A PSO-based End-Member Selection Method for Spectral Unmixing of Multispectral Satellite Images

Mahamed G.H. Omran, Andries P Engelbrecht, and Ayed Salman

Abstract—An end-member selection method for spectral unmixing that is based on Particle Swarm Optimization (PSO) is developed in this paper. The algorithm uses the K-means clustering algorithm and a method of dynamic selection of end-members subsets to find the appropriate set of end-members for a given set of multispectral images. The proposed algorithm has been successfully applied to test image sets from various platforms such as LANDSAT 5 MSS and NOAA's AVHRR. The experimental results of the proposed algorithm are encouraging. The influence of different values of the algorithm control parameters on performance is studied. Furthermore, the performance of different versions of PSO is also investigated.

Keywords—End-members Selection, Multispectral Satellite Imagery, Particle Swarm Optimization, Spectral unmixing.

I. INTRODUCTION

In remote sensing, classification is the main tool for extracting information about the surface cover type. Conventional classification methods assign each pixel to one class (or species). This class can represent water, vegetation, soil, etc. The classification methods generate a map showing the species with highest concentration. This map is known as the thematic map. A thematic map is useful when the pixels in the image represent pure species (i.e. each pixel represents the spectral signature of one species). Hence, thematic maps are suitable for imagery data with a small ground sampling distance (GSD) such as LANDSAT Thematic Mapper (GSD = 30 m). However, thematic maps are not as useful for large GSD imagery such as NOAAa AVHRR (GSD = 1.1 km) because in this type of imagery pixels are usually not pure. Therefore, pixels need to be assigned to several classes along with their respective concentrations. Spectral unmixing provides a more complete and accurate classification than a thematic map generated by conventional classification methods.

Spectral unmixing can be used for the compression of multispectral imagery. Using spectral unmixing, the user can prioritize the species of interest in the compression process. This is done by first applying the spectral unmixing on the original images to generate the abundance images. The abundance images representing the species of interest are then prioritized by coding them with a relatively high bit rate. Other abundance images are coded using a relatively low bit rate. At the decoder, the species-prioritized reconstructed multispectral imagery is generated via a re-mixing process on the decoded abundance images [1]. This approach is feasible if the spectral unmixing algorithm results in a small (negligible) residual error.

In this paper, an end-member selection method for spectral unmixing that is based on Particle Swarm Optimization is proposed. PSO has successfully been applied to many optimization problems [2]. Hence, PSO is used in this paper to tackle the spectral unmixing problem. The algorithm uses the K-means clustering algorithm and a method of dynamic selection of end-members subsets to find the appropriate set of end-members for a given set of multispectral images.

The remainder of the paper is organized as follows: Section II introduces linear pixel unmixing. End-members selection methods are presented in Section III. Section IV discusses particle swarm optimization. The proposed algorithm is presented in Section V, while an experimental evaluation of the algorithm is provided in Section VI. Finally, Section VII concludes the paper.

II. LINEAR PIXEL UNMIXING (OR LINEAR MIXTURE MODELING)

Spectral unmixing is generally performed using a linear mixture modeling approach. In linear mixture modeling the spectral signature of each pixel vector is assumed to be a linear combination of a limited set of fundamental spectral components known as end-members. Hence, spectral unmixing can be formally defined as follows:
\[ \mathbf{z}_p = \mathbf{X} \mathbf{f} + \mathbf{e} = f_1 \mathbf{z}_1 + f_2 \mathbf{z}_2 + \Lambda + f_N \mathbf{z}_N + \mathbf{e} \quad (1) \]

where \( \mathbf{z}_p \) is a pixel signature of \( N_b \) components, \( \mathbf{X} \) is an \( N_b \times N_e \) matrix of end-members \( \mathbf{z}_{1:N_e} \), \( f_i \) is the fractional component of end-member \( i \) (i.e. proportion of footprint covered by species \( i \)), \( \mathbf{f} \) is the vector of fractional components \( (f_1, f_2, \Lambda, f_1, \Lambda, f_N, \Lambda)^T \), \( \mathbf{z}_i \) is the end-member \( i \) of \( N_b \) components, \( \mathbf{e} \) is the residual error vector of \( N_b \) components, \( N_b \) is the number of spectral bands and \( N_e \) is the number of components, \( N_e \leq N_b \).

Provided that the number of end-members is less than or equal to the true spectral dimensionality of the scene, the solution via classical least-squares estimation is,

\[ \mathbf{f} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{z}_p \]

Therefore, there are two requirements for linear spectral unmixing:

- The spectral signature of the end-members needs to be known.
- The number of end-members is less than or equal to the true spectral dimensionality of the scene (i.e. dimension of the feature space). This is known as the condition of identifiability.

The condition of identifiability restricts the application of the linear spectral unmixing when applied to multispectral imagery because the end-members may not correspond to physically identifiable species on the ground. Moreover, the number of distinct species in the scene may be more than the true spectral dimensionality of the scene. For example: for Landsat TM with seven spectral bands (\( N_b = 7 \)), the true spectral dimension is at most five (\( N_e = 5 \)) based on principal component analysis.

### III. Selection of the End-Members

There are many methods for spectral unmixing and end-member selection proposed in the literature [3]-[18]. In the following, several representative techniques are presented.

Mathematical techniques such as Gram-Schmidt orthogonalization and principal component analysis can be used to obtain orthogonal end-members which can be used to linearly unmix each pixel vector of the scene. There are several advantages for the mathematical techniques, namely:

- They result in minimum residual error.
- There is no human interaction time.

However, mathematical techniques suffer from the following drawbacks:

- They may generate end-members with negative components.

Manual techniques can also be used to obtain end-members which can be used to linearly unmix each pixel vector of the scene. In manual techniques, the user will select the end-members directly from the scene or from a library of end-members. The advantages of manual techniques are the disadvantages of mathematical techniques and vice versa.

Spectral screening is another way to obtain end-members. In this approach, a set of unique pixels are selected from scene, as described next. The selection is based on a user-specified spectral angle threshold. The approach works as follows: The first pixel in the image is assumed to be unique and is added to the set of unique pixels. The pixels in the image are then sequentially scanned and each pixel whose spectral angle with respect to all the unique pixels in the set exceeds the user-specified spectral angle threshold, is added to the set of unique pixels.

Clearly, this technique suffers from two major drawbacks:

- The generated set of unique pixels depends on the order in which the pixels are scanned.
- The generated set depends also on the spectral angle threshold.

To overcome the condition of identifiability, Maselli [7] proposed a method of dynamic selection of an optimum end-member subset. In this technique, an optimum subset of all available end-members is selected for spectral unmixing of each pixel vector in the scene. Thus, although not every pixel vector will have a fractional component for each end-member, the ensemble of all pixel vectors in the scene will collectively have fractional contributions for each end-member.

For each pixel vector, a unique subset of the available end-members is selected which minimizes the residual error after decomposition of that pixel vector. To determine the optimum end-members for pixel vector \( \mathbf{z}_p \), the pixel vector is projected onto all available normalized end-members. The most efficient projection, which corresponds to the highest dot product value \( c_{\max} \), indicates the first selected end-member \( \chi_{\max} \). It can be shown that this procedure is equivalent to finding the end-member with the smallest spectral angle with respect to \( \mathbf{z}_p \). The residual pixel signature, \( \mathbf{r}_p = \mathbf{z}_p - c_{\max} \chi_{\max} \), is then used to identify the second end-member by repeating the projection onto all remaining end-members. The process continues up to the identification of a prefixed \( N_{\max} \) maximum \( N_e \) number of end-members from the total of \( N_m \) available end-members.

More recently, Saghir et al. [9] proposed a method to obtain end-members from the scene with relatively small residual errors. In this paper, the proposed method is referred to as ISO-UNMIX. In ISO-UNMIX, the set of end-members are chosen from a thematic map resulting from a modified

\[ \text{ISO-UNMIX} \]
ISODATA [19]. The modified ISODATA uses the spectral angle measure instead of the Euclidean distance measure to reduce the effect of shadows and sun angle effects. The end-members are then set as the centroids of the compact and well-populated clusters. Maselli’s approach discussed above is then used to find the optimum end-member subset from the set of available end-members for each pixel in the scene. Linear spectral unmixing is then applied to generate the abundance images.

According to Saghri et al. [9], the proposed approach has several advantages: the resulting end-members correspond to physically identifiable (and likely pure) species on the ground, the residual error is relatively small and minimal human interaction time is required. However, this approach has a drawback in that it uses ISODATA which depends on the initial conditions.

IV. PARTICLE SWARM OPTIMIZATION

Particle swarm optimizers are population-based optimization algorithms modeled after the simulation of social behavior of bird flocks [2], [20]. PSO is generally considered to be an evolutionary computation (EC) paradigm. Other EC paradigms include genetic algorithms (GA), genetic programming (GP), evolutionary strategies (ES), and evolutionary programming (EP) [21]. These approaches simulate biological evolution and are population-based. In a PSO system, a swarm of individuals (called particles) fly through the search space. Each particle represents a candidate solution to the optimization problem. The position of a particle is influenced by the best position visited by itself (i.e. its own experience) and the position of the best particle in its neighborhood (i.e. the experience of neighboring particles). When the neighborhood of a particle is the entire swarm, the best position in the neighborhood is referred to as the global best particle, and the resulting algorithm is referred to as the gbest PSO. When smaller neighborhoods are used, the best solution is. Different social structures have been investigated [25], [26], with the star topology being used most.

Each particle in the swarm is represented by the following characteristics:

- \( x_i \): The current position of the particle;
- \( v_i \): The current velocity of the particle;
- \( y_i \): The personal best position of the particle.

The personal best position of particle \( i \) is the best position (i.e. one resulting in the best fitness value) visited by particle \( i \) so far. Let \( f \) denote the objective function. Then the personal best of a particle at time step \( t \) is updated as

\[
y_i(t+1) = \begin{cases} 
  y_i(t), & \text{if } f(x_i(t+1)) \geq f(y_i(t)) \\
  x_i(t+1), & \text{if } f(x_i(t+1)) < f(y_i(t)) 
\end{cases}
\]  

If the position of the global best particle is denoted by the vector \( \hat{y} \), then

\[
j(t) = \{y_{o_1}, y_{o_2}, \ldots, y_{o_k}\} = \min\{f(y_{o_1}), f(y_{o_2}), \ldots, f(y_{o_k})\}
\]  

where \( s \) denotes the size of the swarm. For the lbest model, a swarm is divided into overlapping neighborhoods of particles. For each neighborhood \( N_j \), a best particle is determined with position \( \hat{y}_j \). This particle is referred to as the neighborhood best particle, defined as

\[
\hat{y}_j(t+1) = \{y_j(t+1) | f(\hat{y}_j(t+1)) = \min\{f(y_j(t))\}, \forall y_j \in N_j\}
\]

where

\[
N_j = \{y_{j_1}(t), y_{j_2}(t), \ldots, y_{j_k}(t), y_{j_1}(t+1), y_{j_2}(t+1), \ldots, y_{j_k}(t+1)\}
\]

Neighborhoods are usually determined using particle indices [23], however, topological neighborhoods can also be used [24]. It is clear that gbest is a special case of lbest with \( l = s \); that is, the neighborhood is the entire swarm. While the lbest approach results in a larger diversity, it is still slower than the gbest approach.

For each iteration of a PSO algorithm, the velocity \( v_i \) update step is specified for each dimension \( j = 1, \ldots, N_d \), where \( N_d \) is the dimension of the problem. Hence, \( v_{ij} \) represents the \( j \)th element of the velocity vector of the \( i \)th particle. Thus the velocity of particle \( i \) is updated using the following equation:

\[
v_i(t+1) = w v_i(t) + c_1 r_{1j}(t)(y_j(t) - x_i(t)) + c_2 r_{2j}(t)(\hat{y}_j(t) - x_i(t))
\]  

where \( w \) is the inertia weight [24], \( c_1 \) and \( c_2 \) are the acceleration constants and \( r_{1j}, r_{2j} \sim U(0,1) \). Equation (7) consists of three components, namely

- The inertia weight term, \( w \), which serves as a memory of previous velocities. The inertia weight controls the impact of the previous velocity: a large inertia weight favors exploration, while a small inertia weight favors exploitation [22].
- The cognitive component, \( y_j(t) - x_i(t) \), which represents the particle’s own experience as to where the best solution is.
- The social component, \( \hat{y}_j(t) - x_i(t) \), which represents the belief of the entire swarm as to where the best solution is. Different social structures have been investigated [25], [26], with the star topology being used most.

The position of particle \( i \), \( x_i \), is then updated using the following equation:
\( x_i(t + 1) = x_i(t) + v_i(t + 1) \) (8)

The reader is referred to Van den Bergh [27] and Van den Bergh et al. [28] for a study of the relationship between the inertia weight and acceleration constants, in order to select values which will ensure convergent behavior. Velocity updates can also be clamped through a user defined maximum velocity, \( V_{\text{max}} \), which would prevent them from exploding, thereby causing premature convergence [27].

The PSO algorithm performs the update equations above, repeatedly, until a specified number of iterations have been exceeded, or velocity updates are close to zero. The quality of particles is measured using a fitness function which reflects the optimality of a particular solution.

V. THE PSO-BASED END-MEMBER SELECTION (PSO-EMS) ALGORITHM

This section introduces the PSO-EMS algorithm by first presenting a measure to quantify the quality of a spectral unmixing algorithm, after which the PSO-EMS algorithm is shown.

A. Measure of Quality

To measure the quality of a spectral unmixing algorithm, the root mean square (RMS) residual error can be used, which is defined as follows:

\[ E = \sum_{j=1}^{N_p} \sqrt{M_j} \] (9)

where

\[ \hat{M} = \frac{\sum_{p=1}^{N_p} (z_p - X.f)^2}{N_p} \]

where \( N_p \) is the number of pixels in the image.

B. The PSO-EMS Algorithm

In the context of spectral unmixing, a single particle represents \( N_m \) end-members. That is, each particle \( x_i \) is constructed as \( x_i = (X_{\chi_i,1}, X_{\chi_i,2}, \ldots, X_{\chi_i,N_e}) \) where \( \chi_i,k \) refers to the \( k^{th} \) end-member vector of the \( i^{th} \) particle. Therefore, a swarm represents a number of candidate end-members. The quality of each particle is measured using the RMS residual error (defined in (9)) as follows:

\[ f(x_i) = E \] (10)

The algorithm works as follows: each particle is randomly initialized from the multispectral image set to contain \( N_m \) end-members. The K-means clustering algorithm (using a few iterations) is then applied to a random set of particles with a user-specified probability, \( p_{\text{means}} \). The K-means algorithm is used in order to refine the chosen end-members, reduce the search space. Then for each particle \( i \), the \( N_m \) end-members of the particle form the pool of available candidate end-members for the subsequent spectral unmixing procedure.

Maselli’s approach [7] is used to dynamically select the \( N_e \) optimum end-member subsets from the pool of \( N_m \) end-members. Each pixel vector is then spectrally decomposed as a linear combination of its optimum subset of end-members. The RMS residual error for particle \( i \) is then calculated. The PSO velocity and update equations (7) and (8) are then applied. The procedure is repeated until a stopping criterion is satisfied. The \( N_m \) end-members of the best particle are used to generate the abundance images.

C. The Generation of Abundance Images

For each species represented by an end-member, the ensemble of all fractional components forms a concentration map (i.e. abundance map). The fractional concentration maps are then optimally mapped to an eight-bit integer format for display and storage purposes. This is done using the following non-linear mapping function [9]:

\[ \Omega = 255(f_{\exp} - (f_{\min})_{\exp}) \] (11)

where \( \Omega \) is the mapped integer fractional component in the range of \( 0 \leq \Omega \leq 255 \), \( f \) is the fractional component, \( f_{\min} \) is the minimum fractional component, \( f_{\max} \) is the maximum fractional component and \( \exp \) is the floating-point exponent parameter in the range of \( 0 \leq \exp \leq 1.0 \). In this paper, \( \exp \) is set to 0.6 for the abundance images as suggested by Saghri et al. [9].

The PSO-EMS algorithm is summarized below:

1. Initialize each particle to contain \( N_m \) randomly selected end-members.
2. For \( t = 1 \) to \( t_{\text{max}} \)
   (a) For each particle \( i \)
      i. Apply K-means for a few iterations with a probability \( p_{\text{means}} \).
      ii. For each pixel \( z_p \)
          o Find the \( N_e \) optimum end-member subset.
          o Apply linear spectral unmixing using (1).
      iii. Calculate the fitness, \( f(x_i) \)
   (b) Find the global best solution \( \hat{\mathbf{x}}(t) \)
   (c) Update the end-members using (7) and (8)
3. Generate the abundance images using the \( N_m \) end-members of particle.
In general, the complexity of the PSO-EMS algorithm is $O(t_{max}N_p)$. The parameters $s$ and $t_{max}$ can be fixed in advance. Typically $s$ and $t_{max} << N_p$. Therefore, the time complexity of PSO-EMS is $O(N_p)$. Hence, in general the algorithm has linear time complexity in the size of a data set.

VI. EXPERIMENTAL RESULTS

The PSO-EMS algorithm has been applied to two types of imagery data, namely LANDSAT 5 MSS (79 m GSD) and NOAA's AVHRR (1.1 km GSD) images. These image sets have been selected to test the algorithms on a variety of platforms with a relatively large GSD which represent good candidates for spectral unmixing in order to get sub-pixel resolution. The two image sets are described below:

LANDSAT 5 MSS: Fig. 1 shows band 4 of the four-channel multispectral test image set of the Lake Tahoe region in the U.S. Each channel is comprised of a $300 \times 300$, 8-bit per pixel (remapped from the original 6 bit) image and corresponds to a GSD of 79 m. The test image set is one of the North American Landscape Characterization (NALC) Landsat multispectral scanner data sets obtained from the U.S. Geological Survey (USGS). The result of a preliminary principal component study of this data set indicates that its intrinsic true spectral dimension $N_e$ is 3. As in [9], a total of six end-members were obtained from the data set (i.e. $N_m = 6$).

Fig. 1 Band 4 of the Landsat MSS test image of Lake Tahoe

NOAA's AVHRR: Fig. 2 shows the five-channel multispectral test image set of an almost cloud-free territory of the entire UK. This image set was obtained from the University of Dundee Satellite Receiving Station. Each channel (one visible, one near-infra red and three in the thermal range) is comprised of a $847 \times 1009$, 10-bit per pixel (1024 gray levels) image and corresponds to a GSD of 1.1 km. The result of a preliminary principal component study of this data set indicates that its intrinsic true spectral dimension $N_e$ is 3. As in [9], a total of eight end-members were obtained from the data set (i.e. $N_m = 8$).

The rest of this section is organized as follows: Section VI.A illustrates that the PSO-EMS can be used successfully as an end-member selection method by comparing it to the end-member selection method proposed by Saghri et al. [9] (i.e. ISO-UNMIX). Saghri et al. [9] showed that ISO-UNMIX performed very well compared to other popular spectral unmixing methods. Section VI.B investigates the influence of the different PSO-EMS control parameters. Finally, the use of different PSO models (namely, gbest, lbest and lbest-to-gbest) are investigated in Section VI.C.

The results reported in this section are averages and standard deviations over 10 simulations. In addition, we start with an lbest implementation of the PSO (with zero-radius neighborhood) and linearly increase the neighborhood radius until a gbest implementation of the PSO is reached. In this paper, this approach is referred to as lbest-to-gbest-PSO. This hybrid approach is used in order to initially avoid being trapped in local optima, by initially using an lbest approach [23]. The algorithm then attempts to converge to the best solution found by the initial phase by using a gbest approach. The PSO-EMS parameters were initially set as follows: $p_{max} = 0.1$, $s = 20$, $t_{max} = 100$, number of K-means iterations is 10 (the effect of these values are then investigated), $w = 0.72$, $c_1 = 1.49$ and $V_{max} = 255$ for the Lake Tahoe image set and $V_{max} = 1023$ for the UK image set. No attempt was made to tune the PSO parameters (i.e. $w$, $c_1$ and $c_2$) to each data set. The rationale behind this decision is the fact that in real-world applications the evaluation time is significant and as such parameter tuning is usually a time consuming process. These parameters are used in this section unless otherwise specified.

A. PSO-EMS vs. ISO_UNMIX

This section presents results to compare the performance of the PSO-EMS algorithm with that of the ISO-UNMIX algorithm for each of the test image sets.

Table I summarizes the results for the two image sets. The results are compared based on the RMS residual error (defined in (9)). The results showed that, for both image sets, PSO-EMS performed better than the ISO-UNMIX in terms of the RMS residual error. Figs. 3 and 4 show the abundance images generated from ISO-UNMIX and PSO-EMS, respectively, when applied to the Lake Tahoe image set. In addition, Figs. 5 and 6 show the abundance images generated from ISO-UNMIX and PSO-EMS, respectively, when applied to the UK image set. For display purposes the fractional species concentrations were mapped to 8 bits per pixels abundance images.
Fig. 2 AVHRR Image of UK, Size: 847x1009, 5 bands, 10 bpp

TABLE I
COMPARISON BETWEEN ISO-UNMIX AND PSO-EMS

<table>
<thead>
<tr>
<th>Image</th>
<th>ISO_UNMIX</th>
<th>PSO-EMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LANDSAT 5 MSS</td>
<td>0.491837</td>
<td>0.462197 ± 0.012074</td>
</tr>
<tr>
<td>NOAA's AVHRR</td>
<td>3.725979</td>
<td>3.510287 ± 0.045442</td>
</tr>
</tbody>
</table>

B. Influence of PSO-EMS Parameters

The PSO-EMS algorithm has a number of parameters that have an influence on the performance of the algorithm. These parameters include $V_{\text{max}}$, the swarm size, the number of PSO iterations, $p_{\text{means}}$, and the number of K-means iterations. This section investigates the influence of different values of these parameters using the Lake Tahoe image set.

1) Velocity Clamping

Table II shows that using $V_{\text{max}} = 5$ or $V_{\text{max}} = 255$ generally produces comparable results. However, the standard deviation in the case of $V_{\text{max}} = 5$ is smaller than the standard deviation in the case of $V_{\text{max}} = 255$. Hence, using $V_{\text{max}} = 5$ generates more stable results than using $V_{\text{max}} = 255$.

2) Swarm Size

Increasing the swarm size from 20 to 50 particles improves the performance of the PSO-EMS algorithm as shown in Table III. On the other hand, reducing the swarm size from 20 to 10 particles significantly reduces the efficiency of the PSO-EMS algorithm. The rationale behind these results is that increasing the number of particles increases diversity, thereby limiting the effects of initial conditions and reducing the possibility of being trapped in local minima.

![Image of UK with bands 1 to 5]
Fig. 4 Species concentration maps resulting from the application of PSO-EMS to unmix the Lake Tahoe test image set

Table III

<table>
<thead>
<tr>
<th>$s$</th>
<th>RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.468706 ± 0.004753</td>
</tr>
<tr>
<td>20</td>
<td>0.462197 ± 0.012074</td>
</tr>
<tr>
<td>50</td>
<td>0.459195 ± 0.009389</td>
</tr>
</tbody>
</table>

3) Number of PSO Iterations

Reducing the number of PSO iterations, $t_{\text{max}}$, from 100 to 50 did not reduce the performance of the PSO-EMS algorithm as shown in Table IV. Similarly, increasing $t_{\text{max}}$ from 100 to 150, did not significantly improve the performance of the PSO-EMS.

Table IV

<table>
<thead>
<tr>
<th>$t_{\text{max}}$</th>
<th>RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.468041 ± 0.004735</td>
</tr>
<tr>
<td>100</td>
<td>0.462197 ± 0.012074</td>
</tr>
<tr>
<td>150</td>
<td>0.465614 ± 0.00739</td>
</tr>
</tbody>
</table>

Fig. 5 Species concentration maps resulting from the application of ISO-UNMIX to unmix the UK test image set

4) $p_{\text{means}}$

Applying the K-means clustering algorithm to a larger set of particles is expected to improve the performance of the PSO-EMS algorithm. The rationale behind this expectation is the fact that the K-means algorithm generally reduces the search space and refines the end-members. This expectation is verified by the results shown in Table V which shows that increasing the value of $p_{\text{means}}$ significantly improves the performance of the PSO-EMS algorithm. However, as a trade-off, increasing the value of $p_{\text{means}}$ will increase the computational requirements of the PSO-EMS algorithm.
Fig. 6 Species concentration maps resulting from the application of PSO-EMS to unmix the UK test image set.

TABLE V
EFFECT OF \(p_{\text{kmeans}}\) ON THE PERFORMANCE OF PSO-EMS USING LAKE TAHOE IMAGE SET

<table>
<thead>
<tr>
<th>(p_{\text{kmeans}})</th>
<th>RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.462197 ± 0.012074</td>
</tr>
<tr>
<td>0.25</td>
<td>0.460776 ± 0.009120</td>
</tr>
<tr>
<td>0.5</td>
<td>0.454029 ± 0.007051</td>
</tr>
<tr>
<td>0.9</td>
<td>0.445367 ± 0.012339</td>
</tr>
</tbody>
</table>

5) Number of K-means Iterations
Reducing number of K-means iterations from 10 to 5 degrades the performance of the PSO-EMS as shown in Table VI. On the other hand, increasing the number of K-means iterations from 10 to 50 did not improve the performance of the PSO-EMS as shown in Table VI. These results suggest that using 10 iterations of K-means is a good choice for the Lake Tahoe image set. However, when the number of K-means iterations was reduced to 5 iterations but at the same time \(p_{\text{kmeans}}\) was increased from 0.1 to 0.5 the generated RMS was equal to 0.458149 ± 0.004554 which is significantly better than the results in Table VI. This result suggests that the number of K-means iterations can be reduced without affecting the performance of PSO-EMS given that the \(p_{\text{kmeans}}\) is increased.

TABLE VI
EFFECT OF THE NUMBER OF K-MEANS ITERATIONS ON THE PERFORMANCE OF PSO-EMS USING LAKE TAHOE IMAGE SET \((p_{\text{kmeans}} = 0.1)\)

<table>
<thead>
<tr>
<th>No. of K-means iterations</th>
<th>RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.468407 ± 0.004212</td>
</tr>
<tr>
<td>10</td>
<td>0.462197 ± 0.012074</td>
</tr>
<tr>
<td>50</td>
<td>0.466708 ± 0.004524</td>
</tr>
</tbody>
</table>

C. Comparison of \(g_{\text{best}}, l_{\text{best}}\) and \(l_{\text{best}}\)-to-\(g_{\text{best}}\)-PSO
In this section, the effect of different models of PSO is investigated using the Lake Tahoe image set. A comparison is made between \(g_{\text{best}}\), \(l_{\text{best}}\) and \(l_{\text{best}}\)-to-\(g_{\text{best}}\)-PSO (which has been used in the above experiments) using a swarm size of 20 particles. For \(l_{\text{best}}\) PSO, a neighborhood size of \(l = 2\) was used. Table VII summarizes the result of the comparison. The results show no significant difference in performance. However, the standard deviation in the case of \(l_{\text{best}}\)-to-\(g_{\text{best}}\) PSO is the largest. Hence, using \(l_{\text{best}}\)-to-\(g_{\text{best}}\) PSO generates the least stable result.

TABLE VII
COMPARISON OF \(g_{\text{best}}\), \(l_{\text{best}}\) AND \(l_{\text{best}}\)-TO-\(g_{\text{best}}\) VERSIONS OF PSO-EMS USING LAKE TAHOE IMAGE SET

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>(g_{\text{best}}) PSO</td>
<td>0.465809 ± 0.006562</td>
</tr>
<tr>
<td>(l_{\text{best}}) PSO</td>
<td>0.465020 ± 0.004942</td>
</tr>
<tr>
<td>(l_{\text{best}})-to-(g_{\text{best}}) PSO</td>
<td>0.462197 ± 0.012074</td>
</tr>
</tbody>
</table>

VII. Conclusion
This paper presented a new spectral unmixing approach using PSO. The PSO-EMS algorithm has as objective to determine the appropriate set of end-members for a given multispectral image set. The PSO-EMS algorithm was compared against a relatively recent end-member selection method which was proposed by Saghri et al. [9]. The PSO-EMS algorithm produced better results when applied to test image sets from various platforms such as LANDSAT 5 MSS and NOAA’s AVHRR. The effects of different PSO-EMS control parameters were then studied. Finally, the performance of different versions of PSO was investigated.

Experiments need to be conducted to compare the PSO-EMS with the multi-start K-means (with the best result generated from applying K-means \(s_{\text{max}}p_{\text{kmeans}}\) times, where
each K-means starts from random cluster centroids) and other spectral unmixing techniques on various AVHRR images. The performance of the PSO-EMS when applied to hyperspectral Satellite imagery is a potential topic for future research.

REFERENCES


Mahamed G. H. Omran received his B.Sc. and M.Sc. degrees with Honors in Computer Engineering from Kuwait University, State of Kuwait, in 1998 and 2000, respectively. He completed his Ph.D. in the Department of Computer Science at University of Pretoria, South Africa, in April 2005. He is a member of the Computational Intelligence Research Group (http://cire.cs.up.ac.za/) of the Department of Computer Science at University of Pretoria, South Africa.

His current research interest is in the area of computational intelligence with special emphasis on Particle Swarm Optimization (PSO) and Differential Evolution (DE). In addition, Dr. Omran is interested in the area of data clustering, unsupervised image classification and color image quantization.

Andries Engelbrecht received the M.Sc and PhD degrees from the University of Stellenbosch, Stellenbosch, South Africa, in 1994 and 1999, respectively. He is a Full Professor at the Department of Computer Science, University of Pretoria, Pretoria, South Africa, leading a research group of three staff members and 45 Masters and PhD students.

His research interests include aspects of swarm intelligence, evolutionary computation, artificial immune systems, data mining, swarm robotics, and neural networks, with several publications in those fields. Prof. Engelbrecht is also a member of the IEEE CIS and its task forces on Evolutionary Computation and Games, Swarm Intelligence, and Coevolution. He is an associate editor of IEEE Transactions on Evolutionary Computation. He is the author of the book, "Computational Intelligence: An Introduction", Wiley, 2002, and recently authored the book "Computational Swarm Intelligence" to be published by Wiley.

Aydin Salman received his MS and PhD degrees from Syracuse University, NY, USA, in 1996 and 1999, respectively. He is an Assistant Professor at the Department of Computer Engineering, Kuwait University, Kuwait.

His research interests include Application of evolutionary computations in data mining, web mining and multi-objective optimization, E-Commerce, Neural Network, Machine Learning.