

Chapter 7

Applications

This chapter presents PSO-based approaches to tackle the color image quantization and spectral unmixing problems. The proposed approaches are then applied on different image sets to evaluate their performance, and they are compared with other *state-of-the-art* approaches.

7.1 A PSO-based Color Image Quantization Algorithm

A PSO-based color image quantization algorithm is developed in this section. The algorithm randomly initializes each particle in the swarm to contain K centroids (i.e. color triplets). The K-means clustering algorithm is then applied to each particle at a user-specified probability to refine the chosen centroids. Each pixel is then assigned to the cluster with the closest centroid. The PSO is then applied to refine the centroids obtained from the K-means algorithm. The proposed algorithm is then applied to commonly used images. It is shown from the conducted experiments that the proposed algorithm generally results in a significant improvement of image quality compared to other well-known approaches. The influence of different values of the algorithm control parameters is studied. Furthermore, the performance of different versions of PSO is also investigated.

7.1.1 The PSO-based Color Image Quantization (PSO-CIQ) Algorithm

This section defines the terminology used throughout section 7.1. A measure is given to quantify the quality of the resultant quantized image, after which the PSO-CIQ algorithm is introduced.

Define the following symbols:

N_p denotes the number of image pixels

K denotes the number of clusters (i.e. colors in the colormap)

z_p denotes the coordinates of pixel p

m_k denotes the centroid of cluster k (representing one color triple in the colormap)

In this section, the terms centroid and color triple are used interchangeably.

Measure of Quality

The most general measure of performance is the mean square error (MSE) of the quantized image using a specific colormap. The MSE was defined in equation (3.27), and is repeated here for convenience:

$$MSE = \frac{\sum_{k=1}^K \sum_{\forall z_p \in C_k} (z_p - m_k)^2}{N_p} \quad (7.1)$$

where C_k is the k^{th} cluster.

The PSO-CIQ Algorithm

In this subsection, a new post-clustering color image quantization approach is described. The proposed approach is of the class of quantization techniques that performs clustering of the color space.

In the context of color image quantization, a single particle represents a colormap (i.e. a particle consists of K cluster centroids representing RGB color triplets). The RGB coordinates in each color triple are floating-point numbers. Each particle \mathbf{x}_i is constructed as $\mathbf{x}_i = (\mathbf{m}_{i,1}, \dots, \mathbf{m}_{i,k}, \dots, \mathbf{m}_{i,K})$ where $\mathbf{m}_{i,k}$ refers to the k^{th} cluster centroid vector of the i^{th} particle. Therefore, a swarm represents a number of candidate colormaps. The quality of each particle is measured using the MSE (defined in equation (7.1)) as follows:

$$f(\mathbf{x}_i) = MSE(\mathbf{x}_i) \quad (7.2)$$

The algorithm initializes each particle randomly from the color image to contain K centroids (i.e. color triplets). The set of K color triplets represents the colormap. The K-means clustering algorithm is then applied to each particle at a user-specified probability, p_{kmeans} . The K-means algorithm is used in order to refine the chosen colors and to reduce the search space. Each pixel is then assigned to the cluster with the closest centroid. The fitness function of each particle is calculated using equation (7.2). The PSO velocity and update equations (2.8) and (2.10) are then applied. The procedure is repeated until a stopping criterion is satisfied. The colormap of the global best particle after t_{max} iterations is chosen as the optimal result.

The PSO-CIQ algorithm is summarized in Figure 7.1.

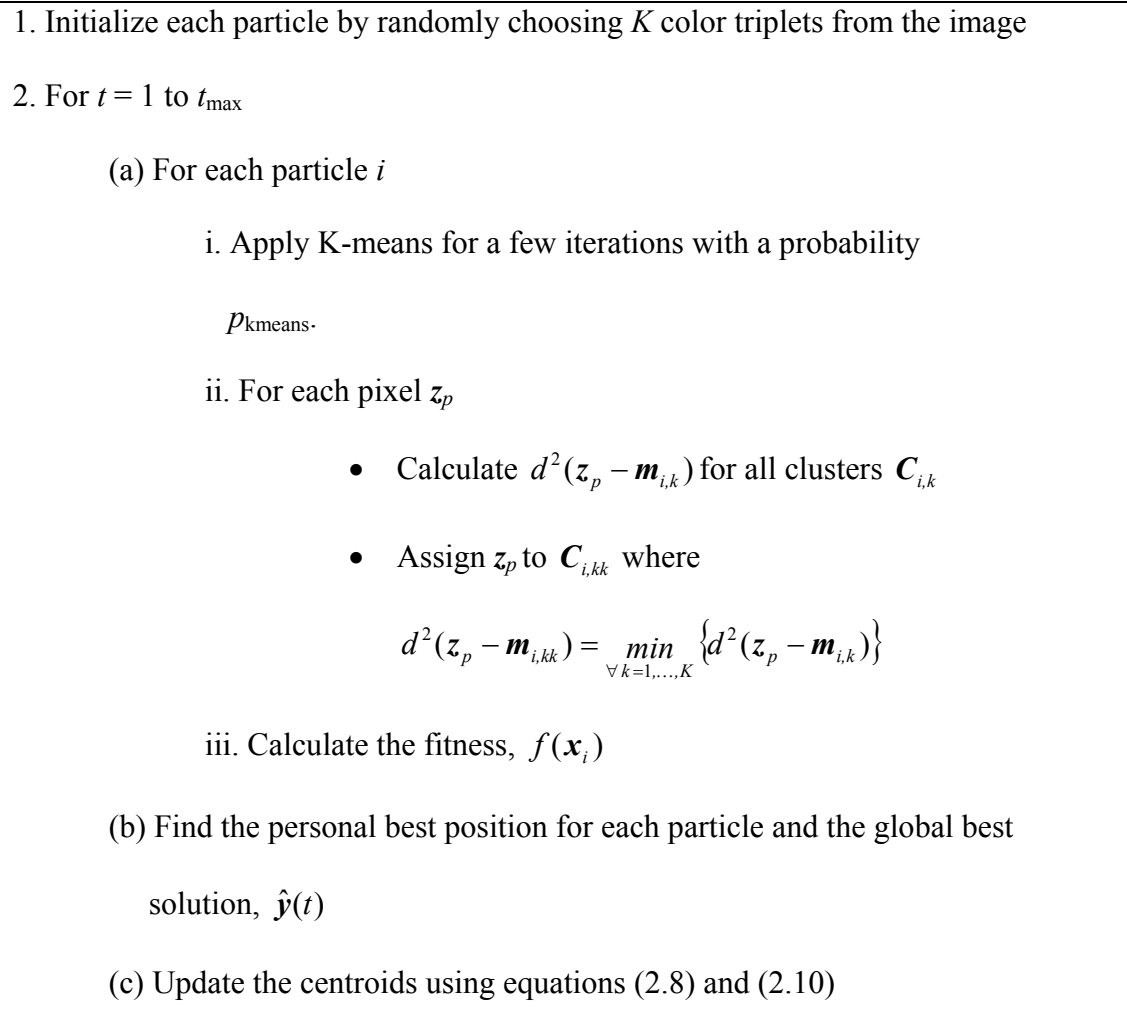


Figure 7.1: The PSO-CIQ algorithm

In general, the complexity of the PSO-CIQ algorithm is $O(sKt_{\max}N_p)$. The parameters s , K and t_{\max} can be fixed in advance. Typically s , K and $t_{\max} \ll N_p$. Therefore, the time complexity of PSO-CIQ is $O(N_p)$. Hence, in general the algorithm has linear time complexity in the size of a data set.

7.1.2 Experimental Results

The PSO-CIQ algorithm was applied to a set of four commonly used color images namely: *Lenna*, *mandrill*, *jet* and *peppers* (shown in Figures 7.1(a), 7.2(a), 7.3(a) and 7.4(a), respectively). The size of each image is 512×512 pixels. All images are quantized to 16, 32 and 64 colors.

The rest of this section is organized as follows: Section 7.1.2.1 illustrates that the PSO-CIQ can be used successfully as a color image quantization algorithm by comparing it to other well-known color image quantization approaches. Section 7.1.2.2 investigates the influence of the different PSO-CIQ control parameters. Finally, the use of different PSO models (namely, *gbest*, *lbest* and *lbest-to-gbest*) are investigated in section 7.1.2.3.

The results reported in this section are averages and standard deviations over 10 simulations. Since *lbest-to-gbest* PSO was generally the best performer in chapter 4, *lbest-to-gbest* PSO is used in this section unless otherwise specified. The PSO-CIQ parameters were initially set as follows: $p_{kmeans} = 0.1$, $s = 20$, $t_{max} = 50$, number of K-means iterations is 10 (the effect of these values are then investigated), $w = 0.72$, $c_1 = c_2 = 1.49$ and $V_{max} = 255$ for all the test images. These parameters were used in this section unless otherwise specified. For the GCMA [Scheunders, A Genetic 1997] a population of 20 chromosomes was used, and evolution continued for 50 generations. For the SOM, a Kohonen network of 4×4 nodes was used when quantizing an image to 16 colors, a Kohonen network of 8×4 nodes was used when quantizing an image to 32 colors, and a Kohonen network of 8×8 nodes was used when quantizing an image to 64 colors. All SOM parameters were set as in Pandya and Macy [1996]: the learning rate $\eta(t)$ was initially set to 0.9 then decreased by 0.005 until it reached

0.005, the neighborhood function $\Delta_w(t)$ was initially set to $(4+4)/4$ for 16 colors, $(8+4)/4$ for 32 colors, and $(8+8)/4$ for 64 colors. The neighborhood function is then decreased by 1 until it reached zero.

7.1.2.1 PSO-CIQ vs. Well-Known Color Image Quantization Algorithms

This section presents results to compare the performance of the PSO-CIQ algorithm with that of SOM and GCMA (both discussed in section 3.3.2) for each of the test images.

Table 7.1 summarizes the results for the four images. The results of the GCMA represent the best case over several runs and are copied from Scheunders [A Genetic 1997]. The results are compared based on the MSE measure (defined in equation 7.1). The results showed that, in general, PSO-CIQ outperformed GCMA in all the test images except for the mandrill image and the case of quantizing the Jet image to 64 colors. Furthermore, PSO-CIQ generally performed better than SOM for both Lenna and peppers images. SOM and PSO-CIQ performed comparably well when applied to the mandrill image. SOM generally performed better than PSO-CIQ when applied to the Jet image. Figures 7.2, 7.3, 7.4 and 7.5 show the visual quality of the quantized images generated by PSO-CIQ when applied to Lenna, peppers, jet and mandrill, respectively.

Table 7.1: Comparison between SOM, GCMA and PSO-CIQ				
Image	<i>K</i>	SOM	GCMA	PSO-CIQ
Lenna	16	235.6 ± 0.490	332	210.203 ± 1.487
	32	126.400 ± 1.200	179	119.167 ± 0.449
	64	74.700 ± 0.458	113	77.846 ± 16.132
Peppers	16	425.600 ± 13.162	471	399.63 ± 2.636
	32	244.500 ± 3.854	263	232.046 ± 2.295
	64	141.600 ± 0.917	148	137.322 ± 3.376
Jet	16	121.700 ± 0.458	199	122.867 ± 2.0837
	32	65.000 ± 0.000	96	71.564 ± 6.089
	64	38.100 ± 0.539	54	56.339 ± 11.15
Mandrill	16	629.000 ± 0.775	606	630.975 ± 2.059
	32	373.600 ± 0.490	348	375.933 ± 3.42
	64	234.000 ± 0.000	213	237.331 ± 2.015



(a) Original



(b) 16 colors



(c) 32 colors



(d) 64 colors

Figure 7.2: Quantization results for the Lenna image using PSO-CIQ



(a) Original



(b) 16 colors



(c) 32 colors



(d) 64 colors

Figure 7.3: Quantization results for the peppers image using PSO-CIQ



(a) Original



(b) 16 colors

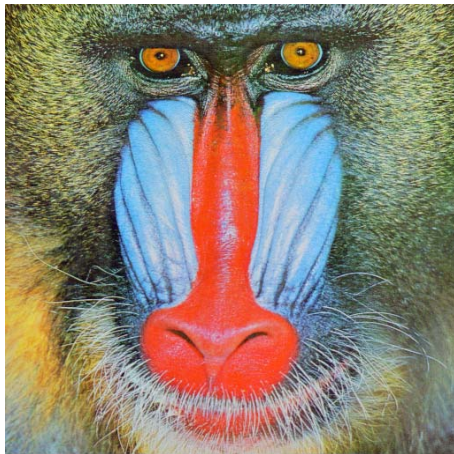


(c) 32 colors



(d) 64 colors

Figure 7.4: Quantization results for the jet image using PSO-CIQ



(a) Original



(b) 16 colors



(c) 32 colors



(d) 64 colors

Figure 7.5: Quantization results for the mandrill image using PSO-CIQ

7.1.2.2 Influence of PSO-CIQ Parameters

The PSO-CIQ algorithm has a number of parameters that have an influence on the performance of the algorithm. These parameters include V_{max} , the swarm size, the number of PSO iterations, p_{kmeans} and the number of K-means iterations. This section investigates the influence of different values of these parameters using the Lenna image when quantized to 16 colors.

Velocity Clamping

Table 7.2 shows that using $V_{max} = 5$ or $V_{max} = 255$ generally produces comparable results.

Table 7.2: Effect of V_{max} on the performance of PSO-CIQ using Lenna image (16 colors)	
	MSE
$V_{max}=5$	209.338 ± 0.402
$V_{max}=255$	210.203 ± 1.487

Swarm Size

Increasing the swarm size from 20 to 50 particles slightly improves the performance of the PSO-CIQ algorithm as shown in Table 7.3. Similarly, increasing the swarm size from 50 to 100 particles slightly improves the performance of the PSO-CIQ algorithm. On the other hand, reducing the swarm size from 20 to 10 particles significantly reduces the efficiency of the PSO-CIQ 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.

Table 7.3: Effect of the swarm size on the performance of PSO-CIQ using Lenna image (16 colors)	
	MSE
$s = 10$	212.196 ± 2.458
$s = 20$	210.203 ± 1.487
$s = 50$	210.06 ± 1.11
$s = 100$	209.468 ± 0.703

Number of PSO iterations

Increasing the number of PSO iterations, t_{\max} , from 50 to 100 slightly improves the performance of the PSO-CIQ algorithm as shown in Table 7.4. Similarly, increasing t_{\max} from 100 to 150 slightly improves the performance of the PSO-CIQ algorithm. Therefore, it can be concluded that increasing t_{\max} generally improves the performance of the PSO-CIQ algorithm.

Table 7.4: Effect of the number of PSO iterations on the performance of PSO-CIQ using Lenna image (16 colors)	
	MSE
$t_{\max} = 50$	210.203 ± 1.487
$t_{\max} = 100$	209.412 ± 0.531
$t_{\max} = 150$	208.866 ± 0.22

p_{kmeans}

Applying the K-means clustering algorithm to a larger set of particles is expected to improve the performance of the PSO-CIQ algorithm. The rationale behind this expectation is the fact that the K-means algorithm generally reduces the search space and refines the chosen colors. This expectation is verified by the results shown in Table 7.5 which shows that increasing the value of p_{kmeans} generally improves the performance of the PSO-CIQ algorithm. However, as a trade-off, increasing the value of p_{kmeans} will increase the computational requirements of the PSO-CIQ algorithm.

Table 7.5: Effect of p_{kmeans} on the performance of PSO-CIQ using Lenna image (16 colors)	
	MSE
$p_{kmeans} = 0.1$	210.203 ± 1.487
$p_{kmeans} = 0.25$	209.238 ± 0.74
$p_{kmeans} = 0.5$	209.045 ± 0.594
$p_{kmeans} = 0.9$	208.886 ± 0.207

Number of K-means iterations

Reducing the number of K-means iterations from 10 to 5 degrades the performance of the PSO-CIQ as shown in Table 7.6. On the other hand, increasing the number of K-means iterations from 10 to 50 significantly improves the performance of the PSO-CIQ as shown in Table 7.6. These results suggest that increasing the number of K-means iterations improves the performance of the PSO-CIQ. However, when the number of K-means iterations was reduced to 5 iterations but at the same time p_{kmeans} was increased from 0.1 to 0.5 the generated MSE was 210.315 ± 1.563 which is significantly better than the corresponding result in Table 7.6. This result suggests that

the number of K-means iterations can be reduced without affecting the performance of PSO-CIQ given that the p_{kmeans} is increased.

Table 7.6: Effect of the number of K-means iterations on the performance of PSO-CIQ using Lenna image (16 colors)	
No. of K-means iterations	MSE
5	212.627 ± 3.7
10	210.203 ± 1.487
50	208.791 ± 0.111

7.1.2.3 Comparison of *gbest*-, *lbest*- and *lbest-to-gbest*-PSO

In this section, the effect of different models of PSO is investigated using the Lenna image when quantized to 16 colors. A comparison is made between *gbest*-, *lbest*- and *lbest-to-gbest*-PSO (which has been used in the above experiments) using a swarm size of 20 particles. For *lbest*-PSO, a neighborhood size of $l = 2$ was used. Table 7.7 shows the result of the comparison. The results show no significant difference in performance.

Table 7.7: Comparison of <i>gbest</i> -, <i>lbest</i> - and <i>lbest-to-gbest</i> - PSO versions of PSO-CIQ using Lenna image (16 colors)	
	MSE
<i>gbest</i> PSO	209.841 ± 0.951
<i>lbest</i> PSO	210.366 ± 1.846
<i>lbest-to-gbest</i> PSO	210.203 ± 1.487

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

An end-member selection method for spectral unmixing that is based on PSO is developed in this section. 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.

7.2.1 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.

Measure of Quality

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

$$E = \sum_{j=1}^{N_b} \sqrt{MS_j} \quad (7.3)$$

where

$$MS = \frac{\sum_{p=1}^{N_p} (z_p - EM \cdot f)^2}{N_p}$$

where N_b is the number of spectral bands, N_p is the number of pixels in the image and $EM \cdot f$ is defined in equation (3.28).

The PSO-EMS Algorithm

In the context of spectral unmixing, a single particle represents N_m end-members. That is, each particle \mathbf{x}_i is constructed as $\mathbf{x}_i = (\mathbf{em}_{i,1}, \dots, \mathbf{em}_{i,k}, \dots, \mathbf{em}_{i,N_m})$ where $\mathbf{em}_{i,k}$ 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 equation 7.3) as follows:

$$f(\mathbf{x}_i) = E(\mathbf{x}_i) \quad (7.4)$$

The algorithm randomly initializes each particle from the multispectral image set to contain N_m end-members. The K-means clustering algorithm is then applied to each particle at a user-specified probability, p_{kmeans} . The K-means algorithm is used in order to refine the chosen end-members and to 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 (refer to section 3.4.2) 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 (2.8) and (2.10) 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.

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. an abundance map). The fractional concentration maps are then optimally mapped to an 8-bit integer format for display and storage purposes. This is done using the following non-linear mapping function [Saghri *et al.* 2000]:

$$\Omega = \frac{255(f^{\text{exp}} - (f_{\min})^{\text{exp}})}{(f_{\max})^{\text{exp}} - (f_{\min})^{\text{exp}}} + 0.5 \quad (7.5)$$

where:

Ω 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

exp is the floating-point exponent parameter in the range of $0 \leq \text{exp} \leq 1.0$. In this chapter, exp is set to 0.6 for the abundance images as suggested by Saghri *et al.* [2000].

The PSO-EMS algorithm is summarized Figure 7.6.

In general, the complexity of the PSO-EMS algorithm is $O(st_{\max}N_p)$. The parameters s and t_{\max} can be fixed in advance. Typically s and $t_{\max} \ll N_p$. 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.

1. Initialize each particle to contain N_m randomly selected end-members
2. For $t = 1$ to t_{\max}
 - (a) For each particle i
 - i. Apply K-means for a few iterations with a probability p_{kmeans} .
 - ii. For each pixel z_p
 - Find the N_e optimum end-member subset
 - Apply linear spectral unmixing using equation (3.28)
 - iii. Calculate the fitness, $f(x_i)$
 - (b) Find the personal best position for each particle and the global best solution, $\hat{y}(t)$
 - (c) Update the end-members using equations (2.8) and (2.10)
3. Generate the abundance images using the N_m end-members of particle $\hat{y}(t)$

Figure 7.6: The PSO-EMS algorithm

7.2.2 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: Figure 4.9 shows the four-channel multispectral test image set of the Lake Tahoe region in the US. Each channel is comprised of a 300×300 , 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 Saghri *et al.* [2002], a total of six end-members were obtained from the data set (i.e. $N_m = 6$).

NOAA's AVHRR: Figure 7.7 shows the five-channel multispectral test image set of an almost cloud-free territory of the entire United Kingdom (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 847×1009 , 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 Saghri *et al.* [2000], a total of eight end-members were obtained from the data set (i.e. $N_m = 8$).

The rest of this subsection is organized as follows: Section 7.2.2.1 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.* [2000] (discussed in section 3.4.2), which is referred to in this chapter as ISO-UNMIX. The time complexity of ISO-UNMIX is $O(N_p)$. Saghri *et al.* [2000] showed that ISO-

UNMIX performed very well compared to other popular spectral unmixing methods. Section 7.2.2.2 investigates the influence of the different PSO-EMS control parameters. Finally, the use of different PSO models (namely, *gbest*, *lbest* and *lbest-to-gbest*) was investigated in section 7.2.2.3.

The results reported in this section are averages and standard deviations over 10 simulations. Since *lbest-to-gbest* PSO was generally the best performer in chapter 4, *lbest-to-gbest* PSO is used in this section unless otherwise specified. The PSO-EMS parameters were initially set as follows: $p_{kmeans} = 0.1$, $s = 20$, $t_{max} = 100$, the number of K-means iterations is 10 (the effect of these values are then investigated), $w = 0.72$, $c_1 = c_2 = 1.49$ and $V_{max} = 255$ for the Lake Tahoe image set, while $V_{max} = 1023$ for the UK image set. These parameters are used in this section unless otherwise specified.

7.2.2.1 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 7.8 summarizes the results for the two image sets. The results are compared based on the RMS residual error defined in equation (7.3). The results showed that, for both image sets, PSO-EMS performed significantly better than the ISO-UNMIX in terms of the RMS residual error. Figures 7.8 and 7.9 show the abundance images generated from ISO-UNMIX and PSO-EMS, respectively, when applied to the Lake Tahoe image set. In addition, Figures 7.10 and 7.11 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.

Table 7.8: Comparison between ISO-UNMIX and PSO-EMS		
Image		RMS
LANDSAT 5 MSS	ISO_UNMIX	0.491837
	PSO-EMS	0.462197 ± 0.012074
NOAA's AVHRR	ISO_UNMIX	3.725979
	PSO-EMS	3.510287 ± 0.045442

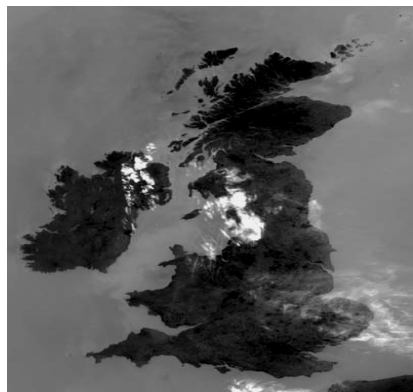
7.2.2.2 Influence of PSO-EMS Parameters

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

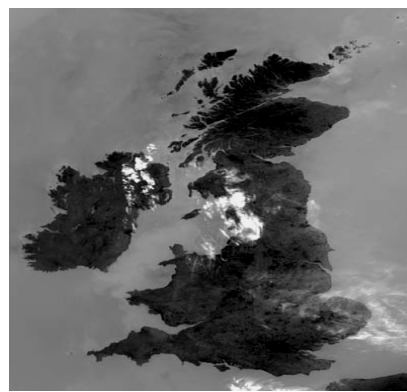
Velocity Clamping

Table 7.9 shows that using $V_{max} = 5$ or $V_{max} = 255$ generally produces comparable results. However, the standard deviation in the case of $V_{max} = 5$ is smaller than the standard deviation in the case of $V_{max} = 255$. Hence, using $V_{max} = 5$ generates more stable results than using $V_{max} = 255$.

Table 7.9: Effect of V_{max} on the performance of PSO-EMS using Lake Tahoe image set	
	RMS
$V_{max}=5$	0.469706 ± 0.000456
$V_{max}=255$	0.462197 ± 0.012074



(a) Band 1



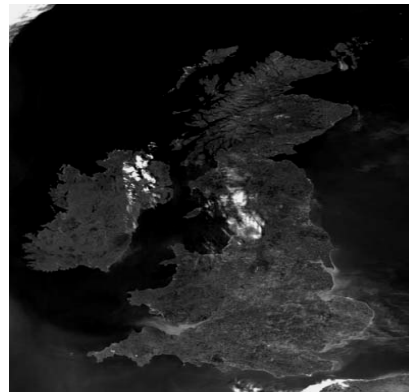
(b) Band 2



(c) Band 3



(d) Band 4



(e) Band 5

Figure 7.7: AVHRR Image of UK, Size: 847x1009 , 5 bands, 10-bits per pixel

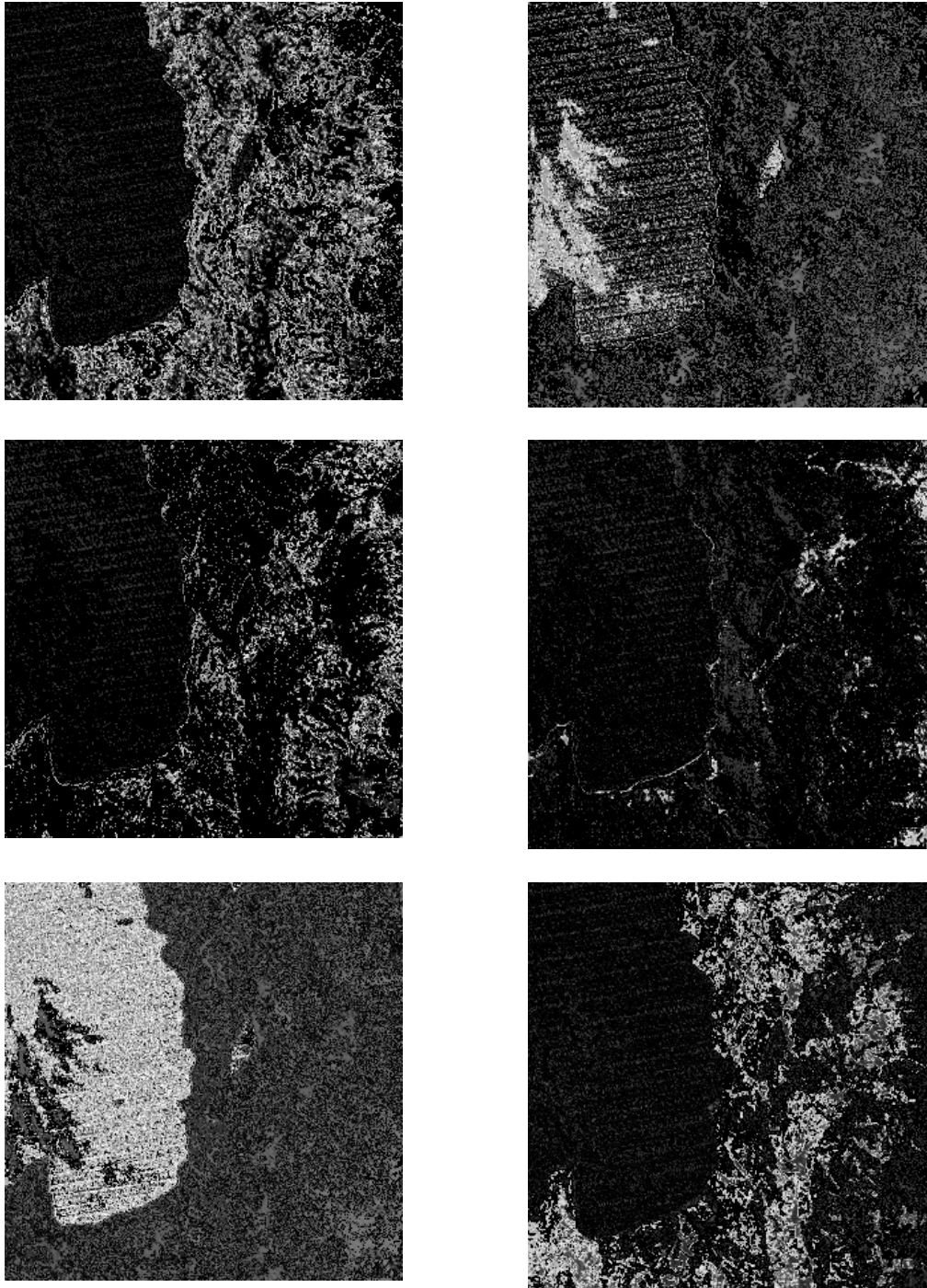


Figure 7.8: Species concentration maps resulting from the application of ISO-UNMIX to unmix the Lake Tahoe test image set

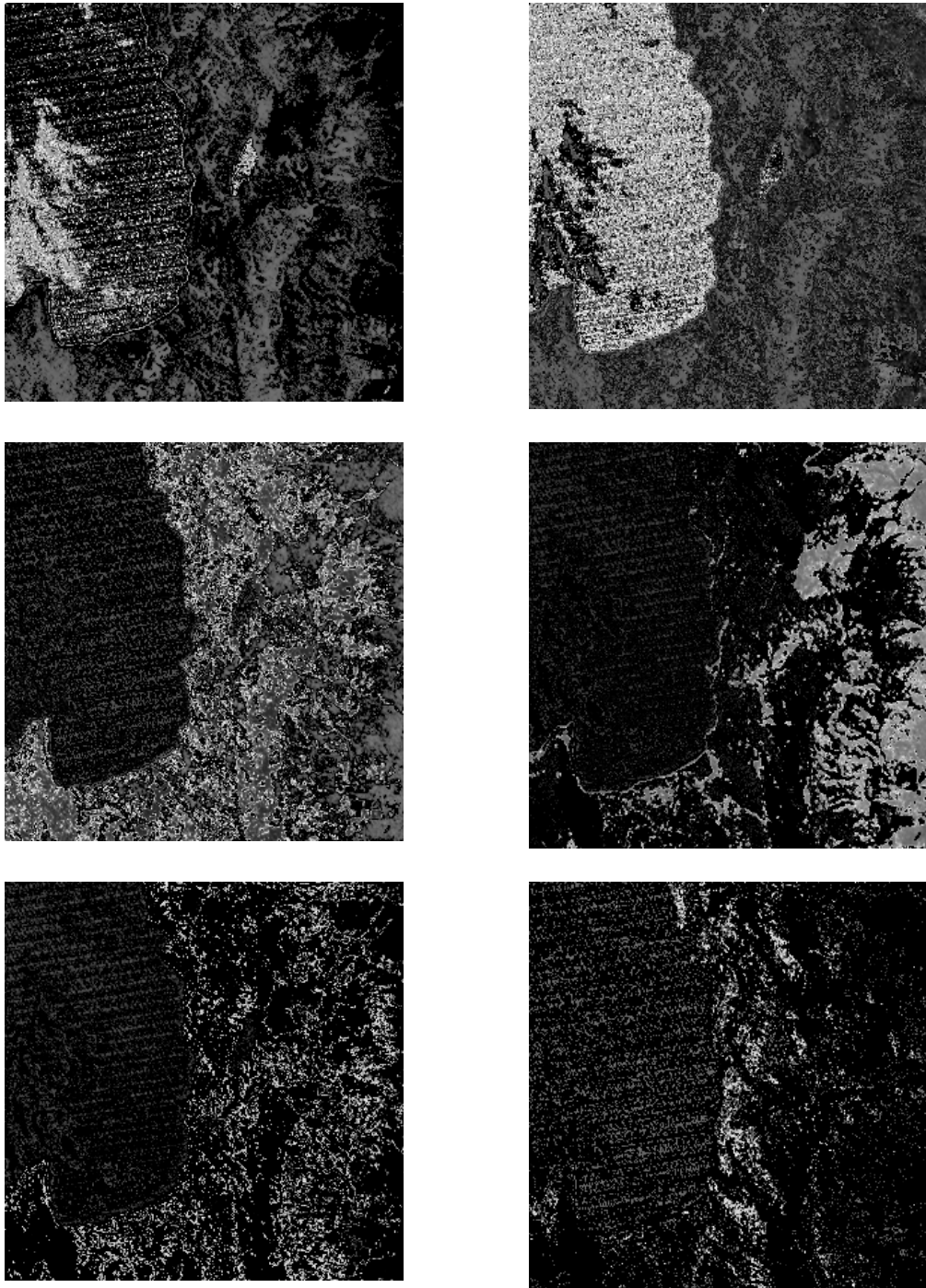


Figure 7.9: Species concentration maps resulting from the application of PSO-EMS to unmix the Lake Tahoe test image set

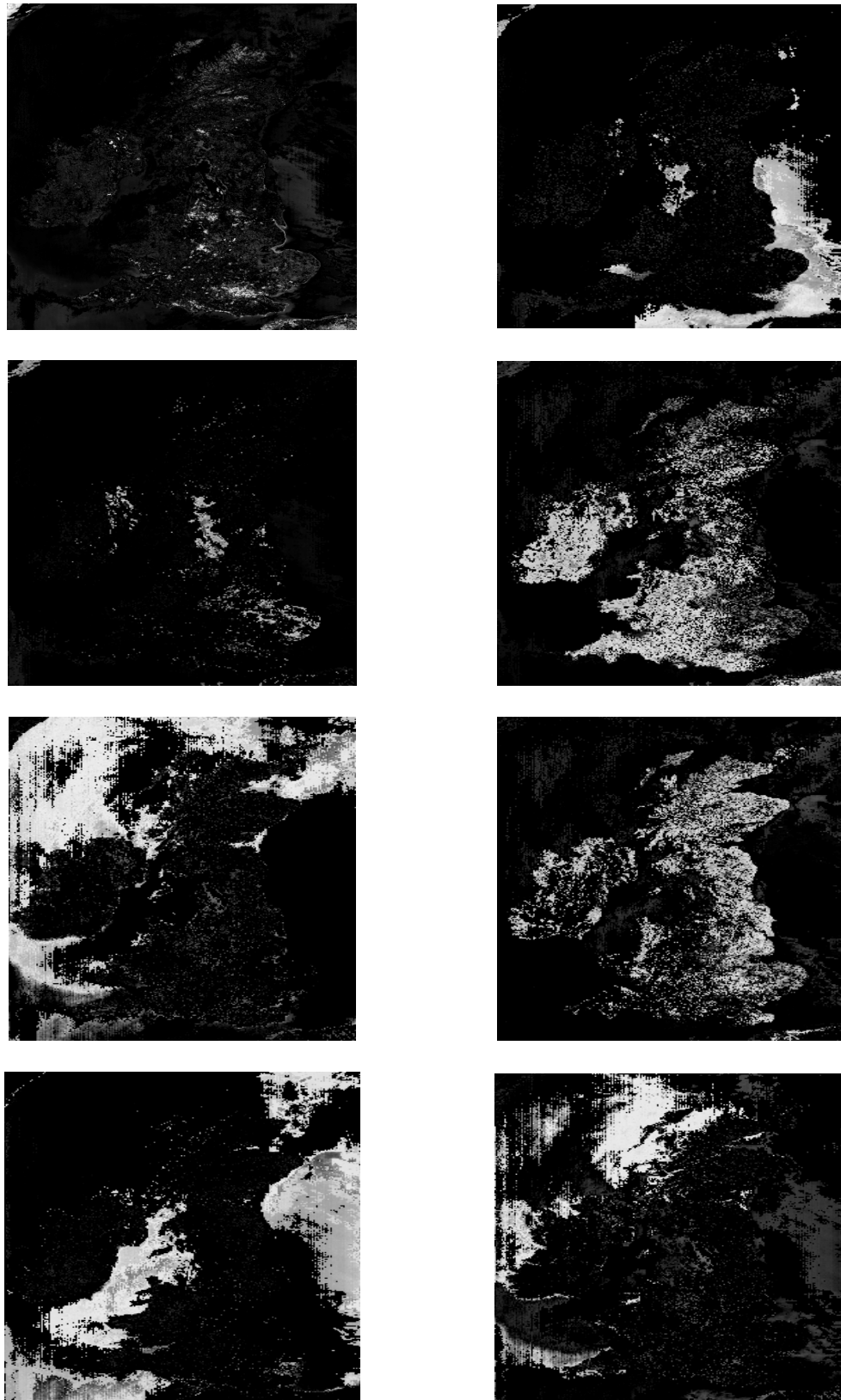


Figure 7.10: Species concentration maps resulting from the application of ISO-UNMIX to unmix the UK test image set

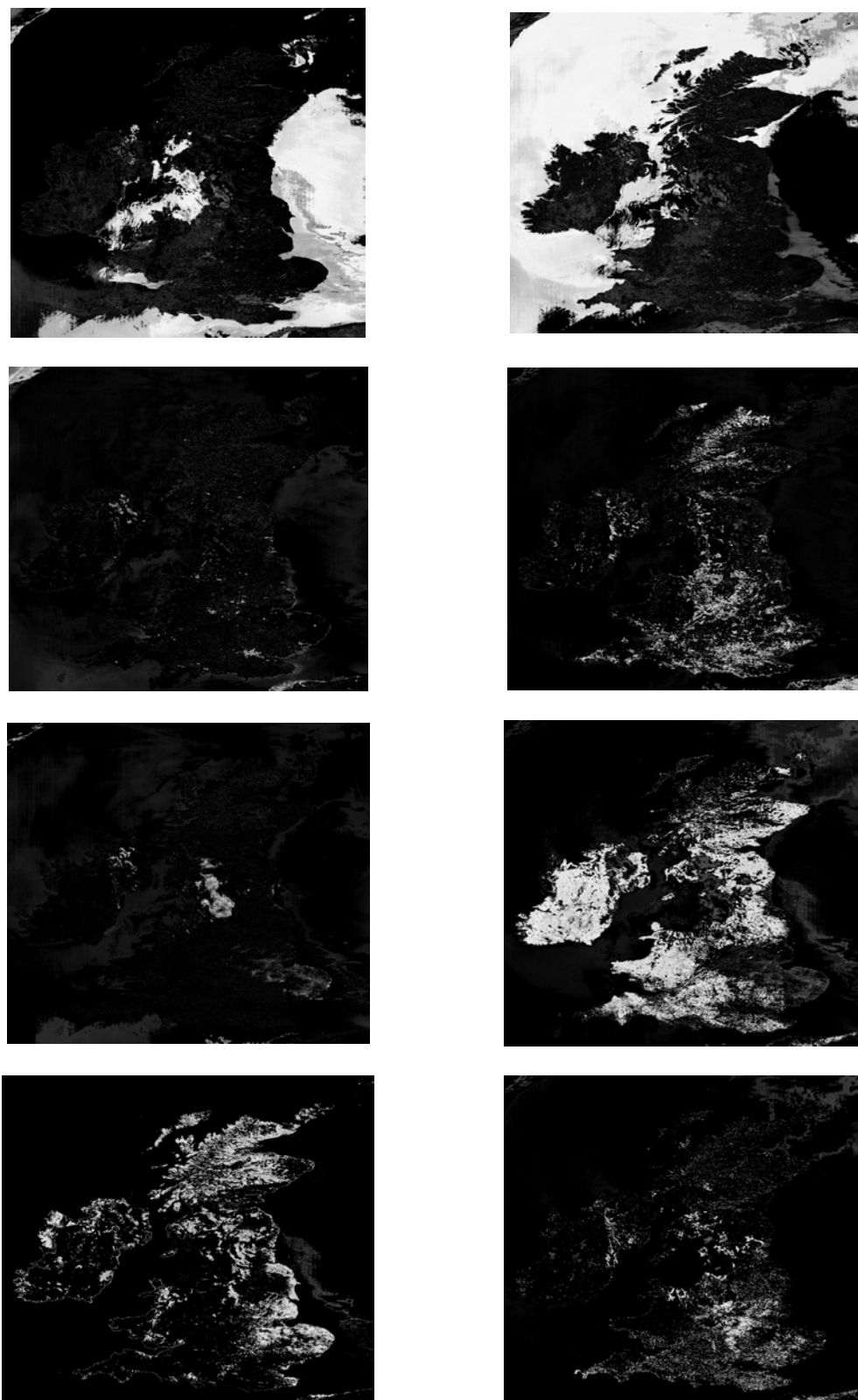


Figure 7.11: Species concentration maps resulting from the application of PSO-EMS to unmix the UK test image set

Swarm Size

Increasing the swarm size from 20 to 50 particles improves the performance of the PSO-EMS algorithm as shown in Table 7.10. 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.

Table 7.10: Effect of the swarm size on the performance of PSO-EMS using Lake Tahoe image set	
	RMS
$s = 10$	0.468706 ± 0.004753
$s = 20$	0.462197 ± 0.012074
$s = 50$	0.459195 ± 0.009389

Number of PSO iterations

Reducing the number of PSO iterations, t_{\max} , from 100 to 50 did not reduce the performance of the PSO-EMS algorithm as shown in Table 7.11. Similarly, increasing t_{\max} from 100 to 150, did not significantly improve the performance of the PSO-EMS.

Table 7.11: Effect of the number of PSO iterations on the performance of PSO-EMS using Lake Tahoe image set	
	RMS
$t_{\max} = 50$	0.468041 ± 0.004735
$t_{\max} = 100$	0.462197 ± 0.012074
$t_{\max} = 150$	0.465614 ± 0.00739

p_{kmeans}

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 7.12 which shows that increasing the value of p_{kmeans} significantly improves the performance of the PSO-EMS algorithm. However, as a trade-off, increasing the value of p_{kmeans} will increase the computational requirements of the PSO-EMS algorithm.

Table 7.12: Effect of p_{kmeans} on the performance of PSO-EMS using Lake Tahoe image set	
	RMS
$p_{kmeans} = 0.1$	0.462197 ± 0.012074
$p_{kmeans} = 0.25$	0.460776 ± 0.009120
$p_{kmeans} = 0.5$	0.454029 ± 0.007051
$p_{kmeans} = 0.9$	0.445367 ± 0.012339

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 7.13. 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 7.13. 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_{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 7.13. This result suggests that the number

of K-means iterations can be reduced without affecting the performance of PSO-EMS given that the p_{kmeans} is increased.

Table 7.13: Effect of the number of K-means iterations on the performance of PSO-EMS using Lake Tahoe image set	
No. of K-means iterations	RMS
5	0.468407 ± 0.004212
10	0.462197 ± 0.012074
50	0.466708 ± 0.004524

7.2.2.3 Comparison of *gbest*-, *lbest*- and *lbest-to-gbest*-PSO

In this section, the effect of different models of PSO is investigated using the Lake Tahoe image set. A comparison is made between *gbest*-, *lbest*- and *lbest-to-gbest*-PSO (which has been used in the above experiments) using a swarm size of 20 particles. For *lbest* PSO, a neighborhood size of $l = 2$ was used. Table 7.14 summarizes the result of the comparison. The results show no significant difference in performance. However, the standard deviation in the case of *lbest-to-gbest* PSO is the largest. Hence, using *lbest-to-gbest* PSO generates the least stable result.

Table 7.14: Comparison of <i>gbest</i> -, <i>lbest</i> - and <i>lbest-to-gbest</i> - PSO versions of PSO-EMS using Lake Tahoe image set	
	RMS
<i>gbest</i> PSO	0.465809 ± 0.006562
<i>lbest</i> PSO	0.465020 ± 0.004942
<i>lbest-to-gbest</i> PSO	0.462197 ± 0.012074

7.3 Conclusions

This chapter addressed two difficult problems in the field of pattern recognition and image processing. The two problems are color image quantization and spectral unmixing. First, the chapter presented a PSO-based color image quantization algorithm (PSO-CIQ). The PSO-CIQ algorithm was compared against other well-known color image quantization techniques. In general, the PSO-CIQ performed better than the other techniques when applied to a set of commonly used images. The effects of different PSO-CIQ control parameters were studied. The performance of different versions of PSO was then investigated.

The chapter then presented a new spectral unmixing approach using PSO (PSO-EMS). 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.* [2000]. 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.

From the results presented, it can be concluded that the PSO is an efficient optimization algorithm for difficult pattern recognition and image processing problems.