Automated Road Extraction from Satellite Imagery Using Hybrid Genetic Algorithms and Cluster Analysis

H. Liu[1*], J. Li[1] and M. A. Chapman[1]

[1]CFI Virtual Environment Laboratory, Department of Civil Engineering, Ryerson University
350 Victoria Street, Toronto, Ontario M5B 2K3, Canada

ABSTRACT. This paper presents a new approach to road extraction from high-resolution satellite imagery based on Genetic Algorithms with fitness calculation of clustering. Traditional segmentation techniques which use clustering require human interaction to fine-tune the clustering algorithm parameters and select good clusters. The proposed approach applies Generic Algorithms to learn the parameters and pick up good clusters automatically. The approach is demonstrated on pansharpened QuickBird imagery and preliminary results are encouraging.

Keywords: Cluster analysis, fuzzy c-means, Genetic Algorithms, road extraction, satellite image processing

1. Introduction

Digital road information is required for a variety of applications ranging from provision of basic topographic infrastructure over transportation planning, traffic and fleet management and optimization, car navigation systems, location-based services, tourism, to web-based emergency response applications and virtual environments. Unsupervised extraction of roads from satellite imagery eliminates the need for human operators to perform the time consuming and expensive process of mapping roads from aerial photographs. As increasing volumes of high spatial resolution satellite imagery (e.g., Ikonos, QuickBird, OrbView-3, etc.) become available, much of which have never even been viewed. What is urgently needed is automation to extract information and analyze image content. It is this problem that motivates our research in automated road extraction. Most of the existing road detection algorithms require user assistance to mark both starting and ending points of road segments. Due to the noise sensitivity, asymmetry of the contrast at the both sides of the edges, and the difficulty of obtaining precise edge directions, edge-based methods are inadequate for multispectral satellite imagery. Our main objective is to develop automatic, robust, and computationally feasible road detection and satellite imagery analysis algorithms.

Automated detection and extraction of roads from space-borne remotely sensed imagery has been an active research and development topic for the last twenty years and in the practical experiments for the last five years with the appearance of the world's first commercial remote sensing satellite, 1m resolution Ikonos in 1999. However, no practically useful tools for the extraction of roads from satellite imagery have been developed yet. Recent discussions on different road extraction methods can be found in Dial et al. (2001) Haverkamp (2002), Zhang (2003), Li et al. (2003), Gibson (2003), Zhang and Couloigner (2004), and Dong et al. (2004).

Pixel classification for partitioning different land-cover types is important in the realm of satellite imagery. Satellite imagery usually has a large number of classes with overlapping and non-linear boundaries. For appropriate modelling of non-linear and overlapping class boundaries, utility of an efficient search technique is evident.

Genetic Algorithms are a randomized search and optimization method, based on the principles of evolution and natural selection. For pixel classification, Genetic Algorithms are efficient, adaptive and robust search processes from a set of image points called population, and can improve the ability of the population by generations to solve problems.

In this paper, such an attempt is made to demonstrate the effectiveness of Genetic Algorithms based classifier, called variable string length Genetic Algorithms (VGA) with fitness evaluation by one of the clustering algorithms, i.e., fuzzy c-means (FCM). We call the hybrid Genetic Algorithms in partitioning different land-cover types, VGA-FCM.

Among the various clustering algorithms, FCM is one of the most popular methods used in chromosome expression data analysis since it is suited to deal with imprecise and uncertain nature of image data sets, such as remote sensing images. Clustering algorithms (e.g., FCM) are used to partition chromosome into groups based on the similarity between their colours. Though FCM is a powerful tool for solving many clustering problems, there are drawbacks. It requires that the desired number of clusters be specified in advance. In

---

* Corresponding author: h5liu@ryerson.ca

most situations, the number of clusters is not known in advance, or sometimes, it is difficult to specify any desired number of clusters. The novel method of VGA-FCM method combines FCM algorithm with variable length string genetic algorithms (VGA) to evolve a desired number of clusters and provide the appropriate clustering.

This paper consists of five sections. After this introduction, Section 2 describes Genetic Algorithms and clustering algorithms as part of the background information. Section 3 describes the hybrid clustering GA-VGA-FCM. Section 4 provides some experimental results of road extractions from pansharpened QuickBird imagery. Conclusions are given in Section 5.

2. Background

2.1. Genetic Algorithms

Holland (1973), who was inspired by the theory of evolution, proposed Genetic Algorithms in the early 1970s. Genetic Algorithms have been applied successfully to a large number of problems in automatic design, pattern recognition, robotic control, and even music generation. The gradual adaptation process of natural chromosomes has occurred primarily to produce a fittest individual who can survive in an ever-changing environment. This process in natural genetics is artificially simulated in Genetic Algorithms to find the optimum solution to an optimization problem. Genetic Algorithms apply some of natural evolution mechanisms such as crossover, mutation, and survival of the fittest to optimization and machine learning. Genetic Algorithms provide an efficient search method working on population and have been applied to many problems of optimization and classification (Goldberg, 1989). In the conventional Genetic Algorithms, each solution is presented by fixed length string.
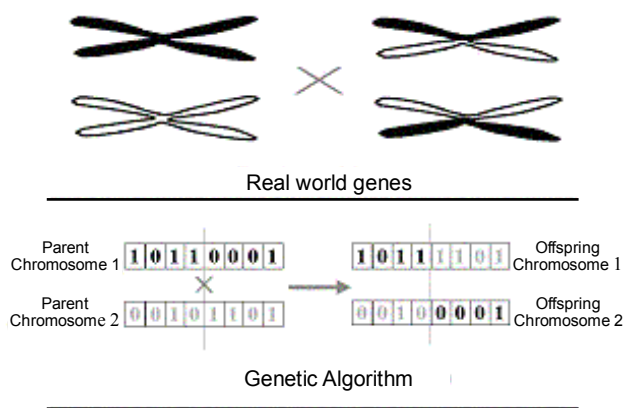

Figure 1. Crossover operation

A general Genetic Algorithms process consists of the following steps (Eberhart et al., 1996):

(1) Initialize the population of chromosomes,

(2) Calculate the fitness for each individual in the population,

(3) Reproduce the individuals selected to form a new population according to each individual's fitness,

(4) Perform crossover and mutation on the population,

(5) Repeat Steps (2) through (4) until some condition is satisfied.

Reproduction makes a copy of a solution. The operator is an artificial version of natural selection. Using reproduction, individual chromosomes are selected according to their fitness, which is evaluated by using an objective function. The result of reproduction means that each individual chromosome with a higher fitness value will have a higher probability to contribute one or more chromosomes in the next generation.

Crossover operation swaps some part of genetic bit string within parents. It emulates just as crossover of genes (chromosomes) in real world that descendants have inherited characteristics from both parents. There are two simple ways in terms of the number of crossover points that are used to crossover operation: single-point and two-point crossover. Figure 1 shows an example of one-point crossover. Mutation operation inverts some bits from whole bit string at very low rate. In real world we can see that some mutants come out. Figure 1 shows the way of applying crossover operation and Figure 2 shows mutation operation to Genetic Algorithms. Each individual in the population evolves to getting higher fitness generation by generation.
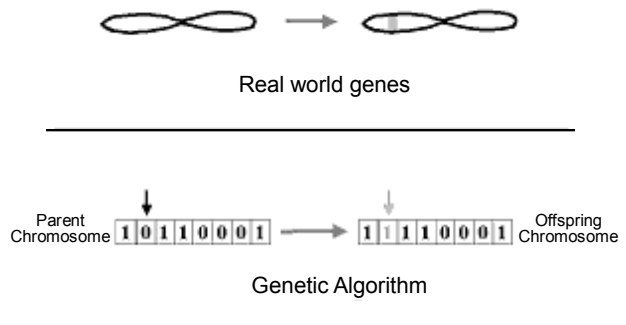

Figure 2. Mutation operation.

A more detailed description of Generic Algorithms can be found in Goldberg (1989), Rawlins (1991), Michalewicz (1994), Adeli and Hung (1995), Chambers (1995), and Mitchell (1996).

2.2. Clustering Algorithms

Clustering is the process of automatically grouping a given set of data into separate clusters such that data points with similar characteristics will belong to the same cluster. While there are many different algorithms for clustering, this paper focuses on the fuzzy c-means algorithm, which is described briefly below.

Clustering algorithm refers to the process of grouping

samples so that the samples are similar within one group (Gose et al., 1996). These groups are called clusters. In applying clustering algorithm, it is very important to decide the similarity measure. Most common methods use distance measures such as city block distance, Euclidean distance, and Minkowski distance. These methods compute the distance from the notation:

$$d_{ij} = d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sqrt[m]{\sum_{k=1}^{p} \left| \boldsymbol{x}_{ik} - \boldsymbol{x}_{jk} \right|^m} \tag{1}$$

where $m = 1$: city block distance, $m = 2$: Euclidean distance, and $m \geq 3$: Minkowski distance.

Three Categories of Clustering

There are three general categories of clustering techniques: hierarchical clustering, partitional clustering, and overlapping clustering.

Hierarchical clustering algorithm constructs a structure of clusters. In this structure a cluster can have several substructures which are composed of other clusters. Hierarchical clustering algorithm takes two different approaches: agglomerative algorithm of a bottom-up approach and divisive algorithm of a top-down approach. Agglomerative algorithm starts with n clusters, consisting of one sample, and continues to merge most similar clusters. On the other hand, divisive algorithm starts with one huge cluster consisting of all individuals and continues to divide them. There are several hierarchical clustering algorithms such as single-linkage algorithm, complete-linkage algorithm, average-linkage algorithm, and Ward's method, which is also called minimum-variance method.

Partitional clustering, different from hierarchical clustering which creates a series of nested clusters, usually creates one set of clusters that partition the data into similar groups. Samples close to one another are assumed to be similar and the goal of the partitional clustering algorithms is to group data that are close together. In many of the partitional algorithms, the number of clusters to be constructed is specified in advance. Hard c-means (HCM) algorithm, k-means algorithm, Forgy's algorithm, and ISODATA algorithm are good examples of partitional clustering (Gose et al., 1996; Haritigan, 1975).

Overlapping clustering algorithm has no hierarchical structure between clusters similar to partitional clustering. However, this approach does not define closed clusters. In overlapping clustering, each cluster can be overlapped partially with others. The FCM algorithm and b-clump algorithm are in this category (Hoeppner et al., 1999; Xie and Beni, 1991). The following describes the FCM algorithm.

Fuzzy C-Means Algorithm

Fuzzy c-means (FCM) clustering is one of well-known unsupervised clustering techniques for image segmentation. For a p by p positive definite matrix, p is the dimen-

sion of the vectors $\boldsymbol{x}_j (j = 1, 2, \cdots, N)$, C is the number of clusters, N is the number of vectors (or data points), fuzzy partition matrix is $U = [\mu_{ij}]$, and $m > 1$ is the fuzziness index. To minimize the objective function J_m

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} \left(\mu_{ij} \right)^m \left\| \boldsymbol{x}_j - \boldsymbol{v}_i \right\|^2 \tag{2}$$

(a) Initialize memberships μ_{ij} of x_i belonging to cluster i such that

$$\sum_{i=1}^{C} \mu_{ij} = 1 \tag{3}$$

(b) Compute the fuzzy centroid $\boldsymbol{v}_i$ for $i = 1, 2, \cdots, C$ using

$$\boldsymbol{v}_i = \frac{\sum_{j=1}^{N} (\mu_{ij})^m \boldsymbol{x}_j}{\sum_{j=1}^{N} (\mu_{ij})^m} \tag{4}$$

(c) Update the fuzzy membership μ_{ij} using

$$\mu_{ij} = \frac{\left(\dfrac{1}{d^2(\boldsymbol{x}_j, \boldsymbol{v}_i)} \right)^{\frac{1}{(m-1)}}}{\sum_{i=1}^{C} \left(\dfrac{1}{d^2(\boldsymbol{x}_j, \boldsymbol{v}_i)} \right)^{\frac{1}{(m-1)}}} \tag{5}$$

(d) Repeat Steps (b) and (c) until the value of the objective function J_m is no longer decreasing.

Clustering techniques allow the division into subgroups to be done automatically, without any preconception about what kinds of groupings should be found. Cluster analysis has been applied in many fields. In VGA-FCM, clustering is used to find the groups of pixels with the similar gray levels and colours, in order to evaluate the fitness of each chromosome in the population.

3. Vga-Fcm Method

3.1. Data Set

In this study, pansharpened colour QuickBird satellite imagery with a spatial resolution of 61 cm was used as input data set. To perform colour image processing, a colour model must be selected. The purpose of a colour model is to facilitate the specification of image colours in a standard manner. Basically, a colour model is a specification of a three-dimensional (3-D) coordinate system, i.e., red, green, and blue (RGB colour space). As mentioned above, the RGB model describes a colour in terms of the percentage of red, green, and blue. The representation of a satellite image has each

pixel specified by three 8 bit (24 bits total) colour values (ranging from 0-255) defining the amount of red, green, and blue, respectively, in each pixel. For example, combining 100% red, 100% green, and 100% blue creates white, i.e., <255, 255, 255> in RGB values. Conversely, mixing 0% red, 0% green, and 0% blue contributes to black, <0, 0, 0> in RGB values.

According to the RGB colour model, a satellite image can be represented with N-array 3-D vector set in the RGB colour space. In the vector $X = \{x_1, x_2, \cdots, x_N\}$, N is the number of pixels in a colour satellite image, and $x_j = \{x_{j1}, x_{j2}, x_{j3}\} \in X$ is a 3-D vector. $x_{jk}(k=1,2,3)$ is coordinate value of jth pixel in the RGB colour space and takes the values from 0 to 255 discretely.

3.2. Proposed VGA-FCM Method

Overview: The method presented in this paper maintains a pool of solutions that evolve in parallel. Figure 3 illustrates the flowchart of the core algorithm. After initializing each solution, the proposed method enters a cycle that repeats until the stopping criterion is satisfied. Each time the cycle completes, a generation has passed.
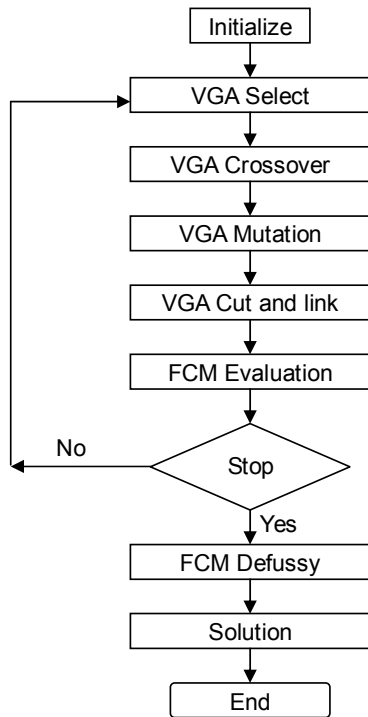

Figure 3. Diagram of the proposed VGA-FCM approach.

Chromosome Representation: In conventional Genetic Algorithms, the optimization problem's variables are usually encoded as binary strings depending on the nature of the problem variables. In our method, the chromosome is directly characterized as a real-coded string. The string is an array of

3-D vectors, which encode the centres of clusters in the RGB colour space. Each element of the vectors representing a cluster centre is an integer that indicates three primary colours in some colour in the RGB colour space between 0 to 255.

As an example the vector string is shown in Figure 4. This example string indicates that there are five clusters and their centres locate on (0, 0, 0), (128, 255, 255), (255, 0, 0), (0, 255, 0), and (0, 0, 255), respectively, in the RGB colour space.

If a string encode C_i cluster centres, then the length of the string is $l_i = 3 * C_i$.

Population Initialization: To generate an initial population, let the population size be M. It is one of the major Generic Algorithms control parameters. In this paper the population size is chosen as an empirical parameter, and the value used has been set after a number of empirical tests. The two special strings that encode the two cluster centres and an estimated maximum number of clusters (C_{max}) are selected into the initial population, to ensure sufficient diversity in the population. And then the lengths of those two strings are 6 and $3 * C_{max}$ bits integers. In theory, the value of C_{max} equals to the number of the entire pixels in an analyzed image, it means that each pixel in the image belongs to different clusters. Virtually, the objects in some scene are limited, so we select the parameter at much smaller value than the number of all pixels. The remaining strings in the initial population are generated randomly, depending on the number of cluster centres C_i ranging from [2, C_{max}], and the value of element of each centre vector ranging from 0 to 255.

Fitness Computation: In order to use the FCM algorithm in Genetic Algorithms, it is necessary to rewrite the objective function presented by Equation (2). Equation (2) can be reformulated by substitution for U using the first-order necessary condition for U. Specifically, for $m > 1$ as long as $\|x_j - v_i\| > 0 \quad \forall i, j$, we can substitute Equation (5) into Equation (2), resulting in the reformulated FCM objective function as following Equation (6)

$$J_m = \sum_{j=1}^{N}\left(\sum_{i=1}^{C}\|x_j - v_i\|^{1/(1-m)}\right)^{1-m} \tag{6}$$

The fitness of each chromosome in the population is evaluated with objection function presented by Equation (6). For each chromosome, the centres encoded in it are first extracted, and then a partition is obtained by computing the membership of each pixel point to the cluster represented by the centre.

Genetic Operation in VGA-FCM: To implement the algorithm, it is necessary for the genetic operators to be newly defined, since the chromosomes in VGA-FCM have an unfixed length. Four kinds of genetic operators are used in the proposed VGA-FCM method.

Reproduction Operator: This operator selects some chromosomes from the parent population to create next

population according to their fitness values. If the fitness value of a chromosome increases then the probability of selecting the chromosome increases too. The fitness value associated to a given chromosome is computed according to Equation (6). In this paper we use the stochastic method that considers a roulette wheel with an unfixed number of sections. Figure 5 shows this roulette wheel and Table 1 gives the fitness value associated to each chromosome and the copies of each chromosome.
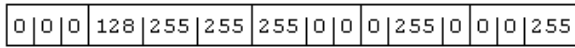
| 0 | 0 | 0 | 128 | 255 | 255 | 255 | 0 | 0 | 0 | 255 | 0 | 0 | 0 | 255 |

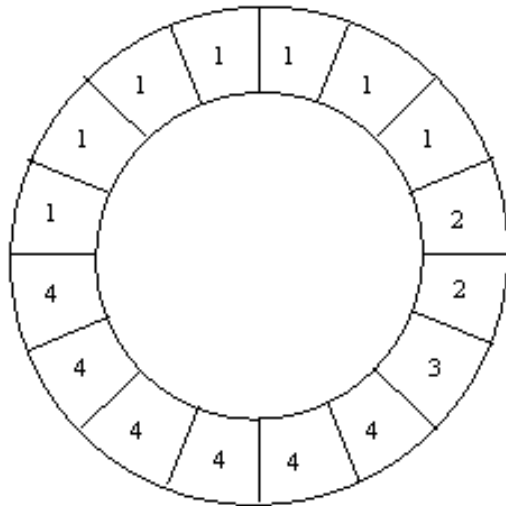**Figure 4.** Example of the cluster centres string.

Figure 5. Illustration of the roulette wheel with sixteen sections for the four chromosomes.

Table 1. Calculate Reproduction Population

Chromosome (i)	Fitness Value (F_i)	Copies ($Cop_i = [F_i / min(F_i)]$)
1	15	7
2	4	2
3	2	1
4	12	6

Figure 5 shows the roulette wheel with sixteen sections for the four chromosomes. The fitness value of a given chromosome determines the number of sections associated to this chromosome. In Table 1, seven copies correspond to the first chromosome, two copies to the second, one to the third,

and six to the fourth.

Crossover: Unlike the conventional Generic Algorithms, the length of string in VGA-FCM is not fixed. To make a crossover between two chromosomes with different lengths, the chromosome with shorter length is extended by randomly inserting null vectors (<#, #, #>) until the same length as the longer chromosome is gained. Two-point crossover is applied to these two strings generating two offspring. In this crossover mechanism, two cuts are made and the portion of the chromosomes between these cuts is swapped. We define that two cut points can only be located between two cluster centre vectors. After crossover, the null vectors in the new chromosomes gained from the crossover operation are eliminated. Figure 7 shows an example of the crossover operation. The crossover operation is applied stochastically with probability μ_c. The value of μ_c is in the range from 0 to 1. In a large population, μ_c gives the fraction of strings actually crossed. If the population size is M, then $M \times \mu_c$ chromosomes suffer the crossover operation. We determine the probability according to our experiment.
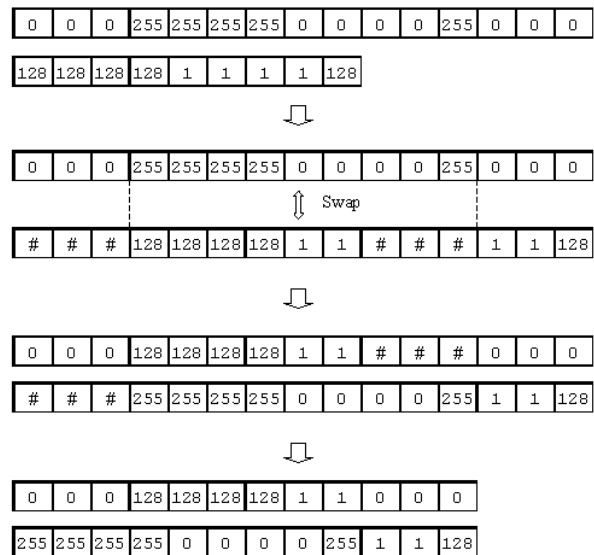

Figure 6. Illustration of two-point crossover operation.

Mutation: After every crossover, some selected chromosomes are considered for mutation. Mutation consists of changing the value of a random bit, which is randomly chosen between 0 and 255, in the chosen string. All strings and bits have the same probability of mutation. Figure 7 illustrates the mutation operation, in which the six number bit of the chosen chromosome is affected by the mutation and the value randomly selected is 128. The mutation rate (μ_m) is the numbers that determinate the mutation that are carried out in a generation. If l is the length of a chromosome and M is the population size, then $\mu_m \times M \times l$ bits change their value in the

generation.

Stopping Criterion: In each iteration, the search can be terminated when there is no significant change between the best fitness values of two successive generations. For example, less than 1%, we choose the difference according to the balance of accuracy and convergence in our experiments.

3.3. VGA-FCM Algorithm with Pseudo Code

In this section, we describe the proposed VGA-FCM algorithm in pseudo code.

```
/* Ng = Number of generations */
/* Pu = Mutation probability */
Procedure VGAFCM ()
{
Initialize ();
For j=1 to Ng
{
VGASelect ();
VGACrossover ();
VGAMutate ();
FCMEvaluate ();
While not end condition do
}
FCMDefuzzy ();
}
Procedure VGASelect()
{
/* Choose parents with probability proportional to fitness value* /
}

Procedure VGACrossover ()
{
/* Perform the crossover operation to generate offspring */
}

Procedure VGAMutate ()
{
/* Apply mutation with Pu */
}

Procedure FCMEvaluate ()
{
/* Calculate fitness value difference between two successive
    generations */
}

Procedure FCMDefuzzy ()
{
/* Convert the image to the clustered image */
}
```

The VGA-FCM algorithm presented in this paper was implemented on a DELL desktop computer using C++ with a user interface. The user can set the parameters for Genetic Algorithms, such as crossover probability μ_c and mutation probability μ_m.

4. Results and Discussion

Several experiments were conducted to demonstrate the performance of the proposed VGA-FCM method with the different parameters selected. The satellite imagery used in the experiments was a subsection of a pansharpened Quick-Bird image with a total of 749,100 pixels (350 x 214), in which 24,910 pixels belong to road regions (see Figure 7). The scene under analysis consists of paved roads, vegetation, and bare soil, which is a typical suburban area in Washington, D. C., the United States. In order to verify the road extraction performance of the proposed VGA-FCM method, road edges obtained through on-screen manually digitizing (white lines shown in Figure 8) were used for overlapping analysis.

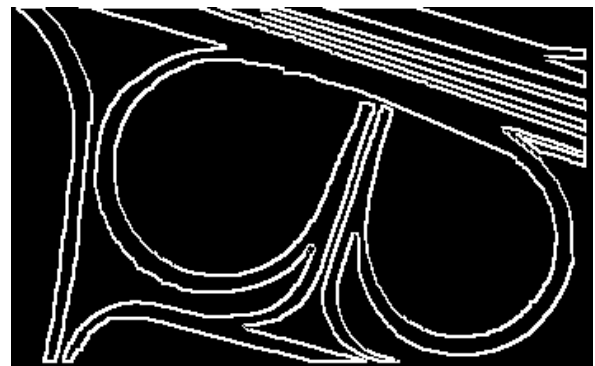**Figure 7.** A pansharpened scene of QuickBird imagery.

Figure 8. Ground-truth data – road edges generated by on-screen manually digitizing (white lines).

Table 2 provides the performance of the VGA-FCM classifier for values of μ_c [10% - 99%], while mutation probability value is kept typical μ_m = 2%. When crossover

probability μ_c is between 30% and 60%, the recognition score shows a good recognition performance.
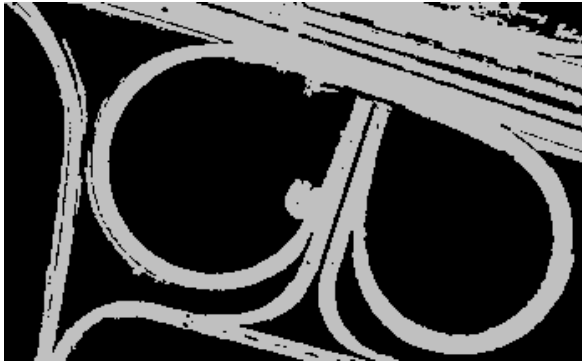

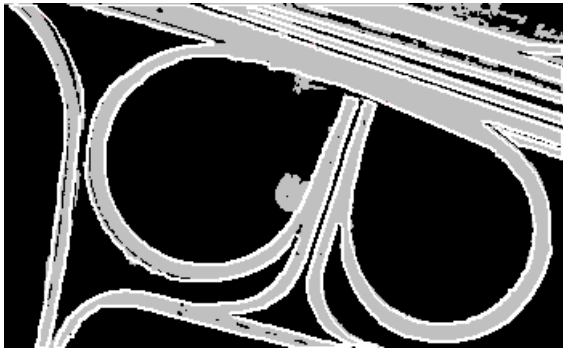**Figure 9.** Extracted road regions (light grey) using the proposed VGA-FCM approach.

Figure 10. Extracted road regions (light grey) overlaid with ground-truth data (white lines).

Table 2. Recognition Score with Different μ_c

μ_c [0,100]%	μ_m [0,100]%	% Recognition Score
10	2.00	79.25
20	2.00	84.46
30	2.00	95.07
40	2.00	97.69
50	2.00	97.08
60	2.00	97.25
70	2.00	93.50
80	2.00	73.17
90	2.00	42.83
99	2.00	30.28

The mutation probability value is kept variable within the range [0.05%, 8.0%]. As we can see from Table 3, the performance of the VGA-FCM classifier is best when $\mu_c =$ 40%.

Table 3. Recognition Score with Different μ_m

μ_c [0,100]%	μ_m [0,100]%	% Recognition Score
40	0.05	76.29
40	0.10	75.49
40	0.15	75.50
40	0.20	75.69
40	0.25	82.01
40	0.30	92.91
40	0.35	93.97
40	0.40	97.30
40	0..45	90.98
40	0.50	90.17
40	1.0	95.95
40	2.0	97.07
40	3.0	96.29
40	4.0	93.14
40	5.0	82.34
40	6.0	69.70
40	7.0	70.14
40	8.0	69.67

Figures 11 and 12 show recognition score with different μ_c and μ_m, respectively. From Figure 11, the recognition is satisfied between 30% and 70%, since the scores are more than 85%. In Figure 12, the satisfied recognition scores are between 0.3% and 4.0%.
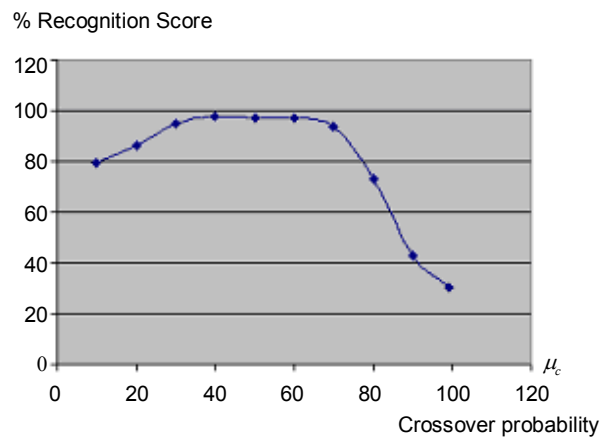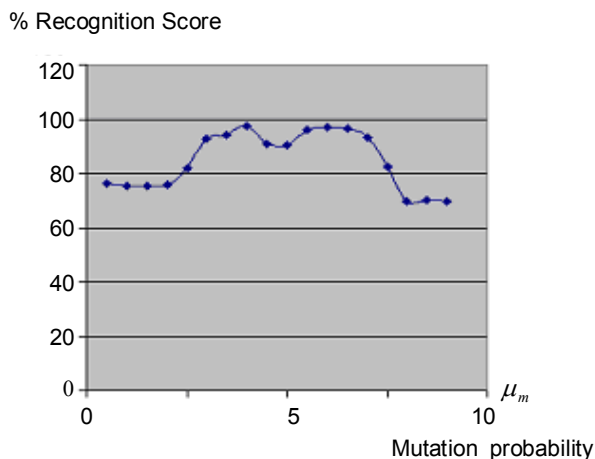

Figure 11. Recognition score with different μ_c.

% Recognition Score

Figure 12. Recognition score with different μ_m.

5. Conclusions

A hybrid approach based on variable length string genetic algorithms and fuzzy c-means clustering (VGA-FCM) has been presented in this paper. The effectiveness of the combination of generic algorithms and cluster analysis is demonstrated by the example of pansharpened colour Quick-Bird imagery. By using the proposed method, it was shown for a single test region that the VGA-FCM algorithm was generally able to distinguish between road and non-road pixels. In addition, the separability between similar classes is improved. Moreover, the segmentation results match better with ground truth. However, the effect of scene type on the road extraction performance needs to be further investigated. To find a fast fuzzy c-means method to calculate the fitness is one priority of the future research, since the fuzzy c-means tend to search all pixels of the image and it costs most of computation time. The chief advantages of this method over traditional clustering algorithms are that it applies Generic Algorithms to learn the parameters and pick up good clusters automatically..

References

Adeli, H. and Hung, S.L. (1995). *Machine Learning-Neural Networks, Genetic Algorithm, and Fuzzy System*, John Wiley & Sons, New York, NY, USA.

Dial, G., Gibson, L. and Poulsen, R. (2001). Ikonos satellite imagery and its use in automated road extraction, *Automated Extraction of Man-Made Objects from Aerial and Space Images (III)*, Balkema Publishers, Lisse, The Netherlands, pp. 349-358.

Dong, H., Li, J. and Chapman, M.A. (2004). A semi-automated approach to extraction of urban road networks from high-resolution satellite imagery, in *Proc. of the Fourth International Symposium on Mobile Mapping Technology*, Kunming, China, 29-31 March 2004 (CD-ROM).

Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*, 1/e, Addison-Wesley, Reading, MA, USA, pp. 412.

Gibson, L. (2003). Finding road networks in Ikonos satellite imagery, in *Proc. of the ASPRS Annual Conference*, Anchorage, AK, 6-9 May 2003 (CD-ROM).

Gose, E., Johnsonbaugh, R. and Jost, S. (1996). *Pattern Recognition and Image Analysis*, Prentice Hall, NJ, USA.

Haritigan, J.A. (1975). *Clustering Algorithms*, John Wiley & Sons, New York, NY, USA.

Haverkamp, D. (2002). Extracting straight road structure in urban environments using Ikonos satellite imagery. *Optical Eng.*, 41(9), 2107-2110.

Holland, J.H. (1973). Genetic Algorithms and the optimal allocation of trials. *SIAM J. on Comput.*, 2(2), 88-105.

Hoppner, F., Klawonn, F., Kruse, R. and Runkler, T. (1999). *Fuzzy Cluster Analysis*, John Wiley & Sons, New York, NY, USA.

Kim, H.S. and Cho, S.B. (2001). An efficient generic algorithm with less fitness evaluation by clustering, in *Proc. of the 2001 IEEE Congress on Evolutionary Computation*, Seoul, Korea, pp. 887-894, 27-30 May 2001.

Li, J., Li, Y. and Dong, H. (2003). Automated extraction of urban road centerlines from Ikonos imagery based on fuzzy mathematical morphology. *Photogramm. Eng. Remote Sens.* (accepted).

Michalewicz, Z. (1994). *Genetic Algorithms + Data Structures = Evolution Programs,* 2/e, Springer-Verlag, Berlin.

Mitchell, M. (1996). *An Introduction to Genetic Algorithm*s, MIT Press, Cambridge, MA, USA.

Rawlins, G.J.E. (Ed.) (1991). *Foundations of Genetic Algorithms,* Morgan Kaufmann.

Xie, X.L. and Beni, G. (1991). A validity measure for fuzzy clustering. *IEEE Trans. on Pattern Anal. Machine Intell.*, 13 (8), 841-847.

Zhang, C. (2003). *Updating of Cartographic Road Databases by Image Analysis*, PhD Dissertation, ETH-Zurich, No.14934, pp.188.

Zhang, Q. and Couloigner, I. (2004). A wavelet approach to road extrcation from high spatial resolution remotely sensed imagery. *Geomatica*, 58(1), 33-39.