International Journal of Engineering & Technology, 7 (2.33) (2018) 1046-1052

SPC

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Classification of Kidney Lesions Using Bee Swarm Optimization

[1]Hima Bindu G, [2]Prasad Reddy Pvgd, [3]M. Ramakrishna Murty, [3]S. Pallam Setty

1Research Scholar, Dept. of Computer Science and Systems Engineering, Andhra University
2Dept. of Computer Science and Systems Engineering, Andhra University
3Dept. of CSE, ANITS(A), Visakhapatnam
4Dept. of Computer Science and Systems Engineering, Andhra University
Email: 1hima.gopalam@gmail.com, 2prasadreddy.vizag@gmail.com,
**Corresponding author Email: 3ramakrishna.malla@gmail.com, 4drsetty@gmail.com*

Abstract

Support Vector Machine (SVM) is extensively used in classification due to its prominent features and better generalisation performance. The classification accuracy is highly dependent on the SVM parameters which are currently selected manually. Therefore, the necessity of an automated, fast and reliable approach to determine optimal SVM parameter and produce high classification accuracy has become important requirement for computer aided detection and diagnostic systems. In the current work, SVM parameters are tuned using Bee Swarm Optimization (BSO) approach to find the probability of achieving better classification accuracy. The approach is studied with two kernel function of SVM – polynomial and Gaussian radial basis function. The algorithm is implemented and executed on kidney CT images for classification of kidney lesions. The BSO-SVM Classification results are compared with SVM classification results obtained on the same dataset and it is found that BSO-SVM classification using BSO optimised SVM parameters produced higher classification accuracy.

Keywords *- Bee Swarm Optimization, regularization parameter, Support Vector Machine, kernel function*.

1. Introduction

Classification plays a significant role in data analysis. There are two types of classification algorithms: Supervised and unsupervised which groups the objects as per their similarity. Unsupervised classification uses clustering methods to check related features and form groups and usually used when the class labels are not beforehand. Supervised classification is built on statistical learning theory and the algorithm builds association between inputs and descried class labels (output) when training dataset is presented to it and produces a classification model which can then be used to predict new test datasets. There are many supervised classifiers used in medical applications, some of them are linear discriminant analysis, naive bayes, decision trees, k nearest neighbors (KNN), Support Vector Machines (SVM), etc. SVM is used both in analysis of classification and regression problems. SVM has gained significant importance in the recent times for classification due to its prominent features and better generalisation performance. Some of the attractive features of SVM include, capability of modelling nonlinear relationships, generalisation performance of SVM being independent of the input space dimensions and SVM being associated to quadratic programming analysis, the solution is global and is generally unique. While SVM has rich features, it suffers from a few setbacks that limit the use of SVM in certain areas. SVM parameters (hyper-parameters and kernel parameters) are required to be defined by user and usually selected from prior knowledge, experimental trails and recommendations [2, 13]. The settings of the SVM parameters influence the quality of generalisation performance and hence it is a challenge to select the best SVM parameters that can produce higher generalisation performance. The complexity, is further complicated as the generalisation ability depends on setting of both hyper-parameters and kernel parameters together and hence it is required to find the optimal parameter setting together and not individually [11]. Bio-inspired optimization techniques can be used for optimization of SVM parameters and these techniques are motivated from biological systems principles.

In the earlier work, SVM based classification is experimented on Kidney lesion classification from CT scan images using linear (LIN), polynomial (POLY) and Gaussian radial basis function (RBF) with different polynomial degrees and kernel scales, keeping the regularization parameter constant. The kidney lesions are classified into four classes namely Cysts, AML (Angiomyolipoma), RCC (Renal Cell Carcinoma) tumors and Normal (portion obtained from normal kidney to help in classification of normal kidney tissues). Statistical feature set was used with SVM and it is found that Cubic SVM produced best classification followed by RBF SVM classification. In the current work, Bee Swarm optimization (BSO) technique is implemented to obtain the optimal values of SVM parameters so that it can help in improving the classification accuracy of SVM of kidney lesions from CT images.

This paper is organised into seven sections: Section-1 is introduction, Section-2 describes the study of literature for the current work, Section-3 presents methodology of the current work, Section-4 details SVM algorithm and its parameters, Section-5 details the BSO algorithm, Section-6 presents the simulation and experimental results and Section-7 presents the concluding remarks.

2. Literature Study

The extensive literature survey reveals that the accuracy of classification algorithm relies on the features selection, the efficiency of the classification algorithm itself and the values of parameter passed to the classification algorithm. In the recent times, parameter optimization using bio-inspired computer algorithms have picked up a lot. In the earlier work, experiments have been performed to compute the performance of SVM and KNN on the kidney lesion dataset and it is observed that SVM classifier produced higher classification accuracy. In this literature study, a study has been conducted to find approaches and methods that can improve the classification accuracy by SVM parameter optimization. There are two major groups of bio-inspired algorithms, which are commonly used. One is the Evolutionary algorithms which are based on natural evolution principle. The second group is the Swarm Intelligence algorithms based on the collective behavior in birds, fish, insects, animals, etc [21].

Genetic algorithm (GA) [6, 7, 21] belongs to the group of successful Evolutionary based algorithms and is a stochastic optimization algorithm. Holland introduced the concept of GA and is based on Darwinan's theory of evolution, which is survival of the fittest. In this algorithm, population of chromosomes are selected and the high fitness chromosomes which form better solutions to the required problem are given higher chance to reproduce. In selecting the best solutions, new fit candidates are considered and less fit ones are removed. GA involves three main genetic operators namely selection, cross-over, and mutation. Cross-over and mutation help the characteristics to be exchanged, while selection operator helps in selecting the best fit candidates. GA is an iterative process and each iteration is termed as generation. Typically, generations can range between 50-500 and GA is stopped after reaching a defined number of generations to examine the best candidates in the population. GA is repeated if satisfactory solution is not found.

Particle-Swarm-Optimization (PSO) belongs to the group of Swarm Intelligence (SI) methods and a population based optimization tool. It was first proposed by Eberhart and Kennedy [7]. PSO is motivated from the social behavior of bird / fish swarms (called particles). These particles are initialised randomly and fly freely across the n-dimensional search space exploring for better solutions. The particles update their velocity and position during the flight which are determined from the best solution of the particle itself and the entire particle population. This helps particle swarm to move towards the point objective function which represent highest value, and eventually all particles will collect around the highest objective function point. PSO is also an iterative process and is based on random decisions. The basic objective of PSO is to accelerate every particle towards its best location using a random weighted acceleration.

Artificial Bee Colony (ABC) is another class of population based Swarm Intelligence algorithms which are inspired by the intelligent foraging behavior of honey bees. ABC method was first introduced by Dr. Dervis Karaboga and is used to solve optimization problems. In this algorithm there are three categories of bees known as employed, onlooker and scout bees. The artificial bees fly in multi-dimensional search space in search of best food source. The bees explore the search space for finding feasible solutions. For discovering best solutions, the bees cooperate among themselves and share information. Using this technique, the bees focus mainly on more prominent areas and slowly discard the less prominent areas. ABC works very well for multi-modal and multi-variable function optimization [8, 9].

Bee Swarm Optimization (BSO) algorithm in yet another variant which is also inspired by the intelligent foraging behaviors of honey bees. This algorithm was first developed by Drias et al. [10] and uses three categories of bees (experienced-forager, onlooker and scout bees) which are sorted based on the fitness value. Honey bees with lowest fitness (scout) are used for exploring new solutions and ones with best fitness values are selected as experienced-forager bees. Onlooker bees follow one of the experienced-forager bees selected as its elite. The experienced-forager bees use both the cognitive and social knowledge to discover superior solutions. The different flying behaviors of bees in BSO are used to find best feasible solutions and provide a great opportunity to create proper balance between exploration and exploitation. BSO differs from other bee algorithms by having experienced-forager bees with both cognitive and social knowledge which helps to cope with early convergence problem [5].

Selvaraj et al [1] have conducted a survey in year 2014 on various bio-inspired SI algorithms including PSO, ant colony optimization (ACO), ABC, Fire-fly and their hybridizations along with the application of these algorithms in various fields. In the survey conducted, it was observed that PSO has been used in wide range of applications. Ren et al [2] have presented SVM parameter optimization using Genetic algorithm(GA) and PSO algorithms and using leave-one-out(loo) cross-validation as objective function. GA-SVM and PSO-SVM are compared with grid search method and it was observed that both the methods produced results with comparable quality to that of grid search and with lesser computation costs while for very small step increments of SVM parameters, grid search produced better quality solution. Secondly, the convergence speed of grid search method proved better to GA/PSO SVM and PSO-SVM resulted in convergence speed closer to that of grid search method.

Liu et al [3] proposed a new method where cuckoo search(CS) and PSO are applied on SVM for disease diagnosis. A few cuckoo species lay egg in the host nest mimicking color and pattern of the eggs similar to that of the host species. Cuckoo search is derived from the breeding behaviour and used for solving various optimization problems. In their work, CS method is first applied on SVM to obtain optimal kernel function parameters. These parameters are used as initial parameters for PSO-SVM method and the best optimal SVM parameters are determined. It is observed that CS-PSO-SVM method achieved better classification accuracy compared to accuracy from individual GA/PSO SVM methods. Chen et al [4] have presented PSO-SVM method where the issue of feature selection and model selection are solved simultaneously using the PSO framework and used in breast cancer diagnosis. In the first stage, optimization of feature selection and SVM parameters are conducted dynamically by using PSO algorithm. In the second stage, SVM model uses these selected features and optimal values and performs the classification tasks through 10-fold CV method. Three fitness values, number of support vectors, number of selected features, and classification accuracy are considered here. This helped to attain both good classification accuracy and generalization ability. Akbari et al [5] proposed another variant to BSO by adding two extensions to BSO. They introduced, repulsion factor to overcome the stagnation problem and time varying weights (TVW) to achieve right balance between exploitation and exploration.

From the extensive literature study it is observed that lot of work in the area of numerical optimization been done using Genetic, PSO and ABC algorithms, however parameter optimization in the medical field is still evolving. Not much work is done in the area of SVM parameter optimization for the kidney dataset or by using BSO algorithm. Hence in the current work, BSO algorithm has been experimented on the CT kidney dataset.

3. Methodology

Kidney lesions are obtained from kidney CT images using segmentation techniques. Feature set using statistical approach (first-order-statistics and second-order-statistics) is extracted from the kidney lesion images and used for classification. In the current work, SVM parameter optimization is proposed using Bee Swarm Optimization (BSO) to improve the overall classification accuracy.

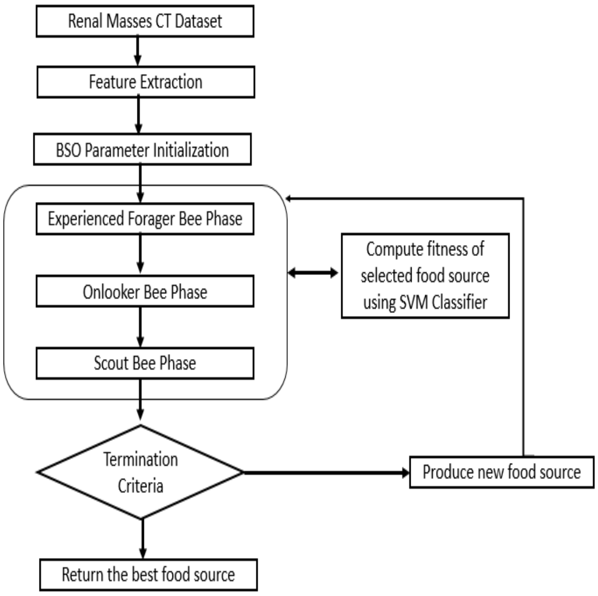

Fig 1: Bso-Svm Workflow

The proposed methodology consists of three main steps:

1. Finding new values of SVM parameters (known as food source / positions in BSO) from the search space depending on the foraging behaviour of bees. Three categories of bees are used in the Bee Swarm optimization and each of them have a different flying pattern.

2. Computing the classification accuracy (fitness) using SVM classifier: Classification model is built based on the SVM parameters and training dataset. The classification model is used on the test dataset to obtain the classification accuracy.

3. Based on the fitness value, the SVM parameters are adjusted and this process is iterated until the best classification accuracy is obtained.

The proposed workflow is executed separately using SVM classifier with polynomial kernel and Gaussian radial basis kernel functions.

4. Svm Algorithm and Svm Parameters

Support Vector Machine (SVM) was developed in 1992 by Vladimir V and used for classification and regression analysis [11-17]. SVM is built on the Vapnik-Chervonenkis (VC) theory and principle of structural risk minimization. To attain best generalization performance, SVM finds a trade-off between maximizing the margin of the hyperplane and minimizing the classification error during training through a value called Error Penalty or Regularization Parameter (C) and remains resilient to over fitting. Higher value of C, will force SVM to minimize the classification errors. Another major benefit of SVM is the usage of convex quadratic programming, which gives only global minima and hence avoids being stuck in local minima. SVM algorithm and parameters are described below.

Let the training data set be represented by (x_i, y_i), where x_i represents input vector and y_i represents the corresponding labels (desired output) and i = {1, 2,….N} and $x_i \in R_D$ and $y_i \in R$. Hyperplane is a decision surface that separates the classes and defined by equation

$$x^T \omega + b = 0 \quad (1)$$

where ω is weight vector and the margin is a non-separable case.

In linear case, the below inequalities hold for all points in training set

$$x_i^T \omega + b \geq 1 - \xi_i \quad \text{for } y_i = 1; \; x_i^T \omega + b \leq 1 + \xi_i \quad \text{for } y_i = -1 \; ; \; \xi_i \geq 0 \quad (2)$$

equation 2 can be combined as

$$y_i (x_i^T \omega + b) \geq 1 - \xi_i; \qquad \xi_i \geq 0 \quad (3)$$

where ξ_i represents the classification error for the misclassified input point x_i.

The objective function for penalized margin maximization can be formulated as:

$$\frac{1}{2}\|\omega\|^2 + C\left(\sum_{i=1}^{N} \xi_i\right)^{\nu} \quad (4)$$

C is regularization parameter which characterizes the generalization performance of the machine, and the positive integer ν ≥1 controls the sensitivity of the SVM to outliers.

Lagrangian Formulation of the primal problem can be formulated as (for ν=1):

$$L_P = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i \{y_i(x_i{}^T\omega + b) - 1 + \xi_i\} - \sum_{i=1}^{n} \mu_i \xi_i \quad (5)$$

where $\mu_i \geq 0$ and $\alpha_i \geq 0$ are Lagrange multipliers

The primal-problem is expressed as:

$$\min_{\omega_k b \xi_i} \max_{\alpha_i} \; Lp \quad (6)$$

Dual lagrangian can be derived using Karush-Kuhn-Tucker(KKT) conditions in the primal lagrangian and formulated as

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i \, y_j \, x_i{}^T x_j \quad (7)$$

The dual-problem is expressed

$$\max_{\alpha_i} L_D \text{ subject to } 0 \leq \alpha_i \leq C \text{ and } \sum_{i=1}^{N} \alpha_i y_i = 0 \quad (8)$$

Support vectors (for data points x_i) can be derived from the equation

$$y_i (x^T \omega + b) \leq 1 \text{ or } \alpha_i (y_i (x^T \omega + b) - 1) = 0 \text{ where } \alpha_i > 0$$

By knowing α_i, we can calculate weight ω for obtaining maximum margin of the hyperplane from the below equation

$$\omega = \sum_{i=1}^{n} \alpha_i y_i x_i; \quad \text{n represents the number of support vectors} \quad (9)$$

Value of b can now be obtained solving the equation $y_i (x_i^T \omega + b) = 1$, where x_i are the support vectors.

Classification can be obtained from the classification rule $h(x) = \text{sign}(x^T \omega + b)$ (10)

Using equation (9), $h(x) = \text{sign}(\sum_{l=1}^{n} \alpha_i y_i x_i x^T + b)$ (11)

Linear separation of the classes is not always possible. Hence for nonlinear cases, a nonlinear mapping function ø is used to transform the initial input space to a higher-dimensional feature space. So the input space $x_i x^T$ is transformed to $ø(x_i)\, ø(x^T)$ feature space and can be represented using the kernel function $k(x_i, x)$. kernel functions are positive symmetric functions that meet mercer's conditions. The classification function can now be expressed as

$$h(x) = \text{sign}(\sum_{l=1}^{n} \alpha_i y_i k(x_i, x) + b) \quad (12)$$

There are many kernel functions that can be used in SVM [14], some of them are as given below:

Linear Kernel: $K(x_i,x_j) = (x_i \cdot x_j)$ (13)

Polynomial Kernel (POLY): $K(x_i,x_j) = (1 + (x_i \cdot x_j))^d$ (14)

d is the polynomial order and equal to 2 and 3 for quadratic for cubic Kernel functions respectively

Gaussian Radial Basis Kernel(RBF):

$$K(x_i, x_j) = \exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2}) \quad (15)$$

$||x_i\text{-}x_j||^2$ is the squared Euclidean distance between the two feature vectors and σ the kernel scale and a free parameter that controls the width of the kernel.

The accuracy of the SVM classification depends on the choice of right parameters (C, kernel function, d, σ) [2]. The regularization parameter C helps SVM to find an optimal hyperplane with maximum margin so that the classification errors are minimized. For larger C value, SVM will select a smaller margin hyperplane and hence helps to reduce the classification errors. Conversely for a smaller C value, larger margin hyperplane is selected which could increase the classification errors. The parameter σ controls the width of the Gaussian kernel. Smaller σ value might force the SVM to over-fit while larger σ value might reduce the flexibility to handle complex function approximation. The kernel function is used in the input space to construct nonlinear decision hyperplane.

In this work, separate optimization is performed to find optimal values of both POLY-SVM and RBF-SVM. Polynomial kernel function SVM and BSO optimization algorithm are used, to obtain the optimal value of (C, d) and classification using these optimal values is termed as BSO-POLY-SVM classification. Secondly Gaussian RBF kernel SVM and BSO optimization to obtain optimal value of (C, σ) and classification using these optimal values is termed as BSO-RBF-SVM classification. BSO algorithm is explained in the next section.

5. Bee Swarm Optimization

Bee Swarm Optimization (BSO) is a technique inspired from nature for optimization and derived from the intelligent behaviour of honey bees [5,18]. In this algorithm, behaviours of three categories of honey bees are considered: experienced-forager, onlooker and scout bees. Each category of bee uses a distinctive flying pattern. The principle of this algorithm is to exploit the richest food source. The scout bees fly to (explore) surrounding regions randomly in search of new food sources. A very low percent of the total bee population is assigned as scout bees. Experienced-forager bees provide information relating to food sources to the onlooker bees by performing dance. The intensity of dance performed by the forager bees is relative to the quality of food source. The onlooker bees use probabilistic approach to select the experienced-forager bee based on the dance performed and adjust its search path towards the richest food location. For optimization problems, the food source is considered as a position in the search space and quality of the food source determines the fitness of the position. The experienced-forager bee can memorize the historical information relating to the position and quality of food sources. Using the historical information, the experienced-forager bees can make intelligent decisions next time. The experienced-forager bee also chooses the best experienced-forager bee as its elite and updates its position based on both the cognitive (information obtained from self by memorizing the best food source visited so far) and social knowledge (information from the elite bee), thus exploiting the richest food sources. These different behaviours of bees in BSO provides proper balance between exploitation and exploration thus helping to find best solutions to optimization problems.

The total number of bees in the swarm is represented as β, number of experienced-forager bees as ε, number of onlooker bees as κ, number of scout bees as υ such that $\beta = (\varepsilon \cup \kappa \cup \upsilon)$. For each bee h, food source is represented by the position vector $\vec{x}(\beta,h)$ and the quality of food source is expressed as fitness function, $fitness(\vec{x}(\beta,h))$. Maximum number of iterations, stop criteria, radius range are defined during initialization. At the beginning of algorithm, random search spaces are assigned to all the bees in the swarm. Bees are categorized into three groups based on the fitness value during every iteration. The fitness values of all bees are calculated and position of lowest fitness food source is assigned as scout bees which is a very small percentage of total bees. The remaining swarm is equally divided as experienced-forager bees and onlooker bees. The ones with the best fitness value are assigned as experienced-forager bees and the remaining are assigned as onlooker bees. Using the flying patterns, the bees adjust the trajectory in the search space for finding new and richer food sources.
An experienced-forager bee h memorizes the best position of the best food source which is denoted as $\vec{b}(\varepsilon,h)$. if the fitness of the new food source is better than the best food source memorized by experienced-forager bee, the position is replaced to the new position. And elite bee is the one of the experienced-forager bee with highest fitness amongst ε. The fitness of elite bee is represented as $\vec{e}(\varepsilon,*)$.

$$If\ (fitness(\vec{x}(\varepsilon, h)) > fitness(\vec{b}(\varepsilon, h))\ \text{then}\ \vec{b}(\varepsilon, h) = \vec{x}(\varepsilon, h) \quad (16)$$

$$If\ (fitness(\vec{b}(\varepsilon, h)) > fitness(\vec{e}(\varepsilon,*))\ \text{then}\ \vec{e}(\varepsilon,*) = \vec{b}(\varepsilon, h) \quad (17)$$

The position of the experienced-forager bee is updated both on the cognitive and social knowledge and represented as

$$\vec{x}_{new}(\varepsilon, h) = \vec{x}_{old}(\varepsilon, h) + w_b r_b(\vec{b}(\varepsilon, h) - \vec{x}_{old}(\varepsilon, h)) + w_e r_e(\vec{e}(\varepsilon,*) - \vec{x}_{old}(\varepsilon, h)) \quad (18)$$

where r_b, r_e are random variables in the range [0,1] and w_b, w_e are control parameters for the best food source of h^{th} bee and the elite bee.

The onlooker bee j uses probabilistic approach to select the experienced-forager bee m from the set ε. The relative fitness of selected experienced-forager bee (m) is represented with the probability function P_m defined as

$$P_m = \frac{fit(\vec{x}(\varepsilon,m))}{\sum_{q=1}^{n(\varepsilon)} fit(\vec{x}(\varepsilon,q))} \quad (19)$$

Onlooker bees use roulette wheel method for selecting their elite bees from the experienced-forager bees. The position of the onlooker bee (j) is updated using the following equation:

$$\vec{x}_{new}(\kappa, j) = \vec{x}_{old}(\kappa, j) + w_e r_e(\vec{e}(\varepsilon, j) - \vec{x}_{old}(\kappa, j)) \quad (20)$$

Scout bees randomly search the region with center as current position and with radius τ to find new food sources. The position of the scout bee (j) is updated using the following equation:

$$\vec{x}_{new}(\upsilon, j) = \vec{x}_{old}(\upsilon, j) + Rw(\tau, \vec{x}_{old}(\upsilon, j)) \quad (21)$$

Where Rw is the random-walk function which is dependent on radius search and scout bee's current position. The value of radius search τ decreases over iterations to allow scout bees to walk more precisely inside smaller regions. The pseudo code for BSO algorithm is given in Fig 2.

Initialization: define

total number of bees

% of experienced-foragers, onlooker and scout bees manually

dimension n

Radius τ

termination condition

max #. of iterations

associate a random position in the search space to all bees in the population

Do

Compute the fitness of all bees

Sort bees based on their fitness

partition the bee population into experienced-foragers (ε), onlookers(κ) and scouts(υ)

#Experienced-forager Bee Phase

for each experienced-forager bee

update prior best position based on fitness

select elite bee which is best among all experienced-forager bees (best position)

for each experienced-forager bee

update position of experienced-forager bee in all dimensions in search space

#Onlooker Bee Phase

for each onlooker bee

select elite bee for onlooker bee

update position of onlooker bee in all dimensions in search space

#Scout Bee Phase

for each scout bee

update position (random-walk in the search space)

Adjust radius τ

until stop criteria

Fig 2: Pseudo code for BSO Algorithm

6. Experimental Study and Results

Dataset: CT images containing kidney lesions are obtained from medical hospitals and from medical databases. Dataset containing four classes is considered: Cysts, AML, RCC Tumor and Normal. Implementation of feature extraction, SVM classification using BSO is performed using Matlab.

14 features (Mean, Standard Deviation, RMS, Variance, Kurtosis, Skewness, L1-Norm, L2-Norm, Smoothness, Contrast, Correlation, Energy, Homogeneity, Entropy) are extracted using First-order-statistics and Second-order statistics for the data set considered in this experimentation. First-order-statistics features are texture measures that are computed from the pixel's values directly while and Second-order statistics features are obtained from the spatial relationship of pixels using Gray-level-co-occurrence matrix.

In the earlier work, SVM classification has been implemented using these features extracted from kidney lesion CT images and classification was performed applying polynomial SVM (quadratic and cubic) and RBF SVM (with σ = 0.9, 3.6 and 14 and termed as Fine-Gaussian, Medium-Gaussian, Coarse-Gaussian respectively based on σ value) and their classification accuracy results are used as baseline data for the current experimentation. In the current work, BSO-POLY-SVM and BSO-RBF-SVM classification approaches are implemented and the results obtained are compared with the baseline data.

Optimization of SVM parameters is performed using BSO-SVM. Both polynomial and Gaussian RBF kernels are included in the search space individually and range of the SVM parameters are defined, (C, d, σ). The search space is explored to obtain optimal values of (C, d) for POLY SVM and optimal values of (C, σ) for RBF SVM separately. The initial values for BSO algorithm (number of bees, number of experienced-forager bees, onlooker bees, scout bees, radius of search, and maximum number of iterations) are configured as given in Table 1. Fitness function selected for the model is classification accuracy and termination condition is defined as when there are no changes in position for three consecutive iterations or when maximum iterations are reached.

The following range of values is chosen for each of the SVM parameters under consideration: $0.1 \leq C \leq 15$; $2 \leq d \leq 6$; $0.1 \leq \sigma \leq 15$. The values are selected from previous knowledge and literature study.

Table 1: Parameter setting for BSO-SVM Algorithm

BSO-SVM Variable	**Value / Range**
Population Size	40
Maximum number of iterations	50
Dimensions	14
Radius Range	[0.2 , 1]
Number of Scout bees	2
Number of Experienced-forager bees	19
Number of Onlooker bees	19

Classification accuracy is calculated using the following formula:

$$\text{Accuracy (ACC)} = \frac{\textit{Correctly classified cases}}{\textit{Total cases}} \quad (22)$$

Firstly, BSO-SVM algorithm is run on polynomial kernel SVM classifier to obtain optimal values of (C, d) for POLY SVM and the corresponding classification accuracy of BSO-POLY-SVM is computed. The results of polynomial SVM are presented in Table 2 and Figure 3. The first two rows of the Table 2 (Quadratic SVM and Cubic SVM) represent test results from earlier and the third row is from the new BSO-POLY-SVM classifier.

Table 2: Classification Accuracy with Polynomial SVM for different Parameters

Classification	**No. of Features**	**C (Regularization Parameter)**	**d (Polynomial)**	**Accuracy**
Quadratic SVM	14	2	2	91.7%
Cubic SVM	14	2	3	93.8%
BSO-POLY-SVM	14	8.8	3	97.4%

Optimal values of C and d obtained are 8.8 and 3 respectively and with classification accuracy of 97.4%. This is a good improvement on the classification accuracy using optimal values of (C, d) for SVM classifier with polynomial kernel function.

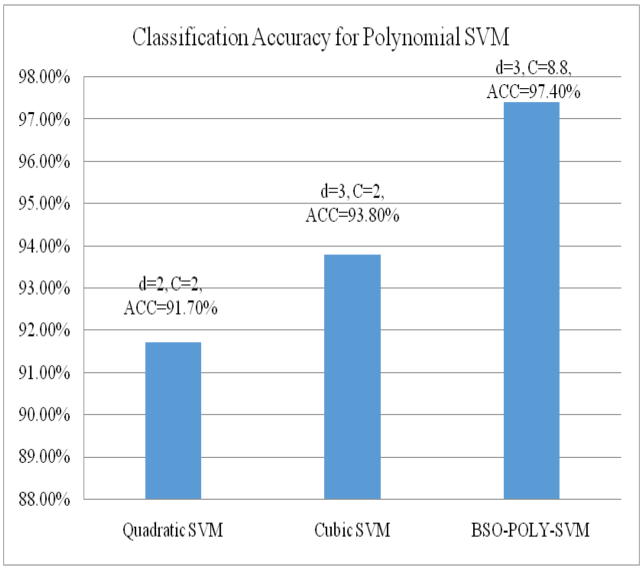

Fig 3: Classification Accuracy for Polynomial SVM

Next BSO-SVM algorithm is run using Gaussian RBF SVM classifier to obtain optimal values of (C, σ) and the corresponding classification accuracy of BSO-RBF-SVM is computed. The results of RBF SVM are presented in Table 3 and Figure 4. The first three rows of the Table 3 (Fine-Gaussian SVM, Medium-Gaussian SVM, Coarse-Gaussian SVM) represent test results earlier row and the last row is from the new BSO-RBF-SVM classifier.

Table 3: Classification Accuracy with RBF SVM for different Parameters

Classification	**No.of Features**	**C (Regularization Parameter)**	**σ**	**Accuracy**
Fine Gaussian SVM	14	2	0.9	68.6%
Medium Gaussian SVM	14	2	3.6	85.3%
Coarse Gaussian SVM	14	2	14	72.5%
BSO-RBF-SVM	14	9.4	4.6	95.7%

Optimal values of C and σ obtained are 9.4 and 4.6 respectively and with classification accuracy of 95.7%. This is a significant improvement on the classification accuracy using optimal values of (C, σ) for SVM classifier with Gaussian radial basis kernel function.

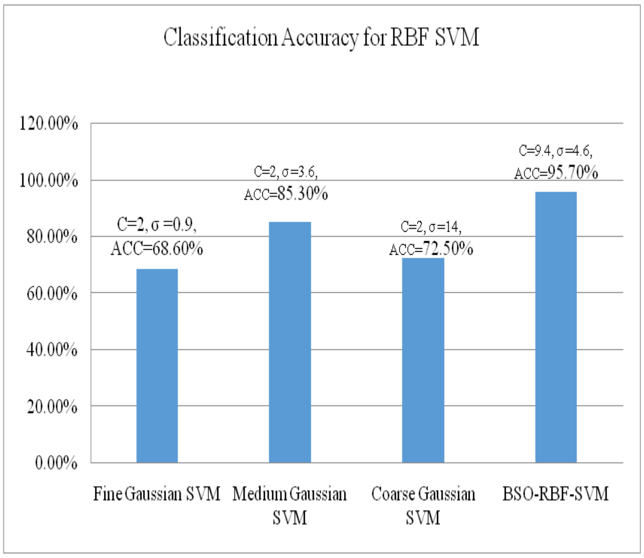

Fig 4: Classification Accuracy for RBF SVM

7. Conclusions and Future Work

In this paper, kidney lesion classification has been implemented using the BSO optimization for SVM classifier with POLY kernel and SVM classifier with RBF kernel separately to obtain optimal values of (C, d) and (C, σ) respectively. It is observed that classification accuracy has improved in both the cases. While for polynomial kernel, the classification accuracy has increased by 3.6%, for Gaussian radial basis kernel function the increase is quite significant and increased by 10.4%.

References:

[1] Selvaraj, Cholavendhan, R. Siva Kumar, and M. Karnan. "A survey on application of bio-inspired algorithms." International Journal of Computer Science and Information Technologies 5.1 (2014): 366-70.
[2] Ren, Yuan, and Guangchen Bai. "Determination of optimal SVM parameters by using GA/PSO." JCP 5, no. 8 (2010): 1160-1168.
[3] Liu, Xiaoyong, and Hui Fu. "PSO-based support vector machine with Cuckoo search technique for clinical disease diagnoses." The Scientific World Journal 2014 (2014).
[4] Chen, Hui-Ling, Bo Yang, Gang Wang, Su-Jing Wang, Jie Liu, and Da-You Liu. "Support vector machine based diagnostic system for breast cancer using swarm intelligence." Journal of medical systems 36, no. 4 (2012): 2505-2519.
[5] Akbari, Reza, Alireza Mohammadi, and Koorush Ziarati. "A novel bee swarm optimization algorithm for numerical function optimization." Communications in Nonlinear Science and Numerical Simulation 15, no. 10 (2010): 3142-3155.
[6] Holland, John Henry. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, 1992.
[7] Jones, Karl O. "Comparison of genetic algorithm and particle swarm optimization." In Proceedings of the International Conference on Computer Systems and Technologies. 2005.
[8] Karaboga, Dervis, and Bahriye Basturk. "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm." Journal of global optimization39, no. 3 (2007): 459-471.
[9] Karaboga, Dervis, and Bahriye Akay. "A comparative study of artificial bee colony algorithm." Applied mathematics and computation 214, no. 1 (2009): 108-132.
[10] Drias, Habiba, Souhila Sadeg, and Safa Yahi. "Cooperative bees swarm for solving the maximum weighted satisfiability problem." In International Work-Conference on Artificial Neural Networks, pp. 318-325. Springer, Berlin, Heidelberg, 2005.
[11] Yu, Hwanjo, and Sungchul Kim. "Svm tutorial—classification, regression and ranking." In Handbook of Natural computing, pp. 479-506. Springer Berlin Heidelberg, 2012.
[12] Jakkula, Vikramaditya. "Tutorial on support vector machine (svm)." School of EECS, Washington State University 37 (2006).
[13] Vapnik V. The nature of statistical learning theory. Springer science & business media; 2013 Jun 29.
[14] Vapnik, Vladimir Naumovich. "An overview of statistical learning theory." IEEE transactions on neural networks 10, no. 5 (1999): 988-999.
[15] Cristianini N, Shawe-Taylor J. An introduction to support vector machines and other kernel-based learning methods. Cambridge university press; 2000 Mar 23.
[16] Huang, Cheng-Lung, and Chieh-Jen Wang. "A GA-based feature selection and parameters optimization for support vector machines." Expert Systems with applications 31, no. 2 (2006): 231-240.
[17] Lin, Shih-Wei, Kuo-Ching Ying, Shih-Chieh Chen, and Zne-Jung Lee. "Particle swarm optimization for parameter determination and feature selection of support vector machines." Expert systems with applications 35, no. 4 (2008): 1817-1824.
[18] Akbari, Reza, Alireza Mohammadi, and Koorush Ziarati. "A powerful bee swarm optimization algorithm." In Multitopic Conference, 2009. INMIC 2009. IEEE 13th International, pp. 1-6. IEEE, 2009.
[19] Lahari, Kannuri, M. Ramakrishna Murty, and Suresh C. Satapathy. "Partition based clustering using genetic algorithm and teaching learning based optimization: performance analysis." In Emerging ICT for Bridging the Future-Proceedings of the 49th Annual

Convention of the Computer Society of India CSI Volume 2, pp. 191-200. Springer, Cham, 2015.

[20] Murty, M. Ramakrishna, J. V. R. Murthy, PVGD Prasad Reddy, Anima Naik, and Suresh Chandra Satapathy. "Performance of teaching learning based optimization algorithm with various teaching factor values for solving optimization problems." In Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2013, pp. 207-216. Springer, Cham, 2014.

[21] Binitha S. and Sathya, S.S., 2012. A survey of bio inspired optimization algorithms. International Journal of Soft Computing and Engineering, 2(2), pp.137-151.