RESEARCH ARTICLE OPEN ACCESS

Particle Swarm Optimization Methods for Image Segmentation Applied In Mammography

N. Gopi Raju[1] & P A Nageswara Rao[2]
P.G student, Department of ECE, Vignan's Institute of Information Technology, Visakhapatnam[1]
Associate Professor, Department of ECE, Vignan's Institute of Information Technology, Visakhapatnam [2]

Abstract
Accurate medical diagnosis requires a segmentation of large number of medical images. The automatic segmentation is still challenging because of low image contrast and ill-defined boundaries. Image segmentation refers to the process that partitions an image into mutually exclusive regions that cover the image. Among the various image segmentation techniques, traditional image segmentation methods are widely used but have certain drawbacks, which cannot be used for accurate result. In this thesis clustering based techniques is employed on images which results into segmentation of images. The performance of Fuzzy C-means (FCM) integrated with Particle Swarm optimization (PSO) technique and its variations are analyzed in different application fields. To analyze techniques in different fields several metrics are used namely global consistency error, probabilistic rand index and variation of information are used. This experimental performance analysis shows that FCM along with fractional order Darwinian PSO give better performance in terms of classification accuracy, as compared to other variation of other techniques used. The integrated algorithm tested on images proves to give better results. Finally, fractional order Darwinian PSO along with neighbourhood Fuzzy C-means and partial differential equation based level set method is an effective image segmentation technique to study the intricate contours.
Keywords— Darwinian PSO (DPSO), Fuzzy C-means (FCM), FCM neighbourhood (FCMN), Fractional Order DPSO (FO-DPSO), Particle Swarm Optimization (PSO),

I. Introduction

As humans, it is easy (even for a child) to recognize letters, objects, numbers, voices of friends, etc. However, making a computer solve these types of problems is a very difficult task. Pattern recognition is the science with the objective to classify objects into different categories and classes. It is a fundamental component of artificial intelligence and computer vision. Pattern recognition methods are used in various areas such as science, engineering, business, medicine, etc. Interest in pattern recognition is fast growing in order to deal with the prohibitive amount of information we encounter in our daily life. Automation is desperately needed to handle this information explosion. This thesis investigates the application of an efficient optimization method, known as Particle Swarm Optimization, to the field of pattern recognition and image processing. PSOs solve optimization problems by simulating the social behaviour of bird flocks. Image segmentation is the process of subdividing an image into its constituent parts and extracting the parts of interest. Segmentation algorithms for monochrome images generally are based on one of the two basic properties of image intensity values: discontinuity and similarity [1]. In the first category, the approach is to partition an image based on abrupt changes in intensity, such as edges in an image. Second category is based on partitioning an image into regions that are similar according to a set of predefined criteria.

1.1 Motivation

There are many difficult problems in the field of pattern recognition and image processing. These problems are the focus of much active research in order to find efficient approaches to address them. However, the outcome of the research is still unsatisfactory.

Local search approaches were generally used to solve difficult problems in the field of pattern recognition and image processing. However, the selected set of problems in this thesis are NP-hard and combinatorial. Hence, evolutionary algorithms are generally more suitable to solve these difficult problems because they are population-based stochastic approaches. Thus, evolutionary algorithms can avoid being trapped in a local optimum and can often find a global optimal solution. A PSO is a population-based stochastic optimization algorithm modelled after the simulation of the social behaviour of bird flocks. PSO is easy to implement and has been successfully applied to solve a wide range of optimization problems [Hu 2004]. Thus, due to its simplicity and efficiency in navigating large search spaces for optimal solutions, PSOs are used in this research to develop efficient, robust and flexible algorithms to solve a selective set of difficult problems in the field of pattern recognition and image processing. Out of these problems, data clustering is elaborately tackled in this thesis specifically image data. The motivation for the focus on data clustering is the fact that data clustering is an important process in pattern recognition and machine learning. Actually, clustering is a primary goal of pattern recognition. Furthermore, it is a central process in Artificial Intelligence. In addition, clustering algorithms are used in many applications, such as image segmentation, vector and colour image quantization, spectral un-mixing, data mining, compression, etc. Therefore, finding an efficient clustering algorithm is very important for researchers in many different disciplines.

1.2 MAMMOGRAPHY

Breast cancer is the leading cause of death among all cancers for middle-aged women in most developed countries. Any diagnostic tool help to improve the sensitivity or specificity of breast cancer would be highly valued. The usefulness of mammography in the symptomatic patient is undisputed; mammography is primarily used to demonstrate the presence of breast cancer and, specifically to indicate the size and location of tumor. There is also considerable evidence indicating the ability of mammography to detect cancer. In addition, randomized controlled trials of screening mammography have demonstrated a significant decline in breast cancer mortality among screened women age 50 and older. In this sense, many automatic and semiautomatic techniques for cancer detection have been studied in the last years including topics such as segmentation, description, and classification.

II. Optimization and Optimization Methods

2.1 Optimization

The objective of optimization is to seek values for a set of parameters that maximize or minimize objective functions subject to certain constraints [Rardin 1998; Van den Bergh 2002]. A choice of values for the set of parameters that satisfy all constraints is called a *feasible solution*. Feasible solutions with objective function value(s) as good as the values of any other feasible solutions are called *optimal solutions* [Rardin 1998]. An example of an optimization problem is the arrangement of the transistors in a computer chip in such a way that the resulting layout occupies the smallest area and that as few as possible components are used. Optimization techniques are used on a daily base for industrial planning, resource allocation, scheduling, decision making, etc. Furthermore, optimization techniques are widely used in many fields such as business, industry, engineering and computer science. Research in the optimization field is very active and new optimization methods are being developed regularly [Chinneck 2000].

Optimization encompasses both maximization and minimization problems. Any maximization problem can be converted into a minimization problem by taking the negative of the objective function, and *vice versa*. Hence, the terms optimization, maximization and minimization are used interchangeably in this thesis. In general, the problems tackled in this thesis are minimization problems. Therefore, the remainder of the discussion focuses on minimization problems.

2.2 Traditional Optimization Algorithms

Traditional optimization algorithms use exact methods to find the best solution. The idea is that if a problem can be solved, then the algorithm should find the global best solution. One exact method is the *brute force* (or *exhaustive*) search method where the algorithm tries every solution in the search space so that the global optimal solution is guaranteed to be found. Obviously, as the search space increases the cost of brute force algorithms increases. Therefore, brute force algorithms are not appropriate for the class of problems known as NP-hard problems. The time to exhaustively search an NP-hard problem increases exponentially with problem size. Other exact methods include linear programming, divide and conquer and dynamic programming. More details about exact methods can be found in Michalewicz and Fogel [2000].

2.3 Stochastic Algorithms

Stochastic search algorithms are used to find near-optimal solutions for NP-hard problems in polynomial time. This is achieved by assuming that good solutions are close to each other in the search space. This assumption is valid for most real world problems [Løvberg 2002; Spall 2003]. Since the objective of a stochastic algorithm is to find a near-optimal solution, stochastic algorithms may fail to find a global optimal solution. While an exact algorithm generates a solution only after the run is completed, a stochastic algorithm can be stopped any time during the run and generate the best solution found so far [Løvberg 2002].

2.4 Evolutionary Algorithms

Evolutionary algorithms (EAs) are general-purpose stochastic search methods simulating natural selection and evolution in the biological world. EAs differ from other optimization methods, such as Hill-Climbing and Simulated Annealing, in the fact that EAs maintain a population of potential (or candidate) solutions to a problem, and not just one solution [Engelbrecht 2002; Salman 1999].

Due to its population-based nature, EAs can avoid being trapped in a local optimum and consequently can often find global optimal solutions. Thus, EAs can be viewed as global optimization algorithms. However, it should be noted that EAs may fail to converge to a global optimum [Gray *et al*. 1997].

EAs have successfully been applied to a wide variety of optimization problems, for example: image processing, pattern recognition, scheduling, engineering design, etc. [Gray *et al* 1997; Goldberg 1989].

2.5 Genetic Algorithms

Genetic Algorithms (GAs) are evolutionary algorithms that use selection, crossover and mutation operators. GAs were first proposed by Holland [1962; 1975] and were inspired by Darwinian evolution and Mendelian genetics [Salman 1999]. GAs follow the same algorithm presented in Figure 2.2. GAs are one of the most popular evolutionary algorithms and have been widely used to solve difficult optimization problems. GAs have been successfully applied in many areas such as pattern recognition, image processing, machine learning, etc. [Goldberg 1989]. In many cases GAs perform better than EP and ESs. However, EP and ESs usually converge better than GAs for real valued function optimization [Weiss 2003]. Individuals in GAs are called *chromosomes*.

III. Particle Swarm Optimization

A particle swarm optimizer (PSO) is a population-based stochastic optimization algorithm modelled after the simulation of the social behaviour of bird flocks [Kennedy and Eberhart 1995; Kennedy and Eberhart 2001]. PSO is similar to EAs in the sense that both approaches are population-based and each individual has a fitness function. Furthermore, the adjustments of the individuals in PSO are relatively similar to the arithmetic crossover operator used in EAs [Coello Coello and Lechuga 2002]. However, PSO is influenced by the simulation of social behavior rather than the survival of the fittest [Shi and Eberhart 2001]. Another major difference is that, in PSO, each individual benefits from its history whereas no such mechanism exists in EAs [Coello Coello and Lechuga 2002].

3.1 A Survey of PSO Techniques

The PSO [1] approach utilizes a cooperative swarm of particles, where each particle represents a candidate solution, to explore the space of possible solutions to an optimization problem. Each particle is randomly or heuristically initialized and then allowed to 'fly'. At each step of the optimization, each particle is allowed to evaluate its own fitness and the fitness of its neighboring particles. Each particle can keep track of its own solution, which resulted in the best fitness, as well as see the candidate solution for the best performing particle in its neighborhood. At each optimization step, indexed by t, each particle, indexed by i, adjusts its candidate solution (flies) according to,

$$\begin{aligned} \bar{v}_i(t+1) &= \bar{v}_i(t) + \phi_1(\bar{x}_{i,p} - \bar{x}_i) + \phi_2(\bar{x}_{i,n} - \bar{x}_i) \\ \bar{x}_i(t+1) &= \bar{x}_i(t) + \bar{v}_i(t+1) \end{aligned} \tag{1}$$

Eqn. 1 may be interpreted as the 'kinematic' equation of motion for one of the par-ticles (test solution) of the swarm. The variables in the dynamical system of Eqn. 1 are summarized in Table 1.

Table 1- List of variables used to evaluate the dynamical swarm response

$\bar{v}_i$	The particle velocity.
$\bar{x}_i$	The particle position (test solution).
t	Time
ϕ_1	A uniform random variable usually distributed over [0,2].
ϕ_2	A uniform random variable usually distributed over [0,2].
$\bar{x}_{i,p}$	The particle's position (previous) that resulted in the best fit-ness so far.
$\bar{x}_{i,n}$	The neighborhood position that resulted in the best fitness so far.

Algorithm 1: Traditional PSO Algorithm

Initialize swarm (Initialize x_t^n, v_t^n, $\bar{x}_t^n$, $\bar{n}_t^n$ and $\bar{g}_t^n$)
Loop:
    for all particles
        Evaluate the fitness of each particle
        Update $\bar{x}_t^n$, $\bar{n}_t^n$ and $\bar{g}_t^n$
        Update v_{t+1}^n and x_{t+1}^n
    end
until stopping criteria (convergence)

Eqn. 1 can be interpreted as follows. Particles combine information from their previous best position and their neighborhood best position to maximize the probability that they are moving toward a region of space that will result in a better fitness.

3.2 Darwinian Particle Swarm Optimization

A general problem with optimization algorithms is that of becoming trapped in a local optimum. A particular algorithm may work well on one problem but may fail on an-other problem. If an algorithm could be designed to adapt to the fitness function, adjusting itself to the fitness landscape, a more robust algorithm with wider applicability, without a need for problem specific engineering would result. Strategies for avoiding local optima include stretching of Parsopoulos[2] and other convexification[3] strategies

In a typical implementation of PSO, a single swarm of test solutions is utilized. To implement natural selection with a single swarm, the algorithm must detect when stagnation has occurred. Since a single swarm is unable to differentiate between a global optimum and a local optimum it cannot simply be extended to model natural selection. One could "time-out" the optimization and restart the algorithm[4] or delete information about the current global optimum in hopes that the swarm will not return to it. Angeline[5] implemented a type of selection process. At the end of each swarm update, the current fitness of the particles are used to order the particles. The top half of the particles is then duplicated and replace the positions and velocities of the bottom half of the particles. The personal bests of the particles are not changed. The author is able to achieve better convergence on some test problems.

Algorithm 2: DPSO Algorithm

Main Program Loop	Evolve Swarm Algorithm
For each swarm in the collection	For each particle in the swarm
Evolve the swarm (Evolve	Update Particles' Fitness
Swarm Algorithm: right)	Update Particles' Best
Allow the swarm to spawn	Move Particle
Delete "failed" swarms	If swarm gets better
	Reward swarm: spawn particle;
	extend swarm life
	If swarm has not improved
	Punish swarm: possibly delete particle;
	reduce swarm life

In search of a better model of natural selection using the PSO algorithm, we formulate what we call a Darwinian PSO, in which many swarms of test solutions may exist at any time. Each swarm individually performs just like an ordinary PSO algorithm with some rules governing the collection of swarms that are designed to simulate natural selection. The selection process implemented is a selection of swarms within a constantly changing collection of swarms.

3.3 Fractional-order Darwinian Particle Swarm Optimization

In this section, a new method to control the DPSO algorithm based on Pires et al. approach to the traditional SO [14] is introduced and denoted as FO-DPSO. Fractional calculus (FC) has attracted the attention of several researchers [4,24,25], being applied in various scientific fields such as engineering, computational mathematics, fluid mechanics, among others [26–29]. The *Grünwald–Letnikov* definition based on the concept of fractional differential with fractional coefficient $\alpha \in C$ of a general signal $x(t)$ is given by:

$$D^{\alpha}[x(t)] = \lim_{h \to 0}\left[\frac{1}{h^{\alpha}}\sum_{k=0}^{+\infty}\frac{(-1)^{k}\,\Gamma(\alpha+1)\,x(t-kh)}{\Gamma(k+1)\,\Gamma(\alpha-k+1)}\right]$$

where Γ is the gamma function.

An important property revealed by the *Grünwald– Letnikov* eqution (4) is that while an integer-order derivative just implies a finite series, the FO derivative requires an infinite number of terms. Therefore, integer derivatives are "local" operators, while fractional derivatives have, implicitly, a "memory" of all past events. However, the influence of past events decreases over time. Based on Eq. (4), a discrete time implementations expression can be defined as:

$$D^{\alpha}[x(t)] = \frac{1}{T^{\alpha}}\sum_{k=0}^{r}\frac{(-1)^{k}\,\Gamma(\alpha+1)\,x(t-kT)}{\Gamma(k+1)\,\Gamma(\alpha-k+1)}$$

where T is the sampling period and r is the truncation order.

The characteristics revealed by fractional calculus make this mathematical tool well suited to describe phenomena such as irreversibility and chaos because of its inherent memory property. In this line of thought, the dynamic phenomena of particle's trajectory configure a case where fractional calculus tools fit adequately.

Considering the inertial influence of Eq. (1) $w = 1$, assuming $T = 1$ and based on [14] work, the following expression can be defined:

$$D^{\alpha}\left[v_{t+1}^{n}\right] = \rho_1 r_1\left(\breve{g}_t^{n} - x_t^{n}\right) + \rho_2 r_2\left(\breve{x}_t^{n} - x_t^{n}\right) + \rho_3 r_3\left(\breve{n}_t^{n} - x_t^{n}\right)$$

Preliminary experimental tests on the algorithm presented similar results for $r \geq 4$. Furthermore, the computational requirements increase linearly with r, *thatis*, the FO-DPSO present a *O(r)* memory complexity. Hence, using only the first $r = 4$ terms of differential derivative given by (5) and Eq. (6) can be rewritten as (7):

$$\begin{aligned} v_{t+1}^{n} = {} & \alpha v_t^{n} + \frac{1}{2}\alpha v_{t-1}^{n} + \frac{1}{6}\alpha(1-\alpha)v_{t-2}^{n} \\ & + \frac{1}{24}\alpha(1-\alpha)(2-\alpha)v_{t-3}^{n} + \rho_1 r_1\left(\breve{g}_t^{n} - x_t^{n}\right) \\ & + \rho_2 r_2\left(\breve{x}_t^{n} - x_t^{n}\right) + \rho_3 r_3\left(\breve{n}_t^{n} - x_t^{n}\right) \end{aligned}$$

Observing Eq. (7), one can conclude that the DPSO is then considered as being a particular case of the FO-DPSO when $\alpha = 1$ (without "memory"). Moreover, the FO-DPSO may also be seen as a collection of FO-PSOs [14], in which each swarm individually performs with some natural selection rules.

IV. EXPERIMENTAL IMPLEMENTATION AND RESULT ANALYSIS

4.1 Experimental Implementation

In this main part of work, images from MIAS database (http://peipa.essex.ac.uk/info/mias.html) are taken which have some area of interest to be analyzed with refinement. Firstly, image is processed through the Particle swarm optimization based on fractional order Darwinian particle swarm optimization which optimizes the candidate clusters for the Fuzzy C-means algorithm which incorporates the spatial information as well. Furthermore, MATLAB tool is used to develop and implement the integration of the above technique with the level set method to analyze the intricate contours of an image. This method is termed as the virtual method or the combined method and its performance the combined performance. On the other hand, other variations of the above integrated method are also used to analyze the improvement in the performance of image segmentation.

Proposed algorithm and its earlier variations are applied to the performance metrics chosen and its effect on classification accuracy is observed. Time and computational complexity of the proposed method is also analyzed. Finally, the visual results of the images and the supporting objective parameters calculated are given in this section.

4.2 Result Analysis

The experiments and performance evaluation were carried on different images. Both algorithms of spatial FCM and the proposed fuzzy level set method were implemented with MATLAB R2009b. The first experiment is designed to evaluate the usefulness of an initial fuzzy clustering for level set segmentation. It adopted the fast level set algorithm as in [35] for the curve optimization, where the initialization was by manual demarcation, intensity threshold and spatial fuzzy clustering. The experiment is carried out in various number of application fields and some of them are shown along with the numerical values. The numerical values of the performance parameters namely global consistency error, variation of information and Probabilistic Rand Index test the efficiency of the algorithm.

4.3 Image Segmentation using PSO, DPSO and FODPSO

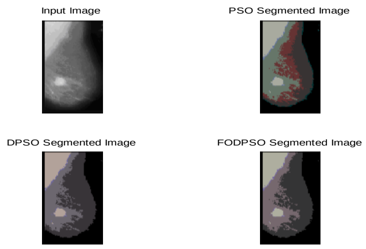

Fig 1: Image segmentation using PSO algorithms

Figure 5.1 depicts the performance comparison on the mammography sample taken from MIAS database. Obviously, due to the weak boundaries, in Figure 5.1 the initial ground truth did not lead to an optimal level set segmentation. On the contrary, both fractional order Darwinian particle swarm optimization and Fuzzy C-means clustering attracted the dynamic curve quickly to the boundaries of interest. It is noteworthy that an image in homogeneity resulted in boundary leakage during Fuzzy C-means clustering. In contrast, the proposed segmentation algorithm with spatial restrictions remedied it substantially in Figure 5.2. to Figure 5.4 depicts the percentage area infected with cancer using the three methods PSO, DPSO and FODPSO.

V. Performance Metrics

5.1 Performance Metrics of PSO segmented image

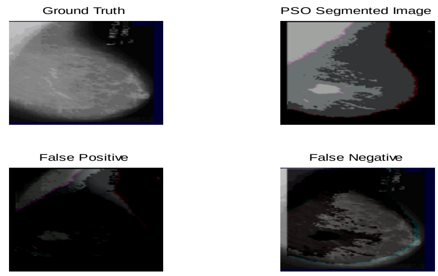

Fig.2: Performance evaluation of PSO image Segmentation

5.2 Performance Metrics of DPSO segmented image

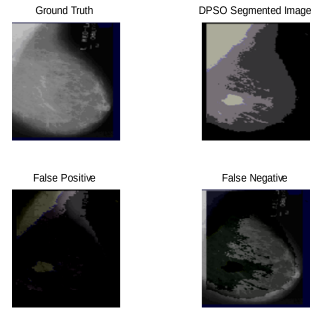

Fig 3: Performance evaluation of DPSO image ***segmentation***

5.3 Performance Metrics of FODPSO segmented image

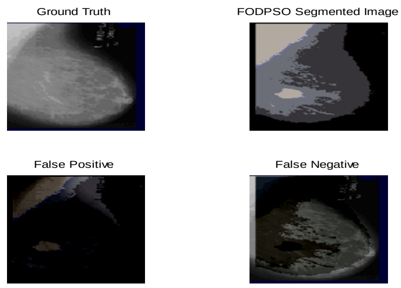

Fig 4: Performance Evaluation of FODPSO image segmentation

5.4 Performance Measures

The performance of medical image segmentation algorithms are evaluated in terms of coefficients of similarity, sensitivity, specificity and segmentation accuracy. The robustness of the algorithms can be tested in the presence of various types of noise.

A large number of literatures on the image segmentation evaluations have been developed in the past few decades. The most common method for evaluating the effectiveness of a segmentation method is subjective evaluation, in which a human visually compares the image segmentation results which is a tedious process. The most important performance criterion is accuracy, that is, the degree to which an algorithm's segmentation matches some reference "gold standard" segmentation.

To perform an evaluation of segmentation it is necessary to verify against a known pixel or voxel wise gold standard [35]. Therefore the radiology community strives to develop realistic and assessable models for research and training. With computer models and manual intervention [36, 37], the true segmentation is known exactly but the images tend to lack realistic anatomical variability and characteristics introduced by the imaging system.

5.4.1 Jaccard Coefficient

A number of similarity coefficients methods [38] are used to specify how well a given segmentation 'A' matches a reference segmentation 'B', where A and B are sets of segmented pixels. The Jaccard coefficient is defined as

$$\text{Jaccard Coefficient} = \frac{(A \cap B)}{(A \cup B)}$$

The Dice Similarity Coefficient (DSC) is often used to measure the accuracy of an automatic segmentation algorithm. The method is simple and appraises the spatial overlap of two binary images. The coefficient ranges from 0 for no overlap, to 1 for a complete overlap. The Dice coefficient [39], is used to assess the accuracy of the semiautomatic segmentation [40] of white matter lesions because, when compared to the Jaccard coefficient, it reflects the intuitive feeling that two regions, of which one fully encompasses the other, are more similar than two partially overlapping regions [41].

5.4.2 Dice similarity coefficient

Dice similarity coefficient expresses the overlap of two regions relative to the sum of the two areas. For two regions A and B, the DSC is given by

$$\text{Dice similarity coefficient} = \frac{2(A \cap B)}{(A)+(B)}$$

In addition to the above, to evaluate the performance of each algorithm several other criteria were used [35]. Respectively TP, TN, FP, FN, stand for the number of samples (i.e., pixels) being labelled as true positive, true negative, false positive and false negative.

5.4.3 True positive (TP): The overlap between the region segmented by the algorithm and the ground truth.

5.4.4 True Negative (TN): The region that is found from both algorithm method and the ground truth.

5.4.5 False positive (FP): The region segmented by the algorithm but is not found in the ground truth.

5.4.6 False negative (FN): The ground truth region which is not included by the algorithm.

Based on the above criteria the following performance measures are defined for validation of 2-D brain tumour segmentation.

Sensitivity

$$\text{Sensitivity} = \frac{TP}{(TP+FN)}$$

Specificity

$$\text{Specificity} = \frac{TN}{(FP+TN)}$$

Accuracy

$$\text{Accuracy} = \frac{TP+TN}{(TP+TN+FP+FN)}$$

Finally, the computational complexity of the algorithms is of interest for using in real time applications.

Evaluating segmentation methods for a particular application, accuracy, reproducibility and running time or amount of user interaction are also important factors to be considered.

Parameter	Segmentation Algorithms		
	PSO	DPSO	FODPSO
Spatial Overlap	0.9653	0.9651	0.9653
Jaccoeff	0.9298	0.9280	0.9282
Accuracy	56.0310	56.4077	57.2386
Sensitivity	93.2918	93.2558	93.2835
Specificity	99.3140	99.0116	99.0229
False Positive	123	179	180
False Negative	2422	2435	2425
Ground Truth	36105	36105	36105

TABLE 1 STATISTICAL PARAMETERS PERFORMANCE FOR THE PROPOSED ALGORITHM IN MEDICAL FIELD

Table 5.1 shows the performance of the parameters chosen for the proposed algorithm. It is clear from this table that minimum value of global consistency error is provided by FO-DPSO based FCMNL algorithm. It continuously improves from neighbourhood information being incorporated to the optimization technique applied. Variation of information is least for fuzzy c-means but probabilistic rand index being the maximum for the proposed algorithm.

Bar-graph given in Figure 5.5. visualizes the performance of the statistical parameters for the proposed algorithm in medical images. It is evident from this figure that the FODPSO based FCMNL algorithm has maximum classification accuracy for the medical image shown.

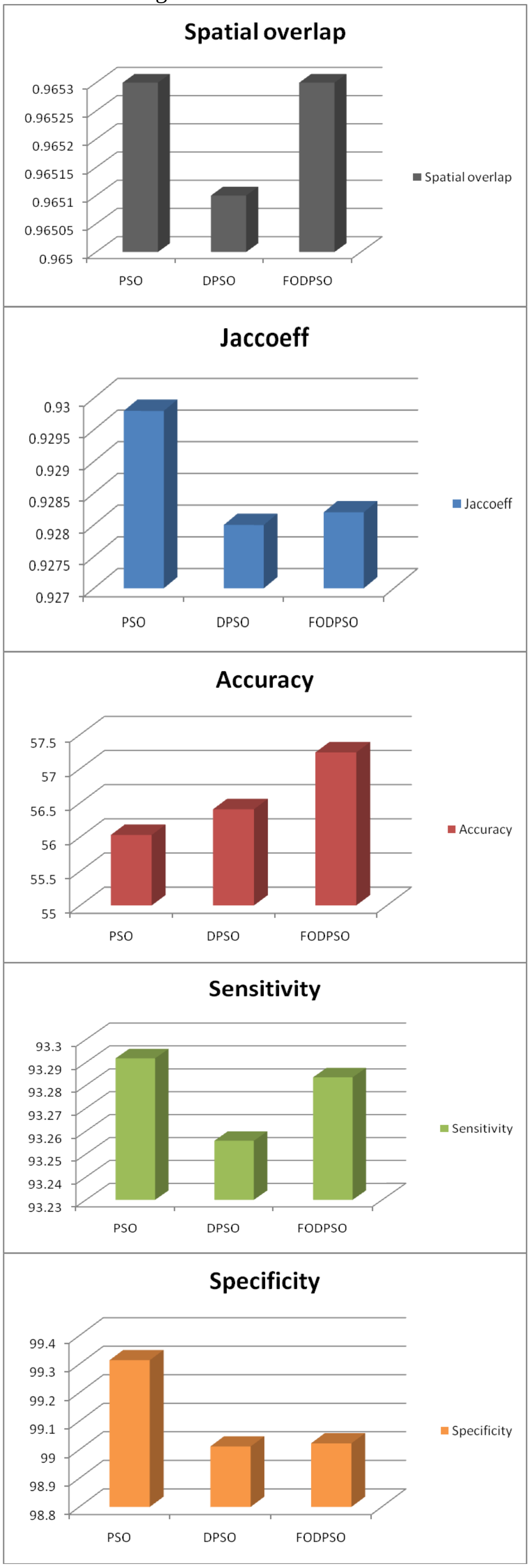

Fig 5: Statistical Parameters performance for the proposed algorithms in mammography

From all this discussion and analysis of various types of images, it is evident that FO-DPSO based FCMNL gives better performance in terms of sensitivity, Specificity, Accuracy. The proposed algorithm seems trivial in some field of images with comparatively clear boundaries. However, in images without distinct boundaries, it would be very important to control the motion of the level set contours. The operator has to monitor level set evolution continuously and adjust various controlling parameters frequently; otherwise inappropriate segmentation would come into being. In contrast, algorithm is able to find out the controlling parameters from Fuzzy C-means clustering automatically. In particular, its solutions are robust and nearly optimal in all cases. But time complexity of the proposed method remains high as compared to the other techniques mentioned. Finally, it is concluded that the proposed algorithm provides a better and improved image segmentation technique.

VI. Conclusions

From all this analysis using four popular image segmentation algorithms: fuzzy c-means, K-means, PDE based level set method and fractional order Darwinian particle swarm optimization, we can conclude following points:

Fuzzy clustering in this algorithm is able to obtain the approximate boundaries of potential components of interest, and is thus suitable to initiate image segmentation. However, the standard FCM algorithms, which are concerned with intensity information only, are not robust enough for all types of image segmentation, due to noise and artefacts. The enhanced spatial FCM attempts to unify intensity and spatial information as a whole.

Level set evolution is subject to various forces from the active curve itself and the image under investigation. It is difficult to coordinate these forces for an optimal image segmentation. Optimal parameters can be achieved only by trial and error for the specific images. The new algorithm is advantageous, because the implicit interface stabilizes once it approaches the genuine boundaries.

Finally, Fuzzy clustering is able to obtain the potential components of interest adaptively. It therefore serves as an effective source of prior knowledge for improved level set segmentation.

Computational complexity is largely reduced using the proposed algorithm as it does not require any more four to five stages processing of an image for the proper image segmentation results.

The proposed algorithm with spatial information can approximate the boundaries of interest well. Therefore, level set evolution will start from a region close to the genuine boundaries.

Bibliography

[1] Gonzalez, Woods, and Eddins, *Digital Image Processing Using MATLAB*, Prentice Hall 2004.

[2] A.K. Jain, M.N. Murty, P.J. Flynn, *Data Clustering: A Review, ACM Computing Surveys, Vol. 31, No. 3, September 1999.*
[3] Hui Zhang, *Improved Clustering-Based Image Segmentation through Learning, A Dissertation, Department of Computer Science and Engineering, August 2007.*
[4] R. Harikumar, B.Vinoth Kumar, G.Karthick, *Performance Analysis for Quality Measures Using K means Clustering and EM models in Segmentation of Medical Images, International Journal of Soft Computing and Engineering, Volume-1, Issue-6, January 2012.*
[5] Feng Zhao, Licheng Jiao, *Spatial improved fuzzy cmeans clustering for image segmentation, International Conference on Electronic and Mechanical Engineering and Information Technology, IEEE 2011, pp 4791-4794.*
[6] Keh-Shih Chuang, Hong-Long Tzen g, Sharon Chen, Jay Wu, Tzong-Jer Chen, *Fuzzy c-means clustering with spatial information for image segmentation, Computerized Medical Imaging and Graphics, 2006, pp 9-15.*
[7] Zhenping Xie Shitong Wang, *A New Level Set Method For Image Segmentation Integrated with FCM, Fourth International Conference on Fuzzy Systems and Knowledge Discovery, IEEE Computer Society- 2007.*
[8] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/
[9] Chuang KS, Hzeng HL, Chen S, Wu J, Chen TJ, *Fuzzy c-means clustering with spatial information for image segmentation, Computerized Medical Imaging and Graphics. 2006, pp. 9–15.*