THE EFFICIENCY OF THE ADAPTED AntClust ALGORITHM FOR SATELITE IMAGES CLUSTERING

Ahmed BENYAMINA[1] and Hadria FIZAZI[2]
[1]Faculty of Sciences and Technology, Bechar University – Algeria
[2]Faculty of Sciences and Technology, USTO University – Algeria
[1]benyaminaa@yahoo.com

ABSTRACT

This paper presents a novel algorithm for images satellite clustering using an adapted algorithm based on self-organization and the collective intelligence of ant colonies. The research aims to partition satellite images automatically by discovering the number of thematic classes in multispectral satellite images. Ants normally move in an array in one dimension and can carry objects. The attachment or removal of an object depends on a lot of similarity between this object and the heap objects. The probability that an ant takes the object is greater than leaving the object isolated. When an ant carries an image pixel, the probability that he deposits it as the element density of the same type in the neighbourhood is great. The experimental results of the AntClust adapted algorithm on satellite images can extract the correct class number.

Keywords: image clustering; ant colonies; AntClust; AntClass; Satellite images; Ant clustering;

1.0 INTRODUCTION

The classification system is designed to gather data into classes so that the data of the same class are as homogeneous as possible. The existing classification methods can be grouped into two broad categories: (1) Supervised classification that operates from a base of training data containing examples of previously treated cases (e.g.: K Nearest Neighbours), and (2) Unsupervised classification designed to automatically split the image into natural clusters, i.e. without any prior classes knowledge (e.g. K-Means).

The second classification type is called partitioning or clustering. Images' clustering is the corner stone of any vision system and an important step in the image analysis process [8]. Clustering is a basic step processing image which aims to partition a dataset into meaningful classes. Image clustering depends on the pixels grouping with similar gray levels in one pixels class. The classes obtained can have various properties in common (intensity, color, texture etc). Satellite image clustering methods provide an image theme by grouping pixels with similar gray levels in one pixel-class.

Clustering, with ant colonies algorithms, is the best known and is a most widely used method due to its implementation simplicity. This paper is organized as follows: Section 2 describes how the founding ant colonies algorithms works. Section 3 presents the basic principle and operation of the chosen ant colonies algorithm, named AntClust. Section 4 presents the AntClust adapted algorithm. Section 5 presents experiments that were performed with appropriate AntClust on satellite imagery in addition to comparison with clustering methods and section 6 is the conclusion.

2.0 SEMINAL WORK

The automatic classification algorithms are inspired by the collective sorting behavior observed in ants. Some studies have shown that some species of ants are able to organize various elements such as brood eggs, larvae, etc [1, 2]. The basic principle of this behavior is as follows:

- When an ant encounters an element of the brood, the more isolated the element is, the more likely it is to pick it up;
- When an ant carries a brood item, the probability that the picked object is from a file that consists of the same element density and type within the neighborhood is great.

Deneubourg et al [1] were the first to model this behavior. In simulation experiments, the objects to be collected are also placed randomly on a grid. Ants are modeled by simple agents that are also placed randomly on the grid representing the environment in which they live. Each agent has only a local perception of its environment and is responsible for moving objects based on similar objects concentration in their immediate environment called the "neighborhood".

The principle is to group similar objects into groups on a grid. Each ant can take an object with a probability based on its similarity with the objects in its neighborhood and deposit it with the same probability. After some iteration, similar objects groups are formed on the grid. The main characteristic of these algorithms is their unsupervised side, which allows the automatic discovery of the number of correct groups without external intervention as is usual in conventional classification algorithms. The objects deposit and collection are biased by probabilities $\boldsymbol{P_p}$ and $\boldsymbol{P_d}$ represented by:

$$P_p = \left(\frac{k_1}{k_1 + f}\right)^2 \quad P_d = \left(\frac{f}{k_2 + f}\right)^2 \qquad (1)$$

f is an estimate objects number placed in the ant vicinity. k_1 and k_2 are positive constants. When $f << k_1$, this means there are few objects in the object vicinity, and therefore, the taking probability P_d is high (close to 1). Conversely, when $f >> k_1$, the chance to take the object is small if it is surrounded by several objects.

The algorithm proposed by Deneubourg was adopted and extended by Lumer and Faita [7] for the classification of digital data.

Lumer and Faieta work inspired other authors to solve the classification problem by ants. Kuntz et al [5] were inspired by clustering graphs. In [6], ant-based classification algorithm was proposed for minimizing communication between processors in a simulation where the processing is spread over multiple processors. In [10], Monmarché s introduces a classification algorithm called AntClass using ant populations. AntClass is based on the Lumer Faieta algorithm with some basic modifications. AntClass uses a toroidal grid and each ant can carry several items at once and bring in many objects on the same grid cell. Moreover AntClass is an ant algorithm hybridization and classification algorithm similar to the classic K-means.

AntClust [11] is an adaptation of the AntClass reiteration with improvements concerning the objects support classifying and movement of ants for images segmentation.

3.0 THE ANTCLUST ALGORITHM

The work of Lumer, Faieta and Monmarché was resumed in order to support the classification of objects and movement of ants.

In most ant-based classification algorithms, like AntClass algorithm, objects are placed on a two-dimensional grid and ants move on a grid cell to another, and use a local similarity measure to group similar objects (Fig.1).

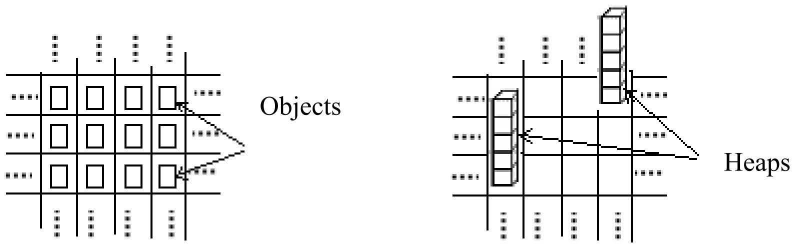


Fig.1. AntClass structure. Initially, each cell of the grid consists of a unique object. Finally, after AntClass treatment the classes are formed with heaps. A heap is a set of similar objects.

In AntClust [12], the grid is abandoned because it is not easy to find the proper settings for several related parameters as listed below:

- The grid size has a great influence on the convergence of the algorithm. The grid should not be too large because the ants would waste time looking for objects, and not too small. Otherwise, there will be no empty cell to drop objects moved by ants.
- Each grid cell can contain only one object at a time. This means that an ant can spend some time finding an available cell on the grid.
- The movement of ants on the grid is random. Some cells may not be visited by ants, and therefore, the objects placed there will not be collected in an acceptable number of iterations.

The results obtained are essentially visual. It must go through a post-processing for use in an object partition.

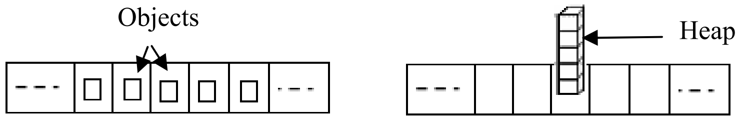


Fig.2. AntClust structure. Initially, each cells consists of unique objects. Finally, the new provision suggested by Ouadfel in AntClust the classes are formed with heaps.

These difficulties have led us to propose the algorithm AntClust, described in the following section.

3.1 PROBLEM FORMALIZATION

Consider a set of N pixels $\{p_1, p_2,..., p_N\}$. We want to group them into classes that are as homogeneous as possible in terms of the gray level of the pixels. We also consider a population A of K ants $\{a_1, a_2,...,a_K\}$ that cooperate together and communicate by stigmergie to provide an optimal classification.

Initially, it was N classes for each pixel. In the process of classification, ants move pixels from one class to another and try to group similar maximum term gray level pixels in the same class.

To do this, we assess a measure of similarity between a pixel p_i with the gray level n_{gi} and the centre of gravity g_k for a class c_k defined as follows:

$$f(p_i, c_k) = \left\{ \frac{1}{1+\left(\frac{ng_i - g_k}{\beta}\right)^2} \right. \qquad (2)$$

β is a parameter that controls the expansion of the function f.

The similarity function $f(p_i, c_k)$ is maximum when $n_{gi} = g_k$ and is standardized between *0* and *1*.

3.2 ANTS ENVIRONMENT

During the AntClust classification process, ants move periodically from their nest to an array of N cells representing classes of pixels as shown in Figure 3. This array has the following properties [12]:

- Each cell of the array is linked to the ant's nest that facilitates the movement of ants on the array,
- Each cell of the array can contain an unlimited number of similar pixels as a measure of similarity,
- Initially, the ants are on the array corresponding to the pixel grouping cell and each cell contains only a single pixel.

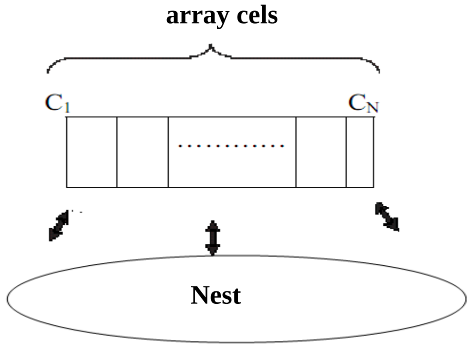


Fig.3. The ants artificial environment

With the used grid in previous works, this array provides two key benefits:

1. It ensures that ants do not waste time searching the pixels on the grid
2. Identification of pixels classes is immediate because a cell can contain more than one pixel. Like in other works, a class is represented by a mass of objects; in this case the class is represented by pixels.

To clarify this point, we will introduce the figure below :

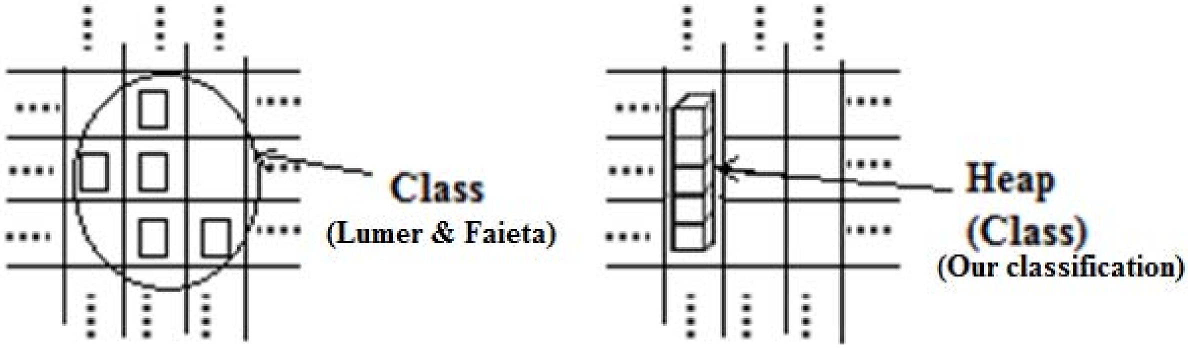


Fig. 4 Lumer & Faieta Classification vs. Adapted AntClust Classification

We refer to Fig.4 which illustrates the class used for different approaches (Lumer & Faieta [7] on the left, and AntClass, Antclust and our Adapted Antclust classification on the right)

Later, the term "cell" refers to a pixel class.

3.3 ANTCLUST ALGORITHM

The algorithm begins with an initial phase in which (1) N pixels are randomly placed on the cells of the grid; (2) The A Ants $\{a_1, a_2, a_K\}$ are moved by their nest and arranged randomly on the cells of the grid by verifying that a cell cannot contain more than a single ant at the same time; and (3) The ant collects a pixel of the cell where it is. As a result of this step, the classification process begins.

This is a simple loop in which (1) an Ant is randomly selected; (2) It returns towards its nest and moves towards a cell guided by an heuristics information; and (3) the Ant decides to drop the pixel where it is transported according to a probabilistic rule. Once it becomes free, it makes new movements between the nest and the cells of the grid to find the next pixel to be carried. Picking up a pixel is also performed based on a probabilistic rule. This loop is repeated for each ant.

During the classification process, no new class is created but a class can be lost if a cell corresponds to an empty cell. At the end of the clustering process, the number of interesting classes of the image corresponds to the number of non-empty grid cells.

AntClust algorithm (1) is as follows:

/***************** STOCHASTIC AntClust

/* Initialisation*/

For each pixel P_i do

    Place p_i in a cell of the array

EndFor

For each ant a_l Do

    Place a_l in a cell chosen randomly and assign its pixel.

    State [a_l]: = carrier;

```
EndDo
Move all the ants towards the nest
/* Main loop */
For t=1 to t_max Do
    For every ant a_1 Do
        If State[a_1]= carrier Then
            Move a_1 towards a cell c_k
            Deposit := false ;
            Deposit := Deposit the pixel p_i which it transports in c_k with a
            probability P_d(p_i,c_k)
            If  Deposit = True Then
                State [a_1] : = free ;
            EndIf
        Else
            Choose randomly a pixel p_i;
            Move a_1 towards the cell c_k containing p_i;
            Carry := false ;
            Carry : = Carry p_i of its cell with a probability P_p (p_i, c_k),
            If Carry = true Then
                State [a_1] := Carrier ;
            EndIf
        EndIf
            Move ants towards the nest
    EndFor
EndFor
Return the obtained partition
```

In the next section, we are going to describe in detail the rules of movements; collection and deposit of pixels which ants are going to use on the array classify the image pixels.

3.3.1 ANTS MOVEMENTS

During the clustering process, the ants move regularly between their nests and grid cells to transport or deposit a pixel. To accelerate the process of grouping, and therefore, the convergence of the algorithm, the ant does not have a completely disorderly movement. For this purpose a modified mechanism version of short-term memory introduced in [7] and [9] is proposed.

In the Lumer and Faieta approach [7], each ant memorizes the m last objects it collected and their location on the grid. Every time it picks up a new object, it compares the object to the objects in its memory. Later, it goes to the location of the most similar object in terms of Euclidean distance. This mechanism has been extended in [3], replacing the Euclidean distance between two objects by the neighbourhood function applied to the positions of all the objects to be classified.

Monmarché [9, 10] incorporates the ideas of Lumer and Faieta, and uses the distance between the centre of gravity of the Heap by the Ant and the heap that he memorized (since in his approach, the ants can transport more than an object at a time) to choose the next location of the object (or the heap) that she transports. We adapt these ideas for the images classification and extend it for the deposit and the collection of pixels as follows:

When an ant carries a pixel, it allows access to its immediate neighbourhood. It calculates the similarity function defined in Eq. 2 for each cells of the 8 neighbours of the pixel it transports and evaluates the ability to directly deposit it in one of it candidate cells. The best location will be the one for which the similarity function is maximum.

The ant decides then to deposit its pixels on this location with a probability $\boldsymbol{p_d}$. If the decision is negative, the Ant keeps the pixel it transports and tries other cells randomly selected until it is able to deposit it.

In this research, when an ant is looking for a pixel to be transported, a common index table containing the free pixels is used (not transported by an ant). Initially, the index table contains all the pixels in the image. We chose to sort the index by an ascending order according to the distance between the gray level of the pixel and the centre of gravity of the class. This index table is a function of the similarity between the gray level of the pixel and the class in which it is located, such that the most dissimilar pixels, which are the farthest from the centre of the classes, are at the top of the index table.

This choice has two advantages: (1) it ensures that only the pixels furthest away from the centers of gravity of the classes to which they belong (the dissimilar pixels) and (2) the update of the index is facilitated each time an ant deposits a pixel in a cell of the array.

When the ant is not carrying any pixels, it searches for a possible pixel to pick up and this search is guided by the index table that contains all free pixels.

Three cases have to be considered:

Case1: If the considered pixel named (p_i) is alone in its designated cell called (c_k),
Case2: If it has one pixel with it in the same cell and
Case3: If there is some others pixels with it the cell.

1. In the first case, the ant picks up it automatically.
2. In the second case, we have an invalid class with only two pixels; the ant will destroy this class by picking up the considered pixel with a probability q.
3. In the third case, the ant has a high probability to pick up a pixel if its similarity with all pixels in the class is low (tend to 0).

The formula in equation (3) describes the ant pixel decision in the three states:

3.3.2 PIXEL REMOVAL

The probability of carrying a pixel p_i of its cell c_k is defined by the following formula:

$$p_p(p_i, c_k) = \begin{cases} 1 & \text{if } |c_k| = 1 \\ q & \text{if } |c_k| = 2 \\ \dfrac{k_p}{k_p + f(p_i, g_k)} & \text{else} \end{cases} \tag{3}$$

where where q is a fixed parameter in [0,1], $|\ c_k\ |$ is the number of pixels in the c_k cell and g_k its center of gravity. If the c_k class contains only one pixel, it is systematically collected by the ant.

If the class contains two pixels, the ant has a probability Q of collecting pixel p_i. Finally, if the cell contains more than two pixels, the p_p probability of transporting pixel p_i is important when the similarity function between the center of class c_k and the gray level of pixel p_i is low (toward 0).

3.3.3 DEPOSIT OF THE PIXEL

If an ant carries a pixel p_i, he explores its immediate vicinity to choose c_k cell, where he will move (see 3.3.1) to deposit with a probability given by the following formula:

$$p_d(p_i,c_k)=\begin{cases}1 & \text{si } f(p_i,g_k)\le f(p_{dissim},g_k)\\ \dfrac{f(p_i,g_k)}{f(p_i,g_k)+k_d} & \text{else}\end{cases} \quad (4)$$

With :

$$f(p_{dissim},g_k)=\min_{p_i\in k}(f(p_i,g_k)) \quad (5)$$

So if the pixel transported by the ant is closer to the center of the c_k class than the pixel most distant from this class, it is deposited there. Otherwise, the smaller the function of similarity between p_i and c_k, the lesser is the probability of deposits.

4.0 THE ANTCLUST ADAPTED ALGORITHM

The AntClust algorithm is not optimal and it is unable to estimate the number of classes considering small homogeneous classes as the objects themselves, increase somewhat the ants carrying capacity. In order to process objects by ants, we adapted the AntClust algorithm and consider the ants can carry a whole heap of objects.

To reduce the number of heap, the ants will be able to carry a heap of objects. The ant drops a heap T_1 on a heap T_2 if the distance between the gravities centres of the two heap is less than T_p (this is a maximum dissimilarity to create a heap consisting of two existing heaps if their interval is between 0.05 and 0.2) (See Algorithm 2); [4]

When T_1 and T_2 are gathered together, they form a single heap T_3. This means that two heaps that have been collected cannot be separated. This accelerates the convergence of AntClust in the second stage. Convergence is reached when the rate of change of pixels classes is low or zero between two consecutive iterations. The AntClust adapted algorithm for image clustering is carried out in the following two steps:

4.1 FIRST STEP

During this step, only the AntClust algorithm is applied where the ants colony is disposed according to a uniform random law to classify images. The ants are deposited to the grid.

The pickup and deposit of pixels are not conducted physically, but virtually, following probability measures calculated in the pixel spatial neighbourhood on which the ant is deposited.

The similarity between two pixels is measured based on the Euclidean distance radiometric. The pickup and deposit parameters P_d and P_p considered are between 0 and 1 allowing for more homogeneous classes, and limiting the clustering errors. The two parameters are determined empirically and the transport capacity of an ant is equated to 1

This first step is an iterative process that ends when the pixels exploration rate is 100% or if it equals a certain threshold. At the end of this step, a labelled image is obtained.

However, there is a risk of two problems:

1. A large number of classes as illustrated in Fig. 5,
2. The ants movement is random, thus, there is a risk of unclassified pixel (pixels free),

In order to solve these problems, we propose a second exploration phase during which another ant-based algorithm is applied.

4.2 SECOND STEP

Assigning a class to each pixel and the reduction of class number occurs following these steps:

1. The colony is composed of a single ant,
2. The displacement of the ant is deterministic,
3. The ant has its own internal memory to directly target the free pixels,
4. The ant transport capacity is infinite and the ant is capable of manipulating heap of objects
5. The basic heuristic rules are adapted to allow the ant to manipulate entire heap,
6. The parameters P_p and P_d are also used.

The adapted AntClust algorithm (algorithm 2) is as follows:

/*********** DETERMINISTIC AntClust

- Initialization parameters of the ant colony (a single ant, speed, infinite transmission capacity $c(a_k) = \infty$).

- Initialization the rate of minimal change.

Begin (Algorithm)

While (Current rate of change < fixed rate of change) Do

Repeat

Determine movement toward a heap
Pick up the heap

Repeat

Determine movement to another heap
Calculate the distance between the gravity centers of the two heaps (picked heap and the new heap)

Until the distance between the gravity centers of the two heaps $<T_p$

Merging of two Tas (formation of a new Tas)

Until All heap are visited

- New configuration

End While

Classified image

End (Algorithm)

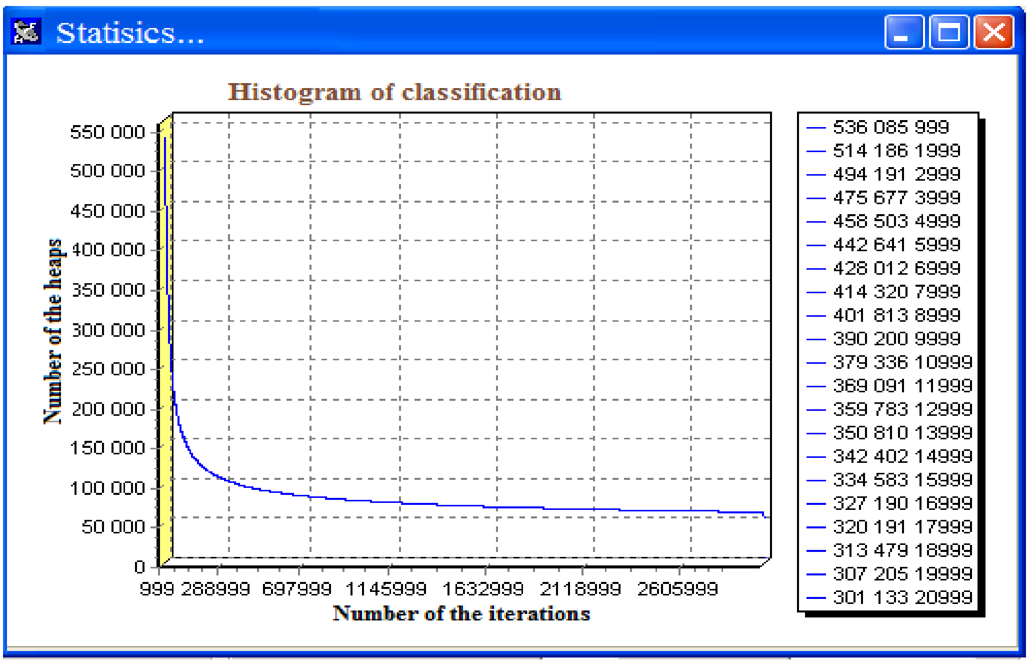
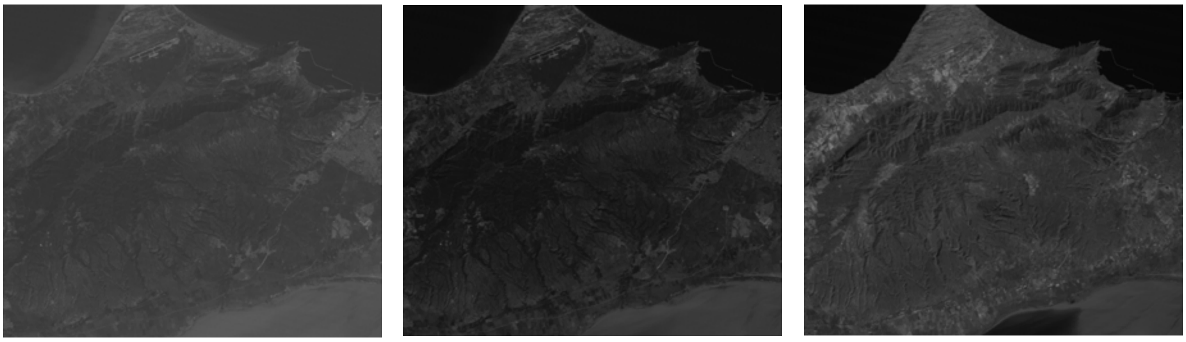


Fig.5. Evolution of the number of heaps during iterations

5.0

This work is implemented using the C++ Builder 6 programming language. To test our approach, we used a multispectral image of the Oran region (West Algeria) of 800×700 = 560 000 pixels (Fig.6), acquired by the LANDSAT satellite in 1993. This image has been chosen for the diversity of themes within it.



Channel 1 channel 3 channel 4

Fig.6 Images of channels 1, 3 and 4 to the region of Oran

We set P_p = 0.015 and P_d = 0.56, empirical parameters [4] with 50 Ants and the iteration number $t_{max}=3*10^6$. In order to evaluate the obtained classes by the adapted AntClust algorithm, we used a supervised database of known number of classes: 12 themes. We defined two measures to perform the evaluation results: Classification Error (CE) and Classification rate (CR).

Classification Error (CE) :

$$CE = \frac{\text{Pixels number incorrectly classified}}{\text{total number of pixel}}$$

Classification rate (CR):
$CR = (1 - CE) * 100\%$

$$CR = \frac{\text{Pixels number correctly classified}}{\text{total number of pixel}} * 100\%$$

Table 1 : comparison of different methods

	ad AntClust.	AntClass	K-means
Error CE	0.00828	0.0691	0.0817
Rate CR	99.172 %	93.089 %	91.83 %

The comparison results between adapted AntClust, AntClass and K-means show that the results of the adapted AntClust are better than that of the other two algorithms. Table 2 shows the pixels number for each class obtained by the three algorithms and Table 4 presents the used parameters by the AntClust adapted algorithm.

After calculating the number of pixels distributed, we obtain the following table:

Table 2. Pixels number of classes for each algorithm

Pixels Number

Classes nam	ad. AntClust	K_Means	AntClass
Sea	74517	72455	71959
Landswell	10039	12345	13203
Sand	3006	87510	93821
Truck farmin	7029	0	16971
Cereal	34067	0	0
Fallow	27218	27339	44580
Drill	216164	75584	107480
Maquis	89049	30451	48352
Urban	17016	43691	46277
Burning	23566	15976	42857
Sebkha1	50323	57267	65378
Sebkha2	8006	8849	9142

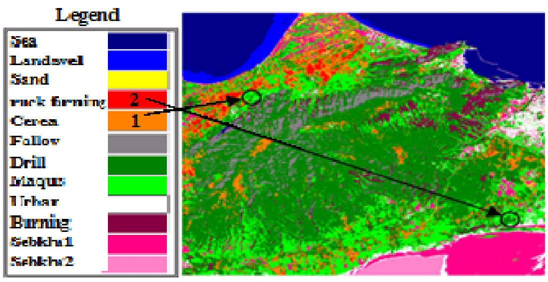
Table 3 : pixel's percentage distribution

	adapted AntClust.	AntClass	K-means
Pixel Number	560000	560000	431437
% exploration	100%	100%	72%

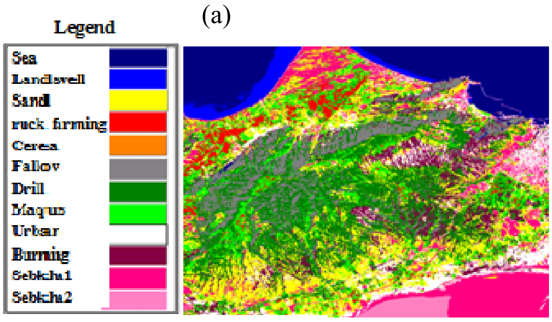
It is noted that the Adapted AntClust algorithm and AntClass have exploited all the pixels of the image (a total of 5600000 pixels) with the existence of 28% of pixels not used for K-means as shown in Table 3. The Adapted AntClust has obtained the number of classes equal to the number of gray levels present in the image (see Fig. 7). Using the example of the truck farming and Cereal, which have 2 and 1 class respectively, the presence of these classes are detected by the AntClust algorithm as shown in Fig. 7.a (and verified by Table 2 in terms of pixels), but not by the other two algorithms confirming that the adapted AntClust algorithm could retrieve the exact number of classes. The results clearly indicate that the adapted Antclust algorithm has knowledge of the probable number of class, successfully identifies an equal number of classes with total distribution of pixels.

Although the AntClass algorithm gave results similar to the adapted AntClust, its contribution in this case is irrelevant. The Antclust Convergence is limited as shown in Figure 5, for number of Classes (Heaps) greater 50000 the algorithm does not give any interesting results which was why the adaptation Process (Adapted Antclust) was introduced in order to reduce the number of Heaps (50000) and the number of existing Themes(12)

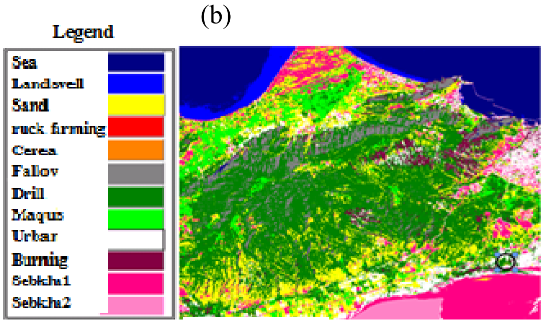
This clearly shows that the Antclust adapted algorithm provides high-performance results and subsequently requires no improvement.



(a)



(b)



(c)

Fig.7. Classified image Adapted AntClust (a), by K-Means (b) and by AntClass (c)

Table 4: Settings of **the** Adapted AntClust

Settings	Description	Value
T_p	Control the probability to deposit a pixel in a cell	0.1
Q	Control the probability to select a pixel in a cell containing two pixels	0.7
β	Control the expansion of the similarity function.	50

6.0 CONCLUSION

The research outlined in this paper shows the efficiency of the adapted Antclust algorithm for image clustering and classifications as compared with the AntClass and K-means algorithms. AntClass group similar objects into groups on a grid

In Antclust, each pixel is placed in a discrete array, which represents the environment of the ants. In our adapted AntClust, we proposed an amelioration of AntClust algorithm to have salient effect in fastening the clustering process. The advantages of the proposed algorithm are that it does not require prior knowledge of the class number or an initial partition of objects. Experimental results on satellite images demonstrated the ability of Adapted AntClust to extract the correct number of clusters and to give better clustering quality compared to those obtained from Antclass algorithm and a classical clustering like k-means.

REFERENCES

[1] J.L. Deneubourg, S. Aron, S. Goss, and J.M.Pasteels, "*The self-organizing exploratory pattern of the argentine ant*". Dans Journal on insect Behavior, 3: 159-168, 1990.

[2] J.L. Deneubourg, S. Goss , N. Franks , A. Sendova-Franks , C. Detrain and L. Chretien [1991]. "*The dynamic of collective sorting robot-like ants and ant-like robots*", in J. A. Meyer and S. W. Wilson (eds), SAB 90 – 1st Conference On Simulation of Adaptive Behavior: From Animals to Animats, MIT Press.

[3] J. Handl, "ant_based methods for tasks of clustering 2nd topograp goc mapping:extensions, analysis and comparison with alternative methods". Master thesis Germany. November 2003.

[4] R. Khedam , "Contribution au développement de méthodologies de fusion/ classification contextuelles d'images satellitaires multi-sources. Application à la cartographie thématique du milieu urbain de la ville d'Alger". Thèse doctorat, 09 février 2008.

[5] P. Kuntz , P. Layzell and D. Snyers, "*A colony of ant- like agents for partitioning in vlsi technology*". In P. Husbands et I. Harvey, éditeurs, Proceedings of the Fourth European Conference on Artificial Life, pages 417-424. MIT Press, 1997.

[7] E. Lumer and B. Faieta, 1994, "*Diversity and Adaptation in Populations of Clustering Ants*". Proceedings of the third International Conference on Simulation of Adaptive Behaviour: From Animals to Animates. MITPress, 501508.

[6] A.E. Langham, P.W. Grant, "Using competing ant colonies to solve k-way partitioning problems with foraging and raiding strategies", in: D. Floreano et al. (Eds.), Advances in Artificial Life, Lecture Notes in Computer Science, 1674, Springer, 1999, pp. 621–625.

[8] O. Monga and B. Wrobel, "Segmentation d'images: vers une méthodologie", Traitement du Signal, vol. 4, n° 3, pp 169-193, 1987.

[9] N. Monmarché , G. Venturini and M. Slimane , "AntClass : discovery of clusters in numeric data by an hybridization of an ant colony with the Kmeans algorithm". Internal Report No 213, E3i, January 1999.

[10] N. Monmarché , G. Venturini and M. Slimane , "*Classification non supervisee par une population de fourmis artificielles*". Actes Coll. Insectes Sociaux, 13 : 43-52 (2000).

[11] S. Ouadfel , M. Batouche , "*Des fourmis pour la segmentation des images*". Journées d'étude en Informatique graphique JIG05, Biskra novembre 2005.

[12] S. Ouadfel , "Contributions à la Segmentation d'images basées sur la résolution collective par colonies de fourmis artificielles".Thèse doctorat, 04 juin 2006

BIOGRAPHY

Ahmed Benyamina is a PhD student in third year computer science University of Science and Technology of Oran (USTO) - ALGERIA. He received the diploma of engineering in Computer Science from the University of Es Senia - Oran-Algiers in 1992. He received the diploma of teaching in Computer Science from the University of Bechar - Algeria, in 2008. Is a lecturer at the University of Bechar - Algeria, in 2010. His research interests are in the field of classification, data analysis and complex algorithms.

Hadria Fizazi is a Lecturer in Computer Science. She teaches at the University of Science and Technology of Oran (USTO) – ALGERIA. His research interests is remote sensing, data processing and analysis of spatial data.