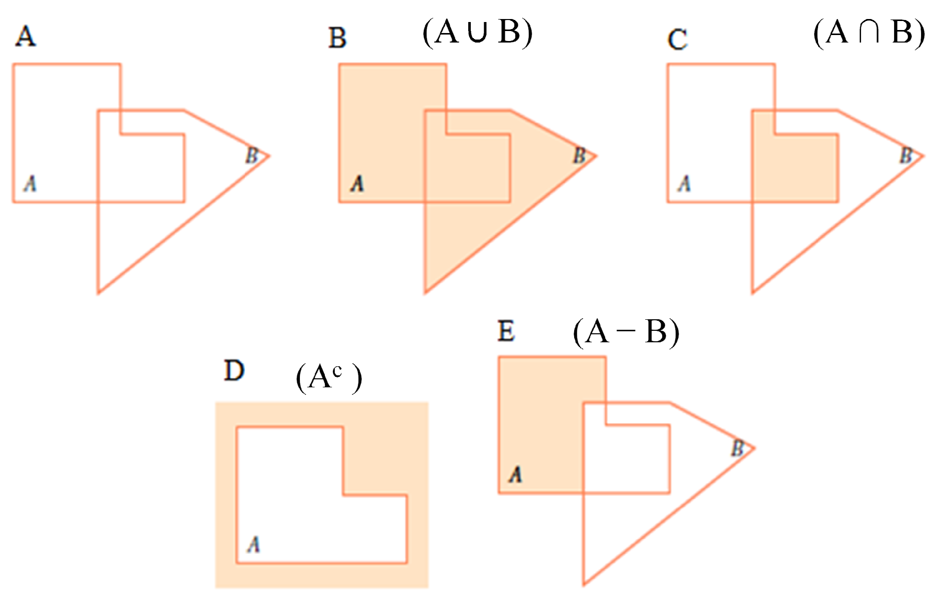


Fig-3.1 Basic concepts from set theory

In addition to the basic operations, morphological operations often require two operations that are specific to sets whose elements are pixel coordinates. The **reflection** of set B, denoted by B^, is defined as

$$\hat{B} = \{w | w = -b, \text{ for } b \in B\}$$

The **translation** of set A by $z = (z_x, z_y)$, denoted by $(A)_z$, is defined as

$$A_z = \{c | c = a + z, \text{ for } a \in A\}$$

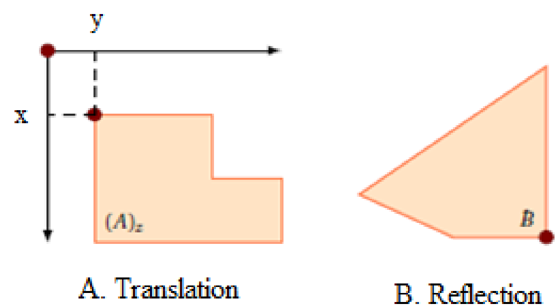


Fig-3.2 Translation and Reflection

The union and intersection can be exchanged by logical duality (DeMorgan's Law) as

$$(A \cap B)^c = A^c \cup B^c$$

and

$$(A \cup B)^c = A^c \cap B^c$$

The cardinality of *A*, indicating the area of the object or the total number of elements in the object set, is denoted as #*A*, |*A*|, or *Card*(*A*). If a set *A* has *n* distinct elements for some natural number *n*, the cardinality of *A* is *n.* [18]

3.3 Binary Images

In binary images, each picture element (pixel) is a tuple (or a 2-dimensional vector) of the (*x*, *y*)-plane. The binary image is a matrix containing object pixels of value 1which can also be called as foreground and nonobject pixels of value 0 which can also be called as background. The binary image can be treated as a set whose elements are composed of all the pixels of value 1. In gray scale images, each element is a triple (or a 3-dimensional vector); that is, (x, y) plus the third dimension which is known as the pixel's intensity value. Sets in higher dimensional spaces can contain other image attributes.

Set operations such as union and intersection can be applied directly to the binary image sets. For example, if A and B are binary images, then C = A U B is also a binary image, where a pixel in C is a foreground pixel if either or both of the corresponding pixels in A and B are foreground pixels. In the first view, that of a function C is given by:

$$C(x,y) = \begin{cases} 1 & if\ either\ A(x,y) or\ B(x,y) is\ 1, or\ if\ both\ are\ 1 \\ 0 & otherwise \end{cases}$$

.

Using the set view, on the other hand, C is given by:

C = {(x, y) | (x, y∈ A or (x, y) ∈B or (x, y) ∈ (A and B)}

3.4 Structuring Element

The basic idea in binary morphology is to probe an image with a simple, pre-defined shape, drawing conclusions on how this shape fits or misses the shapes in the image. This simple "probe" is called structuring element, which is also a binary image (i.e., a subset of the space or grid). Mathematical morphology involves geometric analysis of shapes and textures in images. An image can be represented by a set of pixels. Morphological operators work with two images. The image being processed is referred to as the *active image*, and the other image, being a kernel, is referred to as the *structuring element*. Each structuring element has a designed shape, which can be thought of as a probe or a filter of the active image. The active image can be modified by probing it with the structuring elements of different sizes and shapes.

Mathematical morphology use morphological structuring elements in order to measure and distill corresponding shape of an image to attain objective of analysis, to reduce image data and to keep basic shape character. [17]

3.5 Morphological Operators

There are 4 morphological operators listed below.

3.5.1 Dilation-

Dilation is an operation that grows or thickens objects in a binary image. The specific manner and the extent of this thickening are controlled by structuring element. Dilation is a process that translates the origin of the structuring element throughout the domain of the image and checks to see whether it overlaps with 1-valued pixels. The output image is 1 at each location of the origin of the structuring element if the structuring element overlaps at least one 1-valued pixel in the input image. Mathematically, dilation is defined in terms of set operations. The dilation of A by B, denoted by $A \oplus B$, is defined as

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

Where Φ is the empty set and B is the structuring element. In words, the dilation of A by B is the set consisting of all the structuring element origin locations where the reflected and translated B overlaps at least some portion of A. Dilation is commutative, i.e., $A \oplus B = B \oplus A$. It is a convention in image processing to let the first operand of $A \oplus B$ be the image and the second operand is the structuring element, which is usually much smaller than the image. It is typically applied to binary images, but there are versions that work on gray scale images. The basic effect of the operator on a binary image is to gradually enlarge the boundaries of regions of foreground pixels (i.e. white pixels, typically). Thus areas of foreground pixels grow in size while holes within those regions become smaller.

- **Working of dilation**

The dilation operator takes two pieces of data as inputs. The first is the image which is to be dilated. The second is a (usually small) set of coordinate points known as a structuring element (also known as a kernel). It is this structuring element that determines the precise effect of the dilation on the input image.

As an example of binary dilation, suppose that the structuring element is a 3×3 square, with the origin at its centre, as shown in matrix. In this matrix, foreground pixels are represented by 1's.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Set of coordinate points-{(-1,-1) (0,-1) (1,-1) (-1,0) (0,0) (1,0) (-1,1) (0,1) (1,1)}.

To compute the dilation of a binary input image by this structuring element, the structuring element superimposed on top of the input image for each foreground pixel so that the origin of the structuring element coincides with the input pixel position. If *at least one* pixel in the structuring element coincides with a foreground pixel in the image underneath, then the input pixel is set to the foreground value. If all the corresponding pixels in the image are background, however, the input pixel is left at the background value.

Most implementations of this operator expect the input image to be binary, usually with foreground pixels at pixel value 255, and background pixels at pixel value 0. Such an image can often be produced from a gray scale image using thresholding. The structuring element may have to be supplied as a small binary image, or in a special matrix format, or it may simply be hardwired into the implementation, and not require specifying at all.

The effect of a dilation using this structuring element on a binary image is shown in Fig3.3.

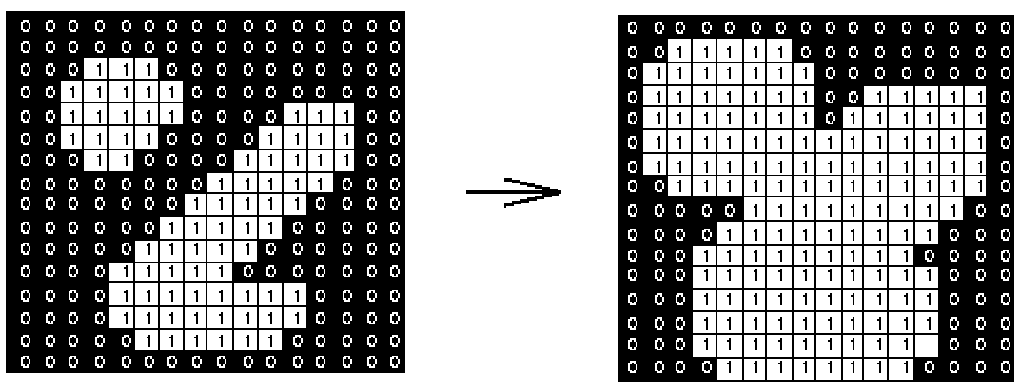


Fig-3.3 Process of dilation

The 3×3 square is probably the most common structuring element used in dilation operations, but others can be used. A larger structuring element produces a more extreme dilation effect, although usually very similar effects can be achieved by repeated dilations using a smaller but similarly shaped structuring element. With larger structuring elements, it is quite common to use an approximately disk shaped structuring element, as opposed to a square one.

Gray scale dilation with a flat disk shaped structuring element will generally brighten the image. Bright regions surrounded by dark regions grow in size, and dark regions surrounded by bright regions shrink in size. Small dark spots in images will disappear as they are `filled in' to the surrounding intensity value. Small bright spots will become larger spots. The effect is most marked at places in the image where the intensity changes rapidly and regions of fairly uniform intensity will be largely unchanged except at their edges.

3.5.2 Erosion

Erosion is the morphological dual to dilation. Erosion shrinks or thins in a binary image. As in dilation, the manner and extent of shrinking is controlled by the structuring element. The output of erosion has a value 1 at each location of the origin of the structuring element, such that the structuring element overlaps *only* 1-valued pixels of the input image.

The mathematical definition of erosion is similar to that of dilation. The erosion of *A* by *B*, denoted by *A* Θ *B*, is defined as

$$A \ \Theta \ B = \{z| \ B_z \cap A^c \neq \Phi\}.$$

In other words, erosion of *A* by *B* is the set of structuring element origin locations where the translated *B* has no overlap with the background of *A*.

It is typically applied to binary images, but there are versions that work on gray scale images. The basic effect of the operator on a binary image is to erode away the boundaries of regions of foreground pixels (i.e. white pixels, typically). Thus areas of foreground pixels shrink in size, and holes within those areas become larger.

- **Working of erosion**

The erosion operator takes two pieces of data as inputs. The first is the image which is to be eroded. The second is a set of coordinate points known as a structuring element. It is this structuring element that determines the precise effect of the erosion on the input image.

As an example of binary erosion, suppose that the structuring element is a 3×3 square, with the origin at its center as shown below. 1's represent foreground pixels here.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Set of coordinate points-{(-1,-1) (0,-1) (1,-1) (-1,0) (0,0) (1,0) (-1,1) (0,1) (1,1)}.

To compute the erosion of a binary input image by this structuring element, the structuring element superimposed on top of the input image for each foreground pixel so that the origin of the structuring element coincides with the input pixel coordinates. If for *every* pixel in the structuring element, the corresponding pixel in the image underneath is a foreground pixel, then the input pixel is left as it is. If any of the corresponding pixels in the image are background, however, the input pixel is also set to background value.

Erosion is the *dual* of dilation, *i.e.* eroding foreground pixels is equivalent to dilating the background pixels.

Most implementations of this operator will expect the input image to be binary, usually with foreground pixels at intensity value 255, and background pixels at intensity value 0. Such an image can often be produced from a gray scale image using thresholding.

The effect of erosion using this type of structuring element on a binary image is shown in Fig3.4.

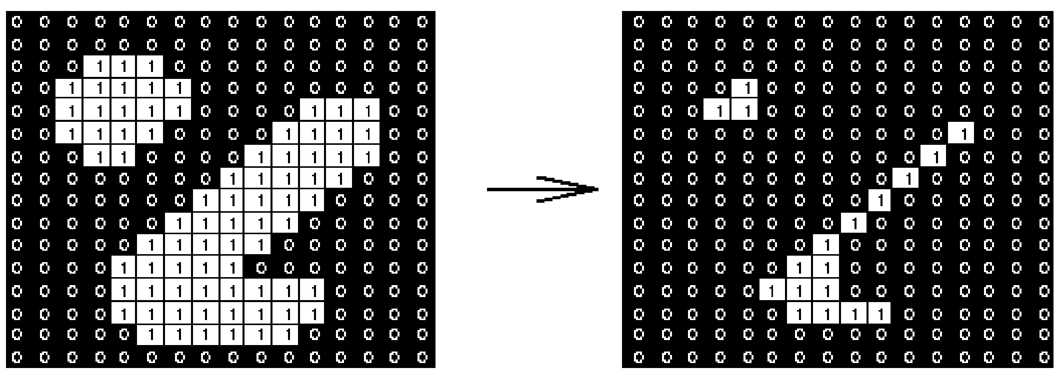


Fig-3.4 Process of erosion

The 3×3 square is probably the most common structuring element used in erosion operations, but others can be used. A larger structuring element produces a more extreme erosion effect, although usually very similar effects can be achieved by repeated erosions using a smaller similarly shaped structuring element. With larger structuring elements, it is quite common to use an approximately disk shaped structuring element, as opposed to a square one.

Erosions can be made directional by using less symmetrical structuring elements. For example, a structuring element that is 10 pixels wide and 1 pixel high will erode in a horizontal direction only. Similarly, a 3×3 square structuring element with the origin in the middle of the top row rather than the center will erode the bottom of a region more severely than the top.

Grayscale erosion with a flat disk shaped structuring element will generally darken the image. Bright regions surrounded by dark regions shrink in size, and dark regions surrounded by bright regions grow in size. Small bright spots in images will disappear as they are eroded away down to the surrounding intensity value, and small dark spots will become larger spots. The effect is most marked at places in the image where the intensity changes rapidly, and regions of fairly uniform intensity will be left more or less unchanged except at their edges.

3.5.3 Opening

The *morphological opening* of A by B, denoted by A *o* B, is simply erosion of A by B followed by the dilation of the result by *B*.

$$A \, o \, B = (A \ominus B) \oplus B.$$

An alternative mathematical formulation of opening is

$$A \, o \, B = \cup \{(B_z) | (B_z) \text{ is a subset of } A\}.$$

Where $\cup \{.\}$ denotes the union of all sets inside braces, and the notation ($C \subseteq D$) means that *C* is a subset of *D*. This formulation has a simple geometric interpretation: A *o* B is the union of all translations of *B* that fit entirely within *A*.

Morphological opening removes completely regions of an object that cannot contain the structuring element, smoothes the object contours, breaks thin connections, and removes the protrusions.

Opening and closing are two important operators from mathematical morphology. They are both derived from the fundamental operations of erosion and dilation. Like those operators they are normally applied to binary images, although there are also gray level versions. The basic effect of an opening is somewhat like erosion in that it tends to remove some of the foreground (bright) pixels from the edges of regions of foreground pixels. However it is less destructive than erosion in general. As with other morphological operators, the exact operation is determined by a structuring element. The effect of the operator is to preserve *foreground* regions that have a similar shape to this structuring element, or that can completely contain the structuring element, while eliminating all other regions of foreground pixels.

- **Working of opening**

An opening is defined as erosion followed by a dilation *using the same structuring element for both operations*. The opening operator requires two inputs: an image to be opened, and a structuring element.

Opening is the *dual* of closing, *i.e.* opening the foreground pixels with a particular structuring element is equivalent to closing the background pixels with the same element.

While erosion can be used to eliminate small clumps of undesirable foreground pixels, quite effectively, it has the big disadvantage that it will affect *all* regions of foreground pixels indiscriminately. Opening gets around this by performing both erosion and a dilation on the image. The effect of opening can be quite easily visualized. Imagine taking the structuring element and sliding it around *inside* each foreground region, without changing its orientation. All pixels which can be covered by the structuring element with the structuring element being entirely within the foreground region will be preserved. However, all foreground pixels which cannot be reached by the structuring element without parts of it moving out of the foreground region will be eroded away. After the opening has been carried out, the new boundaries of foreground regions will all be such that the structuring element fits inside them, and so further openings with the same element have no effect. The property is known as *idempotence*. The effect of an opening on a binary image using a 3×3 square structuring element is illustrated in Fig3.5.

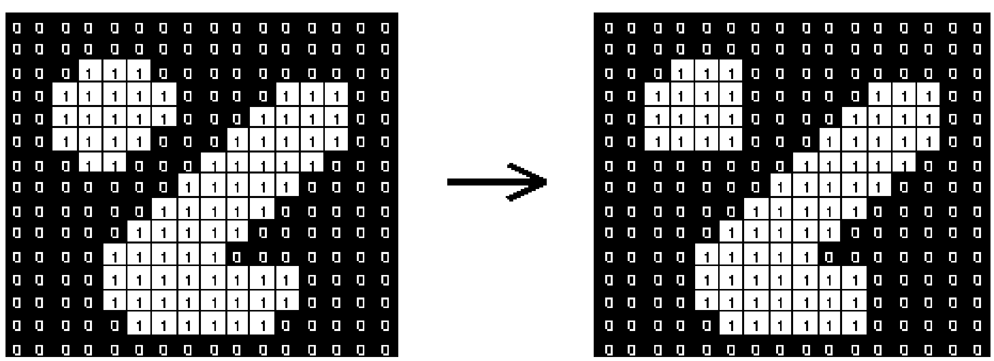


Fig-3.5 Process of opening

As with erosion and dilation, it is very common to use this 3×3 structuring element. The effect in the above figure is rather subtle since the structuring element is quite compact and so it fits into the foreground boundaries quite well even before the opening operation. To increase the effect, multiple erosions are often performed with this element followed by the same number of dilations. This effectively performs an opening with a larger square structuring element.

3.5.4 Closing

The morphological closing of A by B, denoted by A • B, is a dilation followed by erosion:

$$A \bullet B = (A \oplus B) \ominus B.$$

Geometrically, A • B is the complement of the union of all translations of B they do not overlap A.

Like opening, morphological closing smoothes the contours of the objects. Unlike opening, however, it generally joins narrow breaks, filling long thin gulfs, and fills holes smaller than the structuring element.

Closing is an important operator in the field of mathematical morphology. Like its dual operator opening, it can be derived from the fundamental operations of erosion and dilation. Like those operators it is normally applied to binary images, although there are gray level versions. Closing is similar in some ways to dilation in that it tends to enlarge the boundaries of foreground (bright) regions in an image (and shrink background color holes in such regions), but it is less destructive of the original boundary shape. As with other morphological operators, the exact operation is determined by a structuring element. The effect of the operator is to preserve *background* regions that have a similar shape to this structuring element, or that can completely contain the structuring element, while eliminating all other regions of background pixels.

- **Working of closing**

Closing is opening performed in reverse. It is defined simply as dilation followed by erosion *using the same structuring element for both operations*. The closing operator requires two inputs: an image to be closed and a structuring element.

Gray level closing consists straightforwardly of a gray level dilation followed by gray level erosion.

Closing is the dual of opening, *i.e.* closing the foreground pixels with a particular structuring element, is equivalent to closing the background with the same element.

One of the uses of dilation is to fill in small background color holes in images. One of the problems with doing this, however, is that the dilation will also distort *all* regions of pixels indiscriminately. By performing erosion on the image after the dilation, *i.e.* a closing, some of this effect is reduced. The effect of closing can be quite easily visualized. Imagine taking the structuring element and sliding it around *outside* each foreground region, without changing its orientation. For any background boundary point, if the structuring element can be made to touch that point, without any part of the element being inside a foreground region, then that point remains background. If this is not possible, then the pixel is set to foreground. After the closing has been carried out the background region will be such that the structuring element can be made to cover any point in the background without any part of it also covering a foreground point, and so further closings will have no effect. This property is known as idempotence. The effect of a closing on a binary image using a 3×3 square structuring element is illustrated in Fig3.6.

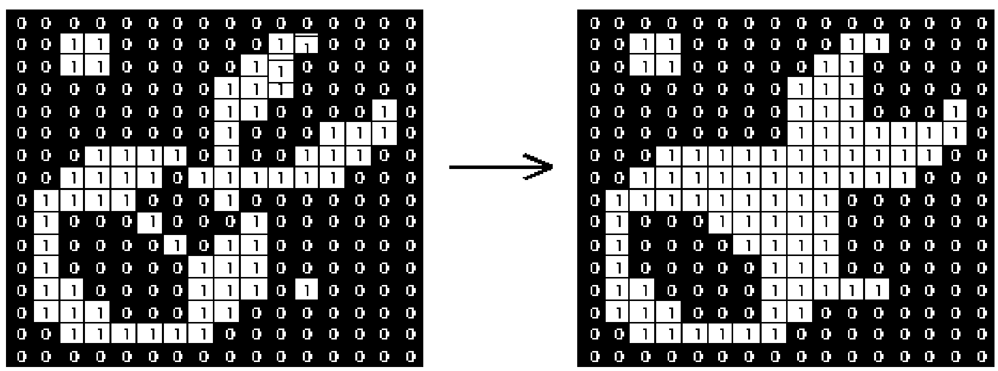


Fig-3.6 Process of closing

As with erosion and dilation, this particular 3×3 structuring element is the most commonly used, and in fact many implementations will have it hardwired into their code, in which case it is obviously not necessary to specify a separate structuring element. To achieve the effect of a closing with a larger structuring element, it is possible to perform multiple dilations followed by the same number of erosions.
Closing can sometimes be used to selectively fill in particular background regions of an image. Whether or not this can be done depends upon whether a suitable structuring element can be found that fits well inside regions that are to be preserved, but doesn't fit inside regions that are to be removed.

3.6 Applications of Mathematical Morphology

Mathematical morphology is used in various applications. Some applications of morphology are described as following.

- 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

CHAPTER-4

METHODOLOGY

4.1 Image File Formats

A Variety of image file formats are available at present. Like TIFF, JPEG, GIF, PNG etc. We mainly using TIFF, PNG and JPEG in our work, these are explained as follows:

4.1.1 TIFF- stands for Tagged Image File Format. Its extension is recognized both as 'tif' and 'tiff'. These are the file formats used for storing images, including photographs and line art. It grew to accommodate gray scale images, then color images. Today, it is a popular format for high-color-depth images, along with JPEG.

4.1.2 JPEG- stands for Joint Photographic Experts Group. It has '.jpg', 'jpeg' as the allowed extensions. It is the most common format for storing and transmitting photographic images on the World Wide Web and is a commonly used method of compression for photographic images.

4.1.3 PNG- stands for Portable Network Graphics (PNG) is a bitmapped image format that employs lossless data compression. PNG was created to improve upon and replace GIF (Graphics Interchange Format) as an image-file format not requiring a patent license. PNG supports palette-based (palettes of 24-bit RGB or 32-bit RGBA colors), gray scale, gray scale with alpha, RGB, or RGBA images. PNG was designed for transferring images on the Internet, not for print graphics, and so does not support non-RGB color spaces.

4.2 Images types

Four types of images are available:

4.2.1 Intensity images

An intensity image is a data matrix whose values represent intensities within some range. For the elements of class uint8 or class uint16 of an intensity image, the integer values lie between [0,255] and [0, 65535], respectively. And if the image is of class double, then the associated values are floating-point numbers. Conventionally, the intensity images with scaled, class double data type have a range of [0, 1]. In MATLAB, an intensity image 36 is stored as a single matrix, with each element of the matrix corresponding to one image pixel.

4.2.2 Binary images

A binary image is a logical array of 0s and 1s. Pixels with the value 0 are displayed as black; pixels with the value 1 are displayed as white. In MATLAB, a binary image must be of class logical that is why the intensity images that happen to contain only 0's and 1's are not taken as binary images.

4.2.3 Indexed images

An indexed image consists of a data matrix, X, and a colormap matrix termed as "map". The "map" is an m-by-3 array of class double containing floating-point values in the range [0, 1]. Its every row specifies the red, green, and blue components of a single color. For these images pixel values are directly mapped to their corresponding colormap values. The color of each image pixel is determined by using the corresponding values of X as an index into map. The value 1 points to the first row in map, the value 2 points to the second row, and so on.

4.2.4 RGB images

An RGB image is also referred as a true- color image. In MATLAB these images are stored in the form of an m-by-n-by-3 data array that defines red, green, and blue components for each individual pixel. The color of each pixel is determined by the combination of the red, green and blue intensities stored in each color plane at the pixel's location. Graphics file formats store RGB images as 24-bit images, where the red, green and blue components are 8 bits each. An RGB array can be of class double, uint8, or uint16. In an RGB array of class double, each color component is a value between 0 and 1. A pixel whose color components are (0,0,0) is displayed as black, and a pixel whose color components are (1,1,1) is displayed as white. The three color components for each pixel are stored along the third dimension of the data array.

In this work we are mainly dealing with intensity images that are also known as gray scale images.

4.3 Methodology

Methodology is divided in two parts.

- Working Algorithm using MATLAB for segmentation of ultrasonic images
- Working Algorithm using MATLAB for segmentation followed by counting of images

4.3.1 Working Algorithm using MATLAB for segmentation of ultrasonic images

This working Algorithm is the way of doing our work. This is a step by step procedure that how we implement the images and how we evaluate the segmentation and how our method is compared with other traditional method. The main algorithm, followed in order to fulfil the aim of our work, is as follows:

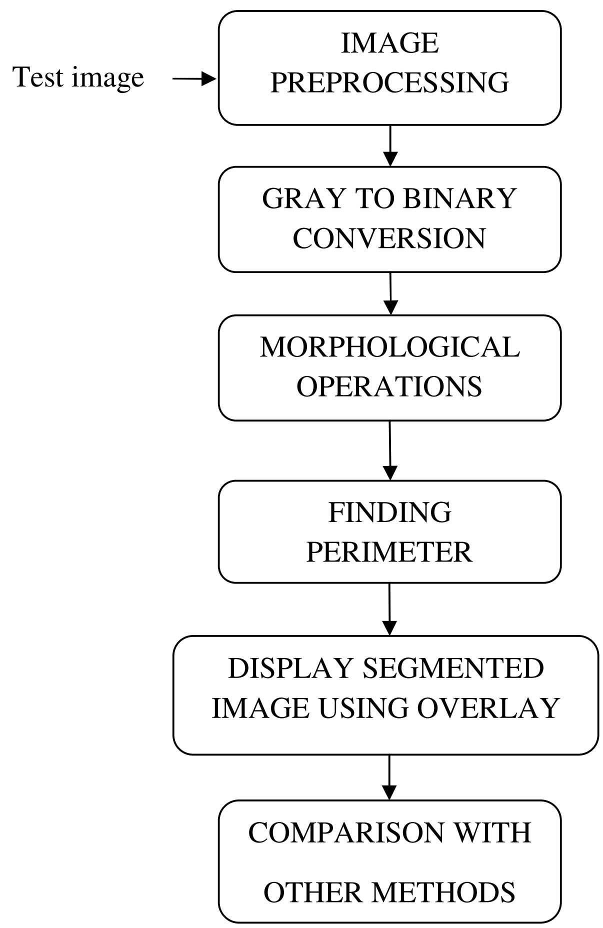


Fig-4.1 Block diagram of Working Algorithm1

4.3.1.1 Image pre-processing

Image preprocessing is the very first stage of any image processing system. It is done in order to correct some unwanted characteristics of image such as noise which is added in the image during image acquisition. Likewise, brightness and contrast of the image may require improvement. Commonly, too, coordinate transformations are needed to restore geometrical distortions introduced during image formation.

- **Increase contrast**

Contrast adjustment isn't usually necessary for segmentation, but it can help to see and understand the image data better. For fairly low-contrast images, it might help. The contrast of an image is adjusted by automatic contrast adjustment technique. The contrast, especially in homogeneous areas, can be limited to avoid amplifying any noise that might be present in the image.

- **Filter the image**

Adequate filtering procedures must be applied in order to distinguish the objects of interest from other objects and the background. A whole chain of processing steps is necessary to analyze and identify objects. Multidimensional image also has to be filtered.

4.3.1.2 Conversion to binary image

For applying morphology the gray image is converted to binary image in order to make the processing easier by a method in which the threshold is selected by using graythresh function which uses Otsu's method. Graythresh has a value of threshold between 0 and 1.

4.3.1.3 Applying morphology

Following morphological operations are performed on the image.

- Filling interior gaps.
- Perform erosion in order to smooth the image by disk structuring element.
- Perform opening by disk structuring element in order to remove regions of an object that cannot contain the structuring element, to smooth the object contours, to breaks thin connections, and to remove the protrusions.
- Finding perimeter.

4.3.1.4 Displaying segmented object

Overlay the perimeter on the original test image so that we could get the segmentation on the initial image.

4.3.1.5 Comparisons

Results obtained with our method are compared with other traditional techniques of segmenting an image. These techniques are segmentation using canny edge detector, sobel edge detector, Laplacian of Gaussian edge detector, prewitt edge detector etc.

4.3.2 Working Algorithm using MATLAB for counting and segmentation of images-

This working algorithm is the way of doing counting and segmentation and a step by step procedure for our work.

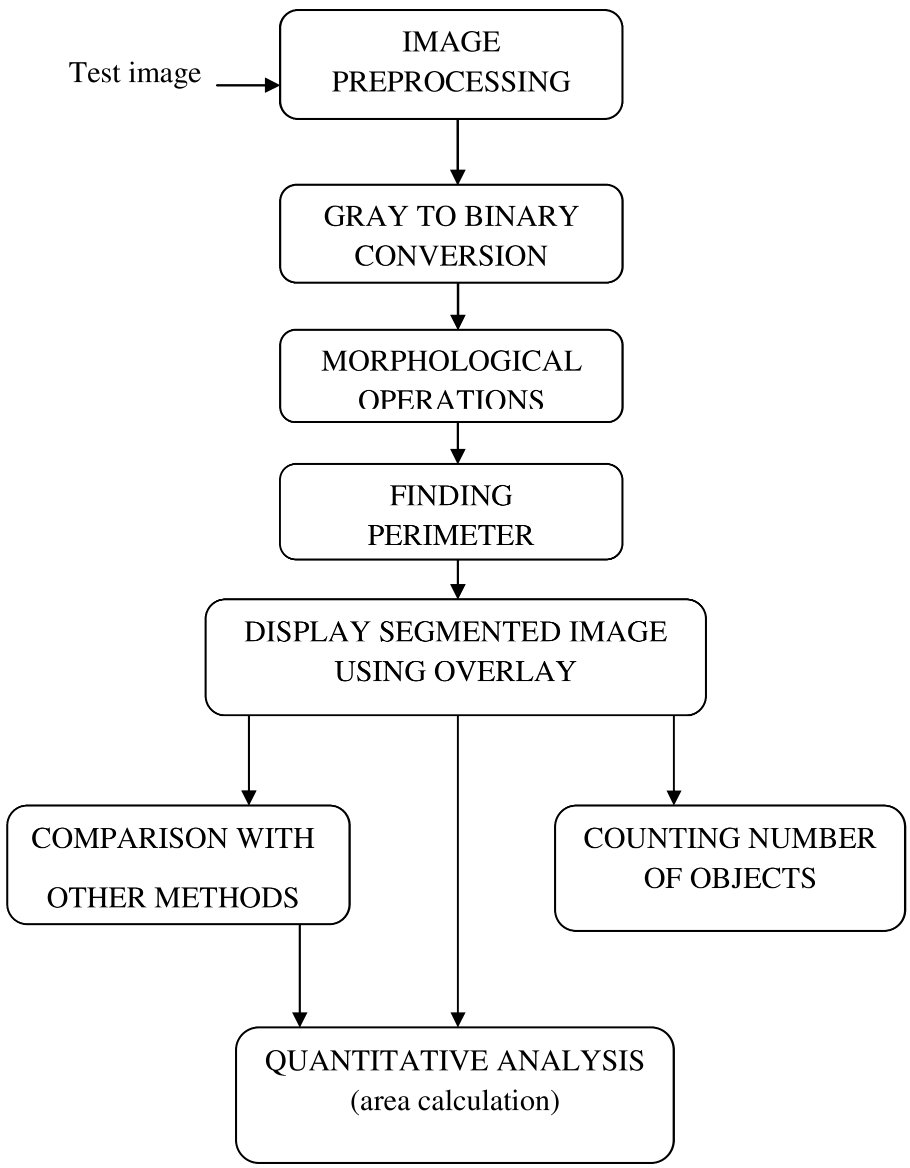


Fig-4.2 Block diagram of working algorithm2

4.3.2.1 Image preprocessing

Image preprocessing is the very first stage of any image processing system. It is done in order to correct some unwanted characteristics of image such as noise which is added in the image during image acquisition. Likewise, brightness and contrast of the image may require improvement. Commonly, too, coordinate transformations are needed to restore geometrical distortions introduced during image formation.

- **Increase contrast**

Contrast adjustment isn't usually necessary for segmentation, but it can help to see and understand the image data better. For fairly low-contrast images, it might help. The contrast of an image is adjusted by automatic contrast adjustment technique. The contrast, especially in homogeneous areas, can be limited to avoid amplifying any noise that might be present in the image.

- **Filter the image**

Adequate filtering procedures must be applied in order to distinguish the objects of interest from other objects and the background. A whole chain of processing steps is necessary to analyze and identify objects. Multidimensional image also has to be filtered.

4.3.2.2 Conversion to binary image

For applying morphology the gray image is converted to binary image in order to make the processing easier by a method in which the threshold is selected by using graythresh function which uses Otsu's method. Graythresh has a value of threshold between 0 and 1.

4.3.2.3 Applying morphology

Following morphological operations are performed on the image.

- Filling interior gaps.
- Perform opening in order to remove regions of an object that cannot contain the structuring element, to smooth the object contours, to breaks thin connections, and to remove the protrusions.
- Filling interior gaps again so that there is not any hole present in the image.
- Perform erosion in order to smooth the image by disk structuring element.
- Perform opening again so that object contours may be smoothen.
- Finding perimeter.

4.3.2.4 Displaying segmented object

Overlay the perimeter on the original test image so that we could get the segmentation over the initial image.

4.3.2.5 Comparisons

Results obtained with our method are compared with other traditional techniques of segmenting an image. These techniques are segmentation using canny edge detector, sobel edge detector, Laplacian of Gaussian edge detector, prewitt edge detector etc.

4.3.2.6 Counting number of objects

The number of objects are counted by using morphology and displayed. It separates each object from each other if they joined and count each object efficiently.

4.3.2.7 Quantitative analysis

A numerical analysis is carried out by calculating area of 10 objects by available methods. And a percentage error is calculated by finding the ratio of segmented area to the actual area.

CHAPTER-5

RESULTS AND DISCUSSIONS

5.1 Introduction

Segmentation is an image specific process i.e. segmenting an image is depends on application, imaging modality and type of image part to be studied. Here we are segmenting two types of images: firstly images of ultrasound and secondly images with similar objects in order to separate them.

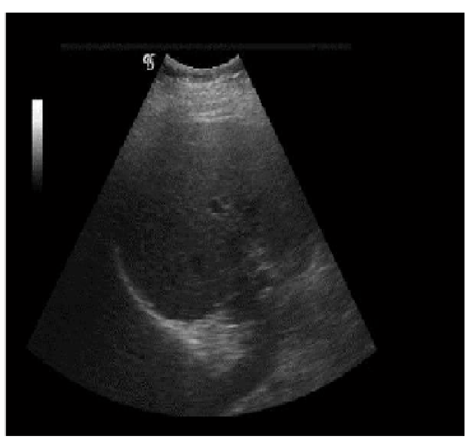
This chapter is divided in following two parts on the basis of images used:

- Segmentation of ultrasound images and
- Segmentation followed by counting of objects in images

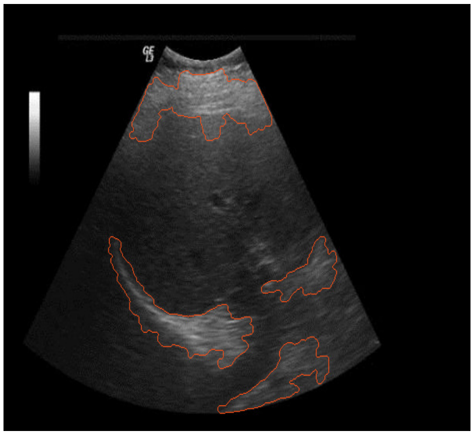
5.2 Segmentation of ultrasound images

25 ultrasound images are segmented using morphological approach and all images are compared with the traditional techniques of segmentation e.g. CANNY, LOG, SOBEL, PREWITT and ROBERTS techniques of segmentation.

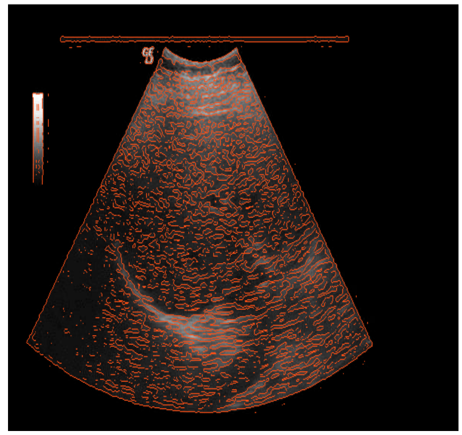
Following test images are taken from "KUKA DIAGNOSTIC CENTRE LUDHIANA" and their results are shown.



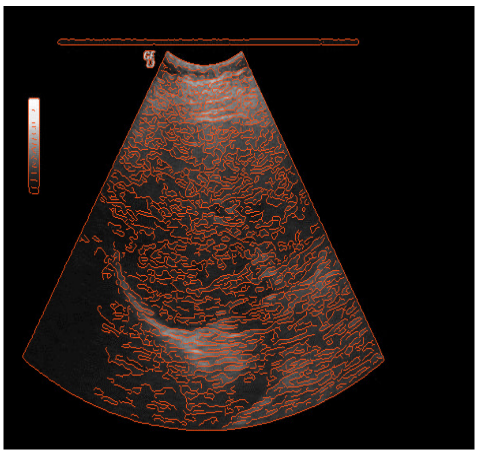
(A) Test Image



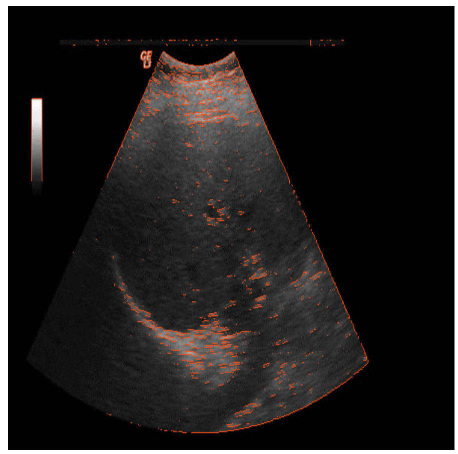
(B) Proposed Method



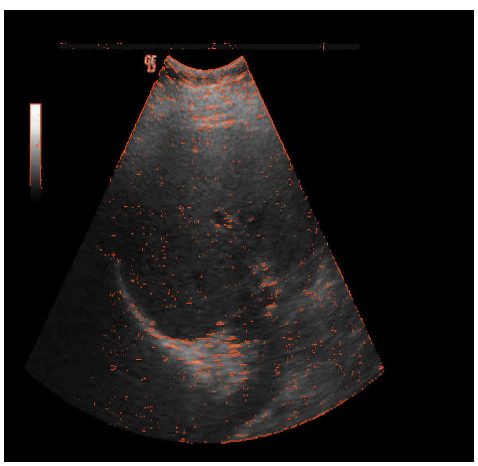
(C) Log Method



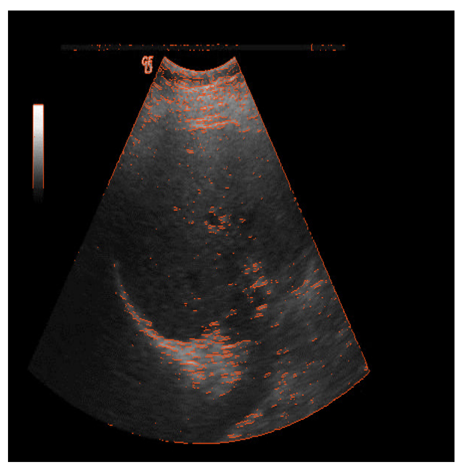
(D) Canny Method



(E) Prewitt Method



(F) Roberts Method

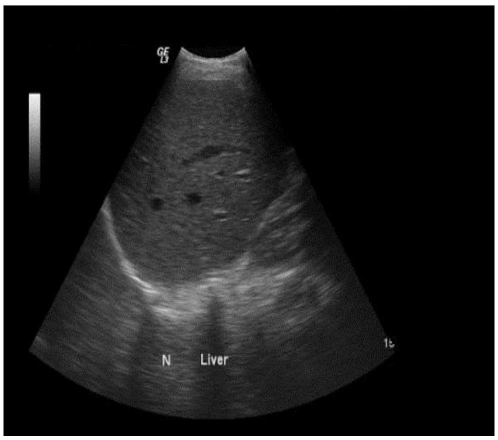


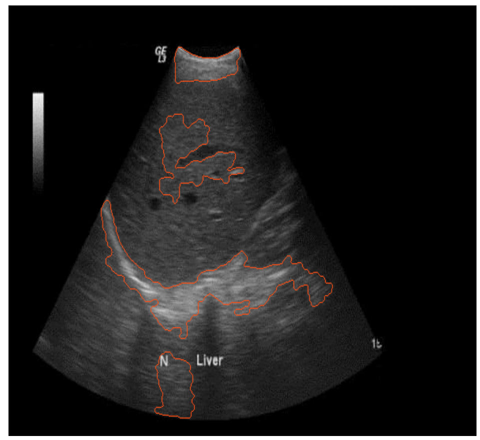
(G) Sobel Method

Fig-5.1 Segmentation of Test Image (1) By Different Methods

As shown from the above figure proposed method is segmenting the ultrasound image efficiently. All the methods of image segmentation are not working efficiently on ultrasound image. As obvious only our algorithm is segmenting the desired portions of the image while other methods are not segmenting the desired portions.

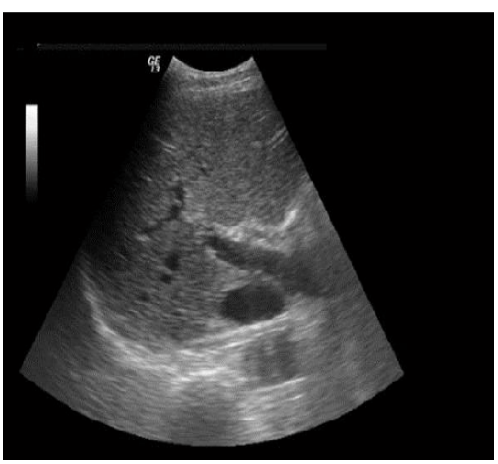
Other 24 ultrasound images are segmented in this way by using morphology. Some of the test images and their results after segmentation by proposed method are shown in figures below.

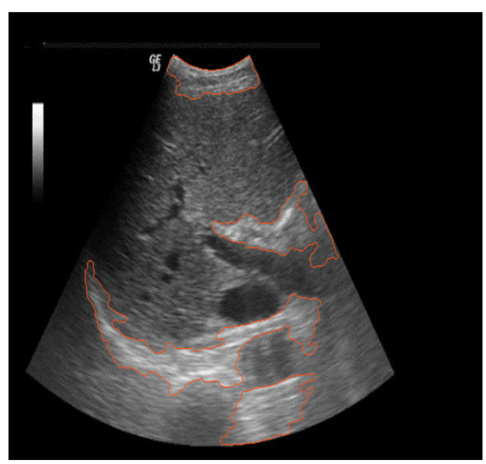




(A) Test Image (B) Segmented Image

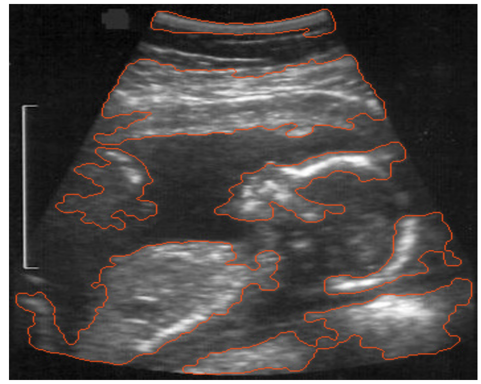
Fig-5.2 Segmentation of Test Image (2) By Proposed Method





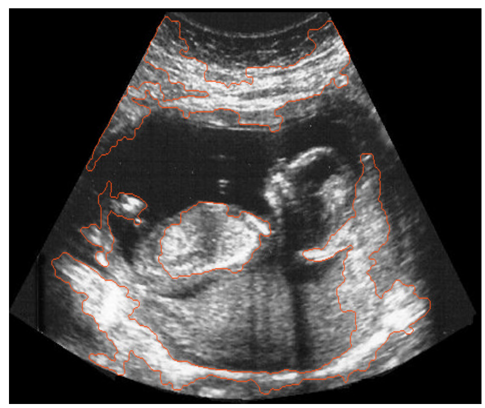
(A) Test Image (B) Segmented Image

Fig-5.3 Segmentation of Test Image (3) By Proposed Method



(A) Test Image (B) Segmented Image

Fig-5.4 Segmentation of Test Image (4) By Proposed Method



(A) Test Image (B) Segmented Image

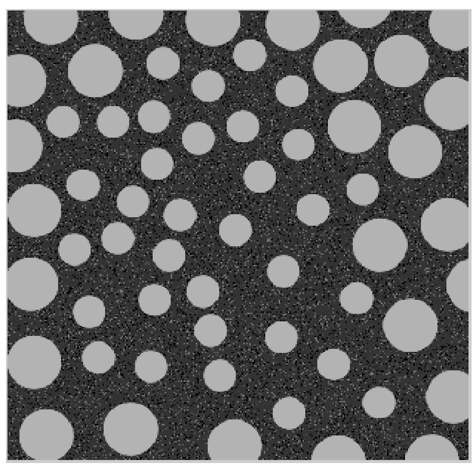
Fig-5.5 Segmentation of Test Image (5) By Proposed Method

Thus in this way algorithm is tested on 25 images. From each figure it is obvious that algorithm is working efficiently on ultrasound images as compared to other segmentation methods.

5.3 Segmentation followed by Counting of objects in images

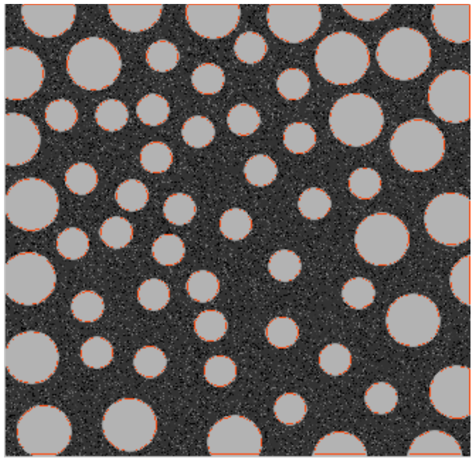
20 images are taken and segmented as well as objects in the image are counted using morphological approach and all images are compared with the traditional techniques of segmentation e.g. CANNY, LOG, SOBEL, PREWITT and ROBERTS as well as region growing techniques of segmentation. Due to limitation of space only 3 images are taken here. A quantitative analysis is done by calculating area of 10 objects in images by each method.

Following test images are taken for experiment. These images are segmented by using morphological approach as well as numbers of objects are counted in same images. A quantitative analysis is done by calculating the area of 10 objects by each method. The test image, segmented image, and quantitative analysis are shown in figures and tables respectively.

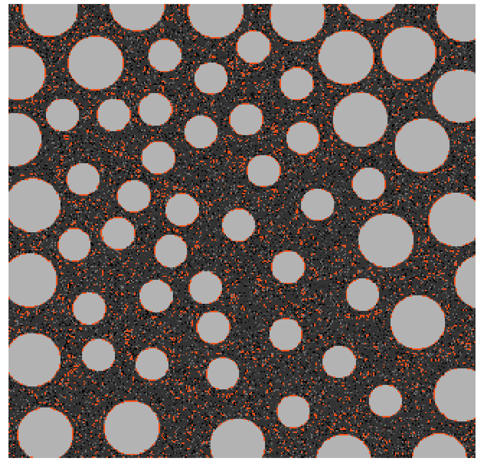


(A) Test Image

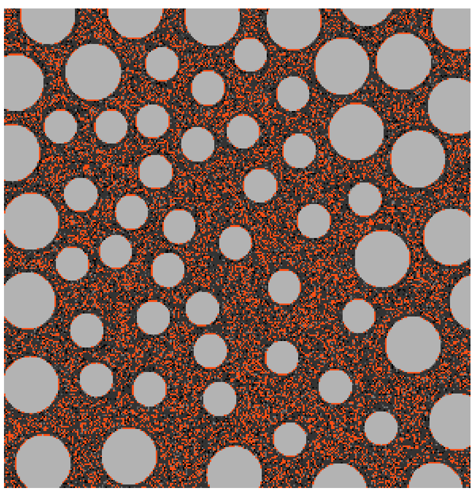
Total number of objects in image=61



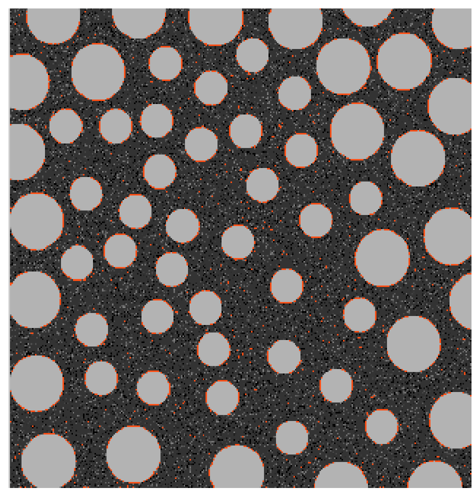
(B) Proposed Method



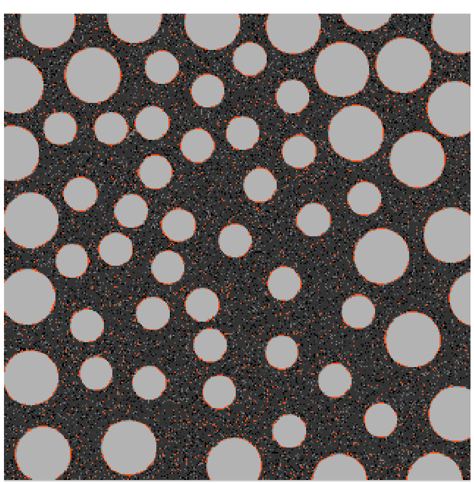
(C) Log Method



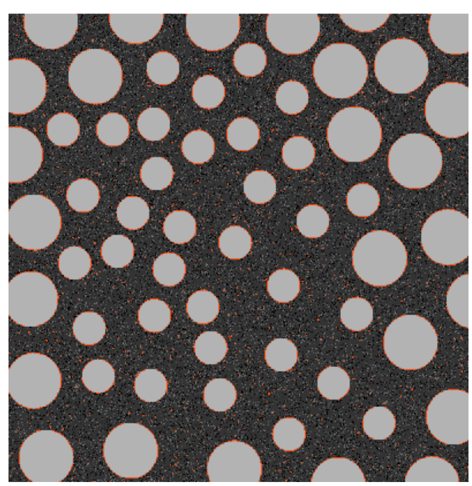
(D) Canny Method



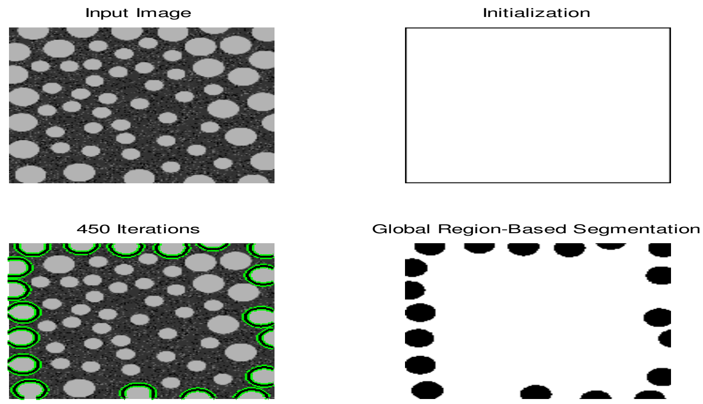
(E) Prewitt Method



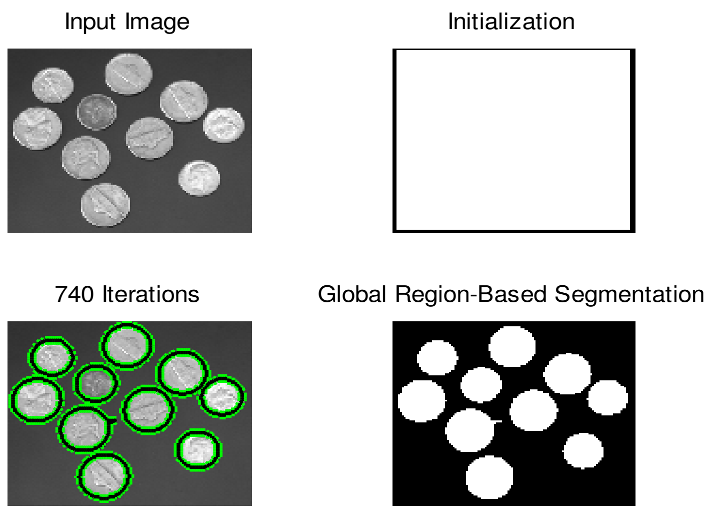
(F) Roberts Method



(G) Sobel Method



(H) After 450 Iterations When Image Is Complex



(I) After 740 iterations when image is not complex

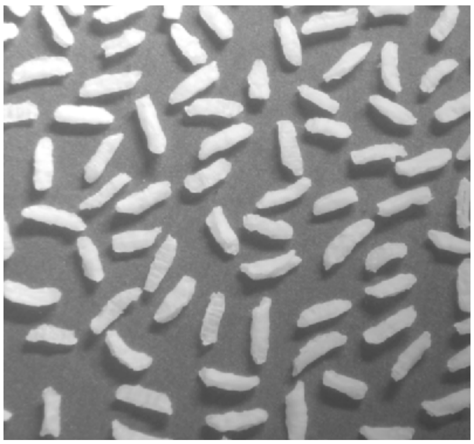
Fig-5.6 Segmentation of Test Image (1) By Different Methods

Table5.1. Quantitative Analysis of Image 1

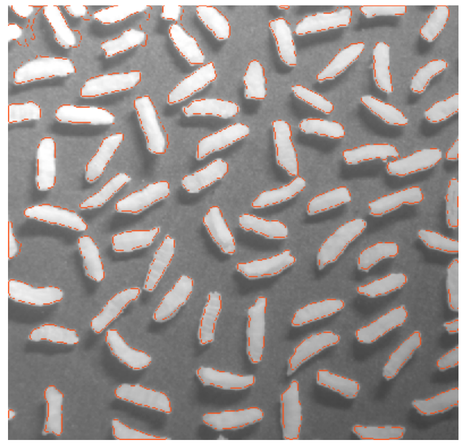
Method → Objects ↓	Actual area	Area by proposed method	Area by Log method	Area by Canny method	Area by Prewitt method	Area by Roberts method	Area by Sobel method
Object1	3051	2769	NA	NA	NA	NA	NA
Error (%)		9.24	-	-	-	-	-
Object2	2717	2435	NA	NA	NA	NA	NA
Error (%)		9.24	-	-	-	-	-
Object3	3942	3686	NA	NA	NA	NA	NA
Error (%)		6.49	-	-	-	-	
Object4	3824	3580	NA	NA	NA	NA	NA
Error (%)		6.38	-	-	-	-	-
Object5	3934	3697	NA	NA	NA	NA	NA
Error (%)		6.02	-	-	-		-
Object6	3903	3652	NA	NA	NA	NA	NA
Error (%)		6.43	-	-	-	-	-
Object7	2785	2512	NA	NA	NA	NA	NA
Error (%)		9.80	-	-	-	-	-
Object8	1460	1308	1463	1491	1502	NA	1505
Error (%)		10.41	-0.205	-2.12	-2.87	-	-3.08
Object9	1454	1316	1463	1490	1499	NA	1497
Error (%)		9.49	-0.618	-2.47	-3.09	-	-2.95
Object10	1453	1324	1463	1500	1505	NA	1506
Error (%)		8.87	-0.688	-3.23	-3.57	-	-3.64

As shown from the above figures and table, our method's performance is very efficient as compared to other method. As in all gradient based methods of segmentation, weak and noisy edges are detected so segmentation is not perfect as well as it cannot count the number of objects. In region growing method of segmentation, segmentation is performed very well, but it can find application only where time is not a constraint. It works when images are not complex and when a small portion is selected for segmentation. Also it has a limitation in synchronizing the number of iterations according to image as shown above in figure.

5.4 List of test images and their segmentation results by proposed method



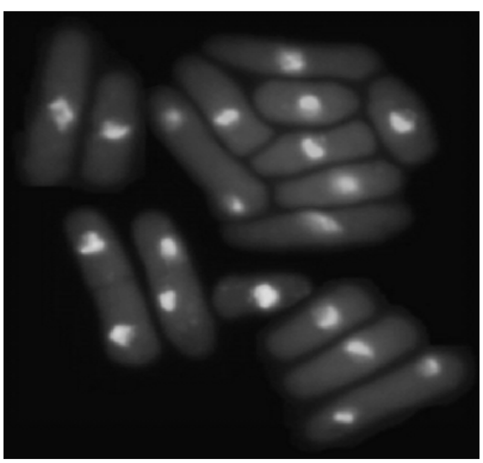
(A) Test Image



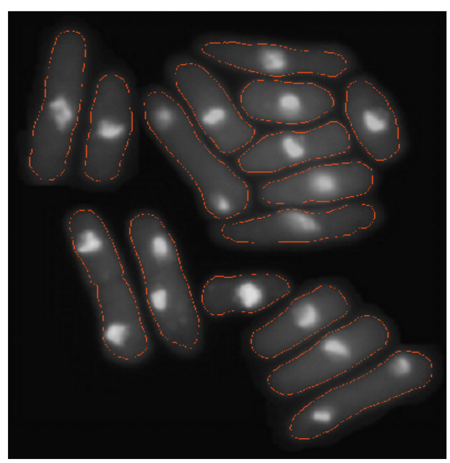
(B) Segmented Image

Fig-5.7 Segmentation of Test Image (2) By Proposed Method

Total number of objects in image=76



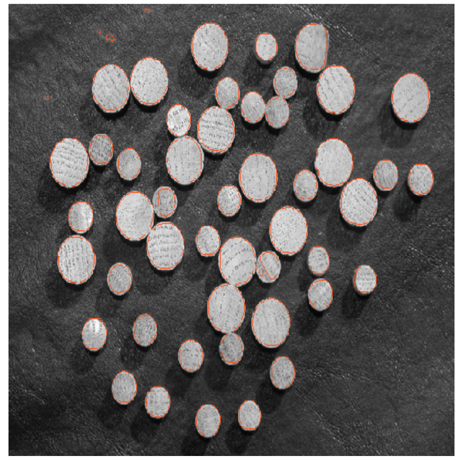
(A) Test Image



(B) Segmented Image

Fig-5.8 Segmentation of Test Image (3) By Proposed Method

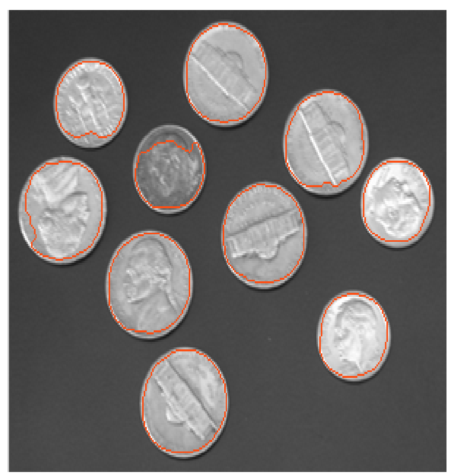
Total number of objects in image=16



(A) Test Image (B) Segmented Image

Fig-5.9 Segmentation of Test Image (4) By Proposed Method

Total number of objects counted=48



(A) Test Image (B) Segmented Image

Fig-5.10 Segmentation of Test Image (5) By Proposed Method

Total number of objects counted= 10

5.5 List of Respective Tables for Quantitative Analysis of Images

Table5.2. Quantitative Analysis of Image 2

Method → Objects ↓	Actual area (in pixels)	Area by proposed method	Area by Log method	Area by Canny method	Area by Prewitt method	Area by Roberts method	Area by Sobel method
Object1	556	514	NA	NA	NA	NA	NA
Error (%)		7.55	-	-	-	-	-
Object2	1172	978	NA	NA	NA	NA	NA
Error (%)		16.55	-	-	-	-	-
Object3	530	502	NA	NA	NA	NA	NA
Error (%)		5.57	-	-	-	-	-
Object4	1783	1705	NA	NA	NA	NA	NA
Error (%)		4.48	-	-	-	-	-
Object5	2226	2035	NA	NA	NA	NA	NA
Error (%)		8.58	-	-	-	-	-
Object6	2242	2037	NA	NA	NA	NA	NA
Error (%)		9.14	-	-	-	-	-
Object7	1730	1634	1668	NA	1720	1771	1722
Error (%)		5.55	3.58	-	0.578	-2.36	0.462
Object8	1234	1227	1281	1317	1312	NA	1312
Error (%)		0.567	-3.81	-6.72	-6.32	-	-6.32
Object9	1733	1584	1572	1624	1634	NA	1633
Error (%)		8.59	9.29	6.29	5.71	-	5.77
Object10	1306	1291	1332	1374	1382	NA	1382
Error (%)		1.14	-1.99	-5.20	-5.82	-	-5.82

Note-Here negative sign indicates the particular object is oversegmented.

Table5.3. Quantitative Analysis of Image 3

Method ↓ Objects →	Actual area	Area by proposed method	Area by Log method	Area by Canny method	Area by Prewitt method	Area by Roberts method	Area by Sobel method
Object1	13384	15327	NA	NA	1453	NA	1444
Error (%)		14.52	-	-	89.14	-	89.21
Object2	16547	15720	934	785	1257	652	1256
Error (%)		4.99	94.35	95.25	92.40	96.05	92.40
Object3	9638	11034	NA	NA	1151	1086	1147
Error (%)		14.48	-	-	88.05	88.73	88.09
Object4	14519	15104	725	687	719	NA	722
Error (%)		4.03	95.00	95.26	95.04	-	95.02
Object5	20063	14859	NA	NA	562	NA	558
Error (%)		25.94	-	-	97.19	-	97.21
Object6	7011	8160	1174	1105	1199	502	1228
Error (%)		16.38	83.25	84.24	82.90	92.84	82.48
Object7	13886	10277	NA	NA	NA	NA	NA
Error (%)		25.99	-	-	-	-	-
Object8	10039	13070	NA	689	1074	1080	1041
Error (%)		30.19		96.13	89.30	89.24	89.60
Object9	14117	14795	NA	NA	491	NA	488
Error (%)		4.80	-	-	96.52	-	96.54
Object10	8603	8645	941	NA	NA	968	NA
Error (%)		0.49	89.06	-	-	88.75	-

Table5.4. Quantitative Analysis of Image 4

Method → Objects ↓	Actual area	Area by proposed method	Area by Log method	Area by Canny method	Area by Prewitt method	Area by Roberts method	Area by Sobel method
Object1	2548	2394	2573	3791	2780	NA	2777
Error (%)		6.04	-0.98	-48.78	-9.10	-	-8.99
Object2	1099	1054	1098	887	NA	NA	NA
Error (%)		4.09	9.09	19.29	-	-	-
Object3	1085	1098	1111	1143	1144	1161	1141
Error (%)		-1.19	-2.39	-5.34	-5.43	-7.00	-5.16
Object4	1085	924	1130	1159	1158	NA	1156
Error (%)		14.83	-4.14	-6.82	-6.73	-	-6.54
Object5	1099	1053	1150	1178	1180	1185	1181
Error (%)		4.18	-4.64	-7.18	-7.37	-7.82	-7.46
Object6	1067	948	1095	1126	1128	NA	1127
Error (%)		11.15	-2.62	-5.52	-5.72	-	-5.62
Object7	1053	972	1113	1142	1143	NA	1141
Error (%)		7.69	-5.70	-8.45	-8.55	-	-8.36
Object8	1094	1037	1139	1178	1174	NA	1173
Error (%)		5.21	-4.11	-7.68	-7.31	-	-7.22
Object9	1106	1031	1154	1183	1187	NA	1185
Error (%)		6.78	-4.34	-6.96	-7.32	-	-7.14
Object10	2457	2321	2470	2537	NA	2560	2532
Error (%)		5.53	-0.53	-4.88	-	-4.19	-3.05

Table5.5. Quantitative Analysis of Image 5

Method → Objects ↓	Actual area	Area by proposed method	Area by Log method	Area by Canny method	Area by Prewitt method	Area by Roberts method	Area by Sobel method
Object1	2647	2256	2664	2790	2753	2775	2752
Error (%)		14.7	-0.64	-5.40	-4.0	-4.83	-3.96
Object2	1852	1578	1863	1920	1927	1964	1933
Error (%)		14.8	-0.60	-3.67	-4.04	-6.04	-4.37
Object3	2670	2435	2715	2828	2773	2801	2773
Error (%)		8.80	-1.68	-5.91	-3.85	-4.90	-3.85
Object4	796	1378	1924	1947	1908	1917	1903
Error (%)		-73.11	-89.70	-84.59	-69.69	-90.83	-79.07
Object5	2751	2540	2774	2874	2836	2868	2834
Error (%)		7.67	-0.836	-4.47	-3.09	-4.25	-3.02
Object6	2509	2321	2541	2626	2605	2643	2601
Error (%)		7.49	-1.27	-4.66	-3.82	-5.34	-3.67
Object7	2592	2377	2623	2693	2683	2724	2683
Error (%)		8.29	-1.19	-3.89	-3.51	-5.09	-3.51
Object8	2542	2231	2570	2625	2635	2667	2632
Error (%)		12.23	-1.10	-3.26	-3.66	-4.91	-3.54
Object9	1851	1696	1864	1920	1956	1978	1955
Error (%)		8.37	-0.702	-3.72	-5.67	-6.86	-5.62
Object10	1863	1628	1839	1909	1923	1948	1924
Error (%)		12.6	1.29	-2.47	-3.22	-4.56	-3.27

5.6 Summary of results

After doing experiments for both methods we conclude that the proposed method is segmenting the ultrasonic images with greater efficiency as shown from figures as compared to other methods or techniques of image segmentation. The proposed method is segmenting the desired portion of ultrasonic images while other methods segmenting noises as well as false and weak edges.

After quantitative analysis it is clear that the proposed method is segmenting efficiently as compared to other methods of segmentation as well as it is counting number of objects in images even they are overlap each other to some extent. The proposed method is separating them and counting number of objects.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

The proposed algorithm is compared with other image segmentation techniques like edge based techniques for example CANNY, LOG, PREWITT methods of segmenting images and region growing techniques. After experiments it is found that the proposed algorithm is efficient, reliable, consumes less time and gives result without noise as well as works on complex images. The proposed algorithm can be applied to many practical areas, such as military objective detection and medical application etc.

This proposed method is also segmenting some portions in the image that are not desired. This method can be made adaptive by using some work so that it segments only the portion which is selected, not the entire image. Efficiency of this proposed method can be increased by doing work in this way.

Generally there is not any universal method for segmenting all kinds of images. So by applying some technique like changing of structuring elements, it is possible to automatically change it depending on the type of images it is made possible to segment all kinds of practical images.

REFERENCES

[1] Kelly, P.A. and Chen, G., "Iterative Segmentation Algorithms Using Morphological Operations", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol-5, PP: 49 – 52, 1993.

[2] Deng Shiwei and Yuan Baozong, "Range Image Segmentation Using Mathematical Morphology", *IEEE Region 10 Conference On TENCON Proceedings on Computer, Communication, Control and Power Engineering*, Vol-2, PP: 1009 – 1011, 1993.

[3] P Salembier and M Pardas, "Hierarchical Morphological Segmentation for Image Sequence Coding", *IEEE Transactions on Image Processing*, Vol-3, PP: 639 – 651, 1994.

[4] Zhao Yuqian, Gui Wei-hua, Chen Zhencheng, Tang Jing-tian, and Li Ling-yun," Medical Images Edge Detection Based on Mathematical Morphology*", Proceedings of the IEEE Engineering in Medicine and Biology 27th Annual International Conference Shanghai, China,* PP: 6492 – 6495,2005.

[5] Ying-Tung Hsiao, Cheng-Long Chuang, Joe-Air Jiang and Cheng-Chih Chien, "A Contour Based Image Segmentation Algorithm Using Morphological Edge Detection", *IEEE International Conference on Systems, Man and Cybernetics*, Vol - 3, PP: 2962 – 2967, 2005.

[6] Jin Zou, Hongsong Li, Bin Liu, and Renfei Zhang, "Color Edge Detection Based on Morphology" *First International Conference on Communications and Electronics, ICCE,* PP: 291 – 293, 2006.

[7] Yuqian Zhao, Weihua Gui and Zhencheng Chen," Edge Detection Based on Multi-Structure Elements Morphology", *Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China,* Vol-2, PP: 9795 – 9798, 2006.

[8] Yuqiu Sun, Jinwen Tian, Jian Liu," Dual band Infrared Image Target Detection Algorithm Based on Multiscale Morphology", *8th International Conference on Signal Processing,* Vol-1, 2006.

[9] Dawei Qi, Fan Guo, and Lei Yu," Medical Image Edge Detection Based on Omnidirectional Multi-Scale Structure Element of Mathematical Morphology", *Proceedings of the IEEE International Conference on Automation and Logistics, Jinan, China, 2007.*

[10] Zhenhua Li, Yingping Yang, and Wei Jiang, "Multi-scale Morphologic Tracking Approach for Edge Detection", *Fourth International Conference on Image and Graphics, 2007.*

[11] Peijun Li and Hongtao Hu, "Segmentation of High-resolution Multispectral Image Based on Extended Morphological Profiles" *IEEE International Conference on Geoscience and Remote Sensing Symposium, IGARSS*, PP: 1481 – 1484, 2007.

[12] Xueshun Wang, Dawei Qi and Yuanxiang Li, "Edge Detection of Decayed Wood Image Based on Mathematical Morphological Double Gradient Algorithm", *IEEE International Conference on Automation and Logistics, ICAL,* PP: 1226 – 1231, 2008.

[13] Yanlei Xu, Jiyin Zhao, Yubin Jiao, "Gray-scale Image Edge Detection Based on Order Morphology Transformation", *Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China,* PP: 5970 – 5975, 2008.

[14] Dawei Qi, Chao Kong, and Lei Yu, "Improved Morphological method in the detection of Log CT Image with Defect", *Proceedings of the IEEE International Conference on Automation and Logistics, Qingdao, China*, PP: 1238 – 1243, 2008.

[15] Yanlei Xu Jiyin Zhao Yubin Jiao," Noisy Image Edge Detection Based on Multi-scale and Multi- structuring Element Order Morphology Transformation", *Conference on Image and Signal Processing*, *CISP*, Vol-3, PP: 379 – 383, 2008.

[16] Wenju Li and Ling Li, "A Novel Approach for Vehicle-logo Location Based on Edge Detection and Morphological Filter" *Second International Symposium on Electronic Commerce and Security,* Vol-1, PP: 343 – 345, 2009.

[17] Liu Yucheng and Liu Yubin, "An Algorithm of Image Segmentation Based on Fuzzy Mathematical Morphology", *International Forum on Information Technology and Applications, IFITA,* Vol-2, PP: 517 – 520, 2009.

[18] Rafael C. Gonzalez, Richard E. Woodes, and Steven L. Eddins, DIGITAL IMAGE PROCESSING, published by Pearson Education(Singapore) Pte. Ltd.

[19] Tinku Acharya and Ajoy K. Ray IMAGE PROCESSING PRINCIPLES AND APPLICATIONS published by A JOHN WILEY & SONS, MC., PUBLICATION.

[20] Handbook Of Biomedical Imaging Processing And Analysis, Isaac N. Bankman, PhD, APPLIED PHYSICS LABORATORY JOHN HOPKINS UNIVERSITY, LAUREL, MARYLAND.