Mapping from Fused Aerial Images and LIDAR Data Using GA-Based Classification

El-Nokrashy, Mahmoud[1], Esmat, Adel[1], Salah, Mahmoud[2], Hamdy, Ahmed[1]

[1]Department of Civil Engineering, Faculty of Engineering, Al-Azhar University, Egypt.
nokrashydr@yahoo.com, adelesmat@yahoo.com, ahmedhamdii@yahoo.com
[2]Department of Surveying Engineering, Shoubra Faculty of Engineering, Benha University,Egypt.
mahmoud.goma@feng.bu.edu.eg

Abstract

The availability of high quality RGB images and LIDAR data provides efficient image classification using the complementary properties of these data sources. The paper objective is to introduce an automated urban unsupervised classification technique using combined semantic (from RGB image) and spatial (from LIDAR data) information leading to the ability to extract different features rabidly and efficiently. In this study, new concepts and techniques for mapping urban areas using aerial images and LIDAR data fusion was developed and tested based on available data. Genetic Algorithms (GAs) were integrated with different fitness functions to produce two proposed techniques. K-Means (KM) and Fuzzy C-means (FCM) algorithms were tested and compared, as fitness functions for GAs. Three groups of data were applied which include: RGB group; RGB/ LIDAR data group and RGB/LIDAR/attributes group. Error matrix and K-HAT (kappa) statistics were adopted as well as visual inspection to evaluate the classification results. FCM proved to be a preferable fitness function for GAs-based classification from aerial images and LIDAR data with accurate average classifications of 87.84%. The feature classification techniques developed in this study are automated, efficient and present a suitable method for extracting different features while overcoming most of the problems in situations of similarity existing in texture or height information accompanied by fast and reliable results. During the vectorization phase, the classified images were processed through a series of image processing techniques to produce the digital vector buildings map. An accurate estimation of buildings was carried out from the reference data and compared against the corresponding results. The extracted building data coordinates were compared against the Global Positioning System (GPS) observations and the standard deviation was 0.43m.

Keywords: LIDAR**,** Unsupervised Classification, Genetic Algorithms, K-means, Fuzzy C-means.

1. Introduction and related work

Land-cover classification including supervised and unsupervised classification is one of the important applications of digital imagery. The main goal of the image classification is the associations of each pixel in the image with a specific land-cover class to produce accurate classification maps from the data. Unsupervised classification divides pixels within an image into a corresponding cluster pixel by pixel. Typically, the only input is the number of clusters of the

scene. In general, unsupervised classification is performed through the well-known method, k-means (waske, 2007).

Genetic algorithms (GAs), introduced by John Holland in 1975, are suitable method to produce heuristic unsupervised classification (Coley, 1999; Pham and Karraboga, 2000). Image classification has been widely and successfully applied by optimization algorithms specifically GAs (Rothlauf, 2006), which confirm the potential of GAs to produce high level of quality results especially when applied without any ground truth (Coley, 1999). Numerous studies have shown that the GAs technique is very efficient in dealing with large datasets and has a large chance to avoid a local optimal solution than other methods (Huang and Wang, 2006; Zhou et al. 2010). Another advantage of the GAs is its capability to search for input features and parameters of classifier simultaneously.

Almeida (2012) studied the use of GAs routine with decision trees for the object-based land-cover classification. The study showed a satisfactory performance for the automatic assessment of the optimal segmentation parameters. Nevertheless, the shape complexity of some targets, the internal spectral variability of certain classes, and the diverse conditions of ageing and maintenance of some roof classes found in the study area led to an over-segmentation of some targets. Adding height information derived from laser scanning to the imagery discriminates targets with similar spectral behavior but diverging values. Chu (2012) used the integration of feature selection using GAs and multi classifier system with Dempster-Shafer theory for classifying different combined datasets. Results of classification revealed that proposed method (FS-GA-DS model) always gave significantly higher accuracy, than any single classifier. Ge (2012) proposed GA-SVM model to classify multiple combined datasets, consisting of Landsat 5 TM, Multi-date dual polarization ALOS/POLSAR images and their multi-scale textural information. The performance of the proposed method was compared with that of the traditional steck-vector approach. It revealed that the proposed method is efficient for handling multisource data. The highest classification accuracy achieved was 96.47% with only 81 out of 189 features being selected which demonstrate the advantages of using multi-source data over single source data. Li (2013) introduced GAs and SVM to high resolution POLSAR image classification. This model was applied with the contrast of three classifiers without using additional polarimetric information, and thus its overall accuracy is only 74.85%. The method gets the best results when using additional information, the accuracy is up to 97.49%. Yang (2006) studied the influence of changing GAs parameters on classification results. Two of these parameters, namely population size and the crossover probability were considered. He pointed out in his results that the population size proofed to be significantly more important than the crossover probability. The effectiveness of this technique was evaluated using IKONOS satellite images. An overall accuracy of 71.1% was reached using (DBI) index as compared to 65.1% when using the ISODATA algorithm. Jamshidpour (2012) suggested a framework to combine filter and wrapper feature selection methods to find feature subset and optimize the SVM kernel parameters at the same time. GAs have been used as global optimizer to obtain the optimal solution after a series of iterative computations. Samaher (2007) attempted to classify 6 different kinds of forest scenes using genetic algorithms clustering and neural network.

The randomly estimation of the number of the clusters that are found in the image may lead to error in classification process. Therefore, the proposed methodology can solve this problem by determining it automatically. Optimal results will depend on the selection of parameters.

The work presented in this paper is probably the first attempt to use GAs for combination of aerial images and LIDAR data for improvement of land cover classification. To further improve the classification performances and overcome the shortcomings of the previous approaches, GAs have been proposed in this research.

The main objective of this research is to develop an accurate, time-effective and automatic method to classify building, roads, trees and grass by the fusion of aerial images, LIDAR data and some effective attributes. This objective is achieved by comparing and combining the outputs from KM based GAs and FCM based ones in order to improve the quality of the classification results. In this regard, GAs was applied with K-means and Fuzzy c-means as fitness functions. A Combined method was applied to fuse the results from both techniques. For the rest of the research, the KM based GAs will be referred to as KMGA while the FCM based one will be referred to as FCMGA. All the methods proposed in this research were implemented through software package generated by the authors in Matlab environment. The study areas and data sources is described, the used classification methods are outlined; then the results are presented and evaluated.

2. Study area and data properties

For this study a very high resolution digital image was available. The image covers an area of approximately 500x500m of the region surrounding the University of New South Wales (UNSW) campus, Sydney Australia. The area is a largely urban area that contains residential buildings, large Campus buildings, and a network of main roads as well as minor roads, trees, open areas and green areas. The color imagery was captured by film camera at a scale of 1:6000. The film was scanned in three color bands (red, green and blue) in TIFF format, with 15μm pixel size (GSD of 0.09m) and radiometric resolution of 16-bit as shown in Figure 1. The characteristics of image datasets are provided in Table 1.

Table 1: Characteristics of image datasets.

Test area	*Size(Km)*	*bands*	*pixel size (cm)*	*Camera*	*Look Angle (deg.)*	
					along track	across track
UNSW	0.5 x 0.5	RGB	9	LMK1000	±30	±30

In order to perform the classification process, three main object classes: trees, roads, and buildings were selected.

Figure 1: An Orthophoto of UNSW campus.

3. Genetic algorithms based classification

GAs is adaptive methods which may be used to solve a variety of optimization problems by the principles of biological organisms' evolution. Recently, GAs has been used in wide variety of optimization problems, specifically in classifying digital data sets (Bandyopadhyay and Maulik, 2003; Majida et al., 2010). GAs is very different from most of the traditional optimization methods. GAs generates a population of solutions at each iteration to approach an optimal solution. This means that GAs can process a number of designs at the same time. In addition it selects the next population by computations that involve random choices. Figure 2 summarizes the working principles of GAs. The following sections describe how to establish a GA classifier for automatically clustering a data set.

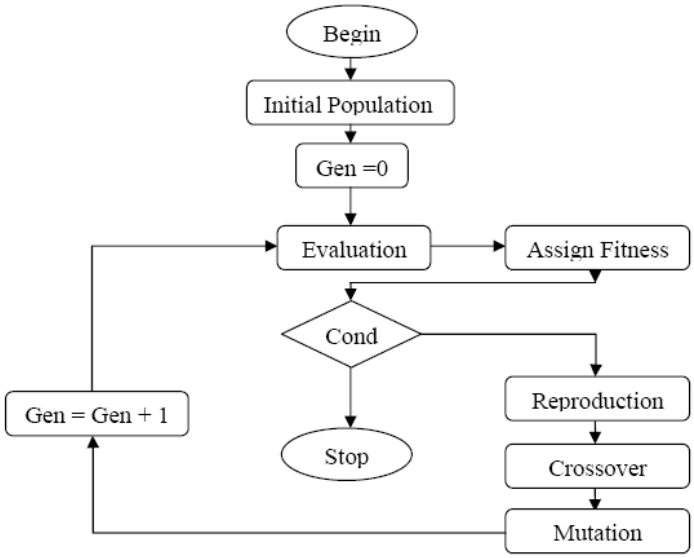


Figure 2: Working principle flowchart of genetic algorithm.

3.1. String representation

In GA applications, the parameters of the searched space are encoded in the form of strings, so-called chromosomes. Each chromosome, representing an answer for the problem, is encoded by a binary, integer or real number. A variable string length is designed without assigning the number of classes a priori. A chromosome is encoded by positive real numbers and a negative integer '-1' which represents a non-existent cluster. The value of K (valid clusters) is randomly assumed in the range (K_{min}, K_{max}), where K_{min} is usually assigned to 2 unless special cases are considered. The length of a chromosome is taken to be K_{max} where each individual gene represents either a cluster center or a non-existent cluster. For each chromosome i in the population (i =1, 2, 3….P, where P is the size of population). The typical size of the population can range from 20 to 1000 (Coley, 1999). K_i are chosen randomly from the data set, and then are randomly allocated in the chromosome (Bandyopadhyay and Maulik, 2003). As an example, assume an image including 3 bands, N pixels for each layer, K_{min}=2 and K_{max}=6, the number K_i equal to 5 for the chromosome i. Let the 3 clusters centers are: (120, 66, 228); (170, 66, 264); and (21, 114, 6). Randomly, the classification centers can be encoded into a chromosome as: -1 (120, 66, 228) (170, 66, 264) -1 (21, 114, 6) -1

3.2. Selection and fitness function identification

The genetic algorithm should succeed two goals: maximizing the classification accuracy; and minimizing the number of selected features. These criteria used to create a single objective function as follows:

$$F = w * C(x) + (1-w) * \frac{1}{N(x)} \quad (1)$$

Where x is the feature subset, $C(x)$ represents the classification accuracy, $N(x)$ is the size of selected feature subset, and w is a parameter between 0 and 1 which adjusts the influence of each criterion. As value of w is higher the weight of classification accuracy in fitness function is greater. On the other hand, reducing the value of w will give more penalties on the size of x (Tan and Fu, 2008). By adjusting w, a trade-off between the accuracy and the size of the feature subset obtained can be achieved. For this research, w was adjusted to 0.8 to avoid large decrease in classification accuracy. Before a GA is operated, an objective function needs to be defined to measure the fitness of each chromosome. The fitness function assigns an adaptability degree to each chromosome in the population. The following Sections summarize the used fitness indices K-means and Fuzzy C-means respectively.

3.3. Genetic operators

In general, GAs is composed of three main operations: reproduction; crossover; and mutation. Reproduction calculates a survival probability of each chromosome which is a criterion to reproduce better chromosomes for next generations. Crossover is a swapping process to create new chromosomes between the reproduced chromosomes. To avoid sticking to a local optimal, mutation is assigned to explore the possible optimal in all the space. The mutation probability is usually set smaller than the crossover, and controls the percentage to introduce new genes for trial. If the mutation probability is too low, some useful genes are never found out. On the other hand, if it is too high, there will be severely random perturbation (Gen and Cheng, 1997).

4. Methodology

4.1. GA application for unsupervised classification

There are basically seven parameters that influence the result of classification using Genetic Algorithm. These parameters include: the maximum length of the chromosome, the way to encode the chromosome units (binary, real number and so on), the population size, the crossover type and probability, the mutation probability, and the employed fitness function (Pham and Karaboga, 2000).

4.2. Description of optimization procedures

In this research, a maximum chromosome length of K_{max}=8 was chosen, which is above the maximum number of clusters in the test image. As mentioned above, chromosome coding was done using positive integers. Only two-point crossover operations were considered with fixed probability. The other parameters were systematically varied in order to study their influence on the result. For selecting the actual parameter values we took advice from general GA references (Coley, 1999). More specifically, the population size was set to 20, 30, 40:200 chromosomes, respectively. Mutation probability values were set to 0.005, 0.05, and 0.5 then to 0.001, 0.003: 0.009 then to 0.01, 0.03:0.09. One set of parameters, namely a population size of 100with a mutation probability of 0.005 was considered as the baseline set, against which the other parameters were varied. Iterations were terminated as soon as they arrive the 100 generation. In this way, the K-means index as a fitness function was used and investigated, resulting in a wide variety of results. Table 2 & 3 display the numerical results for the variation of population size and mutation probability

Table 2: Numerical results for population size variation.

Probability of Mutation = 0.005 & No. of Generations = 100								
Population Size	Overall Accuracy %	Class Accuracy						K-HAT %
		Roads		Trees		Buildings		
		PA %	UA %	PA %	UA %	PA %	UA %	
20	non	non	non	non	non	non	non	non
50	non	non	non	non	non	non	non	non
60	non	non	non	non	non	non	non	non
70	non	non	non	non	non	non	non	non
80	non	non	non	non	non	non	non	non
90	33.33	100	48.39	non	non	00.00	00.00	0.1089
100	44.44	46.67	37.84	33.33	90.91	53.33	100	0.2718
110	non	non	non	non	non	non	non	non
120	16.67	00.00	00.00	non	non	50.00	27.27	-0.1250
130	12.22	non	non	non	non	36.67	15.49	-0.1910
140	non	non	non	non	non	non	non	non
150	33.33	100	33.3	non	non	non	non	00.00
180	22.22	3.33	5.56	non	non	63.33	95.00	0.0948
200	18.89	non	non	56.67	94.44	00.00	00.00	0.1275

Table 3: Numerical results for mutation probability variation.

Population Size = 100 & No. of Generations = 100								
Mutation Probability	Overall Accuracy %	Class Accuracy						K-HAT %
		Roads		Trees		Buildings		
		PA %	UA %	PA %	UA %	PA %	UA %	
0.005	44.44	46.67	37.84	33.33	90.91	53.33	100	0.2718
0.05	68.89	96.67	52.73	43.33	92.86	66.67	100	0.5359
0.5	00.00	non	non	00	00	00	00	-0.1638
0.001	non	non	non	non	non	non	non	non
0.003	21.11	non	non	non	non	63.33	100	0.1514
0.007	6.67	00	00	non	non	20.00	9.84	-0.3622
0.009	34.44	non	non	40.00	92.31	63.33	100	0.2563
0.01	47.87	non	non	96.67	90.63	46.67	87.50	0.3649
0.03	38.89	non	non	40.00	85.71	76.67	82.14	0.2763
0.07	00.00	non	non	non	non	00.00	00.00	-0.0345
0.09	20.00	6.67	14.29	53.33	34.78	00.00	00.00	-0.0385

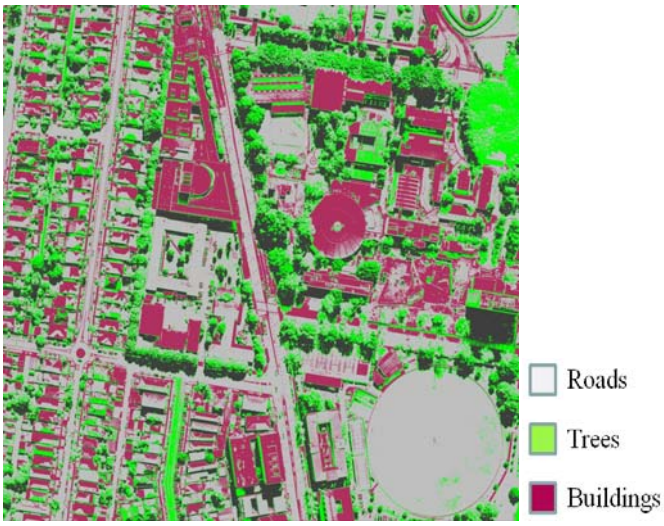


Figure 3: Classified image of Population Size and Mutation Probability 0.05

From on the obtained results the following conclusions can be drawn:

- Few results showed only one class (building) in cases of Population size 130 and mutation probability 0.003 and 0.07.
- In cases of Population size 90, 120, 150, and 180,and mutation probability of 0.007, the results showed two clusters (building and roads). Population size 200 and mutation probability 0.009 and 0.01 showed similar results.
- The results were improved to show three clusters (roads, building and trees) in case of Population size 100 and mutation probability 0.05, 0.03 and 0.09.

- The best results were shown in the case of Population size 100 with mutation probability 0.05 with overall accuracy of 68.89% (see Figure 3).
- For all cases, the K-HAT value is rather low. This indicates a number of errors of omission and commission, which can also be observed when investigating the full error matrices. These matrices will be given in Hamdy (2015).

4.3. Description of validation procedures

4.3.1. Data pre-processing

Data preparation of the study area was implemented in several stages as follow:

4.3.1.1 Filtering of LIDAR data

LIDAR data filtering is the process of separating on-terrain points (DTM) from points falling onto natural objects. A grid method based on support vector machines has been applied for filtering of LIDAR Data (Salah and Trinder, 2010). After filtering of LIDAR points, it converted into digital terrain model (DTM) image. Digital surface model (DSM) is generated from the original LIDAR point clouds (First and last pulses). Finally, normalized digital surface model (nDSM) is generated by subtracting the DTM from the DSM.

4.3.1.2. Generation of attributes

Attributes are necessary to recover some common problems associated with high resolution image data emerge, notably shadows caused by tall buildings or trees; and the spectral variability within the same land-cover class. These disadvantages may cause lower classification accuracy if the classification procedure cannot effectively handle them (Zhou et al., 2008; Lu and Weng, 2007). Before generating the attributes, the aerial photographs (already orthorectified by AAMHatch) were registered to the LIDAR intensity image using a projective transformation. The polymorphic texture strength based on the Förstner operator has been generated and used as input for the classification process.

4.3.1.3. Generation of ground truth data

Accuracy assessment allows evaluating a classified image. In order to assess the performances of the proposed methods, two sets of class values for randomly selected points (900 points) in the classified image were applied. One set of class values was automatically assigned to these random points as they are selected, and the other set of class values (reference values) is input manually. These reference values are based on the digital aerial image of the test area. The accuracy assessment was performed by comparing the classification results with the ground truth data.

4.3.2. Test description and workflow

In this research, a maximum chromosome length of K_{max}=8 was chosen, which is above the maximum number of clusters in the test image. As mentioned above, chromosome coding was done using positive integers. Only two-point crossover operations were considered with fixed probability. The other parameters were chosen to be 100 for population size and 0.05 for mutation probability. In this regard, three groups of data sources, as shown in table 4, were tested and evaluated.

Table 4: Groups of dataset applied for the experiments.

Group 1	Group 2	Group 3
- RGB	- RGB - DSM	- RGB - DSM - Intensity - polymorphic texture strength from nDSM

First, the K-means index as a fitness function was used and investigated (KMGA). After that, Fuzzy c-means index was tested and applied with GA as a fitness function (FCMGA). Finally, an approach was proposed to combine the results from both KMGA and FCMGA. Figure 4 summarize the workflow for the proposed techniques.

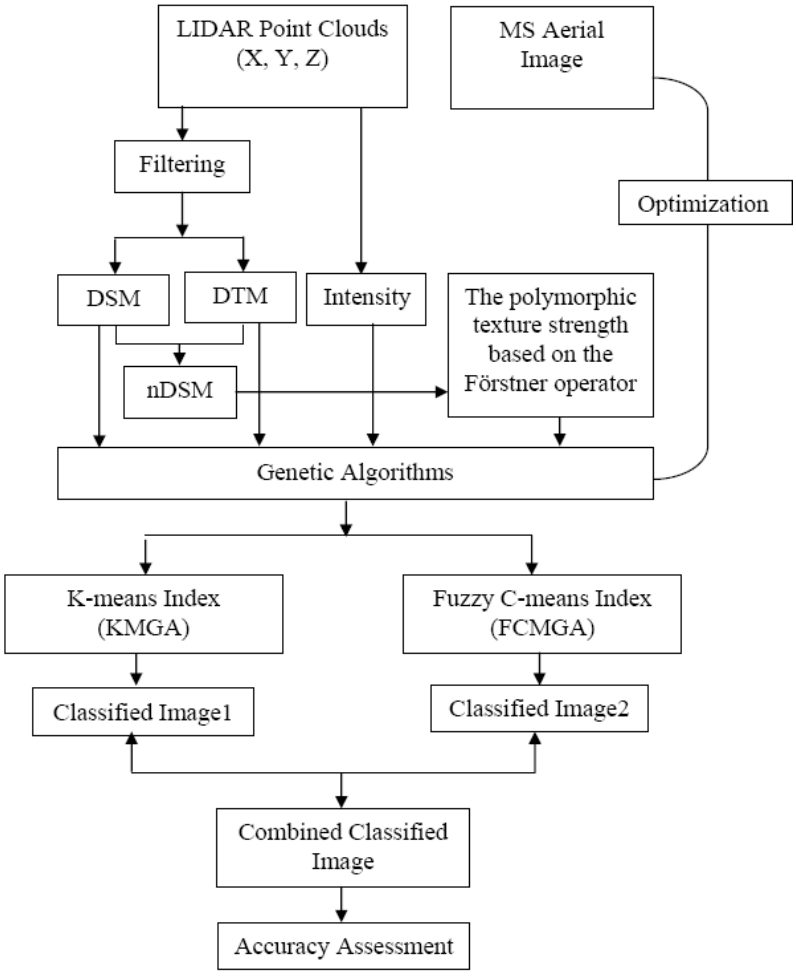


Figure 4: Workflow for image classification by image and LIDAR data fusion.

4.3.3. Genetic *k*-means algorithms based classification

The application of genetic algorithms in the area of classification takes advantage of extensive optimum search capabilities. General genetic procedure in the case of determining the best *k* centers for clusters consists of setting of parameters which include: number of clusters, population initialization, initial population fitness calculation and repeated selection, cross over and mutation operations until termination criteria are met (Hall et al., 1999).

4.3.3.1. Genetically optimized *k*-means clustering algorithms

For genetic k-means selection of cluster number and other algorithm specific parameter values is required. Next, the population should be initialized with randomly created cluster centers. New populations are created by operations of selection, cross-over and mutation. For every solution in population, fitness value is calculated according to the specific fitness function. Solutions with high fitness values come into mating pool. The process is repeated until termination criteria are met. Below some implementation details are given:

Chromosomes: represent solutions consisting of centers of k clusters – each cluster center is a vector of values in the range between 0 and 255 representing intensity of color component.

Population initialization and fitness computation: Cluster centers are initialized randomly to k points with values in the range 0 – 255. Next, fitness value is calculated for each chromosome in the population.

Selection: The operation of selection tries to choose the best suited chromosomes from parent population that come into mating pool. After cross-over and mutation operation child population is created. Most frequently genetic algorithms make use of tournament selection that selects into mating pool the best individual from predefined number of randomly chosen population chromosomes. This process is repeated for each parental chromosome.

Crossover: The operation of crossover presents probabilistic process exchanging information between two parent chromosomes during formation of two child chromosomes. Mostly, two-point crossover operation is used. According to previous studies crossover rate 0.8 - 1.0 yields the best results (Yang and Wu, 2006).

Mutation: The operation of mutation is applied to each created child chromosome with a given probability. After crossover operation children chromosomes that undergo mutation operation flip the value of the chosen bit or change the value of the chosen byte to other in the range from 0 to 255. According to Pham and Karaboga (2000), mutation rate 0.05 yields the best results.

Termination criterion: Termination criterion determines when algorithm completes execution and final results are presented to the user. Most often termination criterion is that algorithm terminates after predefined number of iterations. Other possible conditions for termination of the k-means algorithms depend on degree of population diversity or situation when no further cluster reassignment takes place.

4.3.4. FCM based genetic algorithm classification

GAs searching capability can be used for the purpose of appropriately clustering a set of n unlabeled points in *N*-dimension into *K* clusters. In the following, the same technique can be applied on image data. Considering an image of size $m \times n$, the basic steps of the FCM-based GAs for clustering image data are as follows:

4.3.4.1. Encoding

For an *N*-dimensional space, each cluster center is mapped to *N* consecutive genes in the chromosome. Each chromosome represents a solution which is a sequence of *K* cluster centers. For image datasets each gene is an integer representing an intensity value.

4.3.4.2. Population initialization

In the proposed method, the FCM is run *P* times for generating population of *P* chromosomes; each chromosome is of size *K*. So, each of the *P* chromosomes is obtained by the FCM algorithm which is described below. Each chromosome of the population is a potential solution by FCM algorithm with number of clusters c=K. FCM algorithm assigns pixels to each category by using fuzzy memberships. Let $X=(x_1, x_2,\ldots, x_N)$ denotes an image with *N* pixels to be partitioned into c clusters, where x_i represents multispectral data. The algorithm is an iterative optimization that minimizes the cost function defined as follows (Halder, et al., 2011):

$$J=\sum_{j=1}^{N}\sum_{i=1}^{c} u_{ij}^{m}\left\|x_j - z_i\right\|^2 \tag{3}$$

Where u_{ij} represents the membership of pixel x_j in the i^{th} cluster, z_i is the i^{th} cluster center, || || is a norm metric, and *m* is a constant. The parameter *m* controls the fuzziness of the resulting partition, m: any real number greater than 1. The membership function represents the probability that a pixel belongs to a specific cluster. The cost function is minimized when pixels close to the centroid of their clusters are assigned high membership values, and low membership values are assigned to pixels with data far from the centroid. In the FCM algorithm, the probability is dependent solely on the distance between the pixel and each individual cluster center in the feature domain. The membership functions and cluster centers are updated as follow:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{\left\|x_j - z_i\right\|}{x_j - z_k}\right)^{\frac{2}{m-1}}} \tag{4}$$

Where:

$$z_i = \frac{\sum_{j=1}^{N} u_{ij}^{m} x_j}{\sum_{j=1}^{N} u_{ij}^{m}} \tag{5}$$

Convergence of FCM can be detected by comparing the changes in the membership function or the cluster center at two successive iteration steps.

4.3.4.3. Fitness computation

Two steps to accomplish the fitness computation. First, the pixel dataset is clustered according to the centers encoded in the chromosome under consideration, such that each intensity value x_i, i = 1, 2, ..., $m \times n$ is assigned to cluster with center z_j, $_j$ = 1, 2, ..., *K*.

$$if \ ||xi - zi|| < ||xi - zp|| , p = 1,2, \dots \dots k, and \ p \neq j \quad (6)$$

The next step involves adjusting the values of the cluster centers encoded in the chromosome, replacing them by the mean points of the respective clusters. The new center z_i^* for the cluster c_i is gives by:

$$z_i^* = \frac{1}{n_i} \sum_{x_j \in C_i} x_j, i = 1, 2, \dots \mathrm{K} \quad (7)$$

Subsequently, the clustering metric M is computed as the sum of Euclidean distances of each point from their respective cluster centers and given by:

$$M = \sum_{i=1}^{k} M_i, i = 1, 2, \dots .. K \quad (8)$$

Where:

$$M = \sum_{x, \in C_1} \left\| x_j - z_i \right\| \quad (9)$$

The fitness function is defined as follow:

$$f = \frac{1}{M} \quad (10)$$

Hence the objective is to minimize the clustering metric M i.e. maximize f.

4.3.4.4. Selection

Fitness level is used to associate a probability of selection with each individual chromosome. Roulette Wheel selection was applied, if f_i is the fitness of individual c_i in the population, its probability of being selected is:

$$p_i = \frac{f_i}{\sum_{j=1}^{N} f_i} \quad (11)$$

N is the number of individuals in the population.

4.3.4.5. Crossover and mutation

In this research, a two-point crossover with a fixed crossover probability of μc is used. Also, each chromosome undergoes mutation with a fixed probability 0.05. A number δ in the range [0, 1] is generated with uniform distribution. If the value at a gene position is v, after mutation it becomes:

$$v \pm \delta * v, v \neq 0$$

$$v \pm \delta, \qquad v = 0$$

4.3.4.6. Termination criterion

For each generational iteration, the fittest chromosome is preserved elitism. Thus on termination, this chromosome gives the best solution encountered during the search. The algorithm is a two pass process. In the first pass standard FCM algorithm is used to generate the population. In the second pass, Genetic algorithm is applied on the population generated by the FCM algorithm.

4.3.4.7. Validity measure for clustering

In order to determine the validity of the clustering on the given dataset, the cluster validity index for the fittest chromosome for particular value of K is computed using equation 12. The cluster validity index used in this research is the one proposed by Turi (2001). It aims at minimizing the validity index given by the function as follow:

$$V = yx \frac{\text{int}\, ra}{\text{int}\, er} \tag{12}$$

The term *intra* is the average of all the distances between each pixel x and its cluster centroid z_i which is defined as:

$$\text{int}\, ra = \frac{1}{N} \sum_{i=1}^{k} \sum_{x \in C_i} \left\| x - z_i \right\|^2 \tag{13}$$

The *inter* term is the minimum distances between the cluster centroids which is defined as:

$$\text{int}\, er = \min(\left\| z_i - z_j \right\|^2), \tag{14}$$

Where i=1,2,..,K-1 and j=i+1,2,..K. This term is used to measure the separation of the clusters. Also, y is given as:

$$y = c * N(2, 1) + 1 \tag{15}$$

Where c is a user specified parameter and N (2, 1) is a Gaussian distribution function with mean 2 and standard deviation 1, where the variable is the cluster number and is given as:

$$N(\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{(\mathrm{k} - \mu)^2}{2\sigma^2}} \tag{16}$$

Where k is the cluster number and μ=2 and σ=1. This validity measure serves the dual purpose of minimizing the intra-cluster spread and maximizing the inter-cluster distance.

4.3.5. Classifier combinations

The limitation of LIDAR data on hand in Egypt motivated the authors to combine the results obtained for RGB/KMGA and RGB/FCMGA in order to take advantages of each classifier and improve the overall accuracy because each type of classification algorithm has its own merits and limitations. In this research, the RGB/KMGA and RGB/FCMGA were combined according to their reliability for each class. The class that receives the maximum KIA is taken as the final classification.

It is worth mentioning that accuracy assessments of the proposed system were undertaken using confusion matrices and Kappa statistics. The Kappa Index of Agreement (KIA) is a statistical measure adapted for accuracy assessment in remote sensing field by Congalton and Mead (1983). KIA is a means to test two images, if their differences are due to 'chance' or 'real disagreement'. It is often used to check for accuracy of classified images versus some 'real' ground-truth data.

5. Results and discussion

For the purpose of testing the efficiency of KMGA and FCMGA, the proposed algorithms have been applied to each of the three groups of data. Numerical and visual results from KMGA, FCMGA and combined results for the three groups of data are shown in figure 5 and numerically expressed in graphs 6 through 9. Column 1 of Figure 5 illustrates the KMGA results while Column 2 of the same figure shows the FCMGA results. Moreover, a close watch of Figures 4 reveals that both the approaches are able to yield the distinct clusters but the clusters determined by the FCMGA algorithm are found to be more compact compared to those provided by the KMGA algorithm. The performances of KMGA have been compared to those of FCMGA, in terms of the overall accuracy and their class-accuracies values which are discussed below.

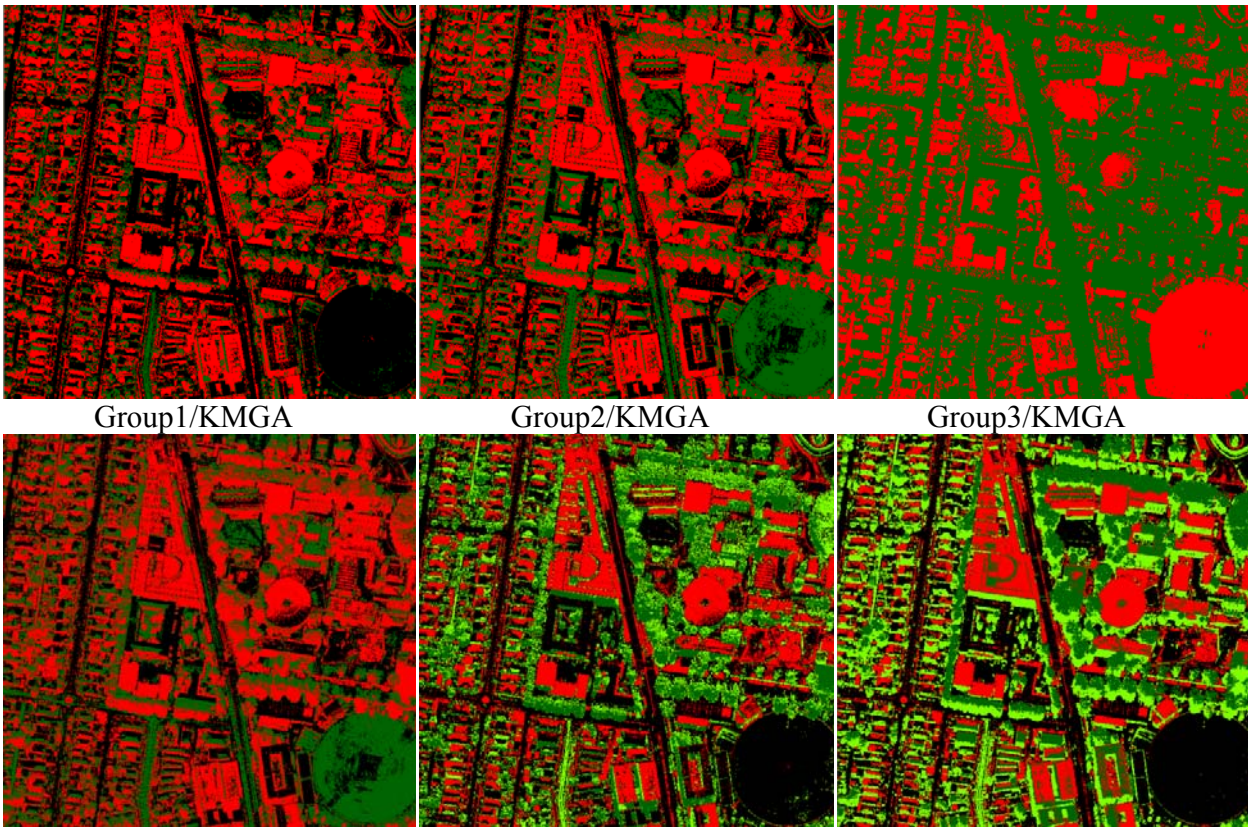


Figure 5: Classifications results. The colors indicate the different classes: Black stands for *roads*, dark Green for *trees,* light green for *Grass* and Red for *buildings.*

5.1 Overall KIA

The results indicate a clear dependence on the range of input data included in the tests. Using only aerial images, many buildings were classified as roads because they have the same spectral

reflectance and the class-accuracies were low. Using aerial image and the LIDAR data increased the classification accuracy due to the suitability of LIDAR data for accurately detecting planes, but still some errors occurred due to the poor horizontal accuracy of edge detection in the LIDAR data. The use of aerial image, LIDAR data and extracted attributes increased the classification accuracy further since the attributes compensated for the weakness of LIDAR for edge detection. FCMGA was very stable in all situations achieving higher classification accuracy. The results also indicate the ability of the FCMGA to take advantage of the input data. The FCMGA outperformed the KMGA for the three groups of data. In the case of FCMGA, group three has yielded the best set of classes with 87.84% overall accuracy. One important note is that the results obtained for FCMGA improved when the elevation data was incorporated into the classification process. Once again, the results improved when the spectral attributes were applied. On the other hand, the results obtained for KMGA has improved when the elevation data is applied, and deteriorated suddenly to 45.82% when the spectral attributes are added. In this case, only two instead of four classes are clearly found. These results indicate that KMGA is a simple model which is inappropriate for real data sets in which there are no definite boundaries between the clusters. Therefore, the authors believe that the bad performance of KMGA was not a result of any inadequate learning process of the algorithm but due to its own structure.

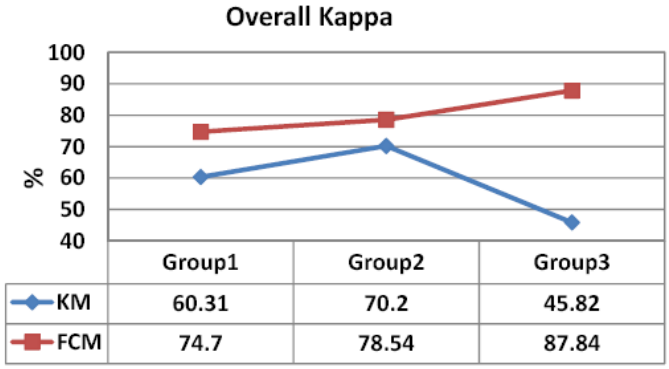


Figure 6: Overall Kappa results obtained for KMGA and FCMGA.

5.2. Per-class accuracy

An additional measure, Per-class Kappa, was used to evaluate the performance of the two models. Unlike Overall Kappa, Per-class Kappa clearly shows how much the performance of the proposed method improves or deteriorates for each individual class. An assessment of the Per-class Kappa confirms that the FCMGA performed the best in most cases as shown in figures 6 through 8. Another advantage of the FCMGA over KMGA is that the achieved Per-class Kappa errors are less variable.

In case of RGB data, most of the Per-class Kappa is improved by the FCMGA. Whereas the application of FCMGA resulted in average Per-class Kappa of 75.18%, the application of KMGA resulted in average of 60.29 %. The roads class had the greatest increase from 42.6% using the KMGA to 76.36% with the FCMGA. Another advantage of the FCMGA over KMGA is that the achieved errors are less variable as shown in table 5. Whereas the application of FCMGA resulted

in standard deviation of 6.37 for Per-class Kappa, the application of KMGA resulted in a SD of 5.80.

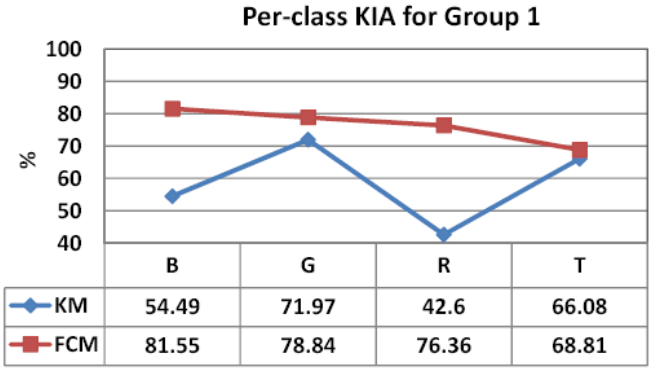


Figure 7: Per-class Kappa results obtained for KMGA and FCMGA in case of RGB dataset. B: Building; G: Grass; R: Roads; and T: trees.

In case of RGB/LIDAR data, most of the Per-class Kappa is improved by the FCMGA. Whereas the application of FCMGA resulted in average Per-class Kappa of 81.68 %, the application of KMGA resulted in average of 72.68 %. The trees class had the greatest increase from 53.43% using the KMGA to 64.09% with the FCMGA. Contrary, there was a decrease in Per-class Kappa for the grass class from 98.93% using the KMGA to 97.86% with the FCMGA. However, those classes are still classified with relatively high Per-class Kappa. Another advantage of the FCMGA over KMGA is that the achieved errors are less variable. Whereas the application of FCMGA resulted in standard deviation of 17.59 for Per-class Kappa, the application of KMGA resulted in a SD of 19.25.

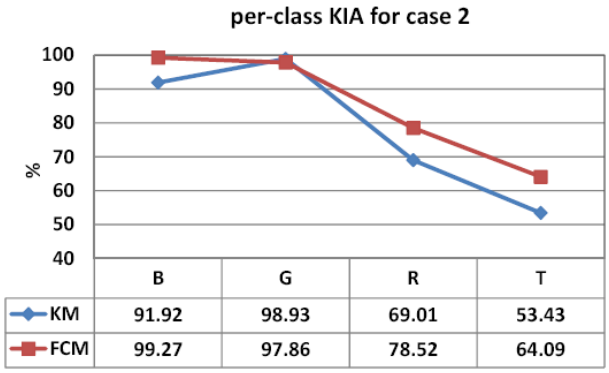


Figure 8: Per-class Kappa results obtained for KMGA and FCMGA in case of RGB/LIDAR dataset. B: Building; G: Grass; R: Roads; and T: trees.

In case of RGB/LIDAR/Attributes data, most of the Per-class Kappa is improved by the FCMGA. Whereas the application of FCMGA resulted in average Per-class Kappa of 87.92 %, the application of KMGA resulted in average of 60.96 %. The grass class had the greatest increase from 28.58% using the KMGA to 89.06% with the FCMGA. Another advantage of the FCMGA

over KMGA is that the achieved errors are less variable. Whereas the application of FCMGA resulted in standard deviation of 4.41 for Per-class Kappa, the application of KMGA resulted in a SD of 13.56.

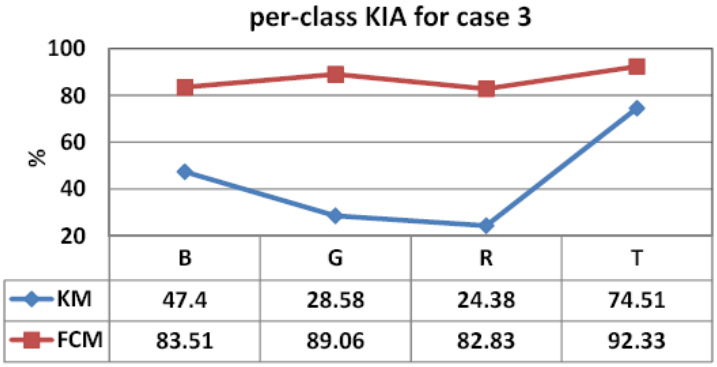
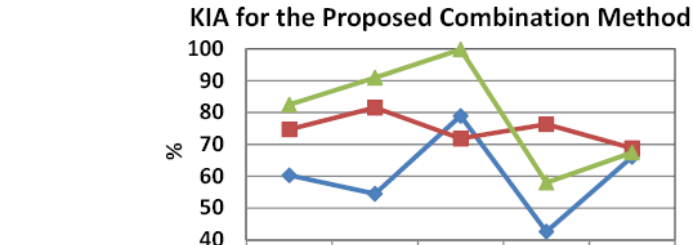


Figure 9: Per-class Kappa results obtained for KMGA and FCMGA in case of RGB/DSM/Intensity/nDSM/polymorphic texture strength dataset. B: Building; G: Grass; R: Roads; and T: trees.

5.3. Combined results

The improvement in overall Kappa achieved by the combination method compared with the KMGA and FCMGA was determined as shown in figure 9. It is clear that the overall performances of the combined method are better than those of the KMGA and FCMGA. It can be seen that a considerable amount of the misclassified pixels have been recovered by the combined classification process. It can be concluded that the application of the combination process results in the most significant improvement in classification accuracy. The strengths of each classifier has compensated for the weaknesses of the other. In general fusing KMGA and FCMGA improve classification accuracies. This demonstrates the benefit of combining different sensor sources at different classification levels.

Some of the Per-class Kappa is improved by the combination method. Whereas the application of KMGA and FCMGA resulted in average Per-class Kappa of 75.18 and 60.29% respectively, the application of combined method resulted in average of 77.69%. The grass class had the greatest increase from 71.84% using the FCMGA to 99.84% with the combined method. Contrary, there was a decrease in Per-class Kappa for the roads and trees classes from 76.36 and 68.61% for roads and trees respectively, using the FCMGA, to 58 and 64.41% with the combined method. As a result, the achieved errors are more variable. Whereas the application of FCMGA resulted in standard deviation of 4.41 for Per-class Kappa, the application of the combined method resulted in a SD of 13.56. However, no patterns can be derived that demonstrates that a certain classifier is better for a particular class. If a particular class is very important, FCMGA, KMGA and the combined results should be tested to select the best result for that class in a given study area. In general, the results indicate that the FCMGA is a more appropriate mapping technique for high resolution imagery. On the other hand, the KMGA method is still an unsatisfactory to classify high resolution images.



	KIA	B	G	R	T
KM	60.31	54.49	78.97	42.6	66.08
FCM	74.7	81.55	71.84	76.36	68.81
Combined Method	82.49	90.96	99.84	58	67.41

Figure 10: Overall and Per-class Kappa results obtained RGB dataset.

The visual interpretation of the final classification result clearly shows a relatively high degree of noise in the KMGA and FCMGA results. In contrast to this, the classification that is based on the combination process appears more homogenous. Within the main land cover classes, some pixels in the aerial image are misclassified, whereas these pixels are correctly classified by the combination process. Figure 11, which is an enlarged portion from Figure 5, is a typical example showing the results from the KMGA classifier in Figure 11(*b*), FCMGA in Figure 11(*c*), and the combined results in Figure 11(*d*). The misclassified pixels are corrected and the noise is significantly reduced by the combination process. This clearly illustrates the different and complementary information provided by the two classifiers.

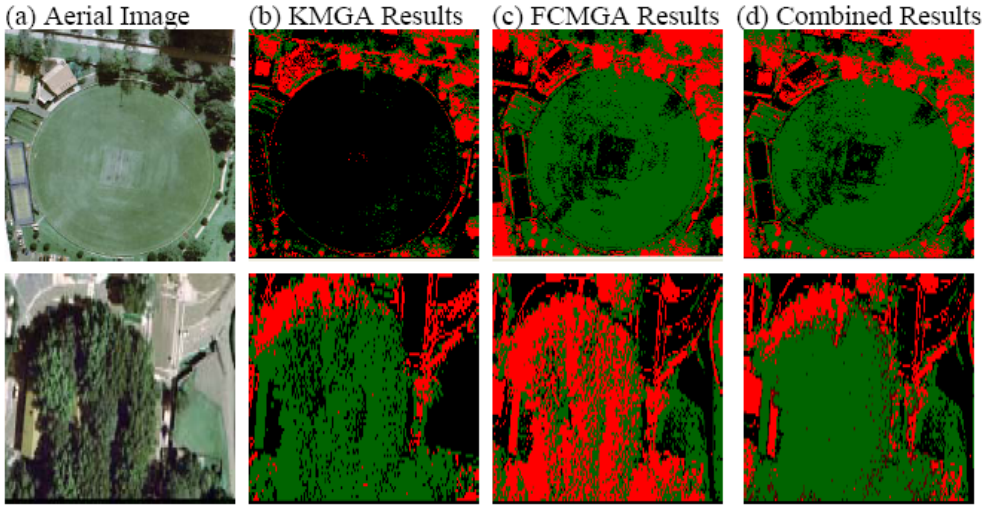


Figure 11: (a) Aerial Image, (b) Classification results of the KMGA; (c) classification results of the FCMGA; (d) Error correction after applying the fusion algorithm.

6. Linear feature extraction and graphic representation

As a result of GA implementation, Group3/FCMGA yields land cover maps in the form of pixels. Although these data sometimes are used directly, standard practice is to convert this form of data into vector format that are considerably more useful for the display of digital spatial information and in the computation of spatial information such as coordinates, lengths, directions, perimeters and areas. This section describes a method of extracting the final vector building maps.

6. 1. Post classification smoothing

In order to extract buildings from the classified data, the classified image is converted to a binary image, with pixels representing buildings displayed with one, and non-building pixels (roads, trees and ground) as zeros. Figure 12 shows the result of this step:

Then, the smaller raster homogeneous building regions were merged into the larger neighboring homogeneous regions or deleted according to an arbitrary 1m distance and 30m^2 area thresholds, respectively. The area threshold represents the expected minimum building area that can be reliably extracted, while the distance threshold was set to 1m to fill in any gaps produced through the classification process. Building regions were retained if at least one of two conditions was satisfied: (1) the building region was larger than the predefined area threshold; and/or (2) more than 1m distance from a larger homogeneous region. As a last step, building borders were cleaned by removing structures that were smaller than 8 pixels and that were connected to building borders. There was a compromise between cleaning thresholds less than 8 pixels, which may leave the original buildings unclean, and thresholds greater than 8 pixels which may remove parts of the buildings. The result was a black and white image that represents the detected buildings without noisy features and also without holes as shown in Figure 13. Finally, the 2-D binary image was labeled, with the background pixels labeled as 0, and the pixels representing each building labeled consecutively from one onwards.

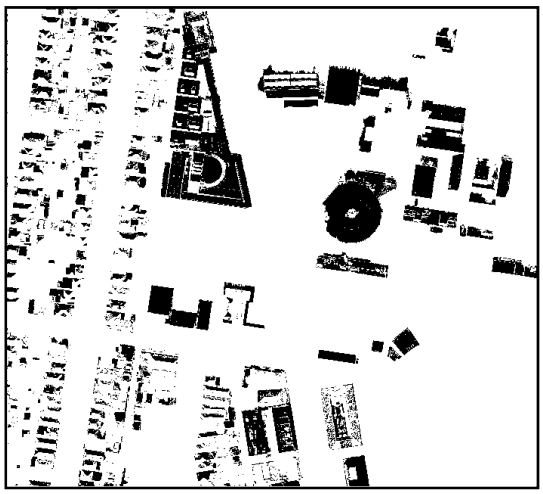


Figure 12: The initial binary image for UNSW test area.

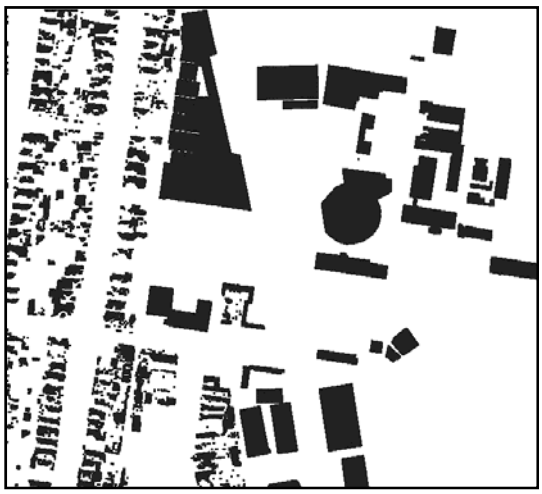


Figure 13: The final labeled 2-D binary image for UNSW test area.

6.2. Boundary extraction

Starting from the upper left corner of the image and using a 3 X 3 moving window, the exterior boundaries of buildings are extracted in the labeled binary image. The central pixel of the window is considered to be part of the perimeter if it is nonzero and it is connected to at least one zero-valued pixel, as shown in Figure 14. This returns a two column vector containing only the rows and columns of the perimeter pixels of each building in the binary image.

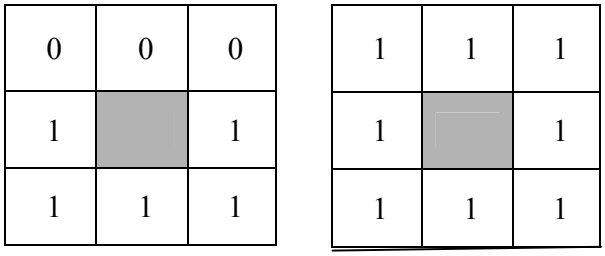


Figure 14: Perimeter pixel (left), non-perimeter pixel (right).

6.3. Simplification of the extracted boundary

The extracted boundaries of buildings are still noisy vectors because of the pixel structure of the raster image. Therefore, the extracted building boundaries are processed again through a line simplification routine. This process simplifies the shapes of buildings to remove unnecessary or unwanted details, while maintaining their essential shape and size.

There are several algorithms to produce a simplified polyline that approximates the original one within a specified tolerance. The Douglas–Peucker algorithm, which has been applied her, is the commonly used algorithm for this purpose, since it is fast and works well for 2D planar polylines. The initial form of the algorithm was suggested by Ramer (1972) and by Douglas and Peucker (1973). Initially, the algorithm automatically marks the first and last point to be kept. It then finds the point that is furthest from the line segment defined by the first and last points. If this point is greater than the reduction toleranceε from the line segment then that point must be kept. All points not currently marked to be kept are discarded. Using recursion, this process continues for each edge until all vertices of the original polyline are within ε. When the recursion process is completed a new output is generated consisting of only those points that have been marked as kept. Based on the experiments in this work it can be concluded that the tolerance can be initially specified equal to the pixel size of the data. If the output still contains too much detail, then the tolerance can be doubled and so on. Similarly, if the output lines do not have enough detail, the tolerance can be halved.

6.4. Adjusting the parallelism and rectangularity of buildings

A method based on Freeman chain codes has been proposed to enhance the parallelism and rectangularity of buildings. The method starts with: aligning one of the coordinate axes, x-axes in this case, with the major axis of the building (length of the building); then the building covered

with a grid of an arbitrary resolution, 1% of the building width in this case. This small resolution provides accurate shapes with minimum deviations from the original shapes. Finally, the grid points closest to the boundary were recorded. Based on this method, the boundaries of buildings are enhanced so that all near-90-degree angles become exactly 90 degrees.

Usually buildings have regular shapes, either squares or rectangles with few exceptions, such as buildings with curved boundaries. A common shape measure called the circularity ratio has been applied in these experiments to test the circularity of each building before adjusting the rectangularity. Circularity is the ratio of the area of the shape to the area of a circle having the same perimeter. That ratio is expressed mathematically as $M = 4*\pi*(\text{area})/ (\text{perimeter})^2$. For a circle, the ratio is one; for a square, it is $\pi/4$; for an infinitely long and narrow shape, it is zero (Bottema, 2000). The area is considered as the number pixels of each building multiplied by the pixel size. The circularity ratio is close to one for buildings with curved boundaries or with circular shape. Based on this fact, these buildings are not processed by the rectangularity test, but still considered in the simplification test. Figure 16 show the extracted buildings after the simplification and adjustment of the rectangularity.

Figure 16: Buildings map after adjusting the rectangularity in case of UNSW test area.

6.5. Evaluation of the planimetric accuracy of the final vector buildings

GPS observations were carried out and compared against the extracted building data coordinates as shown in Table 4. It is worth mentioning that only five of the ten GCPs can be observed in the extracted buildings. This is because small buildings as well as buildings covered by trees were poorly extracted using the proposed method.

Figure 17: Spatial distribution of GPS check points

Table 4: The accuracy of the building vectorization process in case of UNSW test area.

E_{map}	N_{map}	E_{GPS}	N_{GPS}	ΔE	ΔN	$\sqrt{\Delta E^2 + \Delta N^2}$
336012.74	6245795.14	336011.44	6245794.54	-1.302	-0.598	1.4328
336047.63	6245505.55	336048.85	6245505.86	1.221	0.312	1.2602
336195.61	6245776.08	336196.64	6245777.47	1.030	1.394	1.7332
336163.60	6245702.81	336163.01	6245701.74	-0.588	-1.068	1.2192
336193.57	6245663.30	336193.18	6245663.48	-0.394	0.182	0.4340

The mean error of the GPS points in UNSW test area as shown in Table 6.4, is 1.216m ± 0.431m, which is consistent with the standard of 1:5000 scale mapping (1.25m).

7. Conclusions

This study tried to optimize and validate GAs for unsupervised classification of very high resolution image. GA provides a possibility to compute the number of clusters present in a scene from the image data by using a particular fitness function. Experimental results were obtained by classifying a high resolution scene for a part of the region surrounding the University of New South Wales campus, Sydney Australia, depicting three different classes, namely buildings, roads and trees with the KMI as a fitness function while varying a number of parameters of the GA. Compared with the reference data, the results were evaluated based on a variety of criteria which includes visual inspection, error matrices and the K-HAT statistics. The results showed that, the GA-KMI model is much more sensitive for parameter tuning. The results were improved considerably in cases of Population size 100 and mutation probability 0.05, 0.03 and 0.09. The best results were obtained in the case of Population size 100 with mutation probability 0.05 with overall accuracy of 68.89%. Thus, GA algorithms seem to be more flexible and therefore advantageous to more traditional unsupervised classification techniques. In order to improve the

results achieved from this study, different data sources have been extended simultaneously. Also, experiment has been done with different fitness functions to get best results. Finally, we want to integrate indices based on fuzzy theory into the investigations. The results of all these researches are given in Hamdy (2015).

The study tried to optimize and validate GAs for unsupervised classification of high resolution digital imagery and LIDAR data. Two techniques have been applied: KMGA; and FCMGA. As well, the results obtained for both techniques were combined in order to improve the accuracy. Compared with the reference data, the results were evaluated based on a variety of criteria which includes visual inspection and the K-HAT statistics. Three groups of data were tested and evaluated. RGB dataset resulted in overall kappa of 60.31% in case of KMGA, and 74.70% in case of FCMGA. RGB/DSM introduced overall kappa of 70.20% in case of KMGA, and 78.54% in case of FCMGA. RGB/DSM/Intensity/nDSM/ polymorphic texture strength achieved overall accuracy of 45.82% in case KMGA, and 87.84% in case of FCMGA. The combination of RGB/KMGA and RGB/FCMGA has improved the results and resulted in overall kappa of 82.49%. On the basis of the result drawn by this experiment it may be safely stated that the FCMGA outperformed the KMGA. In the future it is planned to extend the study to process more and larger scenes and data sources in order to confirm the results found so far.

During the vectorization phase, the classified images were processed through a series of image processing techniques to produce the digital vector map. An accurate estimation of buildings was carried out from the reference data and compared against the corresponding results. From the GPS observations that carried out and compared against the extracted building data coordinates, the mean error of the observed points is 1.22m with standard deviation of 0.43m.

8. Acknowledgements

The authors would like to thank the AAMHatch (www.aamhatch.com.au) and the School of Surveying and Spatial Information Systems, The University of New South Wales, Sydney, Australia for providing the multispectral aerial imagery.

References

Waske, B. and Benediktsson, J.A. (2007) *Fusion of Support Vector Machines for Classification of Multisensor Data.* IEEE Transactions on Geoscience and Remote Sensing, 45(12-1), December 2007.

Holland, J. H. (1975) *Adaption in natural artificial systems*. Ann Arbor, MI: The University of Michigan Press.

Coley, A D. (1999) *An Introduction to Genetic Algorithms for Scientists and Engineers.* World Scientific, Singapore, 188p.

Rothlauf, F. (2006) *Representations for Genetic and Evolutionary Algorithms*. Springer, Netherlands, 314p.

Huang, C.L. and Wang, C.J. (2006) *A GA-based Feature Selection and Parameters Optimization for Support Vector Machines.* Expert Systems with Applications, 31, 231-240.

Zhou, M., Shu, J. and Chen, Z. (2010) *Classification of hyperspetral remote sensing image based on Genetic Algorithm and SVM.* Remote Sensing & Modeling of Ecosystems for Sustainability VII, Proc. of SPIE Vol.7809, 78090A, doi:10.1117/12.860153.

Almeida, C. M. (2012) *Genetic Algorithms and Data Mining Applied to Optical Orbital and Lidar Data for Object-Based Classification of Urban Land Cover.* Proceedings of the 4th GEOBIA, May 7-9, 2012 - Rio de Janeiro - Brazil. p.649.

Chu, H. T. (2012) *Combination of Genetic Algorothm and Dempster-Shafer Theory of Evidence for Land Cover Classification Using Integration of SAR and Optical Satellite Imagery.* International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XXXIX-B7, 2012 XXII ISPRS Congress, 25 August – 01 September 2012, Melbourne, Australia.

Ge, L. (2012) *Application of Genetic Algorithm and Support Vector Machine in Classification of Multisource Remote Sensing Data.* International Journal of Remote Sensing Applications Sept. 2012, Vol. 2 Iss. 3, PP. 1-11.

Li, P. X. (2013) *High Resolution POLSAR Image Classification Based on Genetic Algorithm and Support Vector Machine.* International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-7/W1, 3rd ISPRS IWIDF 2013, 20 – 22 August 2013, Antu, Jilin Province, PR China.

Yang, Y. F. (2006) *Genetic Algorithms for the Unsupervised Classification of Satellite Images.* Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Nienburger Str. 1, D-30167 Hannover, Germany.

Jamshidpour, N. (2012) *Improvement of Hyperspectral Image Classification Using Genetic Algorithm for Feature Selection and SVMs Parameters Optimization.* Geomatics Department, College of Engineering, University of Tehran North Kargar St., Tehran, Iran.

Samaher, H. A. (2007) *Object Oriented Classification of Forest Images UsingSoft Computing Approach.* 4rth International Conference: Sciences of Electronic, Technologies of Information and Telecommunications March 25-29, 2007 – TUNISIA.

Bandyopadhyay, S. and Maulik, U. (2003) *Genetic Clustering for Automatic Evolution of Clusters and Application to Image Classification.* IEEE pattern recognition, Vol.35, pp.1197-1208.

Majida A. A., Ahmad N. I. and ZubadiMatizHazi (2010) *Pattern Recognition Using Genetic Algorithm.* International Journal of Computer and Electrical Engineering. Department of Computer science, university of Tikrit, Iraq, Vol. 2, No. 3.

Tan, f. and X. Fu. (2008) *A Genetic Algorithm-Based Method for Feature Subset Selection.* Soft Compute, p.p 111-120.

Gen, M. and Cheng, R. (1997) *Genetic algorithms and engineering design.* John Wiley & Sons, New York. 379p.

Pham, D.T. and Karaboga, D. (2000) *Intelligent Optimisation Technique.* Springer, London, Great Britain, 337p.

Hamdy A. (2015) *GA for Feature Extraction from Fused Aerial Images and Other Data Sources.* Ph.D. Thesis to be submitted to Civil Eng. Dept. Azhar University, Cairo, Egypt.
Salah, M. and Trinder, J. (2010) *Support vector machines based filtering of LIDAR data: a grid based method*. FIG Congress 2010, Sydney, Australia.

Zhou, W., Troy, A. and Grove, J. M. (2008) *Object-based land-cover classification and change analysis in the baltimore metropolitan area using multi-temporal high resolution remote sensing data*, Sensors; Vol. 8: pp.1613–1636.

Lu, D. and Weng, Q. (2007) *A Survey of Image Classification Methods and Techniques for Improving Classification Performance.* International Journal of Remote Sensing, 28(5): pp. 823 - 870.
Hall, L., Ozyurt, B. and Bezdek, J. (1999) *Clustering with a genetically optimized approach*. IEEE Transactions on Evolutionary computation, Vol. 3, No. 2, pp 103-112.

Yang, M.S. and Wu, K.L. (2006) *A New Validity Index for Fuzzy Clustering*, IEEE International Fuzzy Systems Conference, pp. 89-92.
Halder, A., Pramanik, S. and Kar, A. (2011) *Dynamic image segmentation using fuzzy C-means based genetic algorithm*. International Journal of Computer Applications (0975 – 8887) Volume 28– No.6.

Turi, R. H. (2001) *Clustering-Based Color Image Segmentation*. Ph.D. Thesis, Monash University, Australia.
Congalton, R. G. and Mead, R. A. (1983) *A Quantitative Method to Test for Consistency and Correctness in Photo Interpretation*. Photogrammetric Engineering and Remote Sensing 49, pp 69–74.
Rosenfield, G. H. and Fitzpatrick-Lins, K. (1986) *A coefficient of Agreement as A Measure of Thematic Classification Accuracy*. Photogrammetric Engineering and Remote Sensing, 52(2):223-227.
Ramer (1972) *An Iterative Procedure for the Polygonal Approximation of Plane Curve*. Computer Graphics and Image Processing, 1(3), 244–256 (DOI: 10.1016/S0146-664X (72)80017-0).
Douglas, D. and Peucker, T. (1973) *Algorithms for the Reduction of the Number of Points Required to Represent A Digitized Line or Its Caricature.* The Canadian Cartographer 10(2), 112–122 (1973) (DOI: 10.3138/FM57-6770-U75U-7727).

Bottema, M. J. (2000) *Circularity of Objects in Images*. in: Proceedings of the 2000 IEE International Conference on Acoustics, Speech, and Signal Processing, June 2000, Istanbul, Turkey, pp. 2247-2250.