Discovering Rules with Genetic Algorithms to Classify Urban Remotely Sensed Data

Sheeren D.[1], Quirin A. [1], Puissant A. [2], Gançarski P. [1] and Weber C. [3]

[1] Sheeren D., Quirin A., Gançarski P.
LSIIT – Data Mining Group
UMR 7005 – CNRS / ULP
Strasbourg, France

[2] Puissant A.
Geosyscom Laboratory
FRE2795–CNRS/University of Caen
Caen, France

[3] Weber C.
Image & Ville Laboratory
UMR 7011 – CNRS / ULP
Strasbourg, France

Abstract— **The classification methods applied in the object-oriented image analysis approach are often based on the use of domain knowledge. A key issue in this approach is the acquisition of this knowledge which is generally implicit and not formalized. In this paper, we examine the possibilities of using genetic programming for the automatic extraction of classification rules from urban remotely sensed data. The method proposed is composed of several steps: segmentation, feature extraction, selection of training sets, acquisition of rules, classification. Features related to the spectral, spatial and contextual properties of the objects are used in the classification procedure. Experiments are made on a Quickbird MS image. The quality of the results shows the effectiveness of the proposed genetic classifier in the object-oriented, knowledge-based approach.**

Keywords-genetic classifier, knowledge-based classification method, object-oriented approach.

I. INTRODUCTION

Since the appearance of the very high resolution sensors and the object-oriented image analysis (OOIA), new questions about the acquisition of knowledge for classification procedures are posed. The OOIA approach is characterised by the extraction of object primitives from images where each object corresponds to a group of homogeneous pixels. The object recognition methods are generally based on the use of knowledge related to spectral, spatial and contextual properties (e.g. mean of spectral and textural values, shape, length, area, adjacency and inclusion relationships…).

While there are several studies that compare object-oriented and pixel-based classification techniques [1,2], only few works focus on the development of the knowledge base used to recognize the objects [3]. However, this is an important issue in the OOIA approach since the information required is generally not formalized. No urban objects dictionary or ontology exists to create the knowledge base. Most of the time, the knowledge is implicit and is held only by the domain experts.

Building a knowledge base in this context is not a trivial task. Previous works in the knowledge acquisition field have already proved that it is still difficult to grasp knowledge directly from the experts, by means of elicitation techniques (i.e. interviews, observations…) [4]. The experts are rarely able to supply an explicit description of the knowledge they use in their reasoning. This is the well-known problem of the *knowledge acquisition bottleneck* [5].

The aim of this study is to examine how data mining techniques and in particular, a genetic programming method, can help to derive this knowledge and to extract classification rules automatically. These rules are intended to enrich an ontology in the urban remote sensing imagery domain.

The paper is organized as follows. In section II, we briefly present the principles of the genetic programming and the algorithm we used to make this study. The methodology and the experiments are detailed in section III. We discuss the results and give concluding remarks in section IV.

II. GENETIC PROGRAMMING

Genetic Programming (GP) is a supervised machine learning technique introduced by Koza in 1992 [6]. It has been applied in a variety of application domains, including image-processing problems. In the field of remote sensing, some works have already demonstrated that this evolution-based process could be useful in particular, to detect and eliminate noisy spectral bands (for hyperspectral images), and to produce comprehensible classification rules [7,8,9,10]. GP can provide relevant and robust rules in terms of classification accuracy. The rules are generally represented in a symbolic language. Thus, they are easily comprehensible and can be revised if necessary by human experts. GP can also be employed in case of incomplete or missing data, without dramatically decreasing the quality of the solution.

A GP algorithm is based on an evolutionary approach. The process starts with a set of randomly created potential solutions (*the individuals*) generated in parallel from a training set given by an expert. Each individual is a classification rule which can be represented in the form of a tree (*chromosome*). All the individuals constitute the *population* which evolves during several generations, until some stopping conditions are satisfied. At each generation, evolution operators are applied to the population to reach the desired solution in an iterative way.

We have developed a high-flexible GP algorithm adapted to remote sensing images, called ProgGen [11]. The working of our algorithm is as follows. First, a set of individuals is created from the learning data, i.e. from the raw and expert samples (initialisation step). Then, the individuals are randomly perturbed to fill in a homogeneous way the search space. Several generational loops are developed until a stopping criterion is triggered. At each generation, a selection operator imitating the natural selection selects individuals for crossover and mutation. Crossover requires two classifiers and cuts their chromosome at some randomly chosen positions to produce two offsprings. After the reproduction, the two new classifiers inherit some rule conditions (e.g. parts of the tree) from each parent-classifier. A crossover operator is used in order to exploit the qualities of already generated classifiers. The mutation operator plays a dual role in the system: it provides and maintains diversity in a population of classifiers, and it can work as a search operator in its own right. The mutation processes a single classification rule and it creates another rule with altered condition structure or variables.

Each offspring produced by the evolution operators has to be assessed according to the learning data. The evaluation function (fitness function) serves to differentiate the quality of generated rules and guide the genetic evolution. Usually, this function depends strongly on the application domain. In our case, fitness measure is based on the simplicity of the trees (e.g. number of tree nodes) and the accuracy computed from the confusion matrix using the learning samples. The individuals are then ranked by their performance, some of them replace the worst individuals from the previous generation (replacement step), and a new generation is done until the program finish.

Each rule is described in the form *if <condition> then <class>*. A condition takes several variables (data attributes), random constants, and returns a *true* or *false* value. In our case study, we have used two kinds of conditions. The first one is based on conjunctions of disjunctions of real-values intervals [9]. The second one is based on symbolic regression [11]. In symbolic regression, the condition is an equation containing simple operands like +, -, *, /. If the equation returns a strictly positive value, the condition is defined as *true* (*false* otherwise).

We have developed a rule for each class (this is called *soft* learning). First, this leads to independent rules that could be combined at different hierarchical level. Second, several rules can be activated, describing several classes included in low-resolution areas.

The ProgGen algorithm is included in highly portable libraries set named VPlat and compiles on various platforms (Windows/Linux/Unix). These libraries are written in C for faster treatments and more extensible capabilities. VPlat can be obtained at the following URL address: [12].

III. Methodology and Experiments

We tested the ProgGen algorithm on a very high resolution (VHR) Quickbird MS image. The area examined is located in the urban area of Strasbourg (France) and has an extent of 42km². The image was collected on May 2002.

A class hierarchy of elementary urban objects composed of three levels has been defined for the experiments. This hierarchy is illustrated in figure 2.

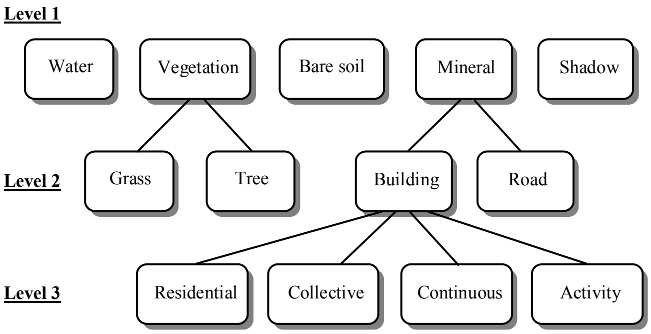

Figure 1. The object classes hierarchy.

A. Segmentation

The VHR-image was segmented to group pixels automatically into homogeneous regions that correspond to geographical objects. The region growing approach proposed by the *eCognition* software (Definiens-Imaging) was used for this task. The parameters of the algorithm were empirically chosen.

The segmentation procedure was applied on the four bands of the image and was supported by a thematic layer. This layer contained buildings coming from a 1m resolution vector database (the BDTOPO from IGN). The introduction of this ancillary data was motivated by the fact that we wanted to discover a set of reusable classification rules based on spectral properties, but also on spatial characteristics. Consequently, it was important to obtain regions with representative shape and dimensions. By introducing this thematic layer, some regions resulting of the segmentation were identical to the polygons of the building theme (fig. 1).

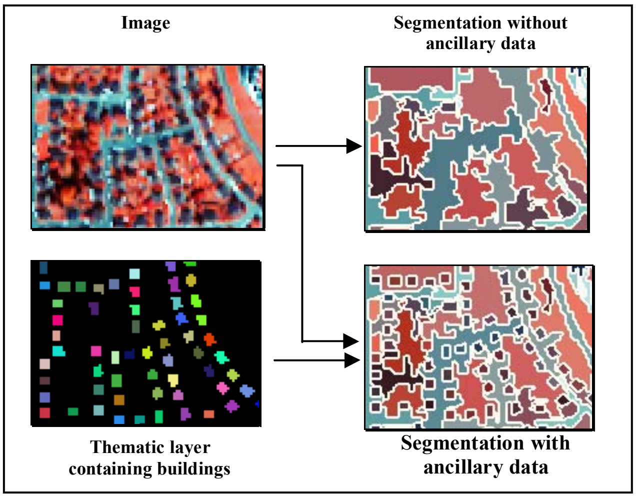
Image
Segmentation without ancillary data
Thematic layer containing buildings
Segmentation with ancillary data

Figure 2. Object extraction supported by a thematic layer.

B. Feature extraction

Each region was characterized by a set a features. For the first two levels of the class hierarchy, we used only the spectral information. We made the assumption that the spectral signatures are sufficient to separate these basic land cover classes. The features retained were as follows: the mean of the spectral values of the regions for each band of the image (blue, green, red, near infrared), the brightness (sum of the mean reflectance values in all bands), and the mean of the values of two index (NDVI and SBI).

For the third level of the class hierarchy, the regions were described with additional information. Confusions can occur between the building classes using the spectral properties of the regions only. Spatial and contextual properties are more relevant to identify the functional character of the buildings. Thus, several shape properties of the regions were computed: area, perimeter, diameter (length of the major axis), compactness (Miller's Index), solidity (ratio of the area to the convex area). The percentage of vegetation around the buildings within a buffer of 20m radius was also determined. All these features were used during the rules acquisition and classification processes.

C. *Selection of training data set and acquisition of rules*

In order to train the genetic classifier, 50 regions of each class were collected and labelled on the basis of expert knowledge. A random sampling method was used to obtain these regions. In this way, a training data set was provided with a representative description of each class.

The genetic classifier was applied on this training set but the classification procedure was made in several steps, in an iterative way. We did not learn the classification rules enabling to discriminate all the classes directly. We rather followed a deductive approach by trying to distinguish successively the objects of one class from all others, starting with the objects easiest to identify. First, we learned rules enabling to recognize the objects 'water'. Thus, the training set was subdivided into two subsets: one containing the samples classified as 'water' and the others reclassified as 'non water'. Then, the set of objects 'non water' were subdivided into 'vegetation' and 'non vegetation'. These last ones were thereafter selected to distinguish successively the classes 'shadow', 'bare soil' and 'mineral'. Finally, this process was continued to discover rules relating to the object classes of levels 2. The learning procedure was therefore split into several steps. In practice, it gives more accurate results since the hypothesis space of the classifier is reduced. After these experiments, another independent test was performed to learn the classification rules relating to the building theme (classes at level 3).

The set of rules provided by the genetic classifier ProgGen is given below. Let us notice that all the features used have values ranging between 0 and 1. The features have been normalized during the learning process. For the following case studies, we used 200 individuals, 500 generations, a crossover rate of 75% and a mutation rate of 15% (parameters of the algorithm).

Class Hierarchy – Level 1:

```
Rule 1:

If mean_IBS < 0.9
And mean_NDVI < 0.14
Then class = water
Else class = non water
```

```
Rule 2:

If mean_IBS < 0.86
And 0.68 < mean_NDVI < 0.99
And mean_brightness < 0.96
Then class = vegetation
Else class = non vegetation
```

```
Rule 3:

If 0 < mean_B1 < 0.8
And 0.04 < mean_B3 < 0.05
And 0 <mean_IBS < 0.04
And 0.02 < mean_NDVI < 0.9
And 0.01 < mean_brightness < 0.96
Then class = shadow
Else class = non shadow
```

```
Rule 4:

If mean_B1 < 0.38
And mean_IBS < 0.93
And 0.3 < mean_NDVI < 0.98
Then class = bare soil
Else class = mineral
```

Class Hierarchy – Level 2:

```
Rule 5:

If 0.07 < mean_B4 < 1
And 0.06 < mean_NDVI < 0.4
And 0.07 < mean_brightness > 0.4
Then class = building
```

```
Rule 6:

If 0.15 < mean_B1 < 0.5
And 0.03 < mean_B2 < 0.97
And 0.25 < mean_B3 < 1
And 0.22 < mean_B4 < 0.98
And mean_IBS < 0.86
And 0.18 < mean_NDVI < 0.26
And 0.04 < brightness < 0.82
Then class = road
```

```
Rule 7:

If 0.02 < mean_B4 < 0.93
And 0.02 < mean_IBS < 0.69
And mean_NDVI < 0.97
Then class = tree
```

```
Rule 8:

If mean_B1 > 0.1
And mean_B2 > 0.1
And mean_B3 > 0.28
And mean_IBS > 0.08
And mean_NDVI > 0.09
And mean_brightness > 0.57
Then class = grass
```

Class Hierarchy – Level 3:

```
Rule 9:

If perimeter < 0.61
And 0.65 < compactness < 1
And diameter < 0.08
Then class = residential B.
```

```
Rule 10:

If area < 0.05
And 0.05 < perimeter < 0.98
And compactness < 0.96
And 0.02 < %_of_V < 0.91
And 0.05 < diameter < 1
Then class = collective
```

```
Rule 11:

If 0.1 < area < 1
And perimeter > 0.84
And 0.3 < compactness < 1
And %_of_V < 0.95
And 0.13 < solidity < 1
And 0.33 < diameter < 1
Then class = activity
```

```
Rule 12:

If 0.05 < area < 0.77
And 0.02 < perimeter < 1
And compactness < 0.46
And %_of_V < 0.25
And solidity < 0.97
And 0.01 < diameter < 0.91
Then class = continuous
```

D. *Classification and results*

The relevance of the learned rules has been verified by introducing them in the *eCognition* software and by applying them on the study area. An excerpt of the classification of the image is illustrated in figure 3.

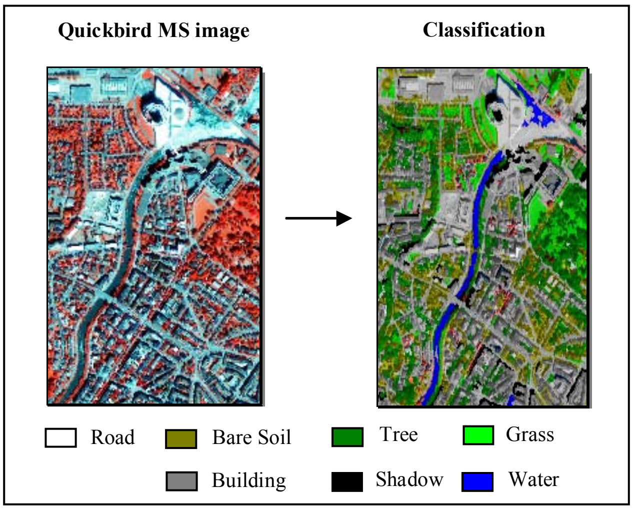

Figure 3. Classification of level 1 and 2.

The accuracy assessment was accomplished with confusion matrices. These matrices have been computed using 50 new test areas derived by visual interpretation. Table 1 shows the results.

TABLE I. CLASSIFICATION ACCURACY ASSESSEMENT (PA = PRODUCER'S ACCURACY; UA = USERS'S ACCURACY ; OA = OVERALL ACCURACY)

Classes (L1 - L2)	**PA**	**UA**	**Classes (L3)**	**PA**	**UA**
Water	0.66%	100%	Residential	0.90%	100%
Grass	0.96%	0.89%	Activity	0.96%	0.75%
Tree	0.86%	0.95%	Collective	0.62%	0.83%
Shadow	0.9%	100%	Continuous	0.88%	0.93%
Bare soil	0.58%	0.80%			
Building	0.92%	0.35%			
Road	--	0.00%			
	OA: O.69%			OA: 0.86%	
	Kappa: 0.64%			Kappa: 0.77%	

Concerning the classes of the first two levels (L1-L2), several important confusions occur. Some water objects are classified as building objects. This is the case when the water is covered with fog. These errors are relatively frequent but well-localized. All the roads are also confused in buildings. These errors are more problematic and could be due to the both fact that we had less road samples than buildings (50 vs. 150) and that a low number of attributes can be used to discriminate each classes (for instance, C4.5, a well known classification algorithm [13] uses only the SBI index and the first band, whereas we try to use all the attributes together). Thus, the learned rule should be revised. Finally, many bare soils are also classified as buildings. This could be explained by the overlapping existing between the spectral ranges.

Results are better for classes at level 3 (the buildings). The accuracy values indicate that the classes are well-discriminated, except for collective buildings. The features used to describe the building objects seem to be adapted.

IV. CONCLUSION

The knowledge acquisition step in the recent OOIA approach is an important issue which has been often neglected. In this paper, we have investigated the possibilities of acquiring classification rules automatically with ProgGen, a new genetic programming method adapted to remote sensing imagery. The results obtained show that ProgGen is a viable approach. However, additional experiments should be made to decrease several confusions between the classes (in particular, concerning the buildings and roads). The spectral signatures are probably not sufficient to separate the basic land cover classes selected. Additional features related to spatial and contextual properties should help to improve the classification rules. The tests we made to discriminate the different kinds of buildings in this way show better classification accuracy.

ACKNOWLEDGMENT

This study is part of the FoDoMuSt (multi-strategies data mining approach to extract and identify urban elements from remote sensing images database) research project financed by the ACI "Masse de Données" (2004-2007).

REFERENCES

[1] Rego L. and Koch B., "Automatic classification of land cover with high resolution data of the Rio de Janeiro city Brazil: comparison between pixel and object classification", Proceedings of the 4th International Symposium on Remote Sensing of Urban Areas, 2003, pp. 1682-1777.

[2] Whiteside T. Ahmad W., "A comparison of object-oriented and pixel-based classification methods for mapping land cover in nothern australia", Proceedings of the SSC2005 Spatial intelligence, innovation and praxis: the national biennal conference of spatial science institute, pp. 1225-1231.

[3] Baltasavias E.P., "Object extraction and revision by image analysis using existing geodata and knowledge: current status and steps toward operational systems," ISPRS Journal of Photogrammetry and Remote Sensing, 2004, vol. 58, pp. 129-151.

[4] David J.-M., Krivine J.-P. and Simmons R. (Eds.), "Second generation expert systems", Springer-Verlag, 1993.

[5] Sester M., "Knowledge acquisition for the automatic interpretation of spatial data", International Journal of Geographical Information Science, 2000, vol. 14, pp. 1-24.

[6] Koza J.R., "Genetic Programming: On the Programming of Computers by Means of Natural Selection", MIT Press, 1992.

[7] Daida J.M, Bersanon-Begey T., Ross S.J. and Vesecky J.F., "Evolving feature-extraction algorithms: adapting genetic programming for image analysis in geoscience and remote sensing," IEEE International Geoscience and Remote Sensing Symposium (IGARSS'96), 1996, pp. 2077-79.

[8] Harvey N., Theiler J., Brumby S., Perkins S., Szymanski J., Bloch J., Porter R., Galassi M., Young A. "Comparison of GENIE and conventional supervised classifiers for multispectrral image feature extraction," IEEE Transactions on Geoscience and Remote Sensing, 2002, vol. 40, pp. 393-404.

[9] Korczak J., Quirin A., "Evolutionary Mining for Image Classification Rules", Proceedings of the 6th International Conference on Artificial Evolution (EA'03), 2003.

[10] Ross B.J., Gualtieri A.G., Fueten F., and Budkewitsch P, "Hyperspectral image analysis using genetic programming," Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'02), pp. 1196-1203, 2002.

[11] Quirin A., "An evolutionary-based approach for classification rules discovery: application to remote sensing imagery" (in French), PhD Thesis in Computer Science, Louis Pasteur University of Strasbourg, 2005.

[12] VPlat 1.3, a set of evolutionary computation libraries, downloadable at http://lsiit.u-strasbg.fr/afd/logiciels/vplat-1.3.zip

[13] Quinlan R., "C4.5: Programs for Machine Learning", Morgan Kaufmann Publishers, San Mateo, CA, 1993.