ANALYSIS AND IMPROVEMENT IN IMAGE SEGMENTATION FOR CT IMAGES

A Thesis submitted in the partial fulfillment of requirements for the award of the degree of

Master of Engineering

In

Electronic Instrumentation and Control Engineering

Submitted by

AMANPREET KAUR

Regn.No-801051001

Under the esteemed guidance of

Mr. M. D. Singh

Asst. Professor

Department of Electrical and Instrumentation Engineering

THAPAR UNIVERSITY

PATIALA (PUNJAB)-147004

June-2012

CERTIFICATE

I hereby declare that the Thesis entitled **Analysis and Improvement in Image Segmentation for CT images** is an authentic record of my own work carried out as the requirements for the award of the degree of M.E. (Electronic Instrumentation and Control Engineering) at Thapar University, Patiala, under the guidance of **Mr. M.D Singh**, Assistant Professor, EIED.

The matter presented in this Thesis has not been submitted for the award of any other degree of this or any other university.

Amanpreet Kaur

Amanpreet Kaur
(Roll no-801051001)

Date 13 July, 2012

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

Mr. M.D Singh
Assistant Professor
Department of Electrical
and Instrumentation
Engineering

Countersigned by:

S. Ghosh

Dr. Smarajit. Ghosh
Professor & Head, EIED
Thapar University, Patiala

Dr. S.K. Mohapatra
Dean, Academic Affairs
Thapar University, Patiala

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my supervisor, **Mr. M.D Singh,** Assistant Professor (EIED), for all his guidance and invaluable advises throughout the progress. He has stimulated my interest in medical image processing and inspired me for doing research on this topic.

I would also like to thank **Dr. Smarajit Ghosh**, Professor & Head, Electrical & Instrumentation Engineering Department, and **Dr. S.K Mohapatra**, Dean, Academic Affairs for giving an opportunity to work in this regard.

Dr. Virender Garg(MD) Sr. Resident, Department of Radiology, Govt. Rajindra hospital, Patiala, for extending all the needed help to carry out this work. Also I would like to thank my parents and all my friends for their continuous support and encouragement.

Amanpreet Kaur

ABSTRACT

Image segmentation in medical images is a very complex task. Currently, brain diseases are detected by imaging only after the appearance of neurological or nervous system symptoms. Manual segmentation is a long and painful task. This method is not reliable and error sensitive. The need for correct segmentation of the ailment is very important for proper medications as any delay or wrong diagnosis may become fatal to the patient. Many methods have been developed to segment tumor. In this work, we have used Computed Tomography (CT) images for segmentation of abnormal portion. Two techniques, modified region based active contour and hybrid level set method, have been used in this thesis work. In the former method, the number of iterations reduced as well as the time consumption drastically reduced. The latter method, overcomes the problem of under segmentation as well as over segmentation.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

CT	Computed Tomography
MRI	Magnetic Resonance Imaging
PSO	Particle Swarm Optimization
PCM	Possibilistic C-Means
FCM	Fuzzy C-Means
SOM	Self-Organizing Map
SDF	Signed Distance Map
ROI	Region Of Interest
FDR	Fisher Discrimination Ratio

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 OVERVIEW

Image segmentation has been used in partitioning various types of fractures and diseases in medical imaging. This particular field of image segmentation is called Medical Image Segmentation. It has revolutionized the field of medicine by providing novel methods to extract and visualize specific information from medical data. Along with this, it helps the physicians to analyse easily the diseased portion and helps in fast detection of abnormal area. The brain is the most complex part of the human body. Medical images are difficult to segment due to complexity of the anatomical features. The anatomy that is of interest may not be separable from its surroundings due to gray level inconsistency and lack of strong edges at its border. There are many techniques available with which one can analyse them such as X-ray, Computed tomography (CT), Ultrasonic Imaging, Magnetic Resonance Imaging (MRI) etc. CT scan is mostly preferred over other techniques due to good detection of haemorrhage and bony detail plus lower cost, short imaging times and widespread availability.

Brain tumor is a physiological disorder caused either by bleeding inside brain or lack of blood supply to some part of the brain. It may result in disability, coma for many days or sudden death. Hence accurate diagnosis, segmentation and proper medication are of prime importance in such case. CT is one such method used for segmentation of diseased part in brain that is preferred over many other techniques. There are various techniques available for detecting the tumor in CT images. The problem with CT is that it does not properly highlight the abnormal area which makes the segmentation more difficult. This problem becomes more prominent when there is a slight difference between normal and abnormal area. This may result in under segmentation or over segmentation. Hence, finding a way to accurately segment different types of abnormalities with greater accuracy and less time consumption would help physicians to accurately detect and medicate patients, and save many lives.

The objective of this thesis is to improve the existing algorithms in terms of time taken for segmentation of abnormalities present inside brain of CT images. As already mentioned, the accurate segmentation of diseased parts is of prime importance here. Region based active contour which has been used, but the problem is that it uses predefined number of iterations which results in more time consumption. Hence an improved method has been proposed which results in faster convergence and the time consumption has been drastically improved. The second improvement has been made in level set evolution without re-initialization. This method has a drawback that it does not segment accurately the diseased portion. In some cases it may lead to under segmentation, while in other cases it may lead to over segmentation. Also this algorithm starts without re-initialization i.e. the contour starts from the boundaries of the image and then converges to diseased portion. In our modified algorithm we have are able to segment diseased portion accurately and with less error.

1.2 ORGANIZATION OF THESIS

The thesis is organized as below:

Chapter 1 includes the brief description of segmentation, CT and objective of the work. Chapter 2 involves literature survey, which explains the various works that have been carried out in the area. Chapter 3 explains image segmentation, various methods used for image segmentation and particle swarm optimization. The two algorithms used, its modified version and texture parameters are explained in chapter 4. Chapter 5 presents the results, and chapter 6 concludes the conclusion.

CHAPTER 2
LITERATURE REVIEW

Minimum cross entropy thresholding method is very time consuming in multilevel thresholding as compared to bi-level thresholding for complex image segmentation. **Yin *et. al.*** [1] proposed a method which uses particle swarm optimization (PSO) together with minimum cross entropy to obtain a threshold for image segmentation. This method overcomes the problem of time consumption. The results of the proposed method are compared with exhaustic search method. It has been concluded that although thresholds of the proposed method with PSO are equivalent to that of exhaustic search method but in latter case computational time increases exponentially with the number thresholds whereas the computational time of former case is negligible.

Nakib *et. al.* [2] proposed two-dimensional survival exponential entropy with PSO for segmentation of magnetic resonance imaging (MRI) images. Initially, the two dimensional histogram is obtained in order to avoid the problem of spatial distribution. The PSO is used for optimization of two-dimensional survival exponential entropy to solve the segmentation problem. Result show that the proposed method produced better results on comparison with two-dimensional shanon entropy.

A successful application introduced by **Djerou *et. al.*** [3] uses thresholding and binary PSO. This approach determines the number of thresholds optimal. The objective function depends on the user's data set. In this paper, mainly two thresholding methods are optimized using proposed algorithm namely, Kapur's method and Otsu's method. The uniformity measure is used to evaluate the quality of threshold images. Experimental results show that computation time of Otsu's method is better than Kapur's method. Therefore the proposed algorithm using Otsu's method is more efficient.

Another approach that uses PSO with entropy has been proposed by **Hongmei *et. al*.** [4]. The author applied improved PSO to predict the multi-level thresholding for image segmentation. The proposed improved PSO overcomes the problem of premature convergence of PSO. One of the problems with standard PSO is parameter selection.

This can be improved by making random numbers instead of acceleration coefficients in PSO and referring the coevolution model. The uniformity measure is used which qualitatively measure the performance of image segmentation method. Results show that on increasing the number of thresholds, the running time of proposed method does not significantly increased and the efficiency as well as the convergence rate of the basic PSO can also be improved.

The multimodal images are more complex to segment as compared to bimodal images. **De *et. al.*** [5] presented a region growing technique along-with thresholding and PSO for the segmentation of multimodal MRI images. The basic idea was that the diseased portion will have different intensity value as compared to non-diseased portion. Initially, the normalised histogram of the diseased MRI image is calculated and then entropy maximization is used to obtain the range of grey level of diseased portion. This range of grey level of diseased portion is optimized using PSO. At last the concept of variable mask is applied over the region of interest to obtain the final segmented image.

Yu *et. al.* [6] proposed an image segmentation technique based on maximum entropy thresholding and quantum behaviour of PSO. This method overcomes the problem of earlier convergence of the PSO and the shortcomings of large calculation of multi thresholding segmentation.

Masooleh *et. al.* [7] proposed a fuzzy system in conjunction with PSO for image segmentation. A sugeno fuzzy system is used in the proposed method. By applying a set of fuzzy rules, each pixel is assigned a colour class. For fuzzy colour classification, HSL colour space is used. The main problem with fuzzy system is the large number of fuzzy rules. Therefore, PSO is used which automatically produces the least number of optimum fuzzy rules and generate the optimized membership function.

Puranik *et. al.* [8] presented a colour image segmentation using comprehensive learning PSO based fuzzy system. Comprehensive learning PSO is an improved version of PSO, in which all particles pbest are used to update the velocity of other particles. A fitness function is used which rates the optimality of each particle. Comprehensive learning PSO is an optimization process which finds the optimal fuzzy rules as well as the membership function. Each particle is assigned a set of fuzzy rules. During this process, each particle tries to maximize the fitness function. Also Comprehensive learning PSO discourages the premature convergence of original PSO. Furthermore, HSL colour space is used in the proposed method as the colour can be presented in three-dimensional for fuzzy colour classification.

Gopal *et. al.* [9] presented two phase of MRI segmentation. The first phase includes pre-processing and enhancement. In order to remove labels and x-ray marks from MRI images, a tracking algorithm has been proposed. Along with this, a median filter is used to remove high frequency components from MRI images. The fuzzy c-means calculates the adaptive threshold, after PSO which automatically determines the optimal threshold value of the given image to select initial cluster seed point.

Murugesan *et. al.* [10] illustrated multi elistic exponential PSO hybridized with fuzzy system in order to perform segmentation of coloured images. Multi elistic exponential PSO is a combination of multi elistic PSO and exponential PSO. Multi elistic PSO employs a kernel induced similarity measure for searching global best of PSO. The standard PSO converges too early in the search space. The multi elistic PSO helps to prevent this convergence behaviour of PSO. On the other hand, exponential PSO avoids the particles from stagnation of local optima by varying the inertia weight exponentially. This hybridized PSO is used to find the optimal fuzzy rules and membership function. Each particle tries to maximize the fitness function. The best fuzzy rule is selected for image segmentation.

The traditional fuzzy c-means clustering algorithm is sensitive to noise. One of the simplest methods is low pass filtering of an image and then applying the fuzzy c-means (FCM) clustering algorithm. The drawback this approach was that it may lead to loss of the important details present in an image. To overcome this drawback, an important fuzzy c-means clustering algorithm has been proposed by **Shen *et. al.*** [11]. An important parameter that can affect the performance of fuzzy c-means clustering method is the parameter optimization. **Forouzanfer *et. al***. [12] proposed a breeding swarm algorithm that helps to find optimum attraction parameters. The breeding swarm algorithm combines the strength of both PSO and genetic algorithm. The algorithm is so designed so that PSO facilitates local search and genetic algorithm performs global search. Experimental results indicate that the proposed breeding swarm with fuzzy c-means clustering algorithm is a good technique for the segmentation of MRI images.

Kole *et. al.* [13] described an approach for image segmentation using hybrid technique based on PSO and genetic algorithm. The PSO based dynamic clustering has been used to find the optimal number of clusters. This information is further used by genetic algorithm to improve the final result of the PSO based method. At last, the best result is obtained by comparing their respective validity indices [14] and the data is partitioned accordingly.

In the standard PSO, the particles are prematurely attracted to the local attractor. **Wei *et. al.*** [15] proposed an inertia adaptive PSO and wavelet mutation algorithm which helps the particles to escape from local minima and increases the speed of the segmentation process. The fitness function of the particles in the swarm is calculated by using fuzzy entropy. The motion of the particles is governed by two dynamical regimes. In the first case, if there is an improvement in the fitness function of the particles from iteration to iteration, in such a case inertia adaptive PSO has been used to sample the particles. In the second case, if there is no improvement, it results in stagnation. In such a case, the wavelet mutation has been proposed. One of the advantages of wavelet mutation is that it exhibits a fine tuning ability.

De *et. al.* [16] illustrated how PSO can be successfully integrated with wavelet mutation and provides a more effective approach to resolve the stagnation problems. Initially, the normalized histogram of the MRI images is obtained. Then entropy maximization is employed to get the expert knowledge of the probable threshold grey level range for segmentation of MRI images. The hybrid PSO together with wavelet mutation is used to optimize the initial value of the threshold. Using this threshold value, the region of interest is obtained. Finally, a variable mask is employed on region of interest to get the segmented MRI images with lesions.

Omran *et. al.* [17] presented a dynamic clustering based on PSO. Initially, the algorithm partitions the data set into the relatively large number of clusters in order to reduce the effect of initial conditions. The binary PSO helps to select the best number of clusters. At last, the centres of the chosen clusters are refined by k-means clustering. One of the advantage of proposed method is that user can choose any validity index according to the given data.

Clustering is an unsupervised technique for image segmentation. **Chun *et. al.*** [18] proposed a method that uses fuzzy c-means clustering together with PSO. The main objective of fuzzy c-means clustering is to find cluster centres that maximizes a similarity function or minimizes the dissimilarity function. The PSO is used for assigning each pixel to a cluster. This hybridized fuzzy c-means clustering and PSO algorithm produce better segmentation results.

The basic fuzzy c-means (FCM) algorithm is affected by the number and initial location of the centre of the predetermined clustering. **Jing *et. al.*** [19] proposed a fast fuzzy c-means (FCM) method together with PSO for image segmentation. The PSO algorithm is an optimization process which automatically determines the number of clusters as well as the centre of the clusters.

The sonar images have low signal to noise ratio. Therefore, it becomes difficult to segregate sonar images. **Liu *et. al.*** [20] presented a PSO based fuzzy cluster for sonar image segmentation. This combination tends to produce strong searching and high speed convergence ability. In addition, the fuzzy measure and fuzzy integral are also calculated to compute the fitness.

Since the possibilistic c-means (PCM) algorithm is very sensitive to initialization and parameters. **Jing *et. al.*** [21] presented an approach to fit clusters which are close to one another. The t-Particle Swarm Optimization (t-PSO) is used to solve the complex computation as well as initial parameter sensitivity problem in order to get accurate segmentation.

Zhang *et. al.* [22] illustrated how possibilistic c-means (PCM) can be integrated with PSO and provides a significant improvement on the efficiency of the segmentation. The possibilistic c-means (PCM) is more accurate as compared to fuzzy c-means (FCM), as it overcomes the relative membership problem of fuzzy c-means (FCM) in image segmentation. The mahalonolis distance is used with possibilistic c-means (PCM) algorithm, since it enhances the performance of the clustering algorithm. The PSO is used to optimize the initial clustering centres.

In order to remove the robustness of fuzzy c-means (FCM) to noise, **Liu *et. al.*** [23] produced a new hybrid algorithm using fuzzy PSO and markov random field. The spatial information described by markov random field model is used to modify the similarity measure of FCM. The segmentation is done corresponding to the global best position of the, since it is less time consuming and also accelerate the speed of the algorithm as compared to the local best position.

The underwater images have low signal to noise ratio. So, it becomes difficult to segment the image. The traditional FCM method does not provide good results and is very time consuming. **Wang *et. al.*** [24] presented segmentation algorithm based on histogram weighted fuzzy c-means. The statistical characteristics of histogram of grey image are taken into account, which produces a fast and effective FCM algorithm for water image segmentation. Since the value of membership affects the convergence affects the convergence rate of the iterative process. The proposed improved fuzzy membership meets the requirements that increases the maximum value membership degree and reduces other memberships. FCM is very sensitive to initial value and improper selection of initial value may lead to fall into the local minimum. The PSO described by sine function has been introduced to overcome the drawback that FCM algorithm cannot reach the global optimum solution. Results indicate that the proposed method can be employed to real time applications and the processing time has also reduced.

An approach that uses rough set entropy with PSO has been proposed by **Feng *et. al.*** [25]. The author applies rough set entropy to segment an grey-scale image. The algorithm obtains the optimal threshold by using PSO and rough set entropy which is based on boundary conditions. Experimental results show that the proposed algorithm is time efficient and the system becomes more stable. Also, the sensibility of the algorithm to partition size image sub-piece is low.

Behera *et. al.* [26] presented a hybrid rough set PSO for partitioning an image into different meaningful segments. In rough c-means, each cluster is treated as an interval or rough set. The k-means clustering algorithm has been used to classify image pixels, which calculates the initial means and their positions in clusters. After obtaining the cluster centres, the upper and lower bounds of the clusters are calculated. The rough set is used to upgrade the cluster centres. PSO is used to tune the parameters of rough c-means. In this approach, a statistical mathematical function called Davies Bouldin [27] index is used for the purpose of the fitness function in PSO. The performance evaluation of PSO shows that this method reduces noisy spots and is less sensitive to noise.

Yi *et. al* [28] illustrated white blood cell image segmentation incorporating an online trained neural network. Initially, a mean shift algorithm [29] has been employed to search the cluster centre. After this, uniform sampling is performed so as to reduce the size of the training set. By using uniform sampling, the statistical data has revealed that subset can represent the entire data set approximately. Furthermore, the PSO algorithm has been employed which helps in faster convergence as well as helps to escape from local optimum.

Lian *et. al.* [30] proposed a new approach for segmenting magnetic resonance imaging (mri) images based on modified adaptive PNN. The MAPNN incorporates SOM (self-organizing map), MPSO (modified particle swarm optimization) and PNN (probabilistic neural network) for segmentation. In this approach, SOM [31] neural network is trained with training feature set, so as to yield a SOM map for PNN. Then, according to this SOM map, MPSO provides the smoothing factor to PNN. The feature extraction is also performed in order to improve training quality of neural network.

Image enhancement and pre-processing is required for ultrasound images, since they have low contrast and speckle noise. **Alamelumangai *et. al*** [32] presented a neuro-fuzzy filter [33] for image enhancement in the proposed algorithm. After pre-processing, an ANFN and EPSO had been employed for image segmentation. The proposed ANFN algorithm is basically a 5-layer network and the inputs are fuzzy values. The EPSO eliminates the weakest particle and searches for optimal solution. The proposed algorithm helps to reduce the computational time without affecting the accuracy of the solution.

CHAPTER 3
BASICS OF IMAGE SEGMENTATION

3.1 INTRODUCTION

In computer vision, image processing is any form of signal processing for which the input is an image, such as photographs or frames of videos. The output of image processing can either be an image or a set of characteristics or parameters related to an image. The image processing involves techniques like image restoration, image enhancement, image segmentation etc.

Image segmentation refers to the process of partitioning a digital image into multiple segments (set of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyse. Image segmentation is typically used to locate object and boundaries in an image. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics. The level to which subdivision is carried depends on the problem being solved. That is, segmentation should stop when the object of interest in an application has been isolated.

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Each of the pixels in a region are similar with respect to some characteristics or computed property, such as color, intensity or texture. Adjacent regions are significantly different with respect to same characteristics.

In other words, segmentation is defined as separation of portions having similar characteristics such as gray level, color, texture, brightness, and contrast. This is a necessary and fundamental concept which plays an important role in digital image processing and biomedical image processing areas. There are many techniques of segmenting an image. The techniques available for segmenting of images is application specific, depend upon imaging modality and type of image part to be studied i.e. it is possible to segment all images by a particular method. Every method has its own merits and demerits depending upon application used.

3.2 IMAGE SEGMENTATION STRATEGIES

Image segmentation algorithms generally are based on one of two basic properties of intensity values:

1. Discontinuity
2. Similarity

Fig. 3.1 Shows difference between similarities and discontinuities

3.2.1 Discontinuity

Discontinuity based approach partitions an image based on abrupt changes in intensity. Techniques based on discontinuity attempt to partition the image by detecting abrupt changes in gray level, e.g. point, line and edge detectors. It should be noted that point detection should be based on second derivative. For line detection we can expect second derivatives to result in a stronger response and to produce thinner lines than first derivatives. Various edge detectors are available for segmenting the images.

Fig. 3.2(a) Original Image

Fig. 3.2(b) Segmentation Based On Discontinuity

3.2.2 Similarity

Similarity based approach partitions an image based on regions that are similar according to a set of predefined criteria. Techniques based on similarity attempt to create the uniform regions by grouping together connected pixels that satisfy predefined similarity criteria, e.g. thresholding, region growing and region splitting and merging.

Fig. 3.3(a) Original Image

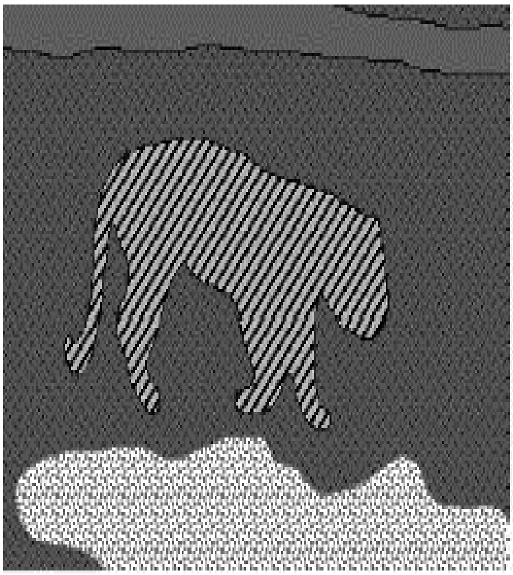


Fig. 3.3(b) Segmentation Based On Similarity

Therefore, the result of segmentation may depend critically on these criteria and on the defined connectivity. The approaches based on discontinuity and similarity mirror one another in the sense that completion of a boundary is equivalent to breaking one region into two.

3.3 TYPES OF IMAGE SEGMENTATION

Image segmentation can proceed on three different ways:

- Manually
- Automatically
- Semi automatically

3.3.1 Manual segmentation

The pixels belonging to the same intensity range could manually be pointed out, but clearly this is very time consuming method if the image is large. A better choice would be to mark the contours of the objects. This could be done discrete from the keyboard, giving high accuracy, but slow speed, or it can be done with the mouse with higher speed but less accuracy. The manual techniques all have in common the amount of time spent in tracking the objects, and human resources are expensive. Tracing algorithms can also make use of geometrical figures like ellipses to approximate the boundaries of the objects. This has been done a lot for medical purposes, but the approximations may not be very good.

3.3.2 Automatic segmentation

Fully automatic segmentation is difficult to implement due to high complexity and variations of images. Most algorithms need some a priori information to carry out the segmentation, and for a method to be automatic, this priori information could for instance be noise level and probability of the objects having a special distribution.

3.3.3 Semiautomatic segmentation

Semiautomatic segmentation combines the benefits of both manual and automatic segmentation. By giving some initial information about the structures, we can proceed with automatic methods.

3.4 CLASSIFICATION OF SEGMENTATION

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

- Region based segmentation technique that finds the regions satisfying a given homogeneity criterion
- 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 homogeneous region. Criteria for homogeneity are most of the time gray levels of the 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 an image. And also there is not any unique thresholding method so these affect the output segmentation depending upon different methods of thresholding.

Fig. 3.4(a) Original image

Fig 3.4(b) Region based segmentation

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 next. This technique divides an image on the basis of boundaries. The limitation of edge based method includes performance degradation by presence of noise. If there is some type of noise in an 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 over segmentation and these techniques are required to be used in conjunction with region based technique for complete segmentation.

Fig. 3.5(a) Original image

Fig. 3.5(b) Discontinuity based segmentation

Table3.1 Difference between region based and edge based methods

S. No.	Region Based	Edge Based
1.	Closed boundaries	Boundaries formed not necessarily closed
2.	Multi spectral images improve segmentation	No significant improvement for multi spectral images
3.	Computation based on similarity	Computation based on difference

3.5 METHODS OF IMAGE SEGMENTATION

Several general purpose algorithms and techniques have been developed for image segmentation. These are listed below:

- Thresholding methods
- Region growing methods
- Level set methods

3.6 THRESHOLDING METHOD

Thresholding is the simplest method of image segmentation. Thresholding consists of segmenting an image into two regions: a particle region and a background region. In its most simple form, this process works by setting to white all pixels that belong to a gray-level interval, called the threshold interval, and setting all other pixels in the image to black. The resulting image is referred to as a binary image.

The threshold can be chosen manually or by using automated techniques. Manual threshold selection is normally done by trial and error, using a histogram as a guide. The example below shows a threshold chosen to isolate the brightest particles from an image:

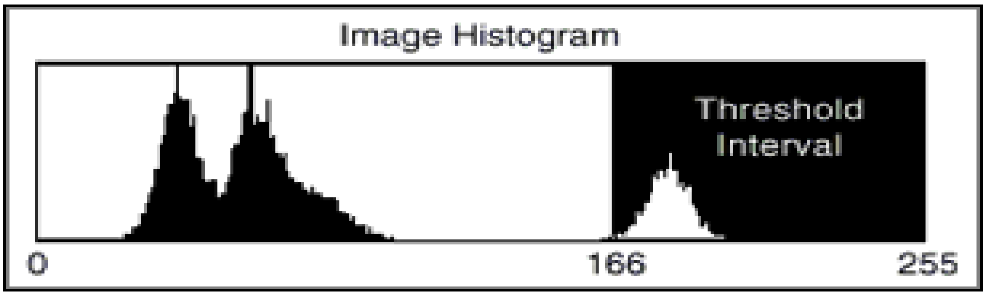


Fig. 3.6 Histogram of image

Automated thresholding techniques select a threshold which optimizes a specified characteristic of the resulting images. These techniques include clustering, entropy, metric, moments, and interclass variance. Clustering is unique in that it is a multi-class thresholding method. In other words, instead of producing only binary images it can specify multiple threshold levels which result in images with three or more gray-level values.

3.6.1 Method

During the thresholding process, individual pixels in an image are marked as "object" pixels if their value is greater than some threshold value (assuming an object to be brighter than the background) and as "background" pixels otherwise. This convention is known as threshold above. Variants include threshold below, which is opposite of threshold above; threshold inside, where a pixel is labelled "object" if its value is between two thresholds; and threshold outside, which is the opposite of threshold inside. Typically, an object pixel is given a value of “1” while a background pixel is given a value of “0.” Finally, a binary image is created by coloring each pixel white or black, depending on a pixel's labels.

Fig 3.7(a) Original Image

Fig. 3.7(b) Threshold Effect on Image

3.6.2 Threshold selection

The key parameter in the thresholding process is the choice of the threshold value (or values, as mentioned earlier). Several different methods for choosing a threshold exist; users can manually choose a threshold value, or a thresholding algorithm can compute a value automatically, which is known as automatic thresholding. A simple method would be to choose the mean or median value, the rationale being that if the object pixels are brighter than the background, they should also be brighter than the average. In a noiseless image with uniform background and object values, the mean or median will work well as the threshold, however, this will generally not be the case. A more sophisticated approach might be to create a histogram of the image pixel intensities and use the valley point as the threshold. The histogram approach assumes that there is some average values for both the background and object pixels, but that the actual pixel values have some variation around these average values. However, this may be computationally expensive, and image histograms may not have clearly defined valley points, often making the selection of an accurate threshold difficult. In such cases a uni-modal threshold selection algorithm may be more appropriate. One method that is relatively simple, does not require much specific knowledge of the image, and is robust against image noise, is the following iterative method:

1. An initial threshold (T) is chosen, this can be done randomly or according to any other method desired.
2. The image is segmented into object and background pixels as described above, creating two sets:
    1. G_1={f(m,n):f(m,n)>T} (object pixels) (3.1)
    2. G_2=={f(m,n):f(m,n)≤T} (background pixels) (3.2)
3. The average of each set is computed.
    1. m_1= average value of G_1
    2. m_2= average value of G_2
4. A new threshold is created that is the average of m_1 and m_2
    1. $T' = (m_1 + m_2)/2$ (3.3)
5. Go back to step two, now using the new threshold computed in step four, keep repeating until the new threshold matches the one before it (i.e. until convergence has been reached).

3.6.3 Classification

Thresholding can be classified as:

- bi-level thresholding
- multi-level thresholding.

Bi-level thresholding classifies the pixels of an image into two classes, one including those pixels with gray levels above a certain threshold, the other including the rest. Bimodal images are given as:

$$g(x,y) = \begin{cases} 1 & if\ f(x,y) > T \\ 0 & if\ \ f(x,y) \leq T \end{cases} \tag{3.4}$$

Where, T is constant applicable over an entire range.

Multi-level thresholding divides the pixels into several classes. The pixels belonging to same class have gray levels within a specific range defined by several threshold. Any multimodal image g(x,y) can be described as:

$$g(x,y) = \begin{cases} a & if\ g(x,y) > T_2 \\ b & if\ T_1 < g(x,y) \leq T_2 \\ c & if\ g(x,y) \leq T_1 \end{cases} \tag{3.5}$$

- **Advantages of thresholding method**
- Very fast
- The threshold segmentation method is easy to grasp
- **Disadvantages of thresholding method**
- The threshold method depends on the possibility to define a threshold that works well everywhere in the image
- Require a region growing or other technique of segmentation if two objects have the same color.

3.7 REGION GROWING METHOD

Region growing is a procedure that group pixels or sub-regions into larger regions based on pre-defined criteria for growth. The basic approach is to start with a set of seed points and from these grow regions by appending to each seed those neighbouring pixels that have predefined properties similar to seed (such as specific ranges of color or intensity). Homogeneity of regions is used as main segmentation criterion in region growing. The criterion for homogeneity is as:

- Gray level
- Color
- Texture
- Shape
- Model

3.7.1 Seed point selection

The first step in region growing is to select a set of seed points. Seed point selection is based on some user criterion (for example, pixels in a certain gray-level range, pixels evenly spaced on a grid, etc.). The initial region begins as the exact location of these seeds.

The regions are then grown from these seed points to adjacent points depending on a region membership criterion. The criterion could be, for example, pixel intensity, gray level texture, or color.

Since the regions are grown on the basis of the criterion, the image information itself is important. For example, if the criterion were a pixel intensity threshold value, knowledge of the histogram of the image would be of use, as one could use it to determine a suitable threshold value for the region membership criterion.

There is a very simple example followed below. Here we use 4-connected neighborhood to grow from the seed points. We can also choose 8-connected neighborhood for our pixels adjacent relationship. And the criteria we make here is the same pixel value. That is, we keep examining the adjacent pixels of seed points. If they have the same intensity value with the seed points, we classify them into the seed points. It is an iterated process until there is no change in two successive iterative stages. Of course, we can make other criteria, but the main goal is to classify the similarity of the image into regions.

Here we show a simple example for region growing.

Figure 3.8(a) is the original image which is a gray-scale lightning image. The gray-scale value of this image is from 0 to 255. The purpose we apply region growing on this image is that we want to mark the strongest lightning part of the image and we also want the result to be connected without being split apart. Therefore, we choose the points having the highest gray-scale value which is 255 as the seed points showed in the figure 3.8(b).

Fig. 3.8(a) Original image

Fig. 3.8(b) Seed point

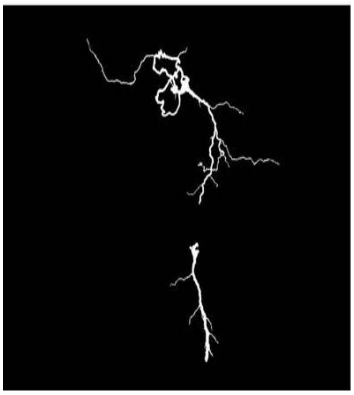


Fig. 3.8(c) Threshold: 225~255

Fig. 3.8(d) Threshold 190~255

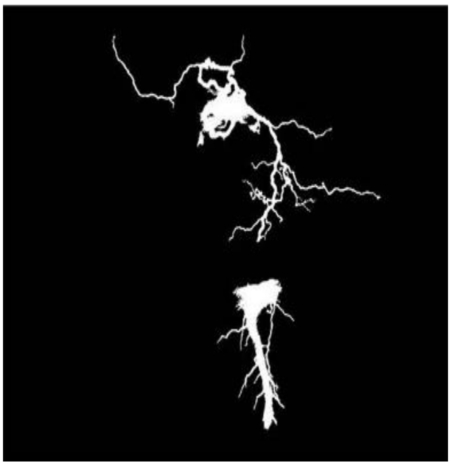


Fig. 3.8(e) Threshold 155~255

After determining the seed points, we have to determine the range of threshold. Always keep in mind that the purpose we want to do is to mark the strongest light in the image. The third figure is the region growing result from choosing the threshold between 225 and the value of seed points (which is 255). It means we only want to mark out the points whose gray-scale values are above 225.

If we make the range of threshold wider, we will get a result having a bigger area of the lightning region show as figure 3.8(c) and figure 3.8(d).

We can observe the difference between the last two figures which have different threshold value showed above. Region growing provides the ability for us to separate the part we want connected.

As we can see in figure 3.8(c) to figure 3.8(e), the segmented result in this example is seed-oriented connected. That means the result grew from the same seed points are the same regions. And the points will not be grown without connected with the seed points in the beginning. Therefore, we can mention that there are still lots of points in the original image having the gray-scale value above 155 which are not marked in Figure. 5. This characteristic ensures the reliability of the segmentation and provides the ability to resist noise. For this example, this characteristic prevents us marking out the non-lightning part in the image because the lightning is always connected as one part.

3.7.2 Method

Start from a set of seed point and from these points grow the regions by appending to each seed those neighboring pixels that have similar properties. The selection of seed points depend on the problem. The selection of similarity criteria depends on the problem under consideration and the type of image data available. Growing a region should stop when no more pixels satisfy the criteria for inclusion in that region.

Let R: the entire image. The segmentation process partition R into n sub-regions R_1,… R_n, such that:

- $\bigcup_{i-1}^{n} R_i$, Where, R_i is a connected region & i=1, 2,…., n. (3.6)
- $R_i \cap R_j = \varphi$ for all i and j, i≠j (3.7)
- $P(R_i)$ = TRUE for i=1,2,…,n (3.8)
- $P(R_i \cup R_j)$ = FALSE for , i≠j (3.9)

Here, P(R_i) is logical predictive defined over all points in R_i.

The region growing technique is an iterative process by which regions are merged starting from individual pixels, or another initial segmentation, and growing iteratively until every pixel is processed. Region based segmentation always provides closed contour regions which is a requirement in many applications. Besides, it is very simple and effective in many applications. Errors in region boundaries are the main drawback of this approach: edge pixels might be joined to any of the neighboring regions. Among region based segmentation approaches is region growing method which is very effective.

- **Advantages**

- Region growing methods can correctly separate the regions that have the same properties we define.
- Region growing methods can provide the original images which have clear edges the good segmentation results.
- The concept is simple. We only need a small numbers of seed point to represent the property we want, then grow the region.
- We can determine the seed points and the criteria we want to make.
- We can choose the multiple criteria at the same time.
- It performs well with respect to noise.

- **Disadvantages**

- The computation is consuming, no matter the time or power.
- Noise or variation of intensity may result in holes or over-segmentation.
- This method may not distinguish the shading of the real images.
- Result is very sensitive to the threshold value

3.8 LEVEL SET METHOD

The level set method was initially proposed to track moving interfaces by Osher and Sethian [34] and has spread across various imaging domains in the late nineties. It can be used to efficiently address the problem of curve/surface/etc. propagation in an implicit manner. The central idea is to represent the evolving contour using a signed function, where its zero level corresponds to the actual contour. Then, according to the motion equation of the contour, one can easily derive a similar flow for the implicit surface that when applied to the zero-level will reflect the propagation of the contour. Furthermore, they can be used to define an optimization framework as proposed by Zhao, Merriman and Osher in 1996. Therefore, one can conclude that it is a very convenient framework to address numerous applications of computer vision and medical image analysis. Furthermore, research into various level set data structures has led to very efficient implementations of this method. The existing level set models can be classified as either parametric level set or geometric level set according to their representation and implementation.

Table 3.2 Comparison between parametric level set and geometric level set

S. No.	Parametric level set	Geometric level set
1	Are represented explicitly as parameterized curves in lagrangian framework	Are represented implicitly as level sets of two-dimensional function that evolves in an eulerian framework
2	Not based on curve evolution theory	Are based on curve evolution theory and level set

3.8.1 Method

The fig 3.10 below illustrates several important ideas about the level set method. In the upper-left corner we see a shape; that is, a bounded region with a well-behaved boundary. Below it, the red surface is the graph of a level set function ϕ determining this shape, and the flat blue region represents the (x-y) plane. The boundary of the shape is then the zero level set of ϕ, while the shape itself is the set of points in the plane for which ϕ is positive (interior of the shape) or zero (at the boundary).

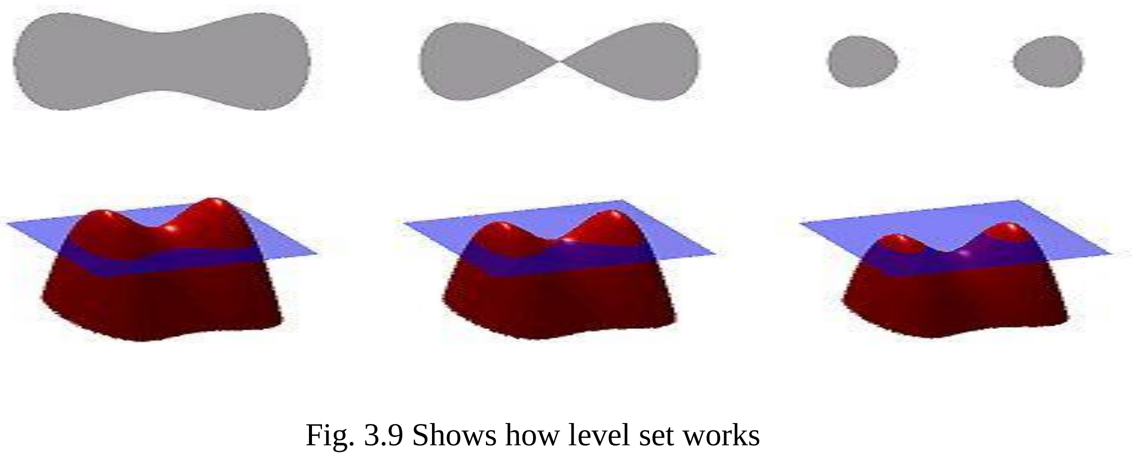


Fig. 3.9 Shows how level set works

In the top row we see the shape changing its topology by splitting in two. It would be quite hard to describe this transformation numerically by parameterizing the boundary of the shape and following its evolution. One would need an algorithm able to detect the moment the shape splits in two, and then construct parameterizations for the two newly obtained curves. On the other hand, if we look at the bottom row, we see that the level set function merely got translated downward. We see that it is much easier to work with a shape through its level set function than with the shape directly, where we would need to watch out for all the possible deformations the shape might undergo.

Thus, in two dimensions, the level set method amounts to representing a closed curve Γ (such as the shape boundary in our example) using an auxiliary function ϕ, called the level set function. Γ is represented as the zero level set of ϕ by

$$\Gamma=\{(x,y) \mid \phi(x,y) = 0\}; \tag{3.10}$$

The level set method manipulates Γ implicitly, through the function ϕ. ϕ is assumed to take positive values inside the region delimited by the curve Γ and negative values outside.

If the curve Γ moves in the normal direction with a speed v, then the level set function ϕ satisfies the level set equation

$$\frac{\partial \phi}{\partial t} = V|\nabla \phi|. \tag{3.11}$$

Here, $|.|$ is the Euclidean norm (denoted customarily by single bars in PDEs), and t is time. This is a partial differential equation, in particular a Hamilton-Jacobi equation, and can be solved numerically, for example by using finite differences on a Cartesian grid.

The numerical solution of the level set equation, however, requires sophisticated techniques. Simple finite difference methods fail quickly. Upwinding methods, such as the Godunov method, fare better; however the level set method does not guarantee the conservation of the volume and the shape of the level set in an advection field that does conserve the shape and size, for example uniform or rotational velocity field. Instead, the shape of the level set may get severely distorted and the level set may vanish over several time steps. For this reason, high-order finite difference schemes are generally required, such as high-order essentially non-oscillatory (ENO) schemes, and even then, the feasibility of long-time simulations is questionable.

- **Advantages**
- It is implicit, parameter free
- Provides a direct way to estimate the geometric properties of the evolving structure
- Can change the topology and is intrinsic.
- **Disadvantages**
- Algorithm is computationally expensive
- In order to get the optimized results, we have to change the values of parameters according to the image

Some other segmentation methods:

3.9 HISTOGRAM BASED METHOD

In computer vision and image processing, Otsu's method is used to automatically perform histogram shape-based image thresholding or, the reduction of a gray level image to a binary image. The algorithm assumes that the image to be thresholded contains two classes of pixels or bi-modal histogram (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal. The extension of the original method to multi-level thresholding is referred to as the Multi Otsu method.

Fig. 3.10(a) Original Image

Fig. 3.10(b) Thresholding Using Otsu's Method

3.10 CLUSTERING METHODS

Clustering in image segmentation is defined as the process of identifying groups of similar primitive. A cluster is therefore a collection of objects which are similar between them and are dissimilar to the objects belonging to other clusters. Clustering techniques can be classified into supervised clustering and unsupervised clustering.

Table 3.3 Comparison between supervised clustering and un-supervised clustering

S. No.	Supervised clustering	Unsupervised clustering
1	Demands human interaction to decide the clustering criteria	Decides the clustering content by itself
2	Includes hierarchical approaches such as relevance feedback techniques	Includes density based clustering methods
3	Grouping is done according to user feedback	Does not require user feedback

An image can be grouped based on keyword or its content. In keyword based clustering, a keyword is a form of font which describes about image keyword of an image refers to its different features. The similar featured images grouped to form a cluster by assigning value to each feature. In content based clustering, content refers to shape, textures or any other information that can be inherited from the image itself. The tools, techniques and algorithms that are used originate from fields such as statistics, pattern recognition, signal processing etc. Most of the clustering techniques are based on two popular techniques known as hierarchical and partitional clustering.

Table 3.4 Comparison between hierarchical and partitional clustering

S. No.	Hierarchical clustering	Partitional clustering
1	The number of clusters need not to be specified	The number of clusters need be specified
2	They are independent of initial clusters	They are dependent on initial clusters
3	They are static, data points assigned to cluster cannot move to another cluster	They are dynamic, data points assigned to cluster can move to

		another cluster
4	They may fail to separate overlapping clusters due to lack of information about the global shape or size of the cluster	They are able to separate overlapping clusters

3.11 EDGE DETECTION METHOD

Edge detection is a well-developed field on its own within image processing. Region boundaries and edges are closely related, since there is often a sharp adjustment in intensity at the region boundaries. Edge detection techniques have therefore been used as the base of another segmentation technique. The edges identified by edge detection are often disconnected. To segment an object from an image however, one needs closed region boundaries. The desired edges are the boundaries between such objects. The main aim of edge detection is to:

- De-noise: suppress image noise without losing the real edges.
- Edge enhancement: use filters that give only large response to edges.
- Edge localization

There are number of edge detectors used for segmentation process. Some of them are roberts, prewitt and sobel

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0		0	−1
0	1		1	0

Roberts

−1	−1	−1		−1	0	1
0	0	0		−1	0	1
1	1	1		−1	0	1

Prewitt

−1	−2	−1		−1	0	1
0	0	0		−2	0	2
1	2	1		−1	0	1

Fig. 3.11 Show various edge detectors

Table 3.5 Difference between various edge detectors

Edge Detector	Advantages	Disadvantages
Roberts	1. Simple, i.e. the kernel is small and contains only integers 2. Works well with binary images	1. Slow 2. High sensitivity to noise 3. Few pixels are used to approximate the gradient
Prewitt	1. Inexpensive in terms of computation	
Sobel	1. Larger convolution kernel smoothens input image to greater extent 2. Less sensitive to noise 3. Detect thicker edges	1. Slower to compute

Fig. 3.12(a) Original Image

Fig. 3.12(b) Gradient magnitude from Sobel

Fig. 3.12(c) Gradient magnitude from Robert

Fig. 3.12(d) Gradient magnitude from Prewitt

3.12 PARTICLE SWARM OPTIMIZATION (PSO)

Particle swarm optimization, first introduced by Kennedy and Eberhart [35], is a stochastic optimization technique that can be likened to the behaviour of a flock of birds or the sociological behaviour of a group of people. It is a kind of swarm intelligence that is based on social-psychological principles and provides insights into social behaviour, as well as contributing to engineering applications.

- Particle swarm optimization is a robust stochastic optimization technique based on the movement and intelligence of swarms.
- Particle swarm optimization applies concept of social interaction to problem solving.

- It uses a number of agents (particles) that constitute a swarm moving around in the search space looking for the best solution.
- Each particle is treated as a point in N-dimensional space.
- Each particle keeps track of its coordinates in the solution space which are associated with the best solution (fitness). This value is called particle best pbest.
- Another best value that is tracked by PSO is the best value obtained so far by any particle in the neighbourhood of that particle. This value is called global best gbest.
- The basic concept of Particle swarm optimization lies in accelerating each particle towards its pbest and the gbest locations**.**

3.12.1 Particle swarm optimization operation:

Consider a scenario in which a flock of birds are searching for a piece of food in an area. All the birds do not exactly know where the food is, but with every iteration, they come to know how far the food is. The best strategy will be to follow the bird which is near to food and also from its own previous best position. This is the basic idea on which PSO works.

The Particle swarm optimization is a population based optimization technique, where the population is called a swarm. A simple explanation of the Particle swarm optimization operation is as follows. Each particle represents a possible solution to the optimization task at hand. With every iteration, each particle accelerates in the direction of its own personal best solution found so far, as well as in the direction of the global best position discovered so far by any of the particles in the swarm. This means that if a particle discovers a promising new solution, all the other particles will move closer to it, exploring the region more thoroughly in the process. In Particle swarm optimization algorithm the five essential parameters [35] that are considered are as tabulated in table 3.6.

Table 3.6 PSO parameters

Parameters	Description
Particle	Candidate solution to a problem
Velocity	Rate of position change
Fitness	The best solution achieved
pbest	Best value obtained in previous particle
gbest	Best value obtained so far by any particle in the population

PSO optimises an objective function by undertaking a population-based search. The population consists of potential solutions, named particles, which are a metaphor of birds in flocks. These particles are randomly initialised and freely fly across the multi-dimensional search space. During flight, each particle updates its own velocity and position based on the best experience of its own and the entire population. The updating policy drives the particle swarm to move toward the region with the higher objective function value, and eventually all particles will gather around the point with the highest objective value. The detailed operation of particle swarm optimisation [36] is given below:

Step 1: Initialisation. The velocity and position of all particles are randomly set to within pre-defined ranges.

Step 2: Velocity Updating. With every iteration, the velocities of all particles are updated according to:

$$\vec{v_i} = w\vec{v_i} + c_1 R_1(\vec{p}_{i,best} - \vec{p_i}) + c_2 R_2(\vec{g}_{i,best} - \vec{p_i}) \tag{3.12}$$

Where $\vec{v_i}$ and $\vec{p_i}$ are the velocity and position of particle I, respectively. $\vec{p}_{i,best}$ and

$\vec{g}_{i,best}$ are the position with the 'best' objective value found so far by particle *i* and the entire population respectively; *w* is a parameter controlling the flying dynamics; R_1 and R_2 are random variables in the range [0, 1]; c_1 and c_2 are factors controlling the related weighting of corresponding terms. The inclusion of random variables endows the PSO with the ability of stochastic searching. The weighting factors, c_1 and c_2, compromises the inevitable trade-off between exploration and exploitation. After updating, velocity should be checked and secured within a pre-specified range to avoid violent random walking.

Step 3: Position Updating. Assuming a unit time interval between successive iterations, the positions of all particles are updated according to:

$$\overrightarrow{p_1} = \overrightarrow{p_\iota} + \overrightarrow{v_\iota} \tag{3.13}$$

After updating, p_i should be checked and limited to the allowed range

Step 4: Memory Updating: Update $\vec{p}_{i,best}$ and $\vec{g}_{i,best}$ when condition is met

$$\vec{p}_{i,best} = \vec{p}_i \quad \text{if } f(\vec{p}_i) > f(\vec{p}_{i,best}) \tag{3.14}$$

$$\vec{g}_{i,best} = \vec{g}_i \quad \text{if } f(\vec{g}_i) > f(\vec{g}_{i,best}) \tag{3.15}$$

Step 5: Termination Checking. The algorithm repeats Steps 2 to 4 until certain termination conditions are met, such as a pre-defined number of iterations or a failure to make progress for a certain number of iterations. Once terminated, the algorithm reports the values of $\vec{g}_{best}$ and f($\vec{g}_{best}$) as its solution.

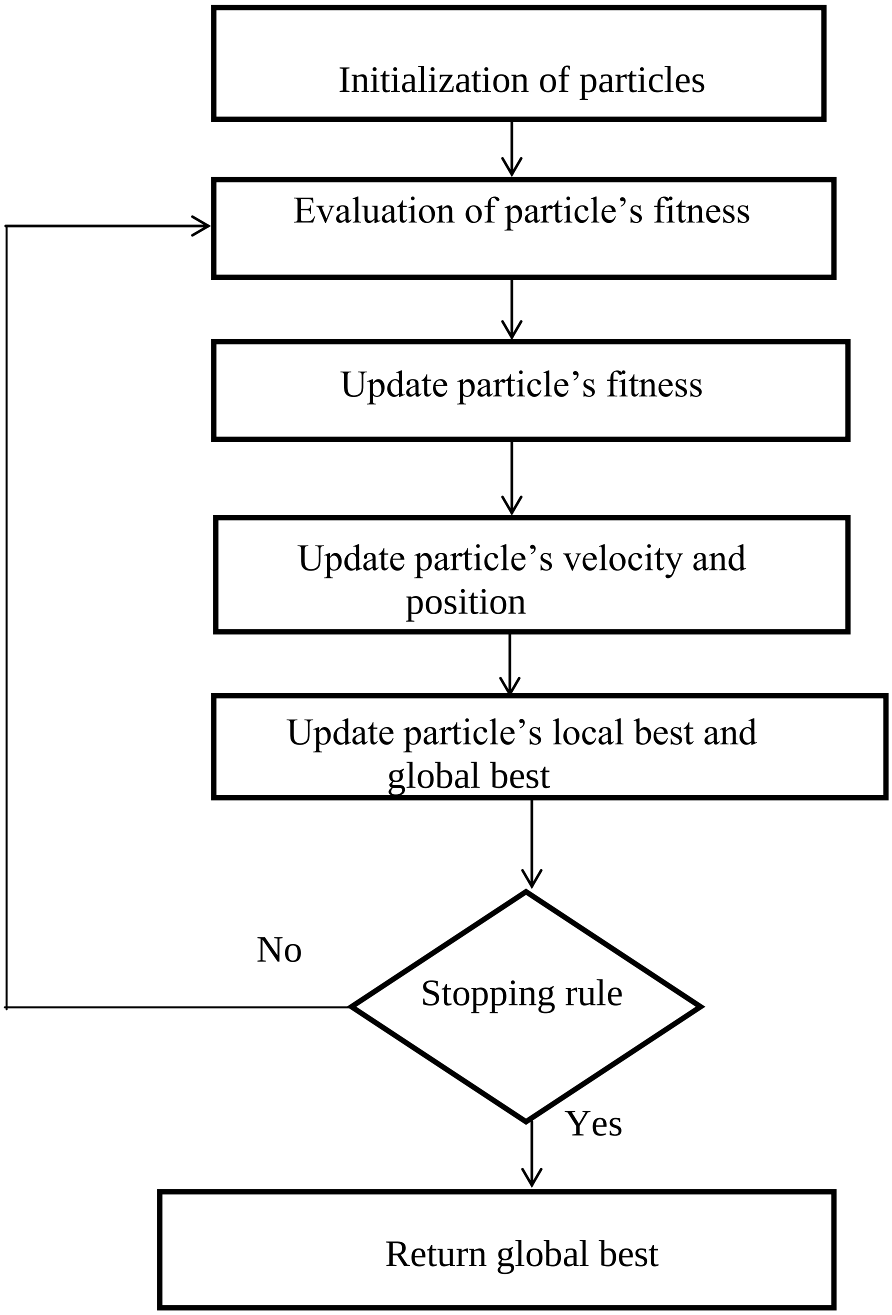


Fig. 3.13 Flowchart for particle swarm optimization

CHAPTER 4
METHODOLOGY

In this thesis work, two techniques for segmenting diseased portion of CT scan images have been used. The techniques used are listed below:

- Region based active contour method
- Hybrid level set method

4.1 METHOD 1

REGION BASED ACTIVE CONTOUR METHOD

Our special thanks to Chan *et. al.* [37] Vese for their region based active contour MATLAB code. It can be used to detect objects whose boundaries are not necessarily defined by gradient. In this technique, energy is minimized which can be seen as a particular case of the minimal partitional problem.

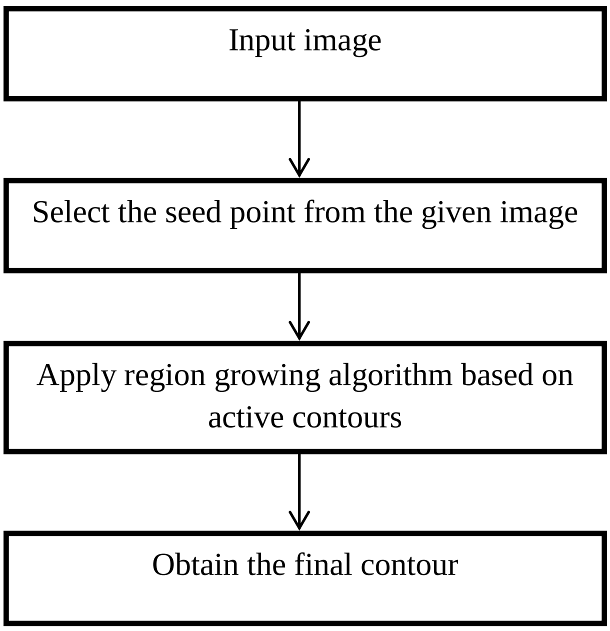


Fig 4.1 Flowchart of original algorithm

The drawback of this algorithm is how to control the number of iterations when the final contour is obtained. Since this may lead to over segmentation or under segmentation. Along with this, more time is required for this algorithm, therefore decreases its efficiency.

4.2 ALGORITHM FOR REGION BASED ACTIVE CONTOUR

This section explains the detailed description of various steps involved in the region based active contour method.

Step 1:- Firstly the image was read with the help of imread command. This command is inbuilt command in MATLAB processing toolbox.

Step 2:- The seed point in the form of mask is selected manually from the given image.

Step 3:- Create the signed distance map (SDF) from the mask.

Step 4:- Initialize the number of iterations.

Step 5:- Obtain the curve's narrow band in this step.

Step 6:- Calculate the interior and exterior points from SDF. With the help of interior and exterior points obtain interior and exterior mean.

Step 7:- Calculate force from image information and curvature along SDF. Obtain gradient to minimize the energy.

Step 8:- Maintain the CFL condition and evolve the curve using SDF function.

Step 9:- Keep SDF smooth using forward and backward difference and obtain mask from level set.

Step 10:- If number of iterations are completed obtain final output, else go to step 5

4.3 MODIFIED ALGORITHM FOR REGION BASED ACTIVE CONTOUR METHOD

The original algorithm does not have automatic convergence. It means that if the segmentation is completed before the given number of iterations, the algorithm will not stop and shall work until the given iterations were over. Therefore, it may lead to more time consumption and decreases the efficiency of the program.

In modified algorithm, automatic convergence for the region based active contour algorithm is proposed. The proposed algorithm overcomes the drawback of over segmentation or under segmentation. With iterations, the area of the segmented region is calculated and then is compared with the previous area. Two conditions arise which are listed below:

- If both the areas are same, then stop further iterations and obtain the final segmented image.
- If both the areas are not same, then go for further iterations until it becomes equal.

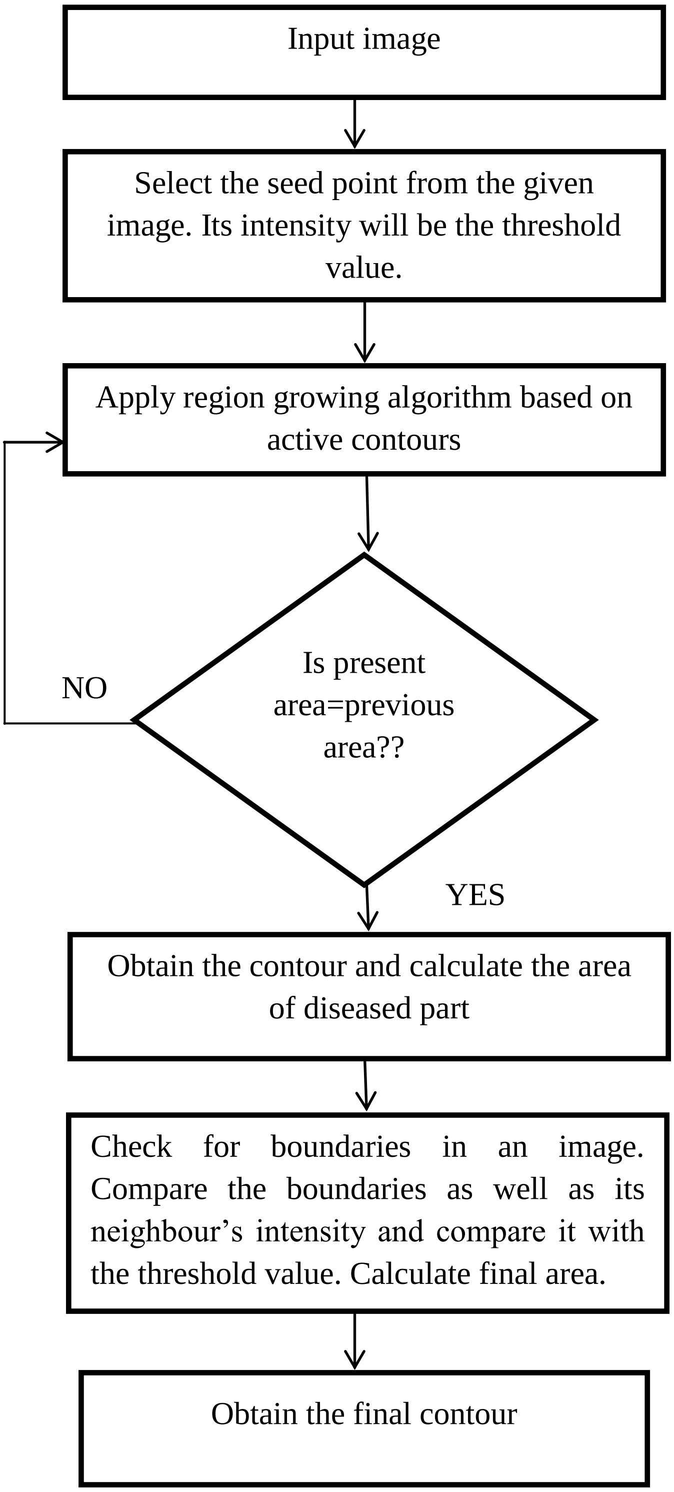


Fig 4.2 Flowchart of modified algorithm

For making the code optimized we have to control the number of iterations. So change is made after step 10, rest step 1 to step 10 are same.

Step 11:- Calculate the area of the diseased portion and compare it with previous area. If the areas are same then break the algorithm and display the output, else return to step 5.

Step 12:- Check for boundaries around the contour obtained in step 11.

Step 13:- Compare the intensity value of pixel at boundaries with the threshold value. Also check for boundaries neighbour's pixels. The threshold value is the intensity value of the selected seed pixel.

Step 14:- If intensity value of pixel is greater than threshold, then include that pixel to diseased part. Else transfer it to non-diseased part.

Step 15:- Display the final segmented image.

4.4 METHOD 2: COMBINATION OF LEVEL SETS AND PARTICLE SWARM OPTIMIZATION

First of all we would like to thanks Li *et. al.* [38] for his MATLAB code for level set without re-initialization. This work is to modify the existing code to get better results. The description and modification that has done is described in this chapter.

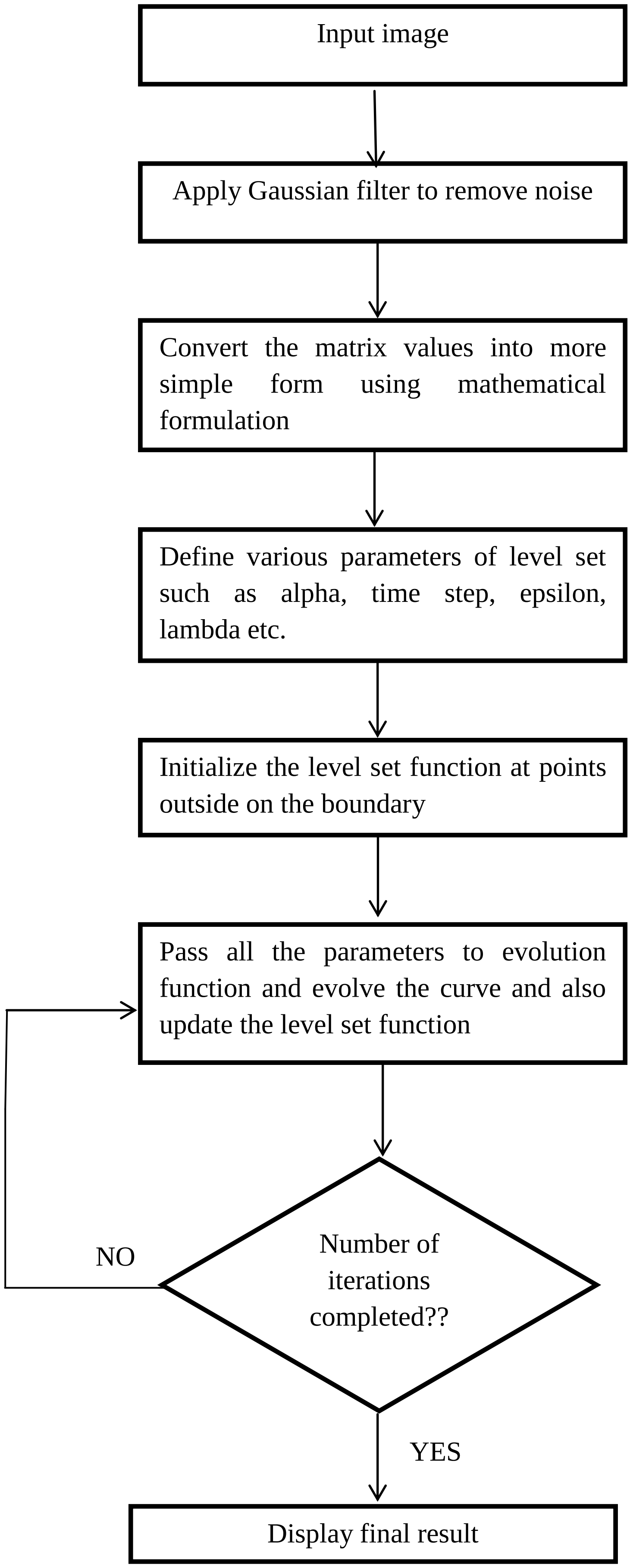


Fig 4.3 Flowchart of original level set based algorithm

4.5 ALGORITHM FOR ORIGINAL LEVEL SET METHOD

Step 1:- Firstly the image was read with the help of imread command. This command is inbuilt command in MATLAB processing toolbox.

Step 2:- The image is filtered with the help of Gaussian filter. The Gaussian filter is the basic filter, used to remove the noise from the image so as to make image more sharp and smooth. The Gaussian filter smoothens or blurs an image by performing a convolution operation with a Gaussian filter kernel.

Step 3:- Now the matrix values are converted into more uniform and simplify form so that further calculation can become easy with the help of mathematical formula

$$f = I\,x\text{^}2+I\,y\text{^}2 \tag{4.1}$$

Step 4:- In this step all parameters are defined which change the topology of the level set speed and stability. The parameters are alpha, time step, MU, lambda and epsilon.

Step 5:- Now pre-processed image is further processed and its gradient is calculated. This gradient image is used to calculate the edges of an image. For calculation of edges following function is used:

$$g=1/1+f \tag{4.2}$$

Step 6:- Initialization of level set means starting shape of contour which depends upon the region. Define the level set function at points outside on the boundary.

Step 7:- In this step we pass all the parameters and initialize level set to the evolution function. In the evolution function, make a function that satisfies the Neumann boundary condition. Calculate the gradient of the above function. Evolve the curve using dirac function and curvature. At last update the level set function.

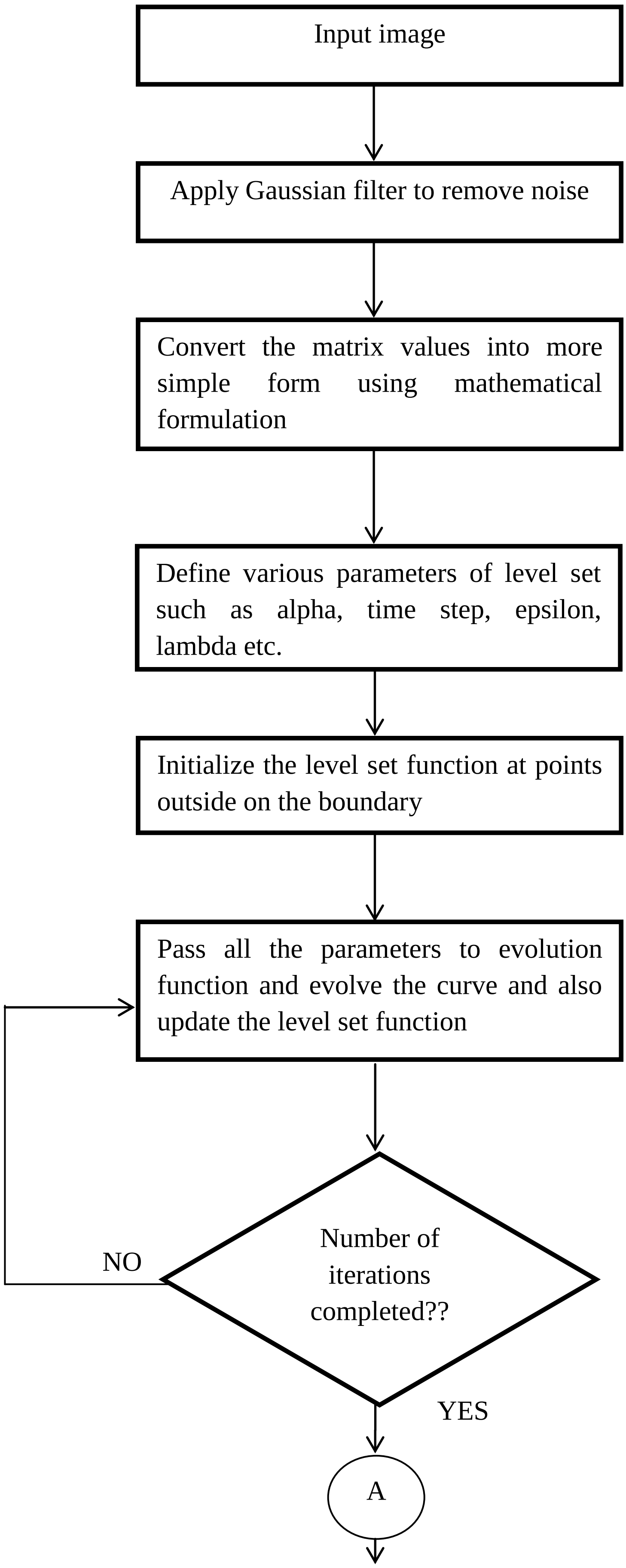
Step 8:- The step 7 repeats until we do not get the final level set. The repetition depends on the given number of iterations.

Step 9:- In this step, final level set will be displayed.

4.6 MODIFIED ALGORITHM FOR LEVEL SET METHOD

The original method has a drawback that it does not segment accurately the diseased portion. In some cases it may lead to under segmentation, while in other cases it may lead to over segmentation. Also this algorithm starts without re-initialization i.e. the contour starts from the boundaries of the image and then converges to diseased portion. So, number of iterations required increases and results in more time consumption.

In this thesis work the original algorithm is modified such that it produces better results. The mean intensity value of diseased portion is obtained using the segmented portion of the original algorithm and this value is taken as threshold value. Across boundaries small region of interest (ROI's) are made and texture parameters are calculated. Texture parameters are compared with the data set. Particle swarm optimization (PSO) is used to optimize the sum of texture parameters. Finally thresholding is applied across the ROI which satisfy the given condition and final segmented image is obtained.


Input image
Apply Gaussian filter to remove noise
Convert the matrix values into more simple form using mathematical formulation
Define various parameters of level set such as alpha, time step, epsilon, lambda etc.
Initialize the level set function at points outside on the boundary
Pass all the parameters to evolution function and evolve the curve and also update the level set function
Number of iterations completed??
NO
YES
A

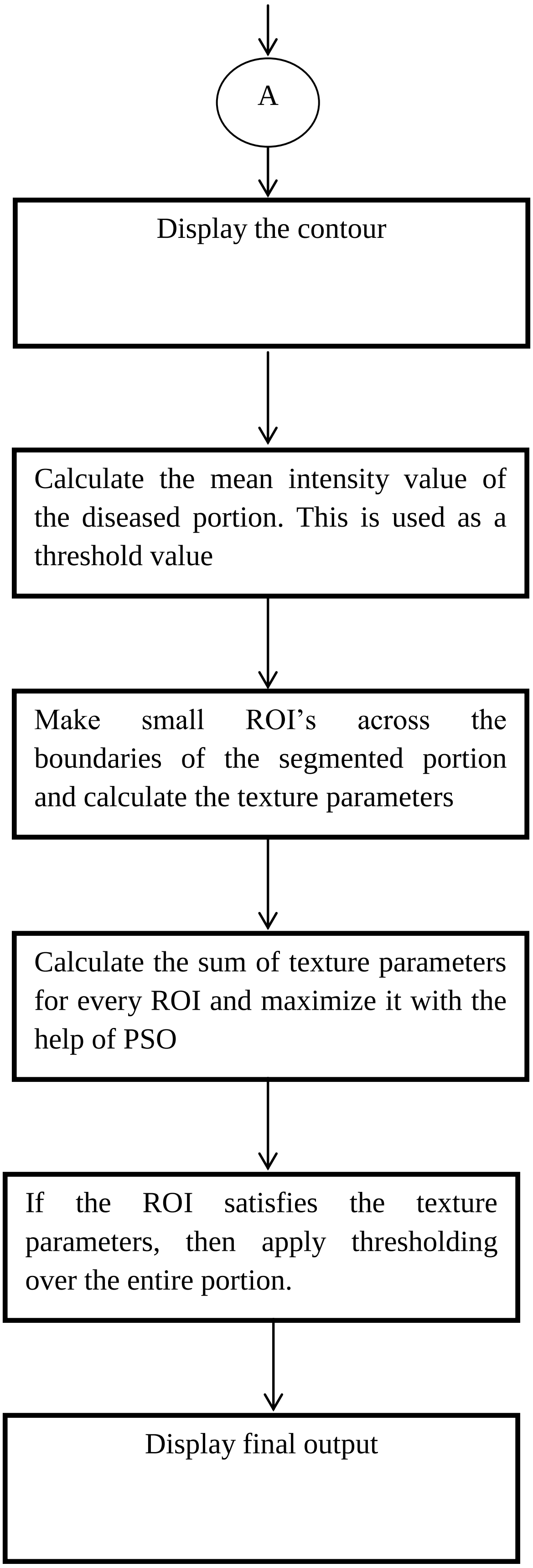


Fig 4.4 Flowchart of modified algorithm

For making the code optimized we have to control over segmentation and under segmentation. So change is made after step 9, rest step 1 to step 9 are same.

Step 10:- Calculate the mean of diseased portion obtained in step 9. We will assume this value as threshold value.

Step 11:- Across boundaries of diseased portion obtained in step 9, draw a number of region of interest (ROI) and calculate the texture parameters. Obtain the sum of texture parameters for each ROI.

Step 12:- Particle swarm optimization (PSO) is used to maximize the sum of texture parameters. The sum of texture parameters of each ROI is compared with the dataset of manually obtained value.

Step 13:- If the ROI's sum of texture parameters equals the manually obtained value, then compare the intensity of each pixel and its neighbour's with the threshold value obtained in step 10. If the value is greater than threshold value then consider it as diseased portion, else put it into non diseased portion.

Step 14:- Display the final output.

4.7 TEXTURE PARAMETERS

Initially we considered 25 texture parameters for our algorithm. Using scattering matrix we concluded that out of these only three parameters were suitable for segmenting the diseased and non-diseased portion from the given image. Consider a two-dimensional image $f(x,y)$ where $x=0,1,.....N-1$ and $y=0,1,....M-1$. The function $f(x,y)$ can take G discrete intensity levels ranging from 0 to G-1.Then the intensity-level histogram is a function showing (for each intensity level) the number of pixels in the whole image, which have this intensity:

$$h(i)=\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}(\delta(f(x,y),i) \tag{4.3}$$

Where δ(j,i) is the Kronecker delta function .

$$\delta(j,i)=\begin{cases}1, & j=i\\0, & j\neq i\end{cases} \tag{4.4}$$

Dividing the values of h(i) by total number of pixels gives probability density of occurrence of the intensity levels. Mathematically it can be expressed as,

$$p(i)=h(i)/NM, \quad i=0,1,.....G-1 \tag{4.5}$$

The various texture parameters are listed below:

Table 4.1 Summary of all the texture measures

Texture Measure	**Mathematical Expression.**
First Order Statistics	
Mean	$\mu=\sum_{i=o}^{G-1} ip(i)$
Variance	$\sigma^2 = \sum_{i=o}^{G-1}(i-\mu)^2$
Skewness	$\mu 3=\sigma^{-3} \sum_{i=o}^{G-1}(i-\sigma)^3 p(i)$
Kurtosis	$\mu_4=\sigma^{-4} \sum_{i=o}^{G-1}(i-\mu)^4 p(i) -3$
Energy	$E=\sum_{i=o}^{G-1}[p(i)]^2$
Entropy	$H=-\sum_{i=o}^{G-1} p(i)^2 log_2[p(i)]$
Law's Texture Energy Measures	
TEM1	texture energy from *LL* kernel
TEM2	texture energy from *EE* kernel
TEM3	texture energy from *SS* kernel
TEM4	$LE=(LE+EL)/2$
TEM5	$ES=(ES+SE)/2$
TEM6	$LS= (LS+SL)/2$
Fourier Power Spectrum	
Radial Sum	$\Phi_{r1,r2} \equiv \sum_{r1*r1 \leq u*u<r2*r2}\|F(u,v)\|^2$
Angular Sum	$\Phi_{\theta 1,\theta 2} \equiv \sum_{\theta \leq(1/tan\left(\frac{v}{u}\right))\leq \theta 2}\|F(u,v)\|^2$
Statistical Feature Matrix	
Coarseness	$F_{CRC}=C/\sum_{(i,j)\epsilon Ne} \frac{DSS(i.j)}{n}$
Roughness	$F_{RGH}=(D_f^{(u)}+ D_f^{(v)}/2$
Periodicity	$F_{PER}= (\bar{M}_{dis}-M_{dis}(valley))/M_{dis}$
Contrast	$F_{CON}=[\sum_{(i,j)\epsilon Nr} CON(i,j)/4]^{1/2}$
Co-Occurrence Matrix Based Texture Features	

Angular Second Moment	$f_1 = \sum_i \sum_j \{p(i,j)\}^2$
Contrast	$f_2 = \sum_{i=1}^{N-1} n^2 \sum_i \sum_j \{p(i,j)\}$
Correlation	$f_3 = (\sum_i \sum_j (i,j)p(i,j) - \mu x * \mu y) / (\sigma x * \sigma y)$
Variance	$f_4 = \sum_i \sum_j (i-\mu)^2 p(i,j)$
Inverse Difference Moment	$f_5 = \sum_i \sum_j p(i,j)^2/(1+(i-j)^2$
Sum Average	$f_6 = \sum_{i=2}^{2Nx} i p_{x+y}(i)$
Sum Variance	$f_7 = \sum_{i=2}^{2Nx} (i - f6) p_{x+y}(i)$

Fischer's discriminate ratio (FDR) is used to calculate the best texture parameters. Out of 25 texture parameters, mean, median and radial sum are the best parameters which can accurately differentiate diseased and non-diseased portion. It has been shown using scattering matrix.

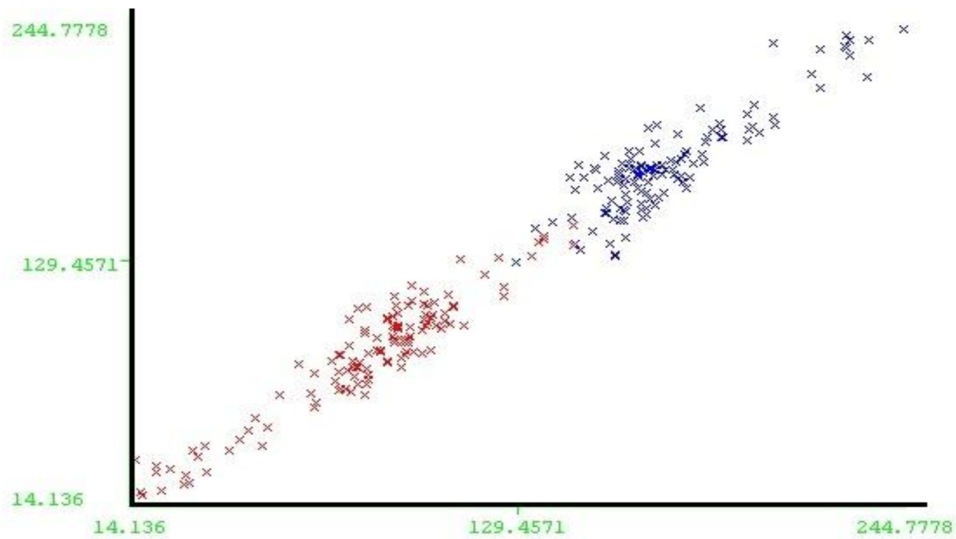


Fig 4.5 Plot of energy and median

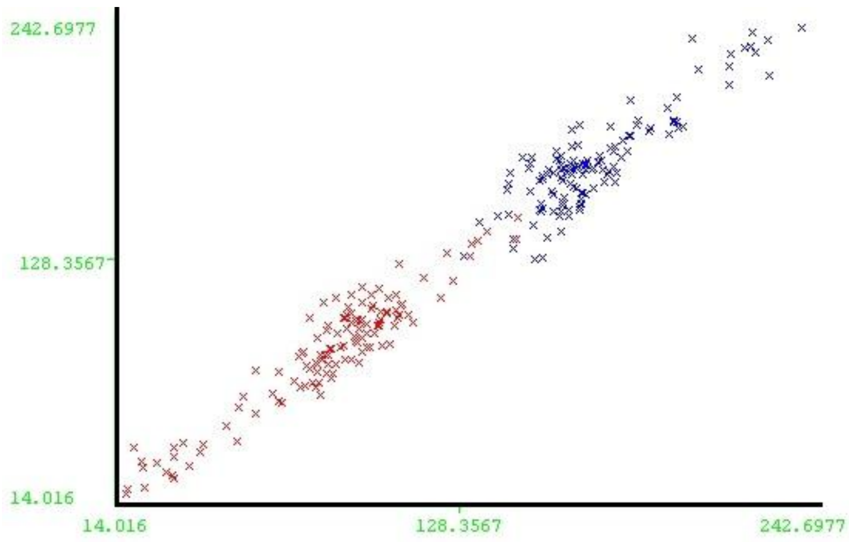


Fig 4.6 Plot of energy and mean

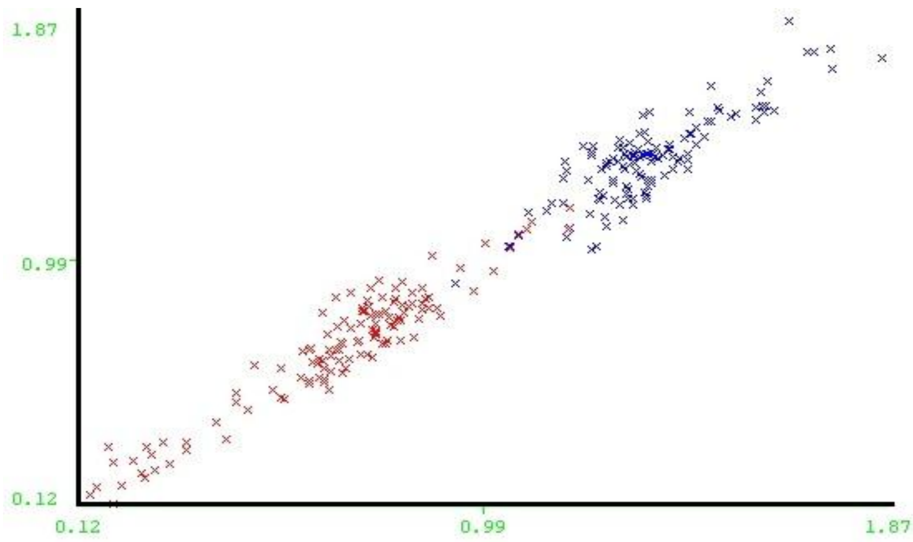


Fig 4.7 Plot of energy and radial sum

CHAPTER 5
RESULTS

This chapter deals with the results of the work at various levels. Firstly the results for region based segmentation are described and hybrid level set will be explained.

5.1 REGION BASED SEGMENTATION

Step 1:- Firstly the image was read with the help of imread command. This command is inbuilt command in MATLAB processing toolbox, fig 5.1(a).

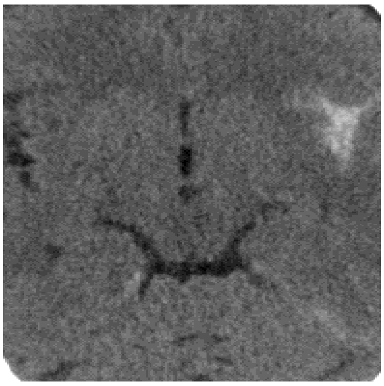


Fig 5.1(a) Original image

Step 2:- The seed point in the form of mask from the given image is selected manually from the given image.

Step 3:- Apply region based active contour to the input image, fig 5.1(b).

Fig 5.1(b) Segmented using region based active contour

Step 4:- Convert image in binary. The white portion is diseased portion. Calculate area of white portion, fig 5.1(c).

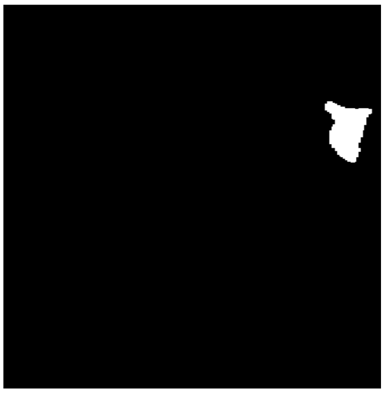


Fig 5.1(c) Binary image of region based active contour

Step 5:- Calculate the mean value of intensity of image obtained in step 4. It will be the threshold value in next step.

Step 6:- Compare the threshold value with the intensity of pixels at boundaries and also their neighbours of image in step 4. If intensity value of pixel is greater than threshold, then include that pixel to diseased part. Else transfer it to non-diseased part. Also calculate the area of white portion, fig 5.1(d).

Fig 5.1(d) Binary image of modified method (thresholding)

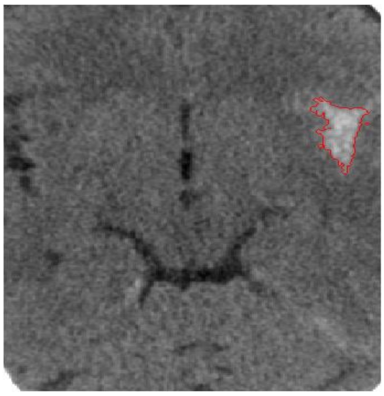


Fig 5.1(e) Final segmentation result

Step 7:- Display the final segmented image, fig 5.1(e).

Some images using this segmentation method are shown below:-

Image 1

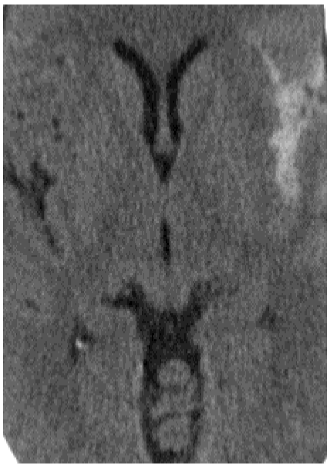


Fig 5.2(a)

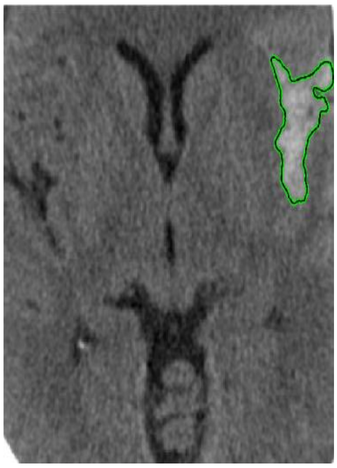


Fig 5.2(b)

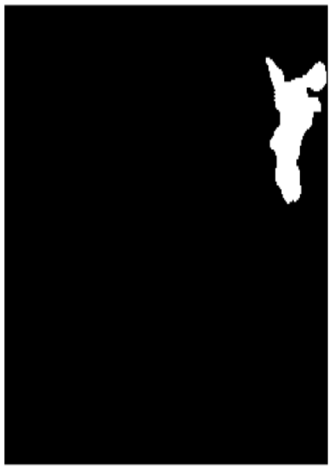


Fig 5.2(c)

Fig 5.2(d)

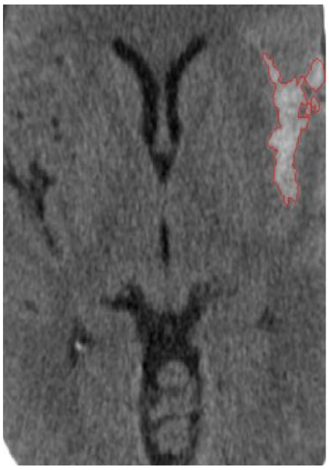


Fig 5.2(e)

Fig 5.2(a) Showing original image, Fig 5.2(b) Segmented using region based active contour, Fig 5.2(c) Binary image of region based active contour, Fig 5.2(d) Binary image of modified method (thresholding), Fig 5.2(e) Final segmentation result

Table 5.1 Results for image 1

Image 1	
Iterations	203
Area before thresholding	1496.25
Area after thresholding	1168.5
Time required	3.1623

Image 2

Fig 5.3(a)

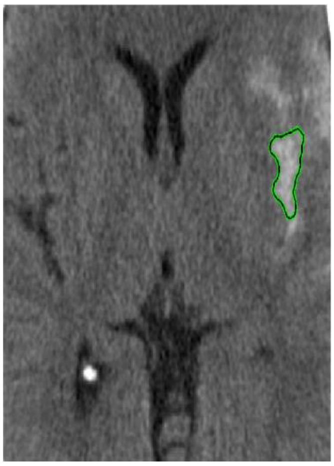


Fig 5.3(b)

Fig 5.3(c)

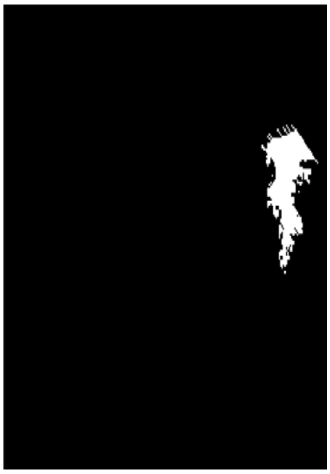


Fig 5.3(d)

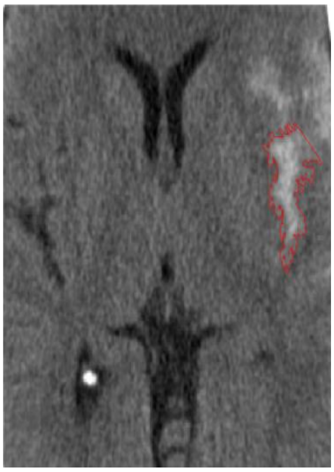


Fig 5.3(e)

Fig 5.3(a) Showing original image, Fig 5.3(b) Segmented using region based active contour, Fig 5.3(c) Binary image of region based active contour, Fig 5.3(d) Binary image of modified method (thresholding), Fig 5.3(e) Final segmentation result

Table 5.2 Results for image 2

Image 2	
Iterations	106
Area before thresholding	577.875
Area after thresholding	1080.75
Time required	2.2263

Image 3

Fig 5.4(a)

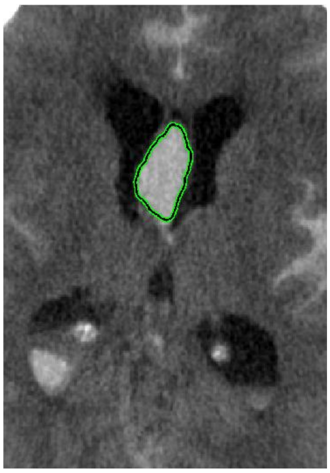


Fig 5.4(b)

Fig 5.4(c)

Fig 5.4(d)

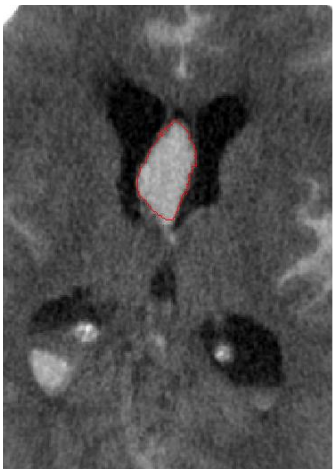


Fig 5.4(e)

Fig 5.4(a) Showing original image, Fig 5.4(b) Segmented using region based active contour, Fig 5.4(c) Binary image of region based active contour, Fig 5.4(d) Binary image of modified method(thresholding), Fig 5.4(e) Final segmentation result

Table 5.3 Result for image 3

Image 3	
Iterations	163
Area before thresholding	1431.375
Area after thresholding	1519.625
Time required	3.1474

Image 4

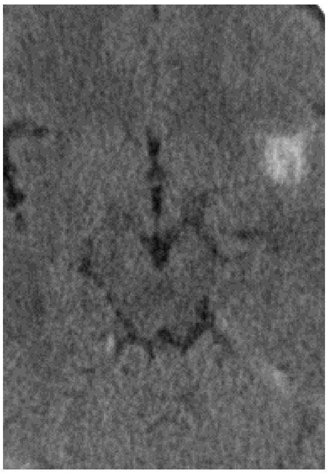


Fig 5.5(a)

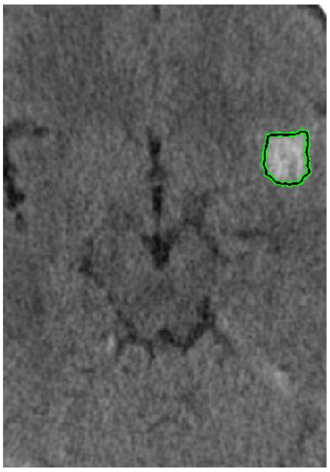


Fig 5.5(b)

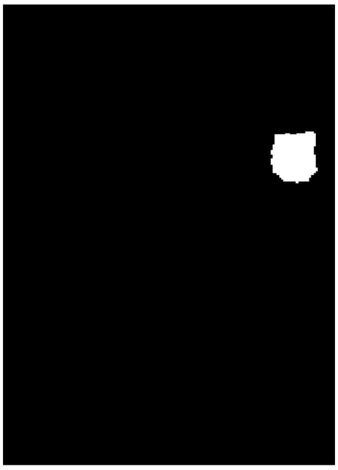


Fig 5.5(c)

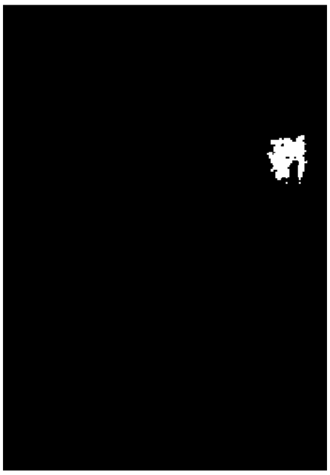


Fig 5.5(d)

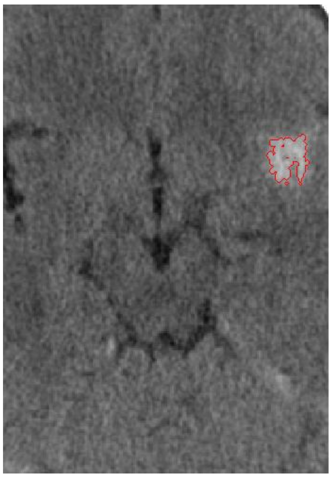


Fig 5.5(e)

Fig 5.5(a) Showing original image, Fig 5.5(b) Segmented using region based active contour, Fig 5.5(c) Binary image of region based active contour, Fig 5.5(d) Binary image of modified method (thresholding), Fig 5.5(e) Final segmentation result

Table 5.4 Result for image 4

Image 4	
Iterations	84
Area before thresholding	685.875
Area after thresholding	398
Time required	1.6543

Image 5

Fig 5.6(a)

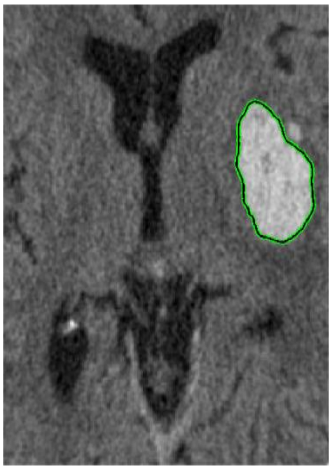


Fig 5.6(b)

Fig 5.6(c)

Fig 5.6(d)

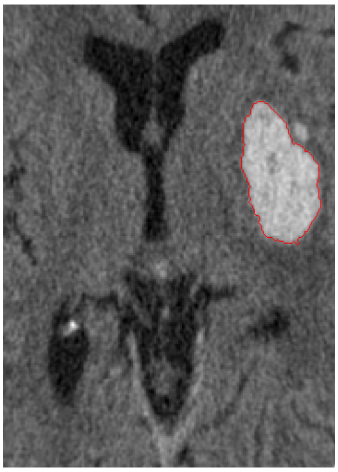


Fig 5.6(e)

Fig 5.6(a) Showing original image, Fig 5.6(b) Segmented using region based active contour, Fig 5.6(c) Binary image of region based active contour, Fig 5.6(d) Binary image of modified method (thresholding), Fig 5.6(e) Final segmentation result

Table 5.5 Results for image 5

Image 5	
Iterations	253
Area before thresholding	2276.125
Area after thresholding	2198.625
Time required	3.2719

Table 5.6 Comparison of original region based contour method and modified method

Image number	Original Area	Original algorithm			Modified algorithm			
		Iterations out of 1000	Area	Time required	Iterations out of 1000	Area after region growing	Area after thresholding	Time required
1	1190.65	1000	1769.5	16.0930	203	1469.25	1168.50	3.1623
2	1109.78	1000	778.5	14.5090	106	577.87	1080.75	2.2263
3	1568.34	1000	1487.1	18.2862	163	1431.37	1519.62	3.1474
4	427	1000	729.7	14.8340	84	685.87	398	1.6543
5	2107.72	1000	2296.1	13.84682	253	2276.12	2198.62	3.2719
6	1389.39	1000	1409.1	10.7439	181	1400.37	1400.37	2.0337
7	6894.59	1000	7431	19.5799	540	6947	6955.12	10.3949
8	699.43	1000	734.5	16.0706	100	654.62	712.75	2.003
9	890.64	1000	956.7	15.9053	152	861.37	864.37	2.5896
10	700.98	1000	1009.3	15.2170	153	992.87	778.25	2.5597
11	267.94	1000	424.7	17.1256	69	402.62	221.12	1.5659
12	310.52	1000	389	17.5760	53	285.50	295.50	1.3498
13	698.74	1000	717	16.0531	85	604.87	604.87	1.6824
14	859.53	1000	923.1	20.4850	139	872.37	883.37	3.0211
15	1198.94	1000	1287.1	18.2862	150	1231.37	1219.62	3.1474
16	510.75	1000	732.8	15.3980	89	709.37	474.25	1.6627
17	419.75	1000	473.75	18.2370	56	437.75	437.75	1.5260
18	2176.43	1000	2296.1	13.8468	253	2276.12	2198.62	3.2719
19	390.87	1000	476.6	12.4652	89	398.16	370.65	2.4519
20	840.54	1000	934.18	13.5389	97	845.32	810.49	4.1582
21	735.76	1000	782.34	15.2987	73	740.56	720.85	3.1976
22	300.73	1000	369.9	18.4516	107	350.54	345.19	3.5632
23	1148.43	1000	1249.5	20.4516	95	1134.92	1100.87	5.2749
24	590.29	1000	589.26	13.4591	142	570.21	560.74	2.4591
25	1312.98	1000	1575.3	15.5972	128	1487.72	1398.43	3.9260

Table 5.7 Comparison of error between original region based active contour method and modified method

Image number	%age error using original method	%age error using modified method
1	48.6163	-1.86033
2	-29.851	-2.61583
3	-5.18	-3.10647
4	70.88993	-6.79157
5	8.93762	4.312717
6	1.418608	0.790275
7	7.780158	0.877935
8	5.014083	1.904408
9	7.417138	-2.94956
10	43.98414	11.02314
11	58.50564	-17.4741
12	25.27373	-4.83705
13	2.613275	-13.4342
14	7.395902	2.773609
15	7.353162	1.724857
16	43.47528	-7.14635
17	12.8648	4.288267
18	5.498454	1.01956
19	21.93312	-5.17308
20	11.14046	-3.57508
21	6.330869	-2.02648
22	23.0007	14.78403
23	8.800711	-4.14131
24	-0.17449	-5.00601
25	19.97898	6.508096

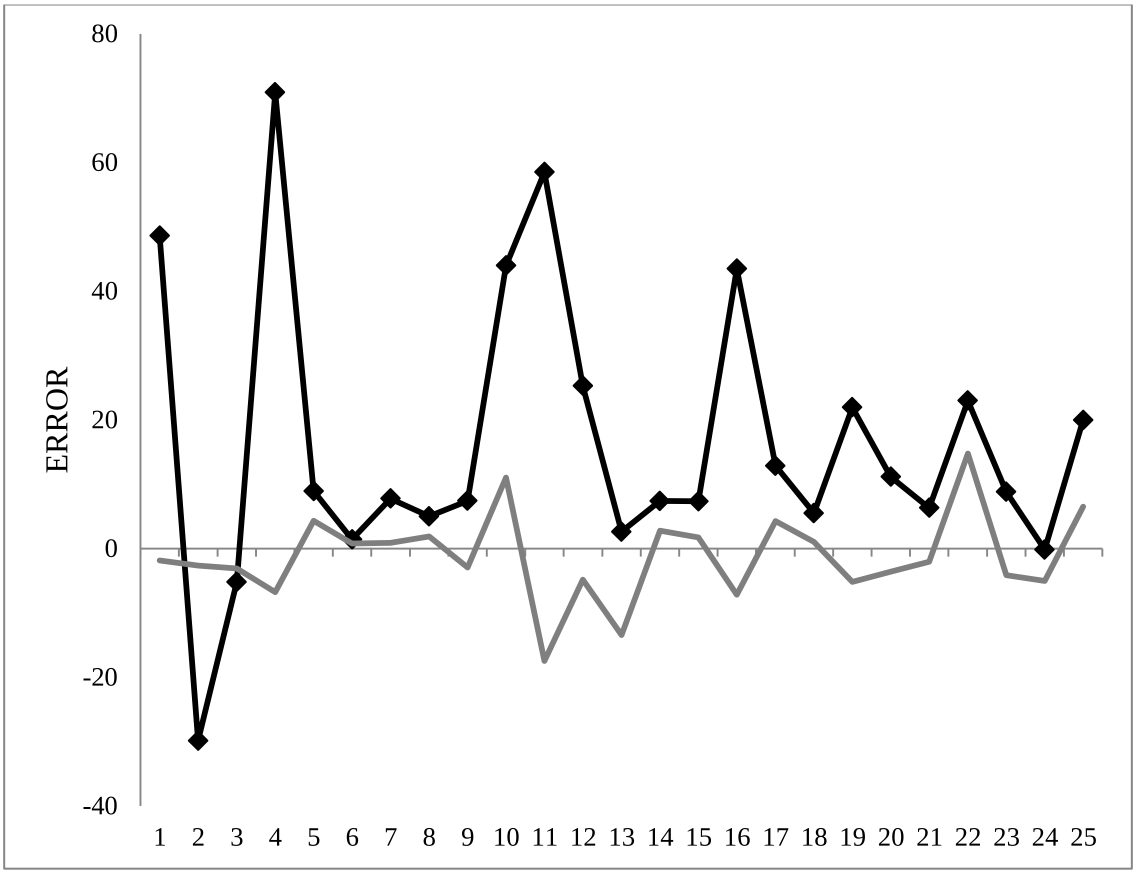


Fig 5.7 Plot of error between original region based active contour method and modified method. Line with marks is the error of original method and the other is error of modified method

5.2 HYBRID LEVEL SET METHOD

Step 1:- Firstly the image was read with the help of imread command, fig 5.8(a). This command is inbuilt command in MATLAB processing toolbox.

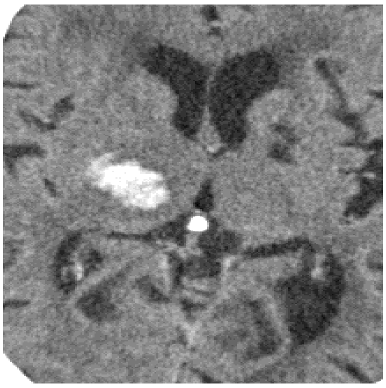


Fig 5.8(a) Original image

Step 2:- Apply the level set method to given input image and obtain the contour, fig 5.8(b).

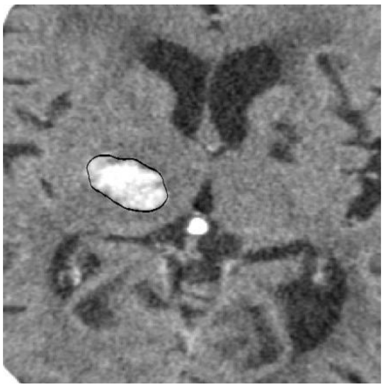


Fig 5.8(b) Original level set based segmentation

Step 3:- Across boundaries of diseased portion obtained in step 2, draw a number of region of interest (ROI) and calculate the texture parameters. Obtain the sum of texture parameters.

Step 4:- Particle swarm optimization (PSO) is used to maximize the sum of texture parameters. The sum of texture parameters of the ROI is compared with the dataset of manually obtained value. If the texture parameters satisfy the condition, merge it into diseased portion else to non-diseased portion, fig 5.8(c).

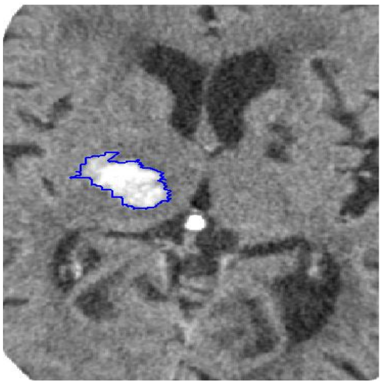


Fig 5.8(c) Modified using texture parameters only

Step 5:- Calculate the mean of diseased portion obtained in step 2. We will assume this value as threshold value. Now, compare the intensity of each pixel of ROI and its immediate neighbour with the threshold value. If the value is greater than threshold value then consider it as diseased portion, else put it into non diseased portion

Step 6:-Obtain the final output, fig 5.8(d).

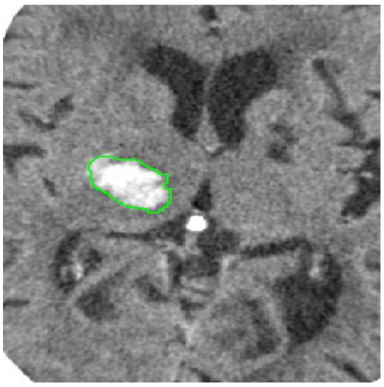


Fig 5.8(d) Modified using texture and thresholding

Some more results of above segmentation method are shown below:

Image 1

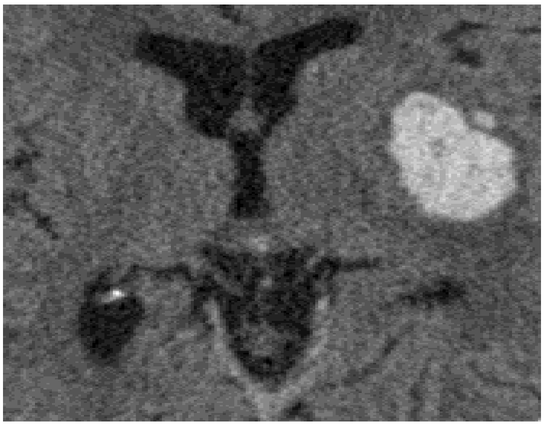


Fig 5.9(a)

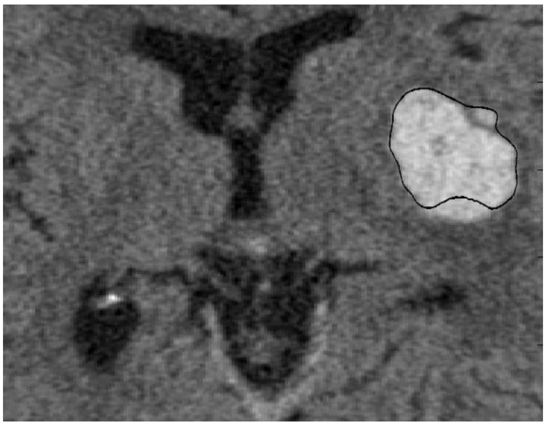


Fig 5.9(b)

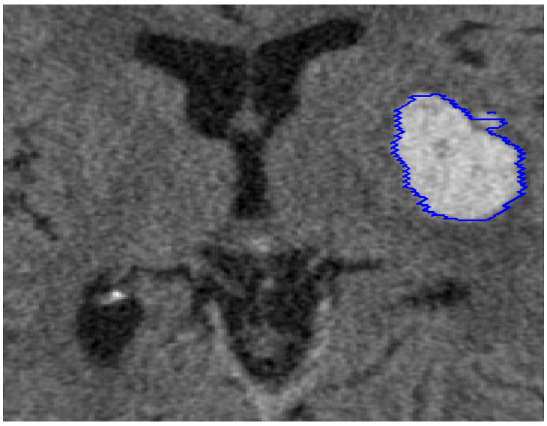


Fig 5.9(d)

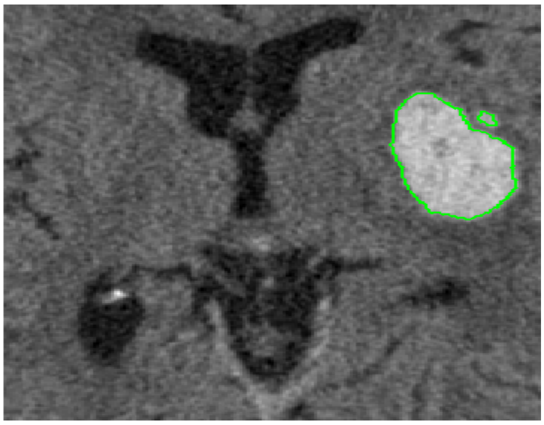


Fig 5.9(d)

Fig 5.9(a) Showing original image, fig 5.9(b) Original level set based segmentation, Fig 5.9(c) Modified using texture parameters only, Fig 5.9(d) Modified using texture and thresholding

Table 5.8 Results for image 1

Original method	
Area	62603.62
Time	3.6794
Modified using texture parameters	
Area	62537.37
Time	4.2902
Modified using texture parameters and thresholding	
Area	62487.12
Time	4.8423

Image 2

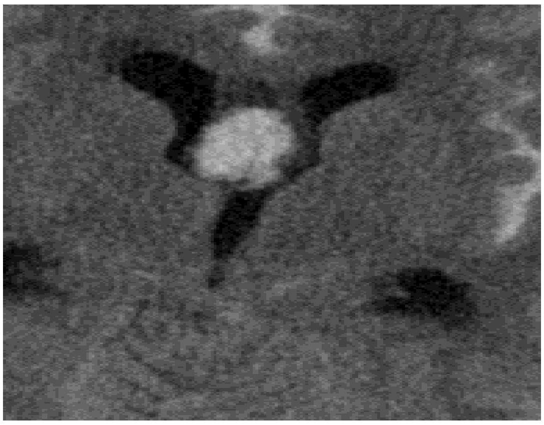


Fig 5.10(a)

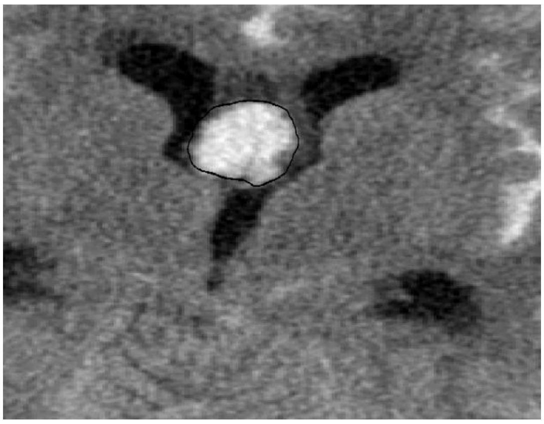


Fig 5.10(b)

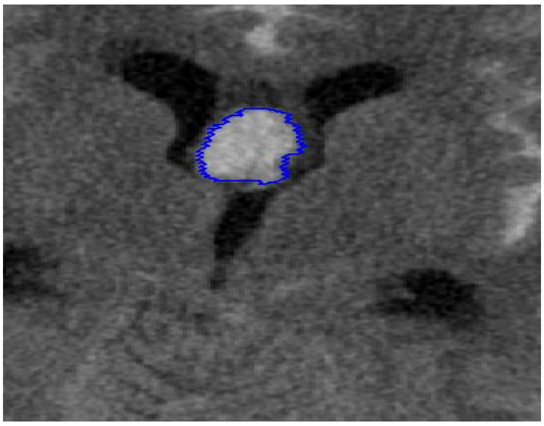


Fig 5.10(c)

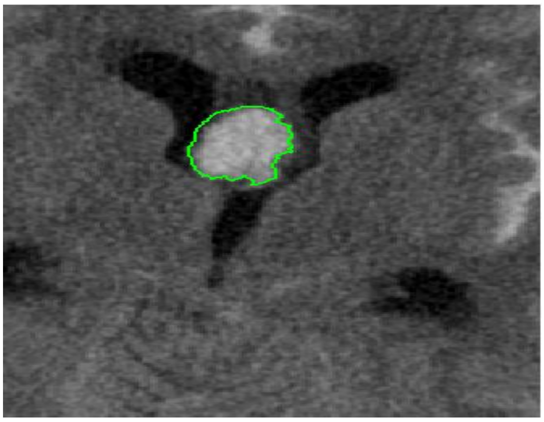


Fig 5.10(d)

Fig 5.10(a) Showing original image, fig 5.10(b) Original level set based segmentation, fig 5.10(c) Modified using texture parameters only, fig 5.10(d) Modified using texture and thresholding

Table 5.9 Results for image 2

Original method	
Area	55281.75
Time	4.0834
Modified using texture parameters	
Area	55556.50
Time	4.6751
Modified using texture parameters and thresholding	
Area	55476.50
Time	5.1322

Image 3

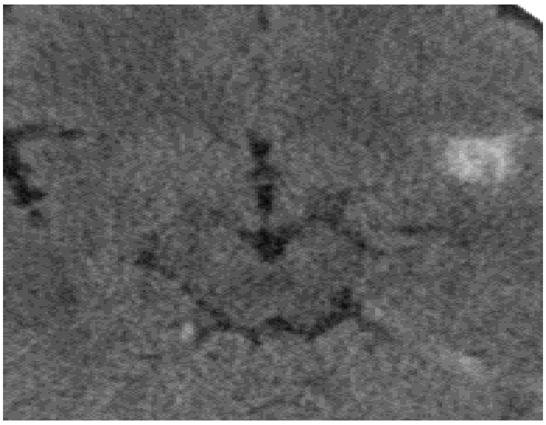


Fig 5.11(a)

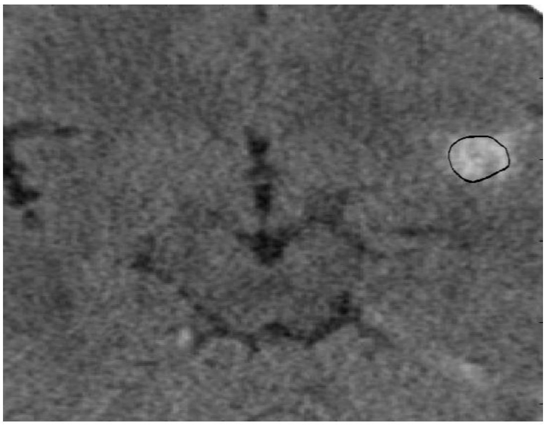


Fig 5.11(b)

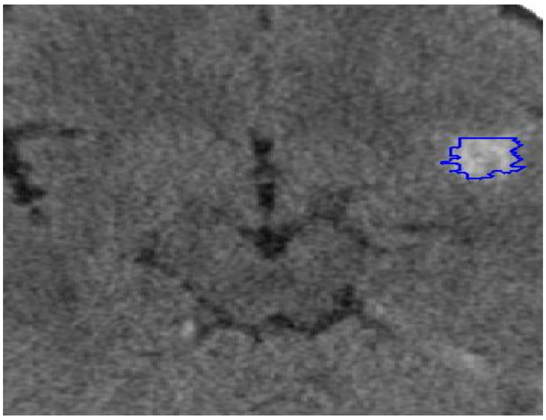


Fig 5.11(c)

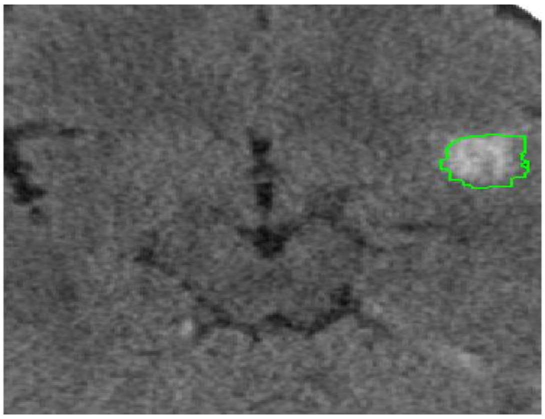


Fig 5.11(d)

Fig 5.11(a) Showing original image, fig 5.11(b) Original level set based segmentation, Fig 5.11(c) Modified using texture parameters only, Fig 5.11(d) Modified using texture and thresholding.

Table 5.10 Results for image 3

Original method	
Area	50684.00
Time	1.9490
Modified using texture parameters	
Area	50597.00
Time	2.4791
Modified using texture parameters and thresholding	
Area	50254.75
Time	2.5307

Image 4

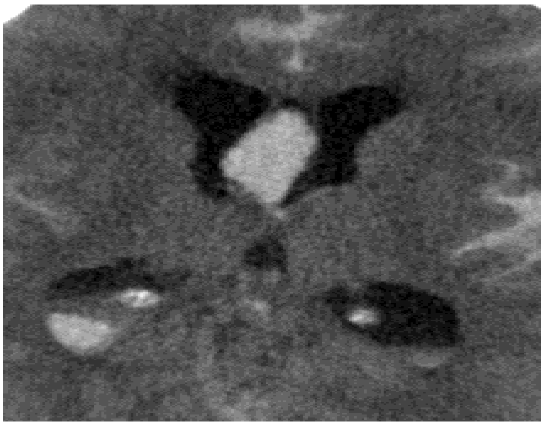


Fig 5.12(a)

Fig 5.12(b)

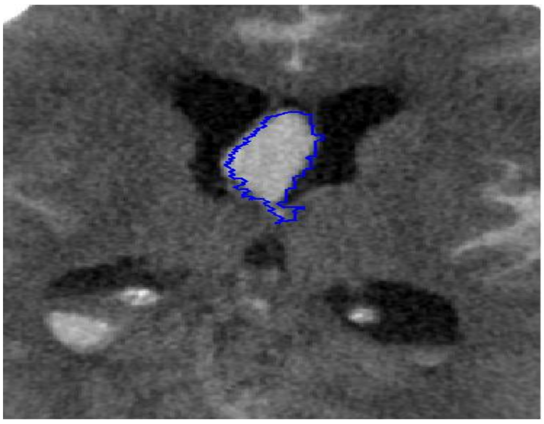


Fig 5.12(c)

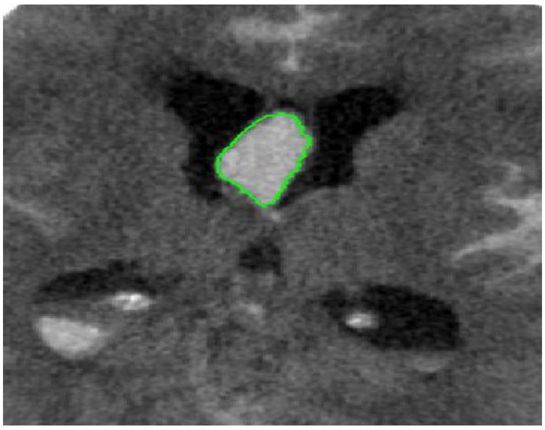


Fig 5.12(d)

Fig 5.12(a) Showing original image, fig 5.12(b) Original level set based segmentation, Fig 5.12(c) Modified using texture parameters only, Fig 5.12(d) Modified using texture and thresholding

Table 5.11 Results for image 4

Original method	
Area	71138.62
Time	6.4303
Modified using texture parameters	
Area	70997.00
Time	7.1158
Modified using texture parameters and thresholding	
Area	70661.50
Time	7.2743

Image 5

Fig 5.13(a)

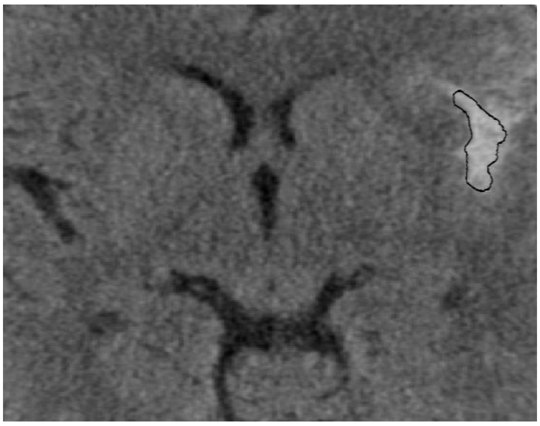


Fig 5.13(b)

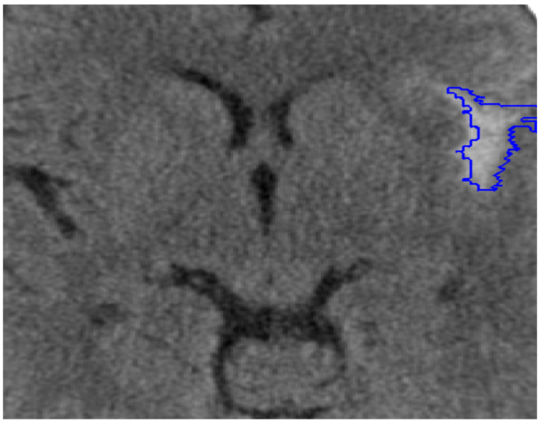


Fig 5.13(c)

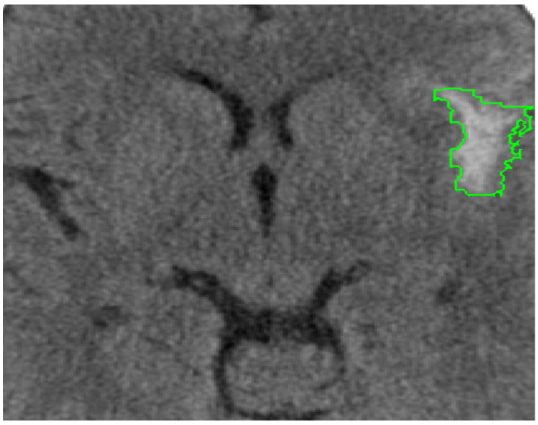


Fig 5.13(d)

Fig 5.13(a) Showing original image, fig 5.13(b) Original level set based segmentation, Fig 5.13(c) Modified using texture parameters only, Fig 5.13(d) Modified using texture and thresholding

Table 5.12 Results for image 5

Original method	
Area	51985.28
Time	1.9472
Modified using texture parameters	
Area	51739.49
Time	2.3154
Modified using texture parameters and thresholding	
Area	51623.54
Time	2.8732

Table 5.13 Comparison between original level set method and modified method

Image	Original area	Original method		Modified using texture only		Modified using texture and thresholding	
		Area	Time	Area	Time	Area	Time
1	63102.91	62603.62	3.6794	62537.37	4.2902	62487.12	4.8423
2	43994.91	44014.00	2.7115	43886.00	3.2509	44026.37	3.857619
3	56123.54	55281.75	4.0834	55556.50	4.6751	55476.50	5.1322
4	51091.62	50684.00	1.9490	50597.00	2.4791	50254.75	2.5307
5	53191.82	52512.62	2.0572	52267.25	2.6228	52276.62	2.5575
6	62987.91	64071.00	5.6358	63713.00	6.2810	63971	6.4984
7	71239.81	71138.62	6.4303	70997.00	7.1158	70661.50	7.2743
8	53109.78	53530.12	2.9752	53447.25	3.5253	53437.25	4.2143
9	61300.18	58576.37	4.3070	58412.25	4.9067	58455.50	4.9695
10	51101.43	52564.25	2.9470	52137.50	3.5990	51697.12	3.9918
11	51525.61	51985.28	1.9472	51739.49	2.3154	51623.54	2.8732
12	4198.65	4582.19	2.4915	4356.89	2.9129	4298.67	3.0235
13	2561.16	2849.15	1.9821	2719.43	2.4819	2619.58	2.5195
14	64191.72	61687.39	3.7162	63197.25	3.9890	63289.13	4.1873
15	4700.87	4817.21	1.9863	4698.42	2.3481	4719.33	2.4353
16	52998.45	54202.34	3.0123	53144.17	3.9843	53998.43	4.0342
17	2499.87	2435.45	2.2398	2487.38	2.9530	2478.35	3.1302
18	45109.34	43219.23	3.0145	44219.77	3.9890	44121.65	4.0372
19	4012.65	3879.78	2.0013	3987.67	2.8712	3997.78	3.0091
20	4200.74	4319.89	2.1198	4298.23	2.9980	4277.35	2.9998
21	58109.64	56341.87	3.6754	57678.66	4.1243	57767.88	4.6753
22	3820.61	3897.73	1.9899	3889.33	2.6782	3876.87	2.9432
23	58342.71	56321.98	2.8973	57876.33	3.3209	57987.67	3.6732
24	62101.49	63217.32	3.4587	62762.45	4.1298	62567.27	4.6513
25	5255.45	5413.84	1.9932	5314.52	2.6713	5299.89	2.7819

Table 5.14 Comparison of error between original level set method and modified method

Image number	%age error using original method	%age error using modified method (Texture parameters only)	%age error using modified method (Texture parameters and thresholding)
1	0.806259	0.699581	0.618667
2	0.043391	-0.24755	0.071508
3	0.28701	0.785436	0.640307
4	1.182593	1.008911	0.325663
5	0.403045	-0.0661	-0.04818
6	0.111078	-0.4483	-0.04517
7	1.27963	1.078007	0.600358
8	0.791455	0.63542	0.616591
9	-4.4434	-4.71113	-4.64057
10	2.862581	2.027478	1.165701
11	0.892119	0.415095	0.190061
12	9.134841	3.76883	2.382194
13	11.24451	6.179622	2.280998
14	-3.90133	-1.54922	-1.40608
15	2.257333	-0.26428	0.179585
16	2.271557	0.274951	1.88681
17	-1.4335	0.668186	0.302727
18	5.132386	7.566237	7.327556
19	-3.31128	-0.62253	-0.37058
20	2.836405	2.320782	1.823726
21	-3.04213	-0.74167	-0.58813
22	2.018526	1.798666	1.47254
23	-3.46355	-0.79938	-0.60854
24	1.796785	1.064322	0.75003
25	3.013824	1.123976	0.845598

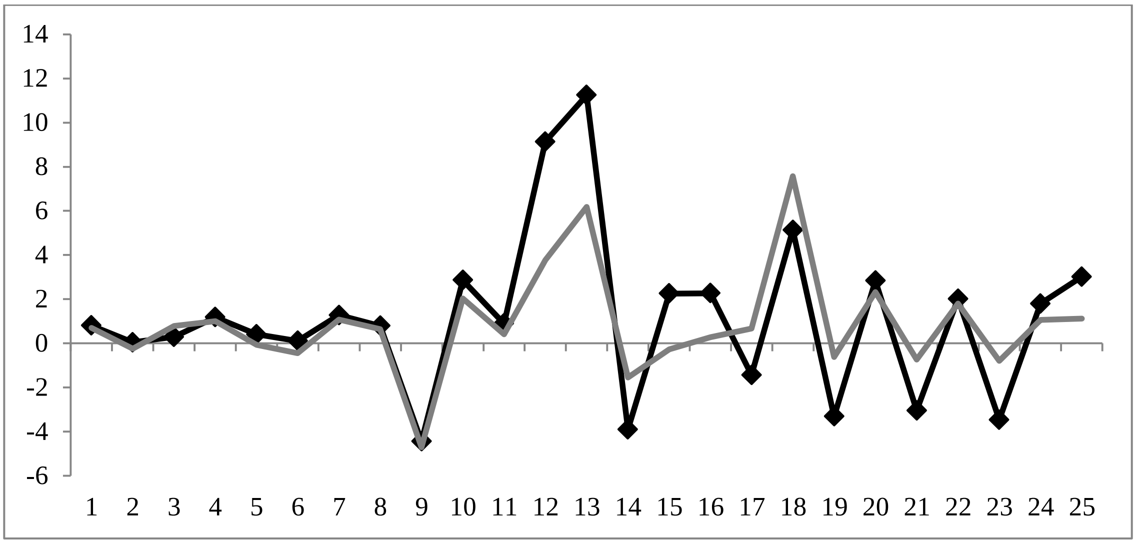


Fig 5.14 Plot of error between original level set method and modified method. Line with marks is the error of original method and the other is error of modified method

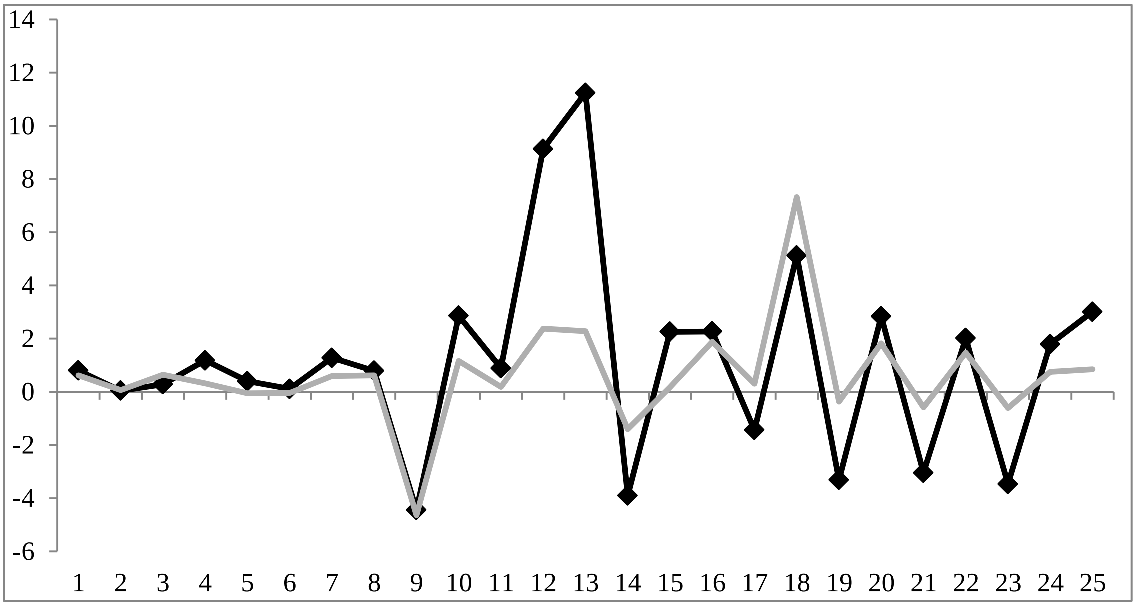


Fig 5.15 Plot of error between original level set method and modified method. Line with marks is the error of original method and the other is error of modified method

CHAPTER 6
CONCLUSION

This thesis work presents segmentation of tumor using two methods which have been modified. A part of total work was done on MATLAB, Version 7.10. In the first method, modified region based active contour, we had applied automatic convergence for the region based active contour. The number of iterations required had been controlled. In the second method, modified level set with re-initialization has been implemented. Initially we applied level set, then made small region of interest (ROI) across the boundaries of image obtained after level set and calculated the sum of three texture parameters. Particle swarm optimization was used to maximize the texture parameters. Then, compare the sum of texture of each ROI with the sum taken as a standard and then updated accordingly. It resulted in good segmentation but with some error. To overcome this drawback, each pixel of ROI which satisfies the texture condition was compared with the mean intensity. The mean intensity is the mean of the diseased portion segmented in the original level set. Initially we had taken 25 textures parameters, but to implement all parameters makes the computation more complex. Using Fisher's Discrimination ratio, the dimension was reduced to 3. Some of the decisive points of the results are listed below:

- Methods used: Modified region based active contour and modified level set.
- Modified region based active contour results in lesser time consumption, since lesser number of iterations were required. Better accurate segmentation has been achieved.
- Modified level set results in better segmentation of diseased part. The parts of over segmentation or under segmentation were segmented properly.

It shows that modified segmentation methods can be effectively used for segmentation of abnormality in CT images. The error has been reduced to a great extent, hence the methods are more reliable. It also shows that the normal Ct images can be effectively used for correct segmentation.

REFERENCES

[1] P.Y. Yin, "Multilevel Minimum Cross Entropy Threshold Selection based on PSO", Applied Mathematics and Computation, Elsevier, Vol. 184, Issue 2, pp. 503-516, 2007.

[2] A. Nakib, S. Roman, H. Oulhadj, P. Siarry, "Fast Brain MRI Segmentation based on Two-Dimensional Survival Exponential Entropy and PSO", 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 5563-5566, 2007.

[3] L. Djerou, N. Khelil, H. E. Dehimi, M. Batouche, "Automatic Multilevel Thresholding using Binary PSO for Image Segmentation", International Conference of Soft Computing And Pattern Recognition, pp. 66-71, 2009.

[4] T. Hongmei, W. Cuixia, H. Liying, W. Xia, "Image Segmentation based on Improved PSO", International Conference on Computer and Communication Technologies in Agriculture Engineering, pp. 191-194, 2010.

[5] A. De, R.L. Das, A.K. Bhattacharjee, D. Sharma, "Masking based Segmentation of Diseased MRI Images", International Conference on Information Science and Applications (ICISA), pp. 1-7, 2010.

[6] C. Yu, J. Zhang, Y. Zhang, "Maximum Entropy Image Segmentation based on Improved QPSO algorithm", International Conference on Electronic and Mechanical Engineering and Information Technology (EMEIT), pp. 3474-3477, 2011.

[7] M.G. Masooleh, S.A.S. Moosavi, "An Improved Fuzzy Algorithm for Image Segmentation", World Academy of Science, Engineering and Technology, pp. 400-404, 2008.

[8] P. Puranik, P. Bajaj, A. Abraham, P. Palsodkar, A.Deshmukh, "Human Perception based Color Image Segmentation using Comprehensive Learning Particle Swarm Optimization", 2nd International Conference on Emerging Trends Engineering and Technology (ICETET), pp. 630-635, 2009.

[9] N.N. Gopal, Dr. M. Karnan, “Diagnose Brain Tumor through MRI using Image Processing Clustering Algorithms such as Fuzzy C Means along with Intelligent Optimization Techniques”, International Conference and Computing Research (ICCIC), pp. 1-4, 2010.

[10] K.M. Murugesan, Dr. S. Palaniswami, “Efficient Colour Image Segmentation using Multi-Elistic-Exponential PSO”, Journal of Theoretical and Applied Information Technology”, Vol. 18, No. 1, pp. 35-41, 2010.

[11] S. Shen, W. Sandham, M. Granat, A. Sterr, “MRI Fuzzy Segmentation of Brain Tissue using Neighbourhood attraction with Neural-Network Optimization”, IEEE Transactions on Information Technology in Biomedicine, Vol. 9, Issue 3, pp. 459-467, 2005.

[12] M. Forouzanfar, N. Forghani, M. Teshnehlab, “Parameter Optimization of Improved Fuzzy C-Means Clustering Algorithm for Brain MR Image Segmentation”, Engineering Applications of Artificial Intelligence, Elsevier, Vol. 23, Issue 2, pp. 160-168, 2010.

[13] D.K. Kole and A. Halder, “An Efficient Dynamic Image Segmentation Algorithm using Hybrid Technique based on PSO and Genetic Algorithm”, International Conference on Advances in Computer Engineering (ACE), pp. 252-255, 2010.

[14] R.H. Turi, “Clustering Based Color Image Segmentation”, Phd Thesis, Monash University, Australia, 2001.

[15] Z. Wei, Z. Y. Zhu, “Image Segmentation Algorithm Based on Wavelet Mutation Inertia Adaptive PSO”, 29th Chinese Control Conference, pp. 2690-2693, 2010.

[16] A. De, A.K. Bhattacharjee, C.K. Chanda, B. Maji, “MRI Segmentation using Entropy Maximization and Hybrid PSO with Wavelet Mutation”, World Congress on Information and Communication Technologies (WICT), pp. 362-367, 2011.

[17] M.G.H. Omran, A. Salman, A. P. Engelbrecht, “Dynamic Clustering using PSO with Application in Image Segmentation”, Pattern Analysis and Applications, Springer, Vol. 8, No. 4, pp. 332-344, 2006.

[18] W. Chun, K. Fang, “A Hybridized Clustering Approach Using PSO for Image Segmentation”, International Conference on Audio, Language and Image Processing (ICALIP), pp. 1365-1368, 2008.

[19] Z. Jing, L. Bo, “Image Segmentation using Fast Fuzzy C Means Based on PSO”, 3rd International Conference on Intelligent Networks and Intelligent Systems (ICINIS), pp. 370-373, 2010.

[20] L. Hongpo, S. Jun, T. Shuhua, T. Zhiguo, “High Resolution Sonar Image Segmentationn by PSO based Fuzzy Cluster Method”, Fourth International Conference on Genetic and Evolutionary Computing, pp. 18-21, 2010.

[21] Z. Jing, S. Kai, “Trembling PSO for Modified Possibilistic C-Means in Image Segmentation”, Second WRI Congress on Intelligent Systems (GCIS), pp. 119-122, 2010.

[22] Y. Zhang, D. Huang, M. Ji, F. Xie, “Image Segmentation using PSO and PCM with Mahalanobis Distance”, Epert Systems with Applications, Elsevier, Vol. 38, Issue 7, pp. 9036-9040, 2011.

[23] G. Liu, A. Wang, Y. Zhao, “An Efficient Image Segmentation Method Based on Fuzzy PSO and Markov Random Field Model”, 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), pp. 1-4, 2011.

[24] S. Wang, Y. Xu, Y. Pang, “A Fast Underwater Optical Image Segmentation Algorithm Based on a Histogram Weighted Fuzzy C-means Improved by PSO”, Journal of Marine and Application, Springer, Vol. 10, No. 1, pp. 70-75, 2011.

[25] Z. X. Feng, S. J. Kui, “Application of Image Segmentation Algorithm Based on PSO and Rough Entropy Standard”, Chinese Control and Decision Conference (CCDC), pp. 2905-2909, 2009.

[26] H.S. Behera, “Segmentation and Classification using Heuristic HRSPSO”, International Journal of Soft Computing and Engineering (IJSCE), Vol. 1, Issue 3, pp. 66-69, 2011.

[27] D.L. Davies, D.W.Bouldin, “A Cluster Separation Measure”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI1, Issue 2, pp. 224-227, 1979.

[28] F. Yi, Z. Chongxun, P. Chen, L. Li, “White Blood Cell Image Segmentation using On-line Trained Neural Network”, 27th Annual International Conference of the Engineering in Medicine and Biology (IEEE-EMBS), pp. 6476-6479, 2005.

[29] K. Fukunaga, L.D. Hosteler, “The Estimation of the Gradient of a Density Function with Applications in Pattern Recognition”, IEEE Trans. Information Theory, Vol. 21, pp. 32-40, 1975.

[30] Y. Lian, Y. Zhao, F. Wu, “Modified Adaptive Probabilistic Neural Network using for MR Image Segmentation”, IEEE Youth Conference on Information Computing and Telecommunications (YC-ICT), pp. 355-358, 2010.

[31] T. Kohonen, “Self-Organizing maps”, Springer, 3rd edition, 2000.

[32] N. Alamelumangai, Dr. J. Devishree, “PSO Aided Fuzzy Inference System for Ultrasound Image Segmentation”, International Journal of Computer Applications, Vol. 7, No. 14, pp. 16-20, 2010.

[33] S.M. Odeh, “Using an Adaptive Neuro Fuzzy Inference System (AnFis) Algorithm for Automatic Diagnosis of Skin Cancer”, Journal of Communication and Computer, Vol. 8, No. 9, 2011.

[34] S. Osher, J.A. Sethian, “Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations”, Journal of Computational Physics, Vol. 79, pp. 12-49, 1988.

[35] R.C. Eberhart and J. Kennedy, “A New Optimizer using Particle Swarm Theory”, Proceedings of the Sixth International Symposium on Micro Machine and Human Science, pp. 39–43, 1995.

[36] S. Ibrahim, N.E.A. Khalid, M. Manaf, “Empirical Study of Brain Segmentation using PSO”, International Conference on Information Retrieval & Knowledge Management (CAMP), pp. 235-239, 2010.

[37] T.F. Chan, L.A. Vese, “Active Contours Without Edges”, IEEE Transactions on

Image Processing, Vol. 10, No. 2, pp. 266-277, 2001.
[38] C. Li, C. Xu, C. Gui, M.D. Fox, “Level Set Evolution Without Re-initialization: A New Variational Formulation”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 1, pp. 430-436, 2005.