ORIGINAL RESEARCH

Hyperspectral image classification incorporating bacterial foraging-optimized spectral weighting

Ankush Chakrabarty[1], Olivia Choudhury[2], Pallab Sarkar[3], Avishek Paul[3], Debarghya Sarkar[3]

1. Electrical Engineering Department, Purdue University, West Lafayette, USA. 2. Computer Science Department, Meghnad Saha Institute of Technology, Kolkata, India. 3. Electrical Engineering Department, Jadavpur University, Kolkata, India.

Correspondence: Ankush Chakrabarty. Address: Electrical Engineering Department, Purdue University, West Lafayette, USA. Telephone: 1-313-455-0253. E-mail: chakraa@purdue.edu

Abstract

The present paper describes the development of a hyperspectral image classification scheme using support vector machines (SVM) with spectrally weighted kernels. The kernels are designed during the training phase of the SVM using optimal spectral weights estimated using the Bacterial Foraging Optimization (BFO) algorithm, a popular modern stochastic optimization algorithm. The optimized kernel functions are then in the SVM paradigm for bi-classification of pixels in hyperspectral images. The effectiveness of the proposed approach is demonstrated by implementing it on three widely used benchmark hyperspectral data sets, two of which were taken over agricultural sites at Indian Pines, Indiana, and Salinas Valley, California, by the Airborne Visible Infra-Red Imaging Spectrometer (AVIRIS) at NASA's Jet Propulsion Laboratory. The third dataset was acquired using the Reflective Optical System Imaging Spectrometer (ROSIS) over an urban scene at Pavia University, Italy to demonstrate the efficacy of the proposed approach in an urban scenario as well as with agricultural data. Classification errors for One-Against-One (OAO) and classification accuracies for One-Against-All (OAA) schemes were computed and compared to other methods developed in recent times. Finally, the use of the BFO-based technique is recommended owing to its superior performance, in comparison to other contemporary stochastic bio-inspired algorithms.

Key words

Support vector machines, Bacterial foraging optimization, Hyperspectral image classification, Spectral weighting

1 Introduction

In recent years, hyperspectral imaging has become extremely popular in remote sensing and aerial reconnaissance activities [1, 2]. The technique involves the acquisition of spectral information over a large region and the consequent analysis and interpretation of the acquired data [8]. Availability of advanced sensor technology such as NASA's Airborne Visible-Infrared Imaging Spectrometer (AVIRIS) has resulted in the collection of spectral data over terrestrial regions at high, medium and low altitudes through adjacent multi-band channels [3], enabling its utilization in a multitude of applications ranging from environmental planning and assessment, monitoring of oil spills, geological research, and target detection in military applications [2-6]. Prolific research has taken place in the last decade in the development of novel mathematical formalisms to solve the image classification problem of hyperspectral data. Hyperspectral image

classification involves the identification of various pixels in the hyperspectral image sequence on the basis of their spectral signatures. These spectral signatures are unique electromagnetic "fingerprints" left uniquely by an object and can thus be used to distinguish between objects in the hyperspectral image. Variability in spectral signatures, presence of noise, high dimensionality of data (Hughes' phenomenon [7]) and the high correlation of features coupled with the limited amount of training data (pixels on the hyperspectral image set whose truth values are known to us from the ground truth data) makes the problem of supervised hyperspectral image classification (an image classification formalism in which a small portion of the image, called the training set whose truth values are known, is used to train an intelligent classification system to identify to which class each pixel belongs) an extremely difficult endeavour [9-12]. In this respect, a methodology using artificial intelligence-based computing has been used to provide a viable solution to this complex problem.

Extensive research in remote sensing applications has shown the superiority of neural computation techniques such as Artificial Neural Networks (ANN) [13-16] as well as kernel-based methods such as Support Vector Machines (SVMs) [17-19, 21-24] in ill-posed hyperspectral image classification problems over older technologies such as maximum likelihood or Bayesian methods [13, 20]. The effectiveness of incorporating a priori knowledge of the hyperspectral data obtained with customized kernels has been explored in [26, 54], demonstrating the discriminatory nature of hyperspectral data over a range of spectral bands and indicating that only a fraction of the spectrum contains information valuable in the classification context. Widely used feature extraction technologies such as PCA, ICA, LDA, 3D-DWT have been utilized previously with the major drawback of these techniques being the inevitable loss of important data [27-30]. A weighting scheme was proposed in [26] which provided a suitable proportion of each spectral band to be used for the purpose of classification while retaining complete original information in all the spectral bands during the employment of various classification strategies. Optimal spectral weights were derived using gradient descent, mutual information and Bhattacharya distance and were integrated into the support vector machine classifier in order to boost the classification accuracy.

The present work describes how a population-based bio-inspired global optimization algorithm seeks optimal spectral weights to effectively customize the SVM kernel in order to leverage the accuracy of hyperspectral image classification without the need for first-derivative information, that is, information from the gradient (or its derivatives) of the function as in the case of gradient-based methods. The Bacterial Foraging Optimization (BFO) technique [49] is a modern population-based evolutionary optimization tool simulating the behavior of biological swarms for solving multi-dimensional global optimization problems. It is based on Darwinian evolution, asserting the "survival of the fittest", and has obtained extraordinary significance in the domain of optimization algorithms [32]. This nature-inspired algorithm is based on the principle of foraging, in which a bacterium seeks out nutrients in minimum time with the minimal use of energy in an environment containing nutrients as well as noxious substances which are analogous to minima and maxima respectively in a minimization problem. The bacteria travel towards nutrients conducive to their metabolism and the fittest bacteria survive. This phenomenon has motivated researchers to develop an optimization technique in which an iterative search is performed on a multi-dimensional search space, making best use of previous best positions encountered by itself, as well as other members of the population [47]. The BFO technique iteratively optimizes a given objective by sharing mutual information about the optimality of each of the previously encountered points on the search space in order to finally arrive at a global optimum.

In the proposed approach, a fraction of the original hyperspectral data is selected for training and the rest is reserved for testing. Next, a radial-basis function (RBF) kernel SVM is employed for bi-classification with randomly initialized spectral weights and the margin of the SVM is used as a viable objective function. Bi-classification of images means the categorization of the pertinent pixels in the hyperspectral image into two classes. This concept may be extrapolated for multi-classification algorithms where more than two classes are formulated. The BFO algorithm iteratively maximizes this objective and seeks the optimal spectral weight vector. Finally, the newly designed customized kernel SVM is used to classify the testing pixels over multiple test runs. The utility of the proposed scheme is demonstrated by implementing it on three popular benchmark hyperspectral image sets, and comparing the effectiveness of the proposed approach vis-à-vis

other methods developed so far. The superiority of the Bacterial Foraging Optimization technique in optimizing spectral weights is conclusively proven in this paper by significantly outperforming many popular contemporary bio-inspired algorithms.

The rest of this paper is organized as follows. Section 2 provides an overview of the theory of the support vector machine classifier and describes its utility in supervised learning problems in the image classification paradigm as well as the development of spectrally weighted kernel functions. Section 3 discusses the Bacterial Foraging Optimization in detail and illustrates how the evolutionary computation tool is implemented in spectral weight estimation. Section 4 presents the performance evaluation of the proposed approach when applied to three popular datasets for OAO and OAA bi-classification for comparison with the findings with that of previous reports as well as the assessment of the efficacy of the BFO-based strategy over competing algorithms such as the Particle Swarm Optimization (PSO), the Artificial Bee Colony (ABC) and the Genetic Algorithms (GA) [31, 33, 34, 50, 53]. Finally, conclusions inferred from the results are discussed in Section 5.

2 Support vector machines

2.1 Classical SVM theory

The support vector machine (SVM) is a controlled bi-classification strategy which was developed in the mid-90's by Cortes and Vapnik [45]. It aims to distinguish between the given samples of two different classes by seeking an optimal hyperplane which introduces maximum distance (margin) between these two classes in the original feature space (linear SVM) or modifies the feature space by a non-linear mapping φ (nonlinear SVM) in order to induce separability amongst them when the classes are not linearly separable [41]. In the present paper, we have utilized SVM for two classification strategies---One-Against-All (OAA), where the objective is to discriminate between a single class amongst all present samples and One-Against-One (OAO), where the target is to differentiate solely between the samples of two unique classes when provided with a limited data set [43]. We present the basic relations required for SVM classification; a detailed discussion of SVMs can be found in [44, 45].

Cover's theorem of separability of patterns states that the chances of linear separability of inherently non-linearly separable patterns is leveraged if the input feature space is projected onto a higher dimensional Hilbert space H by the means of a nonlinear mapping function [36, 39, 40]. This can be expressed mathematically in its inner product form by

$$f(x) = \sum_{i=1}^{N} w < \varphi(x), \varphi(x_i) > + \ b$$

(1)

In accordance with Mercer's theorem [42, 43], we can replace the inner product of the relatively unknown nonlinear mapping φ with a kernel function K of positive magnitude such that

$$\langle \varphi(x_i), \varphi(x_j) \rangle = K(x_i, x_j)$$

(2)

Now let us consider d-dimensional pixel vectors x and x_i which denote hyperspectral data where i denotes the index number of each particular pixel. The corresponding kernel-based SVM classifier can be described by the revised equation:

$$f(x) = \text{sgn}(\sum_{i=1}^{N_s} y_i \alpha_i K(x_i, x) + b)$$

(3)

Where $y_i \in [-1, +1]$ are the labels of the corresponding i th samples, α_i are the Lagrange multipliers, b is a threshold term and N_s is the total number of samples.
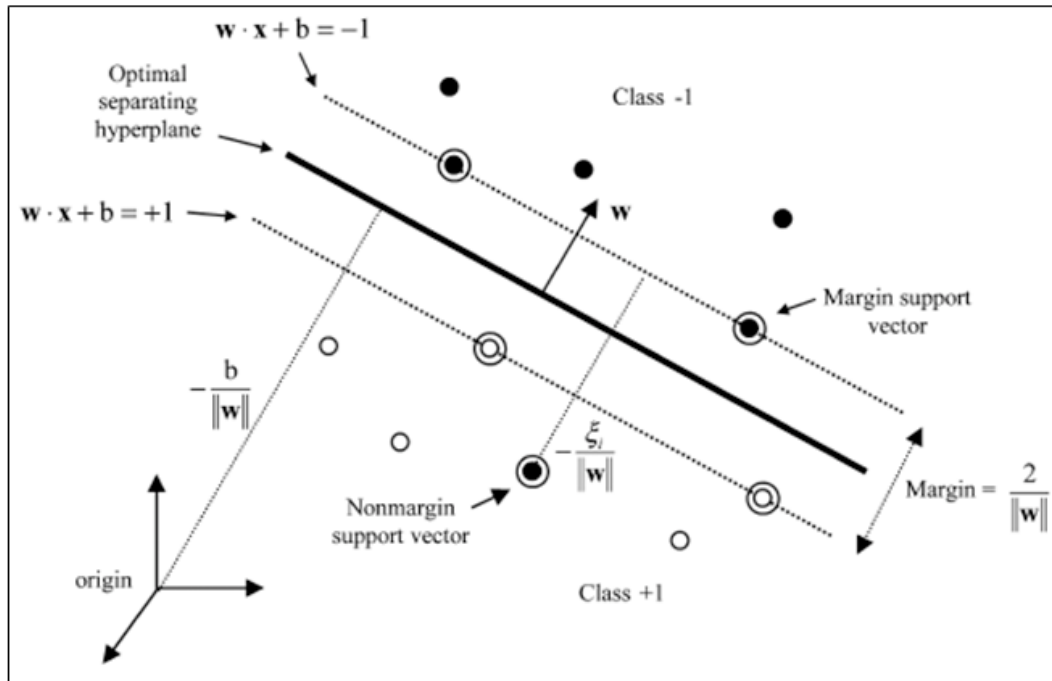
Figure 1. A Graphical Representation of the SVM bi-classifier

Some widely used kernels that satisfy Mercer's conditions are:

Radial Basis Function (RBF) Kernel:

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

(4)

Polynomial Kernel:

$$K^{POLY}(x, x) = (x_i^T x_j + 1)^d$$

(5)

During the training phase of the SVM, the objective is to maximize the margin by seeking optimal SVM parameters. The bi-classification scheme using SVM is depicted pictorially in Figure 1. The margin which must be expanded to ensure high classification accuracy is given by the equation

$$\lambda = \frac{2}{\|w\|}$$

(6)

Where w is the d-dimensional vector orthogonal to the optimal separating hyperplane which is denoted by:

$$w = \sum_{i=1}^{T} \alpha_i y_i \varphi(x_i)$$

(7)

Here, T is the total number of training samples, α_i is the ith Lagrange multiplier, and $\varphi(x_i)$ denotes the mapped ith training sample. It follows from (6) and (7) that in order to maximize the SVM margin, w must be minimized. In mathematical terms, we can write this as an objective function J, where

$$J = \left\| w^2 \right\| = \sum_{i,j=1}^{T} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

(8)

2.2 Spectrally weighted kernels

The general kernels described in (4) and (5) do not have an inherent discriminatory nature with regard to the spectral bands of the input samples. They provide equal emphasis to all the spectral components while projecting onto the modified feature space, thereby retaining the components which have meager quantities of essential information. It was proposed [26] to introduce a spectral weighting scheme by modifying kernel functions using a spectral weight vector $s = \{s_1, s_2, ..., s_d\}$ for d-dimensional data in order to maintain the original information by avoiding feature extraction. Instead the scheme was altered by incorporating *a priori* information in this spectral weight vector, which can be represented in matrix form by $S = diag(s)$. Integrating this spectral weight matrix into the kernels in (4) and (5) results in the following customized kernel functions:

$$K_{SW}^{POLY}(x, x') = (x^T S^T S x' + 1)^d$$

(9)

$$K_{SW}^{RBF}(x, x') = \exp\left\{ \frac{-\left\| S(x - x') \right\|^2}{2\sigma^2} \right\}$$

(10)

Using these tailored kernels, the non-uniformity of spectral distribution in hyperspectral data can be incorporated into the SVM learning scheme thereby enabling lower values of spectral weights to de-emphasize certain bands which are not essential to the classification objective. Placing higher weights on bands containing informative data and reducing the effect of others introduces a measure of feature extraction into the proposed approach. Thus, Hughes phenomenon [7] which relates an increase in classification accuracy with a decrease in dimensionality of data aptly explains the gradual increase of classification accuracy as the tuning of the SVM parameters are carried out with spectral information embedded in its kernels. After the kernel is updated with the optimal spectral weight vector, the objective function in (8) can be reformulated as:

$$J = \left\| w^2 \right\| = \sum_{i,j=1}^{T} \alpha_i \alpha_j y_i y_j K_{SW}(x_i, x_j)$$

(11)

Minimization of this objective involves the seeking of the optimal spectral weight vector s^* and results in the formulation of the trained SVM. The outputs of the classifier on the testing set can be described by the equation (3). In order to minimize the objective function described in equation (11) various possible methods exist such as gradient descent or relevance evaluation [26], but due to the inherent non-linearity of the optimization problem, a derivative-free global optimization method is conducive towards seeking the optimal spectral weights. In this context, a stochastic bio-inspired population-based optimization technique such as the BFO has been proposed as an appropriate optimization method.

3 Bacterial foraging optimization

Foraging strategies denote the phenomenon of search, handling and ingestion of nutrients [49, 56]. The implementation of the Bacterial Foraging Optimization algorithm, which is a modern global optimization tool using iterative stochastic searches, is a computational analogue of the behaviour of intestinal bacteria in their search for nutrients in a hostile environment with minimal loss in energy. This technique, proposed by Kevin Passino in 2002, was inspired by the group foraging behaviour of Escherichia coli in the human intestine [55]. The BFO algorithm comprises four stages, namely: Chemotaxis, Swarming, Reproduction and Elimination-Dispersal. It performs these four actions iteratively to obtain a global optimum if any is present. In this section the bacterial foraging procedure is briefly presented. A full and detailed mathematical discussion can be found in [49].

Let $\theta \in R^p$, where R^p denotes a p-dimensional vector of real numbers which is a trial solution in the search space and $J(\theta)$ be any objective function whose global optimum is of interest and whose gradient may not be determined analytically. Different values of the function denote different conditions of the nutrient surface in the search space which the bacteria currently occupy. Negative, zero and positive values of $J(\theta)$ represent nutrient-rich, neutral and noxious gradients, respectively. Hence, the aim of the BFO is to minimize the objective function to facilitate bacterial growth [56]. The different stages of the Bacterial Foraging Optimization have been described as follows:

3.1 Chemotaxis

Movement in E. coli is enabled by a set of flagella, with the help of which the bacteria can alternate between tumbling and running according to the nutrient gradient – a phenomenon known as Chemotaxis which can be utilized in optimization procedures [35]. A tumble step of the bacteria is exploitative in nature as it searches within a small region in the search space, whereas the alternate swimming step is exploratory as the bacteria "swims" with its flagella through large distances in the nutrient medium. The chemotactic movement is guided by the tendency of bacteria to move towards a high-nutrient gradient (lower values of $J(\theta)$ for a minimization procedure) or away from a noxious gradient (high values of $J(\theta)$). Let the tumbling process be represented by a unit length random direction of movement denoted by $\varphi(j)$ which is kept fixed throughout the procedure. Let $\theta^i(j,k,l)$ represent the distance of the i th bacterium, after the j th chemotactic step, kth reproductive step and l th elimination dispersal step. Let $C(i)$ be the size of the step taken. Then the mathematical expression of the new position occupied by the bacterium after each chemotactic step is:

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\varphi(j) \tag{12}$$

3.2 Swarming

Some species of bacteria including E. coli exhibit a type of intricate pattern (known as swarms) in the presence of nutrient medium. In this particular phase, bacteria can be imagined to either release cell-attractants to attract other cells or release cell-repellents to repel the same. This process of attraction and repulsion can be expressed as –

$$J_{cc}\left(\theta, P(j,k,l)\right) = \sum_{i=1}^{S} J_{cc}^{i}\left(\theta, \theta^i(j,k,l)\right)$$

$$= \sum_{i=1}^{S}\left[-d_{attract}\exp\left(-w_{attract}\sum_{m=1}^{p}\left(\theta_m - \theta_m^i\right)^2\right)\right] + \sum_{i=1}^{S}\left[-h_{repellent}\exp\left(-w_{repellent}\sum_{m=1}^{p}\left(\theta_m - \theta_m^i\right)^2\right)\right]$$

$$\tag{13}$$

Where, $J_{cc}(\theta, P(j,k,l))$ denotes the additional cost function, S is the total number of bacteria, p is the search space's dimension. $d_{attract}$, $w_{attract}$, $h_{repellent}$ and $w_{repellent}$ are the appropriate coefficients whose values must be chosen accordingly. Computationally repulsion and attraction steps are utilized in order to prevent overcrowding of bacteria on local optima. In this paper, the additional cost function has not been used to simplify computational procedures.

3.3 Reproduction

During this phase a fraction of the initial bacterial population S die due to poor health. The remaining healthy bacteria, having lower values of objective function survive and give rise to the next generation of bacteria by splitting up into two. Thus, the overall population of bacteria remains intact throughout the optimization procedure.

3.4 Elimination and dispersal

This step occurs after the reproduction step. In order to simulate real-world phenomena such as wind-dispersal, new bacteria are randomly placed on the search space and a fraction of the older bacterial population is eliminated to ensure swift attainment of the global optimum position. The total population S however, is always maintained constant.

A detailed BFO algorithm has been presented in Algorithm 1.

Algorithm 1: The complete Bacterial Foraging Algorithm [49] applied to a minimization problem

---

BEGIN

Initialize the parameters, $C(i)$, $i = 1,2,…,S$. Also initialize all the counter values to zero.

  REPEAT:

  FOR l = 1 to N_{ed}

    FOR k = 1 to N_{re}

      FOR j = 1 to N_c

        FOR i = 1 to S

          Compute $J(i, j, k, l)$

            Then let $J(i, j, k, l) = J(i, j, k, l) + J_{cc}(\theta^j(j,k,l), P(j,k,l))$

            $J_{last} = J(i, j, k, l)$

            Tumble: Generate a random vector $\Delta(i) \in \Re^p$

            Move: $\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\dfrac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$

            Compute $J(i, j+1, k, l)$

            Then let $J(i, j+1, k, l) = J(i, j+1, k, l) + J_{cc}(\theta^i(j+1,k,l), P(j+1,k,l))$

            $m = 0$

            WHILE $m < N_S$

              $m = m + 1$

              IF $J(i, j+1, k, l) < J_{last}$

                $J_{last} = J(i, j+1, k, l)$

                Move: $\theta^i(j+1,k,l) = \theta^i(j+1,k,l) + C(i)\dfrac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$

Use this $\theta^i(j+1,k,l)$ to compute new $J(i,j+1,k,l)$, with

    cell-to-cell attraction effect

    ELSE

$$m = N_S$$

    ENDIF

    ENDWHILE

ENDFOR

FOR $i = 1$ to S

    Compute $J_{health}^i = \sum_{j=1}^{N_c+1} J(i,j,k,l)$

ENDFOR

Sort bacteria in order of cost values of J_{health}

Destroy S_r bacteria with the highest values of J_{health}

(i.e. least healthy bacteria)

Split each of the S_r bacteria with the lowest values of J_{health} into two and

each such pair resides in the same original location of the parent

ENDFOR

FOR $i = 1$ to S

    Eliminate and disperse each bacterium with probability p_{ed}, keeping

    population of bacteria constant

ENDFOR

ENDFOR

UNTIL termination criterion is satisfied

END

4 Performance evaluation

In this section, the proposed BFO-based optimal spectral weighting scheme applied in conjunction with the SVM bi-classification system is implemented on three standard hyperspectral datasets. The objective of this section is to substantiate the superiority of the proposed approach over other contemporary competing classification approaches as well as to verify the efficacy of the BFO optimization technique over its contemporary competitors, the ABC, PSO and the GA.

4.1 Hyperspectral datasets

Three benchmark hyperspectral datasets were downloaded from [51, 52], and the number of major categories of data were noted. 7 major classes were selected from the Indian Pines and Salinas Valley datasets and 6 major classes were selected from the Pavia University data for One-Against-One (OAO) classification and the rest of the classes were discarded. For One-Against-All (OAA) classification, all classes of data were collected and reserved for experimentation with limited training samples. This subsection discusses briefly each of the three chosen datasets and describes the major classes which were chosen for further experimentation of the proposed approach.

4.1.1 AVIRIS Indian pines, Indiana dataset

The Indian Pines test site in North-western Indiana, USA serves as a popular benchmark hyperspectral dataset collected by the AVIRIS sensor in the early 1990's. Each image comprises of 145×145 samples of an agricultural area. It uses 224 spectral reflectance bands within a wavelength range of 0.4 to 2.5 µm, with a nominal spectral resolution of 10nm, a 16 bit radiometric resolution and a 20m spatial resolution [51]. The number of spectral bands was further reduced to 220 because four spectral bands contain no data. Some structures such as rail lines and highways were ignored as they are not properly

discernible [26]. The ground truth available is designated into 16 classes, namely Alfalfa, Corn-notill, Corn-min, Corn, Grass-pasture, Grass-trees, Grass-pasture-mowed, Hay-windrowed, Oats, Soybean-notill, Soybean-mintill, Soybean-clean, Wheat, Woods, Buildings-Grass-Trees-Drives and Stone-Steel-Towers. The major classes have been listed in Table 1 with their corresponding number of samples. A single band image has been shown in Figure 2(a) with its corresponding reference map in Figure 2(b).
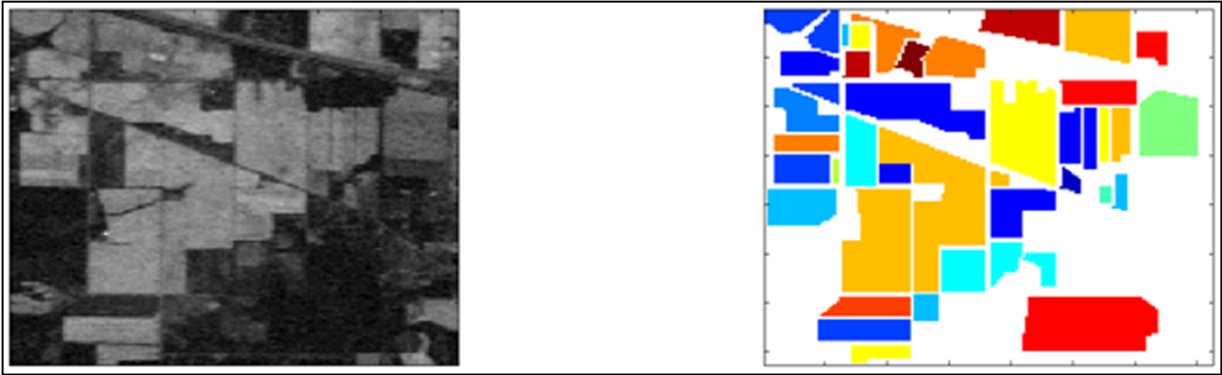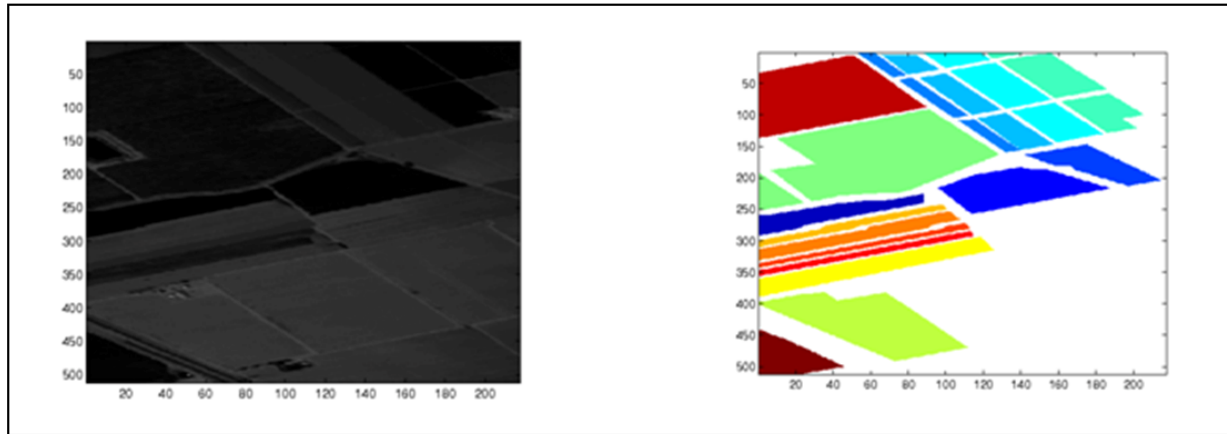

Figure 2. (a) AVIRIS Sample Image of Indian Pines Dataset for a Single Spectral Band (b) Corresponding Color-coded Ground Truth Data

Table 1. Description of the 7 Major Classes of Indian Pine Dataset [51]

Class	Class Description	Number of Pixels
A	Corn-notill	1434
B	Corn-min	834
C	Grass/Trees	747
D	Soybeans-notill	968
E	Soybeans-min	2468
F	Soybeans-clean	614
G	Woods	1294

4.1.2 AVIRIS salinas valley, California dataset

The hyperspectral data contained in this dataset was acquired by the 224-band AVIRIS sensor over Salinas Valley in Southern California, USA at low altitudes resulting in an improved pixel resolution of 3.7 meter per pixel. Each image is made up of 512 lines of 217 samples. 20 spectral bands were removed due to water absorption and noise, resulting in a corrected image containing 204 spectral bands over the range of 0.4 to 2.5µm. A sample band and the corresponding ground truth data has been shown in Figure 3(a) and 3(b) respectively. The Salinas scene consists of the 16 ground truth classes, namely: Broccoli-green-weeds-1, Broccoli-green-weeds-2, Fallow, Fallow-rough-plow, Fallow-smooth, Stubble, Celery, Grapes-untrained, Soil-vinyard-develop, Corn-senesced-green-weeds, Lettuce-romaine-4wk, Lettuce-romaine-5wk, Lettuce-romaine-6wk, Lettuce-romaine-7wk, Vineyard-untrained and Vineyard-vertical-trellis. Due to the similarity in the spectral signatures among these classes, discrimination between these classes proves to be a difficult task. It is our aim to prove the effectiveness of the proposed scheme when faced with a challenging classification problem which is the major reason behind the selection of the Salinas dataset as a benchmark for experimentation. Table 2 provides a detailed list of the major classes of the Salinas Valley dataset.

Figure 3. (a) AVIRIS Sample Image of Salinas Valley Dataset for a Single Spectral Band (b) Corresponding Colour-coded Ground Truth Data

Table 2. Description of the 7 Major Classes of Salinas Valley Dataset [52]

Class	Class Description	Number of Pixels
A	Broccoli-green-weeds-2	3726
B	Stubble	3959
C	Celery	3579
D	Grapes-untrained	11271
E	Soil-vinyard-develop	6203
F	Corn-senesced-green-weeds	3278
G	Vineyard-untrained	7268

4.1.3 ROSIS pavia university dataset

The ROSIS sensor collected this data during a flight campaign over the Pavia district in north Italy. 103 spectral bands were used for data acquisition in this dataset comprising of 610×610 pixel images with a geometric resolution of 1.3m. A few of the samples in these images contain no information and were discarded before classification. A sample image has been portrayed in Figure 4 (a), with the corresponding reference map in Figure 4 (b). The ground truth data shows a total of 9 distinct classes as listed in Table 3 with their respective number of pixels.

Figure 4. (a) ROSIS Sample Image of Pavia University Dataset for a Single Spectral Band (b) Corresponding Colour-coded Ground Truth Data

Table 3. Description of the 9 Major Classes of Pavia University Dataset [52]

Class	Class Description	Number of Pixels
A	Broccoli-green-weeds-2	3726
B	Stubble	3959
C	Celery	3579
D	Grapes-untrained	11271
E	Soil-vinyard-develop	6203
F	Corn-senesced-green-weeds	3278
G	Vineyard-untrained	7268

4.2 Methodology

The major classes, as described in Tables 1, 2 and 3 are collected for OAO classification after comparing with their corresponding reference maps. Pixels collected from each pair of classes are then divided into a training set consisting of 20% of the total number of pixels, and a testing set comprising of the other 80%. A population of initial random spectral weight vectors is initialized in the search space of $s \in (0,1]$ in d space for d spectral bands. Next, the BFO algorithm is implemented in conjunction with an SVM-classifier with a Gaussian RBF kernel using a standard deviation $\sigma = 0.4$, and a regularization parameter $C = 60$, values which are obtained from [26]. The objective in (11) is minimized by the foraging bacteria and the fittest particle is indicative of the optimal combination of spectral weights. For each pair of classes, the optimal weights obtained by the BFO are used to customize the RBF kernel as shown in equation (10). Testing pixels are then applied to the modified SVM to compute the classification results. In order to reduce the effects of randomness, the results are collected for 10 iterations and the mean and standard deviations of the classification errors are tabulated. This scheme is repeated for all three datasets and has proved to yield favourable results.

In order to further evaluate the performance of the proposed method, OAA classification is performed with very limited training samples (2%) and the mean classification accuracies over 10 iterations with randomly selected training samples are noted. The results obtained are presented in the following subsections. All support vector machine based computations were carried out using the LibSVM toolbox [19] in MATLAB 7.7.0.

Algorithm 2 shows how the BFO algorithm is utilized in conjunction with spectrally weighted SVM to develop the proposed hyperspectral image classification method.

Algorithm 2: A complete algorithm of the proposed BFO-based spectrally weighted SVM-kernel scheme for Hyperspectral Image Classification

---

FOR kk = 1: $^{N}C_2$

Collect all pixels of the kk th pair of classes

Distribute these pixels into training sets (20%) and testing sets (80%)

BEGIN BFO

Initialize the parameters, $C(i)$, $i = 1,2,\ldots,S$. Also initialize all the counter values to zero.

  REPEAT:

   FOR $l = 1$ to N_{ed}

    FOR $k = 1$ to N_{re}

      FOR $j = 1$ to N_c

       FOR $i = 1$ to S

         Compute $J(i,j,k,l)$ using equation (11)

            $J_{last} = J(i,j,k,l)$

Tumble: Generate a random vector $\Delta(i) \in \Box^d$

Move: $\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\dfrac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$

Compute $J(i, j+1, k, l)$

$m = 0$

WHILE $m < N_S$

$\quad m = m + 1$

$\quad$ **IF** $J(i, j+1, k, l) < J_{last}$

$\qquad J_{last} = J(i, j+1, k, l)$

$\qquad$ Move: $\theta^i(j+1,k,l) = \theta^i(j+1,k,l) + C(i)\dfrac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$

$\quad$ **ELSE**

$\qquad m = N_S$

$\quad$ **ENDIF**

$\quad$ **ENDWHILE**

ENDFOR

FOR $i = 1$ to S

$\qquad$ Compute $J_{health}^i = \sum\limits_{j=1}^{N_c+1} J(i, j, k, l)$

ENDFOR

Sort bacteria in order of cost values of J_{health}

Destroy S_r bacteria with the highest values of J_{health}

(i.e. least healthy bacteria)

Split each of the S_r bacteria with the lowest values of J_{health} into two and

each such pair resides in the same original location of the parent

ENDFOR

FOR $i = 1$ to S

$\quad$ Eliminate and disperse each bacterium with probability p_{ed}, keeping

$\quad$ population of bacteria constant

ENDFOR

ENDFOR

UNTIL termination criterion is satisfied

END

Optimal spectral weights = $\theta_{best}(n) = s^*$

Customize SVM kernel by pre-multiplying initial dataset with $diag(s^*)$

FOR k = 1:10

Perform training and testing with randomly selected samples

$\qquad$ Compute classification error / accuracy

ENDFOR

Compute mean and standard deviation of classification error / accuracy

ENDFOR

4.3 Results

Spectral weights are considered to be in the range $s \in (0,1]$ in R^d, defining a unit d-dimensional hypercube as the optimization search space in accordance with the values reported in [26]. The total number of iterations for the stochastic iterative techniques is considered to be 400 for all experiments.
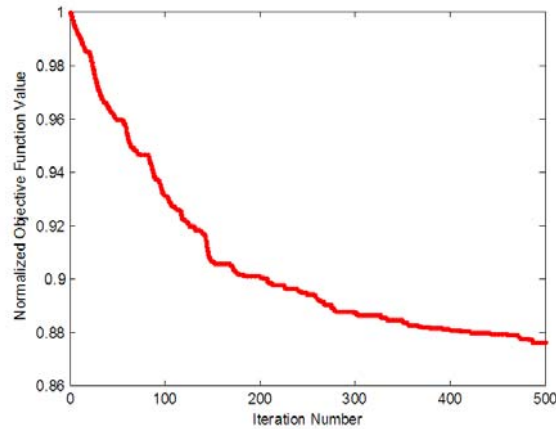
Figure 5. Effect of Optimal Values with Iteration Number (BFO)

Table 4. Performance Evaluation of Proposed Approach on AVIRIS Indian Pines Dataset in comparison with previous results [26]. All values in this table are error mean/error S.D. values.

	SVM (without weighting)[26]	Weights based on Gradient Descent[26]	Weights based on Mutual Information[26]	Weights based on GA-optimization	Weights based on ABC-optimization	Weights based on PSO-optimization	Weights based on BFO-optimization
Class-Pair	Mean/SD (%)	Mean/SD (%)	Mean/SD (%)	Mean/SD (%)	Mean/SD (%)	Mean/SD (%)	Mean/SD (%)
A|B	14.95/1.69	11.21/0.70	8.22/1.13	6.36/0.76	6.15/0.79	5.64/0.97	5.17/0.51
A|C	0.42/0.05	0.57/0.19	0.55/0.07	0.31/0.12	0.32/0.13	0.44/0.14	0.30/0.21
A|D	15.56/0.70	12.32/1.03	13.05/0.61	7.15/0.89	6.95/0.57	6.47/0.70	6.91/0.89
A|E	19.45/0.62	11.46/0.79	15.54/0.51	6.89/0.31	6.34/0.37	6.51/0.51	6.30/0.72
A|F	17.83/0.54	11.99/1.88	10.98/0.71	2.71/0.35	2.46/0.72	3.21/0.75	2.36/0.34
A|G	0.10/0.04	0.10/0.05	0.09/0.05	0.15/0.06	0.14/0.05	0.18/0.10	0.16/0.08
B|C	0.30/0.13	0.22/0.06	0.19/0.04	0.24/0.17	0.34/0.18	0.28/0.31	0.21/0.18
B|D	5.02/0.40	4.68/0.50	3.21/0.28	2.81/0.55	2.83/0.36	2.85/0.32	2.73/0.77
B|E	12.06/0.52	11.43/1.88	12.35/1.05	5.29/0.57	5.01/0.44	4.97/0.35	4.87/0.50
B|F	19.15/1.58	11.44/2.05	14.43/1.45	4.11/0.34	4.76/0.83	4.74/0.91	4.21/0.49
B|G	0.04/0.03	0.06/0.00	0.01/0.03	0.01/0.04	0.01/0.02	0.02/0.01	0.01/0.04
C|D	0.67/0.23	0.58/0.14	0.55/0.13	0.65/0.24	0.63/0.14	0.63/0.20	0.61/0.14
C|E	1.19/0.22	1.07/0.39	1.05/0.20	0.32/0.28	0.31/0.21	0.28/0.16	0.19/0.19
C|F	0.48/0.20	0.50/0.05	0.53/0.12	0.60/0.26	0.37/0.16	0.48/0.46	0.45/0.37
C|G	1.25/0.20	1.41/0.33	1.32/0.15	0.65/0.32	0.46/0.22	0.55/0.22	0.44/0.19
D|E	14.43/0.49	12.87/0.68	11.81/1.17	7.27/0.59	7.26/0.49	7.42/0.73	7.42/0.63
D|F	10.72/0.21	8.87/1.16	9.11/0.63	3.06/0.60	3.64/0.56	3.11/0.60	2.78/0.86
D|G	0.03/0.03	0.00/0.00	0.00/0.00	0.03/0.04	0.02/0.05	0.06/0.05	0.00/0.00
E|F	9.32/0.41	7.89/1.24	10.95/1.19	2.89/0.47	2.72/0.28	2.76/0.56	2.70/0.33
E|G	0.25/0.07	0.39/0.11	0.23/0.10	0.21/0.07	0.21/0.09	0.20/0.07	0.13/0.07
F|G	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.01/0.02	0.00/0.00	0.00/0.00
Mean Error (%)	6.82	5.33	5.44	2.47	2.34	2.38	2.30

Graphically, the variation of J as in equation (11) has been shown with the number of iterations for the BFO program. It is clear from Figure 5 that a 12% decrease in the normalized margin values occurs till the 400th iteration, after which the minimization is 1% in 100 iterations. Thus, 400 is selected after several trials as the optimal iteration number required to perform satisfactorily without adversely increasing the computation time.

Results using un-weighted SVM, gradient descent-based weighted SVM, mutual information-based SVM and stochastically derived spectral weighted SVM are shown in Table 4 when employed on the AVIRIS Indian Pines dataset. It can be observed that the proposed approach outperforms both the previous results as well as other stochastic optimization techniques employed in recent times. It is known that the classes A, B, D, E and F are very similar thereby making classification amongst these classes quite difficult as is indicated by errors greater than 10% in [26]. However, in the proposed approach, the corresponding errors (although larger in comparison to the errors produced by the other pairs of classes) have been lowered to almost half of the errors computed using MI or Gradient Descent-based approaches. Furthermore, spectral weighting shows the effectiveness on class pairs like A|E and B|F whose errors have been reduced from almost 20% with un-weighted kernels to 6% and 4% with the proposed approach. This can be explained by the integration of exploration and exploitation that highlights the BFO algorithm's random search in comparison to the greedier gradient descent which only exploits its current position instead of exploring a larger region and is constrained to differentiable objective functions only thereby increasing the probability of getting stuck in a local minimum [31]. The standard deviations over 10 iterations is used as an index for the scheme's robustness, and the lower values of S.D. as compared to the previous methodologies' results indicate that the proposed scheme is relatively immune to random sampling.

The proposed approach is also implemented for the OAA classification problem. The results obtained in 10 iterations with 2% randomly selected samples are shown in Table 5. The overall mean classification accuracies have been compared and it is observed that the highest value obtained is 95.86% by the BFO-based paradigm.

Table 5. OAA Classification Mean Accuracies for Indian Pines Data

Class	ABC-based Spectral Weighting Mean/S.D. (%)	GA-based Spectral Weighting Mean/S.D. (%)	PSO-based Spectral Weighting Mean/S.D. (%)	BFO-based Spectral Weighting Mean/S.D. (%)
A	97.24/0.19	97.54/0.35	97.88/0.03	97.40/0.63
B	89.70/1.30	89.81/0.97	86.15/0.04	90.24/0.80
C	93.34/0.78	93.49/0.74	91.96/0.03	93.68/0.82
D	97.74/0.33	97.81/0.24	97.75/0.02	97.81/0.23
E	98.28/0.41	98.10/0.50	95.21/0.02	98.39/0.23
F	97.75/0.46	98.11/0.49	92.79/0.04	98.12/0.43
G	97.46/0.53	97.56/0.40	97.68/0.02	97.71/0.22
H	99.06/0.82	99.24/0.29	95.27/0.02	99.28/0.38
I	97.66/0.34	97.52/0.40	97.87/0.01	97.47/0.44
J	91.95/0.97	91.85/0.87	90.67/0.04	92.11/0.94
K	84.92/1.06	85.15/1.03	76.22/0.07	86.83/0.64
L	94.80/0.64	94.81/0.53	94.08/0.05	94.59/0.69
M	99.41/0.20	99.24/0.47	97.88/0.02	99.37/0.17
N	97.93/0.76	97.86/0.76	87.51/0.04	97.95/1.01
O	96.46/0.51	96.58/0.45	96.33/0.02	96.65/0.27
P	98.62/0.10	98.36/0.42	97.88/0.03	98.29/0.54
Mean Overall Accuracy (%)	95.77	95.83	93.33	95.86

From Tables 4 and 5, it is apparent that the proposed BFO-based strategy is generally more effective for both OAA and OAO classification paradigms. Further comparison is made among the overall classification accuracies the stochastic

optimization algorithms with previously reported results [46, 54] on the Indian Pines Dataset. The lowest accuracies are found in the Euclidean and default LibSVM approaches. Window based classification formalisms introduced in [46] prove to be effective, but are outperformed conclusively by the kernel methods optimized by stochastic algorithms, consolidating the efficacy of the spectral weighting scheme using random search algorithms.

Table 6. Comparison of Overall Accuracies of OAA Classification on Indian Pines Dataset

Classifiers Used	Window Type[46]	Overall Classification Accuracy (%)
LibSVM Default [54]		52.79
ENVI Default [54]		82.45
Euclidean		48.23
bLOOC+DAFE+ECHO		82.91
K_ω		87.30
K_ω [46]		88.55
Spatial/Spectral Classifiers		
Mean-based	Spatial	84.55
	Stacked	94.21
	Summation	92.61
	Weighted	95.97
	Cross Terms	94.80
	Summation+Stacked	95.20
	Cross Terms + Stacked	95.10
Mean + Standard Deviation-based	Spatial	88.00
	Stacked	94.21
	Summation	95.45
	Weighted	96.53
	Summation+Stacked	96.20
Spectral Weighting-based using Stochastic Optimization	GA-based	96.04
	PSO-based	
Population Size 40, Training Pixels 5%	(without feature extraction)	96.46
	(with feature extraction)[54]	95.25
	ABC-based	96.01
	BFO-based	96.88

Next, the population size of each of the stochastic optimization techniques are varied in order to gauge the effect of classification accuracy with increasing population members. It can be seen from Figure 6 that while each of the population-based algorithms show a definite increase in accuracy with increments made in the population size, the maximum accuracy resulting from these increases in bacteria takes place in BFO from 95.97% to 96.88%.

However, due to the exorbitant increase in computational time per classification with variation in population size as observed from Table 7, it is preferred to use a population size of 10 to boost computational speeds.

It is also expected for a classification scheme to perform better with the increase of the amount of training samples. Experimentation was performed on the proposed method with gradually increased training samples and the results were plotted graphically as shown in Figure 7. It is observed that the increase of training samples greatly leverages the performance of the BFO-based proposal. It is also noteworthy that the GA and PSO-based algorithms also follow similar accuracy trajectories under identical simulation conditions, but attain maximum accuracies 0.4% less than the proposed approach.
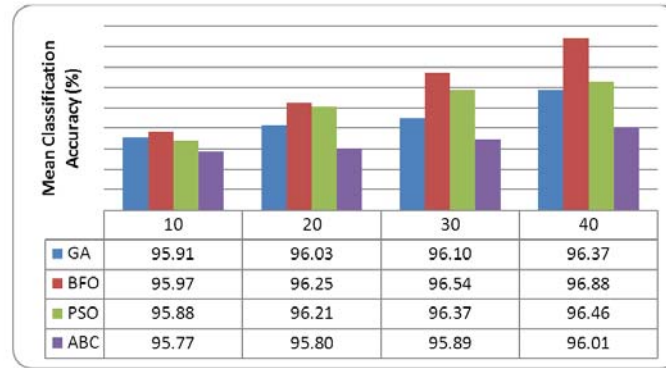
Figure 6. Mean Classification Accuracy of OAA Scheme with Variation of Population Size of all competing algorithms

Table 7. Computation Times for 16 classes OAA Classification with Increase in Population Size

| Population Size	10	20	30	40
GA-based Scheme	418.50	925.71	1248.44	1527.57
BFO-based Scheme	214.28	414.36	687.99	941.29
ABC-based Scheme	201.22	407.41	600.14	838.97
PSO-based Scheme	183.20	257.92	377.44	589.76
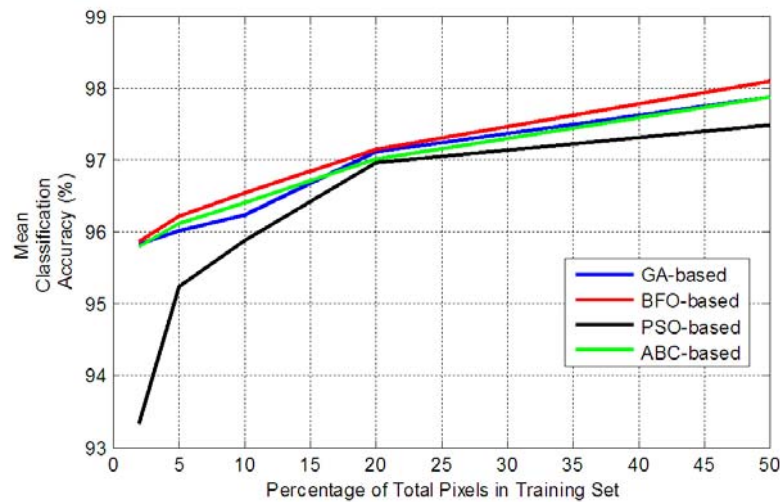
Figure 7. Variation of Mean Classification Accuracy in OAA Scheme with increase in training samples

In order to compare the spectral weight vector obtained by the proposed approach to those reported in [26], an optimal spectral weight vector was plotted.

A closer inspection of the spectral weight plots in [26] confirms the importance of the certain ranges of spectral bands such as from 1 to 20, from 100 to 150 and from 160 to 210. The BFO-optimized spectral weights bear a strong influence to the Bhattacharya distance plot in [26] also, indicating that this technique correctly identified the spectral bands which contained the bulk of essential information. Thus, the Figure 8 consolidates the proposal of approaching the hyperspectral image classification problem as an optimization procedure for the attainment of optimal spectral weights thereby enhancing the contribution of those bands of data which contain essential information.
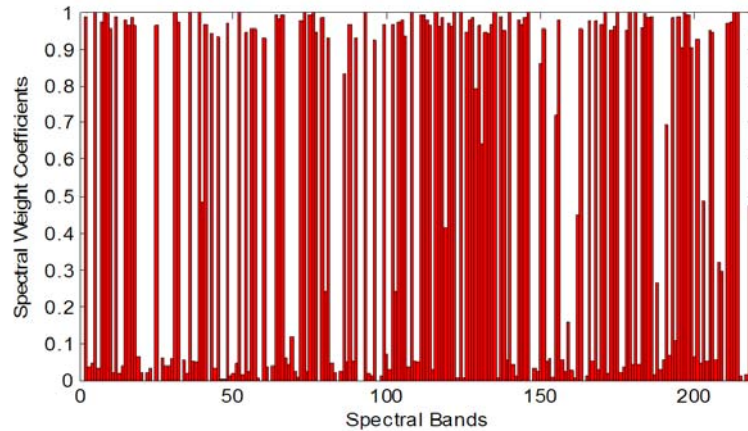
Figure 8. Spectral Weight Vector of Indian Pines Dataset for bi-class A|B

In order to further test the proposed method, two more datasets are selected. The Salinas Valley dataset offers a higher spatial resolution image, and the inherent similarity of spectral features makes it a very difficult benchmark set for image classification purposes. Additional experimentation is also executed on the Pavia University Dataset, which provides an insight into the performance of the proposed approach in the urban land cover context.

Table 8. OAO Classification Results on Salinas Valley Data. All values in this table are error mean/error S.D. values

Class Pair	GA-based Spectral Weighting Classification Error	ABC-based Spectral Weighting Classification Error	PSO-based Spectral Weighting Classification Error	BFO-based Spectral Weighting Classification Error
	Mean/S.D. (%)	Mean/S.D. (%)	Mean/S.D (%)	Mean/S.D. (%)
A|B	0.03/0.06	0.05/0.07	0.02/0.02	0.01/0.01
A|C	0.05/0.05	0.03/0.03	0.05/0.05	0.05/0.05
A|D	0.01/0.01	0.02/0.02	0.01/0.01	0.00/0.04
A|E	0.00/0.01	0.00/0.00	0.00/0.00	0.00/0.00
A|F	0.05/0.03	0.04/0.04	0.04/0.05	0.04/0.03
A|G	0.05/0.03	0.03/0.05	0.04/0.01	0.03/0.05
B|C	0.05/0.03	0.06/0.04	0.04/0.01	0.03/0.03
B|D	0.04/0.02	0.02/0.01	0.01/0.02	0.02/0.03
B|E	0.03/0.03	0.01/0.01	0.06/0.01	0.02/0.03
B|F	0.09/0.06	0.10/0.06	0.09/0.03	0.04/0.07
B|G	0.03/0.02	0.02/0.01	0.03/0.02	0.02/0.01
C|D	0.02/0.01	0.02/0.01	0.02/0.02	0.02/0.01
C|E	0.02/0.01	0.02/0.01	0.04/0.08	0.01/0.01
C|F	0.14/0.11	0.08/0.04	0.08/0.05	0.08/0.04
C|G	0.03/0.04	0.05/0.06	0.02/0.02	0.02/0.10
D|E	0.02/0.01	0.02/0.02	0.02/0.02	0.02/0.02
D|F	0.32/0.07	0.34/0.08	0.32/0.06	0.30/0.03
D|G	16.98/0.13	16.35/0.29	16.72/0.26	16.29/0.19
E|F	0.48/0.07	0.43/0.09	0.48/0.13	0.42/0.07
E|G	0.05/0.01	0.02/0.01	0.04/0.04	0.02/0.01
F|G	0.30/0.09	0.32/0.11	0.20/0.10	0.20/0.07
Mean Overall Error %	0.89	0.86	0.87	0.84

Tables 8 and 9 present the results obtained by the implementation of the proposed approach on the Salinas Valley dataset, and Tables 10 and 11 offer details of the performance of this scheme on urban classification problems. Reported results found in the literature [3] have also been provided for comparison with the devised swarm-based technology.

Table 9. OAA Classification Results on Salinas Valley Data compared to previous research methods [3]

Morphological Feature Extraction[3]	OA (%)
Original	81.25
Reduced (MNF)	87.94
Multi-channel (D-ordering)	91.44
Multi-channel (L-ordering)	80.67
Multi-channel (R-ordering)	89.57
Multi-channel reduced (MNF)	90.22
Mono-channel (MNF)	80.45
Spectral Weighting Scheme	
GA-Based	98.75
PSO-Based	93.75
ABC-Based	98.80
BFO-Based	**98.83**

In Table 8 and 10, best classification results have been represented in boldface. It is easily concluded from these tables that the proposed approach can be applied seamlessly to both agricultural and urban datasets and produce results which are comparable even to the some advanced methods used at present [3, 26, 46]. From Table 9, it is also recommended that the optimal spectral weights be estimated using a global search procedure like the algorithms tested in this paper, as the accuracy of these methods exceeds those of its predecessors by a significant margin.

Table 10. OAO Classification Results on Pavia University Data. All values in this table are error mean/error S.D

Class Pair	ABC-based Spectral Weighting Classification Error (%)	GA-based Spectral Weighting Classification Error (%)	PSO-based Spectral Weighting Classification Error (%)	BFO-based Spectral Weighting Classification Error (%)
	Mean/S.D.	Mean/S.D.	Mean/S.D	Mean/S.D.
A|B	0.11/0.04	0.11/0.05	0.09/0.03	0.08/0.05
A|C	1.46/0.16	1.42/0.11	1.40/0.19	1.40/0.18
A|D	0.09/0.05	0.14/0.03	0.13/0.04	0.09/0.04
A|E	0.32/0.12	0.42/0.10	0.35/0.07	0.33/0.08
A|F	2.45/0.18	2.41/0.19	2.45/0.08	2.53/0.15
B|C	0.10/0.02	0.10/0.02	0.08/0.02	0.08/0.02
B|D	0.89/0.08	0.95/0.07	0.89/0.08	0.88/0.08
B|E	2.87/0.06	2.79/0.06	2.66/0.11	2.51/0.09
B|F	0.13/0.04	0.15/0.03	0.13/0.04	0.13/0.04
C|D	0.08/0.02	0.07/0.03	0.08/0.02	0.07/0.03
C|E	0.30/0.13	0.39/0.08	0.30/0.07	0.30/0.07
C|F	10.17/0.42	11.44/0.47	10.93/0.42	9.90/0.50
D|E	0.39/0.06	0.49/0.12	0.42/0.04	0.39/0.17
D|F	0.04/0.01	0.08/0.04	0.05/0.04	0.05/0.03
E|F	0.73/0.09	0.72/0.08	0.77/0.09	0.70/0.11
Mean Overall Error %	1.88	1.45	1.39	1.31

Table 11. OAA Classification Results on Pavia University Data compared to previous research [46]

Scheme used	OA (%)
Spectral Information[46]	87.17
Morphological Information [46]	91.87
Spectral Weighting Scheme	
GA-Based	97.71
ABC-Based	97.66
PSO-Based	97.61
BFO-Based	97.87

5 Conclusion

The present paper proposes an improved hyperspectral image classification methodology utilizing a modern popular stochastic optimization technique called the Bacterial Foraging Optimization (BFO). The BFO algorithm is employed to maximize the margin of a support vector machine bi-classifier, where the decision variables correspond to the spectral weight vector integrated with the SVM kernel. The quantitative presentation of performance indices and the graphical presentations demonstrate that the BFO based image classification scheme could significantly out-perform the competing classifiers, for varied benchmark datasets, thereby providing a superior alternative to hyperspectral image classification for future applications. Finally, the proposed approach has been shown to conclusively outperform other commonly used stochastic optimization techniques such as Genetic Algorithms, Bee Colonies and Particle Swarms used for similar spectral weight optimization applications [54].

Acknowledgements

REFERENCES

[1]   Goetz AFH, Vane G, Solomon JE, Rock BN. Imaging spectrometry for Earth remote sensing. Science. 1985; 228 (4704): 1147-1153. PMid:17735325 http://dx.doi.org/10.1126/science.228.4704.1147

[2]   Landgrebe D. Signal theory methods in multispectral remote sensing. John Wiley: Hoboken, NJ. 2003. http://dx.doi.org/10.1002/0471723800

[3]   Plaza J, Plaza AJ and Barra C. Multi-Channel Morphological Profiles for Classification of Hyperspectral Images Using Support Vector Machines. Sensors. 2009; 9: 196-218. PMid:22389595 http://dx.doi.org/10.3390/s90100196

[4]   Salem F and Kafatos M. Hyperspectral Image Analysis for Oil Spill Mitigation. 22nd Asian Conference on Remote Sensing, National University of Singapore. 2001.

[5]   Manolakis D, Marden D, Shaw GA. Hyperspectral Image Processing for Automatic Target Detection Applications. Lincoln Laboratory Journal. 2003; 4 (1).

[6]   Aspinall RJ, Marcus WJ, and Boardman AW. Considerations in collecting, processing, and analyzing high spatial resolution hyperspectral data for environmental investigations. Journal of Geographical Systems. 2002; 4: 15-29. http://dx.doi.org/10.1007/s101090100071

[7]   Hughes G. On the mean accuracy of statistical pattern recognizers. IEEE Trans. Inf. Theory. 1968; 14 (1): 55-63. http://dx.doi.org/10.1109/TIT.1968.1054102

[8]   Ye J, Xiong T, Janardan R, Bi J, Cherkassky V, and C. Kambhamettu C. Efficient model selection for regularized linear discriminant analysis. In CIKM'06, Arlington, Virginia, USA. 2006; 532-539.

[9]   Lillesand TM, Kiefer RW, and Chipman JW. Remote Sensing and Image Interpretation. 5th ed. New York: John Wiley & Sons. 2004.

[10] Chi M and Bruzzone L. Semi-supervised classification of hyperspectral images by SVMs optimized in the primal. IEEE Transactions on Geoscience and Remote Sensing. 2007; 45 (6): 1870-1880. http://dx.doi.org/10.1109/TGRS.2007.894550

[11] Bandos TV, Bruzzone L and Camp-Valls G. Classification of Hyperspectral Images with Regularized Linear Discriminant Analysis. IEEE Transactions on Geoscience and Remote Sensing. 2009; 47 (3): 862-873. http://dx.doi.org/10.1109/TGRS.2008.2005729

[12] Li J, Bioucas-Dias JM, and Plaza A. Semi-Supervised Hyperspectral Image Segmentation Using Multinomial Logistic Regression with Active Learning. IEEE Transactions on Geoscience and Remote Sensing. 2010; 48 (11): 4085-4098.

[13] Paula JD, Schowangerdt RA. A detailed comparison of back propagation NN & Max likelihood classifiers for urban land use classification. IEEE Transactions on Geoscience & Remote Sensing. 1995; 33 (4): 981-996. http://dx.doi.org/10.1109/36.406684

[14] Merényi E, Taranik JV, Minor TB, and Farrand WH. Quantitative Comparison of Neural Network and Conventional Classifiers For Hyperspectral Imagery. Summaries of the Sixth Annual JPL Airborne Earth Science Workshop, Pasadena, CA. 1996 March; 4-8.

[15] Abuelgasim A and Gopal S. Classification of multiangle and multispectral ASAS data using a hybrid neural network model. In Proc. Int'l Geosci. and Remote Sensing Symposium, Volume III, Caltech, Pasadena. 1994 August; 8(12): 1670-1672.

[16] Aitkenhead MJ and Dyer R. Improving land-cover classification using recognition threshold neural networks. Photogrammetric Engineering & Remote Sensing. 2007; 73(4): 413-421,

[17] Srivastava AN, and Stroeve J. Onboard Detection of Snow, Ice, Clouds and Other Geophysical Processes Using Kernel Methods. Proc. of the ICML Workshop on Machine Learning technologies for autonomous space applications. 2003.

[18] Gualtieri JA, Chettri SR, Cromp RF, and Johnson LF. Support vector machine classifiers as applied to AVIRIS data. in Proc. of the 8th JPL Airborne Geoscience Workshop. 1999.

[19] Chang CC and Lin CJ. LibSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology [Internet]. 2011; 2(27): 1-27. Available from: http://www.csie.ntu.edu.tw/~cjlin/libsvm

[20] Avrithis YS, and Kollias SD. Fuzzy Image Classification Using Multiresolution Neural Networks with Applications to Remote Sensing, Digital Signal Processing Proceedings, 1997. DSP 97. 1997 13th International Conference. 1997 (2-4 July): 261-264.

[21] Camps-Valls G, and Bruzzone L. Kernel-based methods for hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing,. 2005; 43 (6): 1351-62. http://dx.doi.org/10.1109/TGRS.2005.846154

[22] Brown M, Lewis HG, Gunn SR. Linear spectral mixture models and support vector machines for remote sensing. IEEE Transactions on Geoscience and Remote Sensing. 2000; 38(5): 2346-60. http://dx.doi.org/10.1109/36.868891

[23] Huang C, Davis LS, and Townshend JR. An assessment of support vector machines for land cover classification. Int. J. Remote Sensing. 2002; 23 (4): 725-749. http://dx.doi.org/10.1080/01431160110040323

[24] Shah CA, Watachaturaporn P, Varshney PK, and Arora MK. Some recent results on hyperspectral image classification. Advances in techniques for Analysis of Remotely Sensed Data in IEEE Workshop. 2003; 346-353

[25] Melgani F and Bruzzone L. Classification of hyperspectral remote sensing images with support vector machines. IEEE Transactions on Geoscience and Remote Sensing. 2004; 42(8): 1778-90. http://dx.doi.org/10.1109/TGRS.2004.831865

[26] Guo B, Gunn SR, Damper RI and Nelson JDB. Customizing Kernel Functions for SVM-Based Hyperspectral Image Classification. IEEE Transactions on Image Processing. 2008; 17 (4): 622- 629. PMid:18390369 http://dx.doi.org/10.1109/TIP.2008.918955

[27] Tang X, and Pearlman WA. Three-dimensional wavelet-based compression of hyperspectral images. Hyperspectral Data Compression, Kluwer Academic Publishers. 2008; 273-308.

[28] Jolliffe I. Principal Component Analysis., 2nd edition, Springer. 2002.

[29] Peng G, Ruiliang P, and Bin Y. Conifer species recognition: An exploratory analysis of in situ hyperspectral data. Remote Sensing of Environment. 1997; 62 (2): 189-200. http://dx.doi.org/10.1016/S0034-4257(97)00094-1

[30] Mika S, Ratsch G, Weston, Scholkopf JB, and Muller KR. Fisher's discriminant analysis with kernels. Neural Networks for Signal Processing IX, IEEE. 1999; 27: 41-48.

[31] Kennedy J and Eberhart R. Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks IV. 1995; 1942-1948.

[32] Banerjee S, Chakrabarty A, Maity S and Chatterjee A. Feedback linearizing indirect adaptive fuzzy control with foraging based on-line plant model estimation. Applied Soft Computing. 2011; 11(4): 3441-3450. http://dx.doi.org/10.1016/j.asoc.2011.01.016

[33] Kennedy J. The particle swarm: social adaptation of knowledge. Proceedings of IEEE International Conference on Evolutionary Computation. 1997; 303-308.

[34] Kennedy J, and Eberhart RC. Swarm Intelligence. Morgan Kaufmann. 2011.

[35] Bremermann H. Chemotaxis and optimization. J. Franklin Inst. 1997; 297: 397-404. http://dx.doi.org/10.1016/0016-0032(74)90041-6

[36] Wahba G. Support vector machines, reproducing kernel Hilbert spaces, and randomized GACV. Technical report, University of Wisconsin, Department of Statistics. 1998.

[37] Ray T and Liew K. Society and Civilization: An Optimization Algorithm Based on the Simulation of Social Behavior. IEEE Transactions on Evolutionary Computation. 2003; 7(4): 386-396. http://dx.doi.org/10.1109/TEVC.2003.814902

[38] Bovolo F, Bruzzone L, and Carlin L. A Novel Technique for Sub-pixel image classification based on support vector machine. Image Processing, IEEE Trans. 2010; 19(11): 2983-2999.

[39] Cover TM. Geometrical and statistical properties of systems of linear inequalities with application in pattern recognition. IEEE Trans. Electron. Comput. 1965; 14: 326-334. http://dx.doi.org/10.1109/PGEC.1965.264137

[40] Vapnik V. Statistical Learning Theory. John Wiley. 1998; 732.

[41] Cristianini N and Shawe-Taylor J. An Introduction to Support Vector Machines. Cambridge University Press. 2000.

[42] Mercer J. Functions of positive and negative type and their connection with the theory of integral equations. Philos. Trans. Roy. Soc., London. 1909.

[43] Schölkopf B and Smola A. Learning With Kernels—Support Vector Machines, Regularization, Optimization and Beyond. Cambridge, MA: MIT Press. 2001.

[44] Burges CJC. A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery. 1998; 2(2): 121-167. Kluwer Academic Publishers. http://dx.doi.org/10.1023/A:1009715923555

[45] Cortez C and Vapnik V. Support-Vector Networks. Machine Learning. 1995; 20 (3): 273-297. http://dx.doi.org/10.1007/BF00994018

[46] Plaza, A. et al. Recent advances in techniques for hyperspectral image processing. Remote Sensing of Environment. 2009. http://dx.doi.org/10.1016/j.rse.2007.07.028

[47] Landgrebe D and Biehl L. An Introduction to MultiSpec. School of Electrical and Computer Engineering, Purdue University. 2001.

[48] Mishra S. A hybrid least square-fuzzy bacterial foraging strategy for harmonic estimation. IEEE Trans. On Evolutionary Computation. 2005; 9(1): 61-73. http://dx.doi.org/10.1109/TEVC.2004.840144

[49] Passino KM. Biomimicry of bacterial foraging for distributed optimization and control. IEEE Control System Magazine. 2002; 22(3): 52-67. http://dx.doi.org/10.1109/MCS.2002.1004010

[50] Goldberg DE. Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, New York. 1989.

[51] AVIRIS Indian Pines Hyperspectral Dataset. Available from: https://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html

[52] AVIRIS Salinas Valley and ROSIS Pavia University Hyperspectral Datasets. Available from: http://www.ehu.es/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

[53] Karaboga D and Basturk B. A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm. Journal of Global Optimization. 2007; 39: 459-471. http://dx.doi.org/10.1007/s10898-007-9149-x

[54] Ding S. Spectral and Wavelet-based Feature Selection with Particle Swarm Optimization for Hyperspectral Classification. Journal of Software. 2011; 6 (7): 1248-1256.

[55] Dasgupta S, Das S, Abraham A and Biswas A. Adaptive Computational Chemotaxis in Bacterial Foraging Optimization: An Analysis. IEEE Transactions on Evolutionary Computation. 2009; 13(4): 919. http://dx.doi.org/10.1109/TEVC.2009.2021982

[56] Kim DH, Abraham A, and Cho JH. A hybrid genetic algorithm and bacterial foraging approach for global optimization. Information Sciences. 2007; 177(18): 3918-3937. http://dx.doi.org/10.1016/j.ins.2007.04.002