OBJECT EXTRACTION FROM LIDAR DATA USING AN ARTIFICIAL SWARM BEE COLONY CLUSTERING ALGORITHM

S. Saeedi [a] , F. Samadzadegan [b] , N. El-Sheimy [a]

[a] Department of Geomatics Engineering, University of Calgary, T2N 1N4, AB, Canada - (ssaeedi, elsheimy)@ucalgary.ca
[b] Department of Geomatics Engineering, Faculty of Engineering, University of Tehran, Iran - samadz@ut.ac.ir

Commission III, WG III/4

KEY WORDS: Clustering, LIDAR, Artificial Bee Colony, Urban Area, Object Extraction

ABSTRACT:

Light Detection and Ranging (LIDAR) systems have become a standard data collection technology for capturing object surface information and 3D modeling of urban areas. Although, LIDAR systems provide detailed valuable geometric information, they still require extensive interpretation of their data for object extraction and recognition to make it practical for mapping purposes. A fundamental step in the transformation of the LIDAR data into objects is the segmentation of LIDAR data through a clustering process. Nevertheless, due to scene complexity and the variety of objects in urban areas, e.g. buildings, roads, and trees, clustering using only one single cue will not reach meaningful results. The multi dimensionality nature of LIDAR data, e.g. laser range and intensity information in both first and last echo, allow the use of more information in the data clustering process and ultimately into the reconstruction scheme. Multi dimensionality nature of LIDAR data with a dense sampling interval in urban applications, provide a huge amount of valuable information. However, this amount of information produces a lot of problems for traditional clustering techniques. This paper describes the potential of an artificial swarm bee colony optimization algorithm to find global solutions to the clustering problem of multi dimensional LIDAR data in urban areas. The artificial bee colony algorithm performs neighborhood search combined with random search in a way that is reminiscent of the food foraging behavior of swarms of honey bees. Hence, by integrating the simplicity of the *k*-means algorithm with the capability of the artificial bee colony algorithm, a robust and efficient clustering method for object extraction from LIDAR data is presented in this paper. This algorithm successfully applied to different LIDAR data sets in different urban areas with different size and complexities.

1. INTRODUCTION

The need for rapidly generating high-density digital elevation data for areas of considerable spatial extent has been one of the main motives for the development of commercial airborne laser scanning systems. During the last decade, several clustering and filtering techniques have been developed for the extraction of 3D objects for city modelling applications or removing the “artefacts” of bare terrain (i.e. Buildings and trees) in order to obtain the true Digital Elevation Model (Filin and Pfeifer; 2006; Kraus and Pfeifer, 1998; Lodha et al., 2007; Rottensteiner and Briese, 2002; Tóvári and Vögtle, 2004).

However due to low information content and resolution of available commercial LIDAR scanners, it is difficult to correctly recognize and remove 3D objects exclusively from LIDAR range data in urban areas (Maas, 2001; Samadzadegan, 2004; Tao and Hu, 2001; Vosselman et al., 2004).

In order to improve the performance of 3D object extraction process, additional data should be considered. Most LIDAR systems register, at least, two echoes of the laser beam, the first and the last echo, which generally correspond to the highest and the lowest object point hit by the laser beam. First and last echo data will especially differ in the presence of vegetation (Kraus, 2002). Moreover, LIDAR systems record the intensity of the returned laser beam which is mainly in the infrared part of the electromagnetic spectrum. In addition, an extra powerful source of information is visible image. Digital images can provide additional information through their intensity and spectral content as well as their high spatial resolution which is better than the resolution of laser scanner data.

Therefore, in the context of 3D object extraction in urban areas, various type of information can be fused to overcome the difficulties of classification and identification of complicated objects (Lim and Suter, 2007; Vosselman et al., 2004). Collecting this information, extremely enlarge the size of data sets and proportionally the dimension of feature spaces in clustering process. As a result, most of traditional clustering techniques that have been applied with standard data and low feature space dimension are not efficient enough for object extraction process from LIDAR data (Melzer, 2007; Lodha et al., 2007).

k-means is one of the most popular clustering algorithms for handling massive datasets. The algorithm is efficient at clustering large data sets because its computational complexity only grows linearly with the number of data points (Kotsiantis and Pintelas, 2004). However, the algorithm may converge to solutions that are not optimal. This paper presents an artificial bee colony (ABC) clustering algorithm for overcoming the existing problems of traditional *k*-means.

2. BASIC CONCEPTS IN DATA CLUSTERING

Historically, the notion of finding useful patterns in data has been given a variety of names including data clustering, data mining, knowledge discovery, pattern recognition, information extraction, etc (Ajith et al., 2006). Data clustering is an analytic process designed to explore data by discovering of consistent patterns and/or systematic relationships between variables, and then to validate the findings by applying the detected patterns to new subsets of data.

Data clustering is a difficult problem in unsupervised pattern recognition as the clusters in data may have different shapes and sizes. In the background of clustering techniques, the following terms are used in this paper (Jain et al., 1999):

- A pattern (or feature vector), z, is a single object or data point used by the clustering algorithm.
- A feature (attribute) is an individual component of a pattern.
- A cluster is a set of similar patterns, and patterns from different clusters are not similar.
- A distance measure is a metric used to evaluate the similarity of patterns.

The clustering problem can be formally defined as follows (Jain et al., 1999): Given a data set $Z=\{z_1,z_2, . . . ,z_p, . . . ,z_{Np}\}$ where z_p is a pattern in the N_d-dimensional feature space, and N_p is the number of patterns in *Z*, then the clustering of *Z* is the partitioning of *Z* into *K* clusters $\{C_1,C_2, . . . ,C_K\}$ satisfying the following conditions:

- Each pattern should be assigned to a cluster, i.e.
$$\cup_{j=1}^{k} C_j = Z$$
- Each cluster has at least one pattern assigned to it, i.e.
$$C_k \neq 0, \qquad k = 1, \dots, K$$
- Each pattern is assigned to one and only one cluster
$$C_k \cap C_j = 0, where\ k \neq j$$

As previously mentioned, clustering is the process of identifying natural groupings or clusters within multidimensional data based on feature space through similarity measure. Hence, similarity measures are fundamental components in most clustering algorithms (Jain et al., 1999). The most popular way to evaluate a similarity measure is the use of distance measures. The most widely used distance measure is the Euclidean distance, defined as:

$$d(z_i, z_j) = \sqrt{\sum_{k=1}^{N_d} (z_{i,k} - z_{j,k})^2} = \|z_{i-}z_j\|_2 \qquad (1)$$

Generally, clustering algorithms can be categorized into partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods. An excellent survey of clustering techniques can be found in (Kotsiantis and Pintelas, 2004). In this paper, the focus will be on the partitional clustering algorithms. Partitional clustering algorithms divide the data set into a specified number of clusters and then evaluate them by some criteria. These algorithms try to minimize certain criteria (e.g. a square error function) and can therefore be treated as optimization problems (Harvey et al., 2002; Omran et al., 2005; Wilson et al., 2002).

The most widely used partitional algorithm in clustering techniques is the iterative *k*-means approach (Kotsiantis and Pintelas, 2004). The objective function *J* that the *k*-means optimizes is:

$$J_{K-means} = \sum_{j=1}^{K} \sum_{\forall z_p \in C_k} d^2 (z_p, m_k) \qquad (2)$$

Where m_k is the centroid of the *k*-th cluster. The membership and weight functions *u* for *k*-means are defined as:

$$u(m_k|z_p) = \begin{cases} 1 & if\ d^2(z_p, m_k) = \arg min_k\{d^2(z_p, m_k)\} \\ 0 & otherwise \end{cases} \qquad (3)$$

Consequently, the k-means method minimizes the intra-cluster distance. The *k*-means algorithm starts with *k* centroids (initial values are randomly selected or derived from a priori information). Then, each pattern z_p in the data set is assigned to the closest cluster (i.e. closest centroid). Finally, the centroids are recalculated according to the associated patterns. This procedure is repeated until convergence is achieved.

It is known that the *k*-means algorithm may reach local optimal solutions, depending on the choice of the initial cluster centres. Genetic algorithms have a potentially greater ability to avoid local optima through the localised search employed by most clustering techniques. Maulik and Bandyopadhyay (2004) proposed a genetic algorithm-based clustering technique, called GA-clustering, that proven to be effective in optimal clusters. With this algorithm, solutions (typically, cluster centroids) are represented by bit strings. The search for an appropriate solution begins with a population, or collection, of initial solutions. Members of the current population are used to create the next generation population by applying operations such as random mutation and crossover. At each step, the solutions in the current population are evaluated relative to some measures of fitness (which, typically, is inversely proportional to *d*), with the fittest solutions selected probabilistically as seeds for producing the next generation. The process performs a generate-and-test beam search of the solution space, in which variants of the best current solutions are most likely to be considered next. In the next section, an alternative clustering method to solve the local optimum problem of the *k*-means algorithm is described. The applied method adopts the artificial swarm bees algorithm as it has proved to give a more robust performance than other intelligent optimisation methods for a range of complex problems (Pham, 2006).

3. CLUSTERING OF LIDAR DATA USING SWARM ARTIFICIAL BEE COLONY ALGORITHM

Swarm Intelligence (SI) is an innovative distributed intelligent paradigm for solving optimization problems that originally took its inspiration from the biological examples by swarming, flocking and herding phenomena. These techniques incorporate swarming behaviours observed in flocks of birds, schools of fish, or swarms of bees, and even human social behaviour, from which the idea is emerged (Omran et al., 2002, 2005; Paterlini and Krink, 2005; Pham et al., 2006; Wu and Shi, 2001). Data clustering and swarm intelligence may seem that they do not have many properties in common. However, recent studies suggest that they can be used together for several real world data clustering and mining problems especially when other methods would be too expensive or difficult to implement.

Clustering approaches inspired by the collective behaviours of ants have been proposed by Wu and Shi (2001), Labroche et al. (2001). The main idea of these approaches is that artificial ants are used to pick up items and drop them near similar items resulting in the formation of clusters. Omran et al. (2002) proposed particle swarm optimization (PSO) clustering algorithm. The results of Omran et al. (2002, 2005) show that PSO outperformed *k*-means, fuzzy *c*-means (FCM) and other state-of-the-art clustering algorithms. More recently, Paterlini and Krink (2005) compared the performance of *k*-means, genetic algorithm (GA), PSO and Differential Evolution (DE) for a representative point evaluation approach to partitional clustering. The results show that GAs, PSO and DE outperformed the *k*-means algorithm. Pham et al. (2006) used the artificial bee colony algorithm for clustering of different data sets. The obtained results of their work show that their proposed artificial bee colony algorithm has better performance than both of standard *k*-means as well as GA-based method. In general, the literature review of recent techniques in clustering shows that the swarm-based clustering algorithm performs better than the *k*-means algorithm. Clustering of massive LIDAR data and the unique potential of artificial bee colony algorithm in solving complex optimization problems are the core of this paper. The research work presented in this paper clearly show that the artificial swarm bee colony algorithm has clearly outperform *k*-means method in clustering of LIDAR data.

3.1 Artificial Bee Colony Algorithm

A colony of honey bees can extend itself over long distances in order to exploit a large number of food sources (Camazine et al., 2003; Pham et al., 2006) . The foraging process begins in a colony by scout bees being sent to search for promising flower patches. Flower patches with large amounts of nectar or pollen that can be collected with less effort tend to be visited by more bees, whereas patches with less nectar or pollen receive fewer bees (Camazine et al., 2003).

In the artificial bee algorithms, a food source position represents a possible solution to the problem to be optimized. Therefore, at the initialization step, a set of food source positions are randomly produced and also the values of control parameters of the algorithm are assigned. The nectar mount of a food source corresponds to the quality of the solution represented by that source. So the nectar amounts of the food sources existing at the initial positions are determined. In other words, the quality values of the initial solutions are calculated.

Each employed bee is moved onto her food source area for determining a new food source within the neighbourhood of the present one, and then its nectar amount is evaluated. If the nectar amount of the new one is higher, then the bee forgets the previous one and memorizes the new one. After the employed bees complete their search, they come back into the hive and share their information about the nectar amounts of their sources with the onlookers waiting on the dance area. All onlookers successively determine a food source area with a probability based on their nectar amounts. If the nectar amount of a food source is much higher when compared with other food sources, it means that this source will be chosen by most of the onlookers. This process is similar to the natural selection process in evolutionary algorithms. Each onlooker determines a neighbour food source within the neighbourhood of the one to which she has been assigned and then its nectar amount is evaluated.

3.2 Artificial Swarm Bee Colony Clustering Method

The artificial swarm bee colony clustering method exploits the search capability of the Bees Algorithm to overcome the local optimum problem of the *k*-means algorithm. More specifically, the task is to search for appropriate cluster centres ($c_1, c_2, \ldots, c_k$) such that the clustering metric *d* (equation 1) is minimised. The basic steps of this clustering operation are:

```
1. Initialise the solution population.
2. Evaluate the fitness of the population.
3. While (stopping criterion is not met)
   a. Form new population.
   b. Select sites for neighbourhood search by means of
      information in the neighbourhood of the present one.
   c. Recruit bees for selected sites (more bees for the best
      e sites) and evaluate fitness values.
   d. Select the fittest bee from each site.
   e. Assign remaining bees to search randomly and
      evaluate their fitness values.
End While.
```

Each bee represents a potential clustering solution as set of *k* cluster centres and each site represent the patterns or data objects. The algorithm requires some parameters to be set, namely: number of scout bees (*n*), number of sites selected for neighbourhood searching (*m*), number of top-rated (*elite*) sites among *m* selected sites (*e*), number of bees recruited for the best *e* sites (*nep*), number of bees recruited for the other (*me*) selected sites (*nsp*), and the stopping criterion for the loop.

At the initialization stage, a set of scout bee population (*n*) are randomly selected to define the *k* clusters. The Euclidean distances between each data pattern and all centres are calculated to determine the cluster to which the data pattern belongs. In this way, initial clusters can be constructed. After the clusters have been formed, the original cluster centres are replaced by the actual centroids of the clusters to define a particular clustering solution (i.e. a bee). This initialization process is applied each time new bees are to be created.

In step 2, the fitness computation process is carried out for each site visited by a bee by calculating the clustering metric *d* (equation 1) which is inversely related to fitness. Step 3, is the main step of bee colony optimization, which start by forming new population (step 3-a). In step 3-b, the *m* sites with the highest fitness are designated as “selected sites” and chosen for neighbourhood search. In steps 3-c and 3-d, the algorithm conducts searches around the selected sites, assigning more bees to search in the vicinity of the best *e* sites. Selection of the best sites can be made directly according to the fitness associated with them. Alternatively, the fitness values are used to determine the probability of the sites being selected. Searches in the neighbourhood of the best e sites - those which represent the most promising solutions - are made more detailed. As already mentioned, this is done by recruiting more bees for the best *e* sites than for the other selected sites. Together with scouting, this differential recruitment is a key operation of the bee algorithm. In step 3-d, only the bee that has found the site with the highest fitness (the “fittest” bee) will be selected to form part of the next bee population. In nature, there is no such a restriction. This restriction is introduced here to reduce the number of points to be explored. In step 3-e, the remaining bees in the population are assigned randomly around the search space to scout for new potential solutions. At the end of each loop, the colony will have two stages to its new population: representatives from the selected sites, and scout bees assigned to conduct random searches. These steps are repeated until a stopping criterion is met.

4. EXPERIMENTAL INVESTIGATIONS

The airborne LIDAR data used in the experimental investigations have been recorded with TopScan's Airborne Laser Terrain Mapper system ALTM 1225 (TopScan, 2004). The data are recorded in area of Rheine in Germany. Two different patches with residential and industrial pattern were selected to test the developed algorithm. The selected areas were suitable for the evaluation of the proposed classification strategy because the required complexities (e.g. proximities of different objects e.g. building and tree) were available in the image (figure 1-a, b). The pixel size of the range images is one meter. This reflects the average density of the irregularly recorded 3D points which is fairly close to one point per m^2. Intensity images for the first and last echo data have been also recorded and the intention was to use them in the experimental investigations, Figure 1 shows the details of the test data. The impact of trees in the first and last echo images can be easily recognized by comparing the two images of this figure.

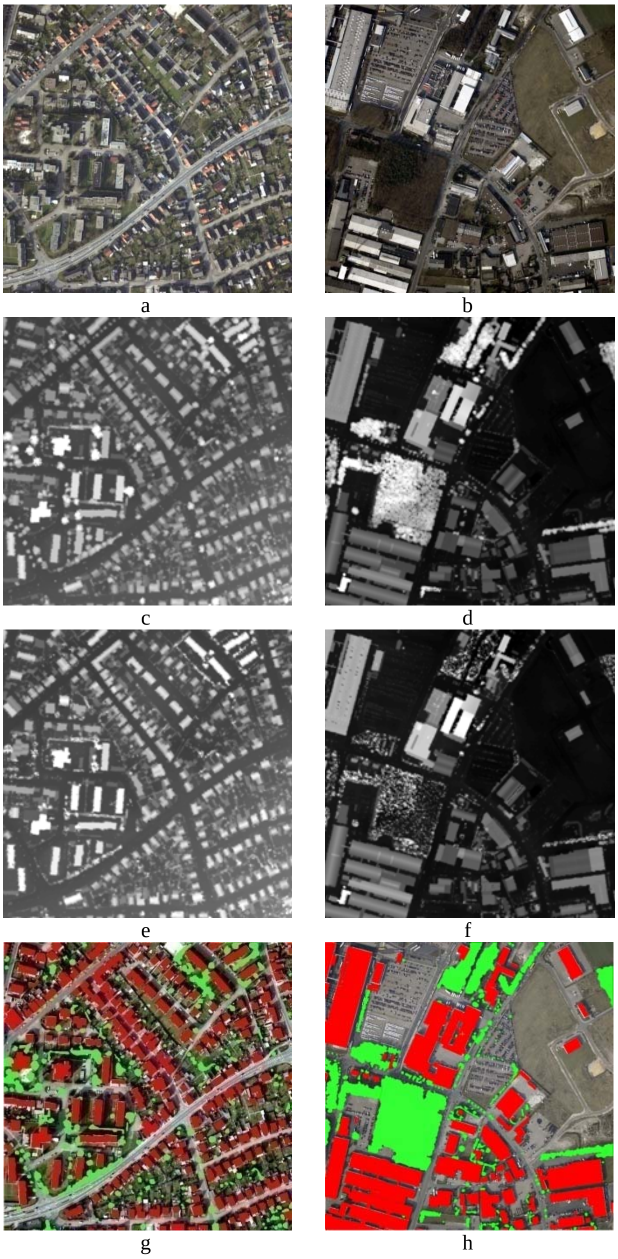

Figure 1. a) Aerial image of residential area. b) Aerial image of industrial area. c) First echo LIDAR range data of residential area. d) First echo LIDAR range data of industrial area. e) Last echo LIDAR range data of residential area. f) Last echo LIDAR range data of industrial area. g) Overlaid of manually digitized objects in residential area; h) Overlaid of manually digitized objects in residential area

The first step in every clustering process is to extract the feature image bands. The features of theses feature bands should carry useful textural or surface related information to differentiate between regions related to the surface. Several features have been proposed for clustering of range data. Axelsson (1999) employs the second derivatives to find textural variations and Maas (1999) utilizes a feature vector including the original height data, the Laplace operator, maximum slope measures and others in order to classify the data. In the following experiments we used five types of features:

- LIDAR range data
- The difference between first and last echo range images
- Top-Hat filtered last echo range image

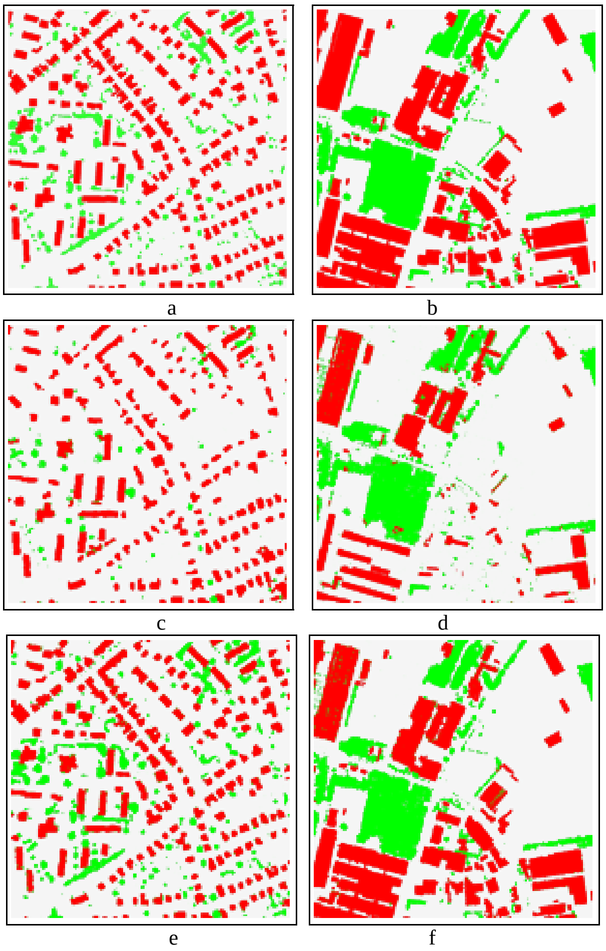

Figure 2. a) Manually digitized objects in residential area. b) Manually digitized objects in industrial area. c) Clustering results of *k*-means in residential area. d) Clustering results of *k*-means in industrial area. e) Clustering results of artificial swarm bee colony algorithm in residential area. f) Clustering results of swarm bee algorithm in industrial area.

- Local height variation which is computed using a small window (3*3) around a data sample.
- Last echo intensity

The normalized difference of the first and last echo range images is used as the major feature band for discrimination of the vegetation pixels from the others. According to the above defined features, the *k*-means and artificial swarm bee algorithm were developed based on the parameters listed in table 1.

Table 1. Parameters used in the clustering of LIDAR datasets

Algorithm	Parameters	Value
***k*-means**	Maximum number of iterations	1000
Artificial swarm bee colony algorithm	Number of scout bees, *n*	35
	Number of sites selected for neighbourhood search, *m*	11
	Number of best “elite” sites out of *m* selected sites, *e*	2
	Number of bees recruited for best *e* sites, *nep*	7
	Number of bees recruited for the other (m-e) selected sites, *nsp*	3
	Number of iterations, *R*	200

Evaluation of these two algorithms for clustering of the data sets into three clusters (ground, tree, and building) is depicted in figure 2. Figures 2c and 2d show the *k*-means clustering results and figures 2e and 2f show the artificial bee colony algorithm clustering results in two evaluation areas. Building class regions are highlighted by red and vegetation class regions by green colour in figure 2. Visual inspections shows that vegetation class is directly associated with trees, bushes or forest and the building class is mainly associated with building regions.

4.1 Accuracy Assessment

Comparative studies on clustering algorithms are difficult due to lack of universally agreed upon quantitative performance evaluation measures. Many similar works in clustering use the classification error as the final quality measurement (Zhong and Ghosh, 2003); so in this research, we adopt a similar approach. In this paper, confusion matrix used to evaluate the true labels and the labels returned by the clustering algorithms as the quality assessment measure. If some ground truth is available, the relation between the "true" classes and the classification result can be quantified. With the clusters the same principle can be applied. Mostly a much higher number of clusters is then related to the given ground truth classes to examine the quality of the clustering algorithm. From the confusion matrix we calculate the *Kappa Coefficient* (Cohen, 1960). Although the accuracy measurements described above, namely, the overall accuracy, producer's accuracy, and user's accuracy, are quite simple to use, they are based on either the principal diagonal, columns, or rows of the confusion matrix only, which does not use the complete information from the confusion matrix. A multivariate index called the Kappa coefficient (Tso and Mather, 2009) overcomes these limitations. The Kappa coefficient uses all of the information in the confusion matrix in order for the chance allocation of labels to be taken into consideration. The Kappa coefficient is defined by:

$$\hat{k} = \frac{N\sum_{i=1}^{r} x_{ii} - \sum_{i=1}^{r}(x_{i+} \times x_{+i})}{N^2 - \sum_{i=1}^{r}(x_{i+} \times x_{+i})} \qquad (4)$$

In this equation, $\hat{k}$ is the kappa coefficient, r is the number of columns (and rows) in a confusion matrix, x_{ii} is entry *(i, i)* of the confusion matrix, x_{i+} and x_{+i} are the marginal totals of row *i* and column *j*, respectively, and *N* is the total number of observations (Tso and Mather, 2009).

Table 2 shows the confusion matrix and Kappa coefficient of *k*-means and artificial swarm bee colony algorithms clustering in residential dataset. The confusion matrix and Kappa coefficient of *k*-means and artificial swarm bee colony algorithms clustering in industrial dataset presented in Table 3.

By comparing the counts in each class, a striking difference to the artificial swarm bee colony algorithm result is clearly observed. For the two classes of major interest in this study, the building class and tree class, the differences are quite significant. Visual interpretation clearly indicates that the building class of k-means not only include building areas but also regions related to roads which supports the smaller number of counts of the artificial swarm bee colony method to be more precise. Similarly the higher number of counts for the tree class indication (3D) vegetation regions (trees, bushes) obtained with the artificial swarm bee colony algorithm method is supported by visual interpretation. Overall performance of artificial bee colony algorithm is outperforming k-means clustering algorithm. This can be observed from the Kapa coefficient.

Table 2. Confusion matrix and Kappa coefficient of *k*-means and artificial swarm bee colony algorithms in residential area.

		Reference Data			
		Building	Tree	Ground	Total
k-means	Building	64338	1551	338	66227
	Tree	3561	58692	5930	68183
	Ground	54341	10509	290740	355590
	Total	122240	70752	297008	490000
	Kappa coefficient = 0.6927				
		Reference Data			
		Building	Tree	Ground	Total
Bee algorithms	Building	114602	3471	5686	123759
	Tree	2124	61123	6144	69391
	Ground	4214	7558	285078	296850
	Total	120940	72152	296908	490000
	Kappa coefficient = 0.8916				

Table 3. Confusion matrix and Kappa coefficient of *k*-means and artificial swarm bee colony algorithms in industrial area.

		Reference Data			
		Building	Tree	Ground	Total
k-means	Building	26878	2168	1108	30154
	Tree	187	3707	105	3999
	Ground	16443	12879	139025	168347
	Total	43508	18754	140238	202500
	Kappa coefficient = 0.584				
		Reference Data			
		Building	Tree	Ground	Total
Bee algorithms	Building	39528	1158	2097	42783
	Tree	839	15641	1290	17770
	Ground	3842	3483	134622	141947
	Total	44209	20282	138009	202500
	Kappa coefficient = 0.866				

5. CONCLUSION

This paper presented and tested a new clustering method based on the artificial bee colony algorithm in extracting buildings and trees form LIDAR data. The method employs the artificial swarm bee colony algorithm to search for the set of cluster centres that minimizes a given clustering metric. One of the advantages of this method is that it does not become trapped at locally optimal solutions. This is due to the ability of the artificial swarm bee colony algorithm to perform local and global search simultaneously. Experimental results for different LIDAR data sets have demonstrated that the artificial swarm bee colony algorithm method produces better performances than those of the *k*-means algorithm. One of the drawbacks of the artificial artificial swarm bee colony algorithm, however, is the number of tunable parameters it employs.

6. ACKNOWLDGMNT

The authors would like to thank Dr. Michael Hahn from Stuttgart University of Applied Sciences for providing the data set used in the paper.

7. REFERENCES

Ajith, A., Crina G., Vitorino R., (Eds.) (2006). Swarm Intelligence in Data Mining, Studies in Computational Intelligence (series), Vol. 34, Springer-Verlag, ISBN: 3-540-34955-3, 267 p., Hardcover.

Axelsson, 1999 P. Axelsson (1999), Processing of laser scanner data-algorithms and applications, ISPRS Journal of Photogrammetry and RS 54 (2–3), pp. 138–147. Article | PDF (1269 K) | View Record in Scopus | Cited By in Scopus (113)

Camazine, S., Deneubourg, J., Franks, N.R., Sneyd, J., Theraula, G. and Bonabeau, E. (2003). Self-Organization in Biological Systems, (Princeton University Press, Princeton).

Filin, S. and Pfeifer, N., (2006). Segmentation of airborne laser scanning data using a slope adaptive neighborhood. *ISPRS Journal of Photogrammetry and Remote Sensing* 60(2): 71-80.

Harvey N. R., J. Theiler, S. P. Brumby, S. Perkins, J. J. Szymanski, J. J. Bloch, R. B. porter, M. Galassi, A. C. Young, (2002). “Comparison of GENIE and conventional supervised classifiers for multispectral image feature extraction,” IEEE Trans. on Geoscience and RS, vol. 40, no. 2, pp. 393-404.

Jain, A.K. , M.N. Murty, P.J. Flynn, (1999). Data Clustering: A Review, ACM Computing Surveys, Vol. 31, No. 3, Sep. 1999.

Kotsiantis S. and Pintelas P., (2004). Recent advances in clustering: a brief survey, WSEAS Transactions on Information Science and Applications 1:73-81.

Kraus and Pfeifer, (1998). Determination of terrain models in wooded areas with ALS data. *ISPRS Journal of Photogrammetry and Remote Sensing* **53** 4 (1998), pp. 193–20.

Kraus, K., (2002). Principles of airborne laser scanning. Journal of the Swedish Society for Photogrammetry and Remote Sensing 2002:1 53–56.

Labroche N, Monmarche N, and Venturini G., (2002). Visual Clustering based on the Chemical Recognition System on Ants. In Proceedings of the European Conf. on AI, 2002.

Lim, E. H. and Suter, D., (2007). Conditional Random Field for 3D Point Clouds with Adaptive Data Reduction. In Proceedings of the 2007 international Conference on Cyberworlds (October 24 - 26, 2007). International Conference on Cyberworlds. IEEE Computer Society, Washington, DC, 404-408. DOI= http://dx.doi.org/10.1109/CW.2007.24

Lodha, S. K., Fitzpatrick, D. M., and Helmbold, D. P., (2007). Aerial Lidar Data Classification using AdaBoost. In Proceedings of the Sixth international Conference on 3-D Digital Imaging and Modeling (August 21 - 23, 2007). 3DIM. IEEE Computer Society, Washington, DC, 435-442. DOI= http://dx.doi.org/10.1109/3DIM.2007.10

Maas, H.-G., (1999). Close solutions for the determination of parametric house models from invariant moments of airborne laserscanner data, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 32(3-2W5): 193-199.

Maas, H.-G., (2001). The suitability of Airborne Laser Scanner Data for Automatic 3D Object Reconstruction, Third Int. Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images, Ascona, Switzerland.

Maulik U. and Bandyopadhyay S., (2004). "Genetic Algorithm Based Clustering Technique", Pattern Recognition, vol. 33, no. 9, pp. 1455-1465, 2004.

Melzer T., (2007); “Non-parametric segmentation of ALS point clouds using mean shift”; Journal of Applied Geodesy. Volume 1, Issue 3, Pages 159–170, ISSN (Online) 1862-9024, ISSN (Print) 1862-9016, DOI: 10.1515/jag.2007.018

Omran M, Salman A, and Engelbrecht AP., (2002). Image Classification using Particle Swarm Optimization. In Conf. on Simulated Evolution and Learning, vol. 1, pp 370–374, 2002.

Omran M., A. Engelbrecht and A. Salman, (2005). Differential evolution methods for unsupervised image classification, In Proc. of the IEEE Cong. on Evolutionary Computation (CEC2005) 2, 966-973, September 2005.

Paterlini S and Krink T., (2006). Differential Evolution and PSO in Partitional Clustering. Computational Statistics and Data Analysis, 50(2006):1220–1247, 2005.

Pham, D.T., Ghanbarzadeh, A., Koç, E., Otri , S., Rahim , S. and Zaidi, M., (2006). The Bees Algorithm – A novel tool for complex optimisation problems. In: Proc. of the 2nd Virtual International Conference on Intelligent Production Machines and Systems (I*PROMS-06), Cardiff, UK, 2006, 454-459.

Rottensteiner, F., Briese, Ch., (2002). "A new method for building extraction in urban areas from high-resolution LIDAR data", International Archives of Photogrammetry and Remote Sensing, Vol. 34, Part 3A, Graz, Austria

Samadzadegan, F., (2004) Object extraction and recognition from LIDAR data based on fuzzy reasoning and information fusion techniques, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXV-B3, Istanbul, Turkey

Tao, C., Hu, Y., (2001). A review of post-processing algorithms for airborne LiDAR data. CD-ROM Proceedings of ASPRS Annual Conference, April 23–27, St. Louis, USA.

Tóvári, D., Vögtle, T., (2004). Classification Methods for 3D Objects in Laserscanning Data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXV-B3, Istanbul, Turkey

Vosselman, G., B. Gorte, G. Sithole and Rabbani, T., (2004). Recognising Structure in Laser Scanner Point Clouds, International Conference NATSCAN “Laser-Scanners for Fores and Landscpae Assessment – Instruments, Processing Methods and Applications, ISPRS working group VIII/2, Freiburg im Breisgau, Germany.

Wilson H. G., B. Boots, A.A. Millward, (2004). A comparison of hierarchical and partitional clustering techniques for multispectral image classification, in Proceedings of the International Geoscience and Remote Sensing Symposium, pp. 1624-1626, Toronto, Canada, 2002.

Wu B and Shi Z., (2001). “A Clustering Algorithm based on Swarm Intelligence”; In Proceedings of the International Conference on Info-tech and Info-net, pages 58–66, 2001.

Zhong S. and J. Ghosh, (2003). A comparative study of generative models for document clustering; in SIAM Int. Conf. Data Mining Workshop on Clustering High Dimensional Data and Its Applications, San Francisco, CA, May 2003.