A platform for Spatial Data Labelling in a Urban Context

Julien Lesbegueries (1), Nicolas Lachiche (1), Agnès Braud (1), Grzegorz Skupinski (1 and 2), Anne Puissant (2) and Julien Perret (3)

Abstract This paper presents a platform based on an opensource framework (Geoxygene) adapted to label urban areas from vector based topographic databases. The development of this platform is part of the GeOpenSim project which aims at analysing and simulating urban dynamics The objective is to detect automatically which elementary areas (urban blocks) can potentially evolve. The proposed labelling process is used in order to characterize and identify several evolution processes of different type of urban fabric. This software provides facilities in the domain of urban planning and management to (i) experimentally refine the modelling of their problem, (ii) collect training data, (iii) automate the labelling.

1 Introduction

The GeOpenSim project aims at developing an opensource framework to study urban evolutions using vector based topographic databases. This framework, based on GeOxygene[1], is composed of several modules spanning from the creation of spatio temporal topographic databases to the simulation of urban dynamics. The novelty of our approach lies in using topographic data for the simulation whereas most research on urban simulation is based on cellular automata or graph cellular automata [2, 3, 4, 6, 9]. Indeed, the analysis of observed urban phenomena using topographic data allows for a more accurate, more realistic approach to urban simulation. Such an analysis is realized at different geographic levels: at the micro level (buildings, roads, etc.) and at several meso geographic levels (urban blocks, districts, cities, etc.). For each on these levels, the context in which the evolutions take place is crucial to the study. The context of an evolution is time-based (evolutions

(1) Université de Strasbourg, LSIIT-FDBT, Bd Sébastien Brant, BP 10413, F-67412 Illkirch Cedex
(2) Université de Strasbourg, LIVE, 3 rue de l'Argonne F-67000 Strasbourg
(3) Laboratoire COGIT, Institut Géographique National, 73, avenue de Paris, 94165 Saint-Mandé Cedex

are different in the 1950s and in the 1970s) as well as spatially-based (evolutions in a peri-urban context are different from the evolutions in an industrial context). Therefore, in the framework of this spatial context, geographic features have to be characterized so they can be labelled depending on their nature. This paper presents our work on the labelling of a specific type of geographic features: elementary areas (or urban blocks).

We developed an opensource add-on to Geoxygene. This labelling add-on fulfills several needs concerning the management of geographic data visualization and labelling. It uses a high-level connection, providing analysis and adding information. Its input consists in geographic data and a list of labels (screenshot in figure 1). Geographic data consists in several layers of topographic data, at micro and meso levels (figure 1 (A)). A target layer is defined in order to be labelled (figure 1 (C)). The other ones are used in the visualization. Moreover a list of labels is defined (figure 1 (D)) and the module generates a dynamic interface with items or sliders for each label.

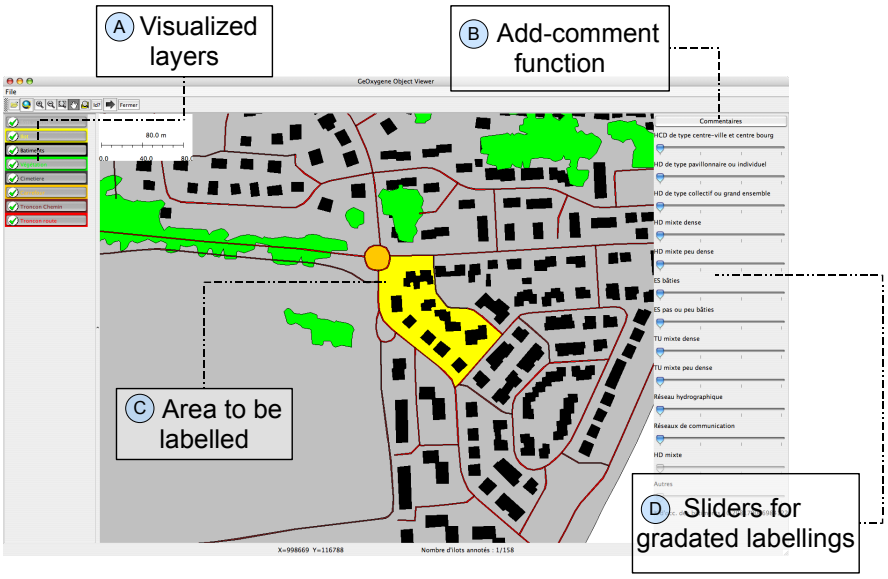


Fig. 1 Global screenshot of the labelling module.

The module provides a labelling process with several options in order to perform various experiments and to fit to different problems:

- several visualizations are provided,
- several ways of labelling,
- several procedures to complete the labelling.

The different options are explained and illustrated with the specific problem of urban labelling.

The figure 2 summarizes the main procedure of labelling. A map visualization is created from a geographic database thanks to high-level connection functions provided by Geoxygene [1] for specific geographic data. A user logs in to the interface and labels selected areas. The labels are stored in a database and export functions produce appropriate output files or output storing in training databases. This data is then used to perform a learning process on the entire dataset. One way is to integrate learning functionalities in our platform with the help of the Open Source Data Mining toolkit Weka [7]. This allows us to analyze learning results directly in the platform.

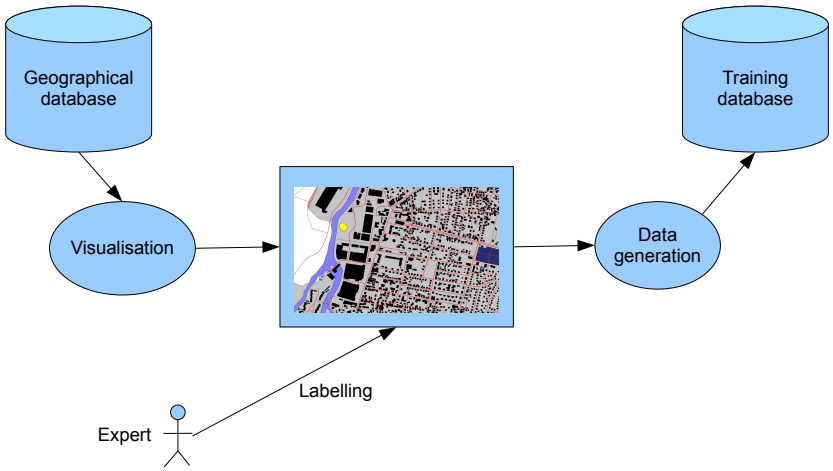


Fig. 2 Labelling procedure schema.

In the next section, facilities concerning the problem modelling are depicted: how to carry out the labelling process and the different ways of labelling. Section 3 presents the labelling acquisition, its analysis and its use for a supervised learning purpose.

2 Modelling the problem

This section describes how the platform helps modelling the problem. A first step consists in defining the geographic input data and once the geographic parcels are defined, the second step consists in defining the labels.

In the framework of our problem, we must define adequate geographic parcels in a city in order to locate potential evolutions. The first step is to partition it in relevant elementary areas (districts). This partitioning must be generic and automatic. Then labels must be defined to categorize these areas according to their evolution

potential. The former function of urban partitioning is performed within Geoxygene and the latter is performed by the labelling module.

2.1 Input data

Input data must be geographic layers stored in a geographic database e.g. PostGIS[1]. One of the layers must be the target one, in which geographic parcels have to be labelled.

To provide a generic labelling module, we chose to use the topographic database, provided by the French National Geographic Institute (IGN). Indeed there is a vector based topographic database, namely the BDTopo©, available for the entire french territory and thus for every french city. This database is composed of layers for each kind of *micro* geographic objects (buildings, vegetation, networks, ...).

The partitioning of the city is computed according to the communication network layers: main roads, country roads, railroads, watercourses. Figure 3 shows the resulting elementary areas that correspond to the spatial unit to label (the yellow one for example).

2.2 Labels definition

The platform allows to dynamically define labels, because they can change all along the labelling campaign. A property file configured *a priori* is used to create a label index in a database (figure 4) and to build the list of sliders (figure 1 (C)).

We illustrate this step on our specific urban problem. Labels defined are:

1. Continuous urban fabric (city center),
2. Discontinuous urban fabric with individual houses,
3. Discontinuous urban fabric with with collective buildings,
4. High density mixed housing surface (mix of 2 and 3),
5. Low density mixed housing surface (mix of 2 and 3),
6. Specific urban surface (industry buildings),
7. Not or little built specific urban surface (industrial wasteland),
8. High density mixed urban surface (mix of 2,3 and 6),
9. Low density mixed urban surface (mix of 2,3 and 6),
10. Hydrographic network (canals, rivers),
11. Communication network (roads, country roads, railroads).

For each label, a slider and a tuple in the database are created. This dynamic building is necessary because the labelling procedure is intrinsically an iterative, possibly backtracking procedure. Next section details this characteristic.

[1] http://postgis.refractions.net/

2.3 Refining the labels

The module provides different functionalities to refine the labelling. A relevant question is the manner labels are collected.

The easiest way is to force the expert to choose one class for each visualized area during the labelling process. However, in the case of a random choice of next areas to be labelled, the expert may have difficulties to label every area, given that the areas generation is automatic and can produce artefacts. An additional class “others” can solve the problem.

A second implemented solution consists in allowing a gradation the expert can express with a confidence degree from 1 to 4. In this case the value of each class is stored for each labelled area.

Another manner to face the difficulty of labelling is to consider areas labels as overlapping classes (the area is no longer “of one kind” but “made of”). Then the expert is allowed to associate several classes by area (with gradations for each class).

Along our labelling campaign, some mixed classes were added in order to disambiguate confusing areas (4, 5 and 8,9).

Fig. 3 Elementary areas (in grey and yellow) are built from communication networks (red lines). The black polygons represent buildings and the green ones represent vegetation.

2.4 Confidence level and exclusive / overlapping classes

This section describes the various labelling functionalities implementation. The user can label in a binary manner or in a more gradated manner. The binary manner can be sufficient for easy-to-label procedures, when human experts haven’t got any doubt. When the labelling procedure consists in a more complex problem, implying confusing areas to label, the gradated manner implemented by sliders and representing a confidence degree can be a solution. By default, the module expects 4 gradations (from 0 to 3). The figure 4 presents a schema of the labelling database (storing binary and gradated labellings). The *labelling index* table stores the expert

identifier, the concerned area identifier and its labelling. In the binary way, the *label_id* column is used whereas in the gradated way it is the *gradated_label_id* one. Then the *Gradated index* stores the sliders values.

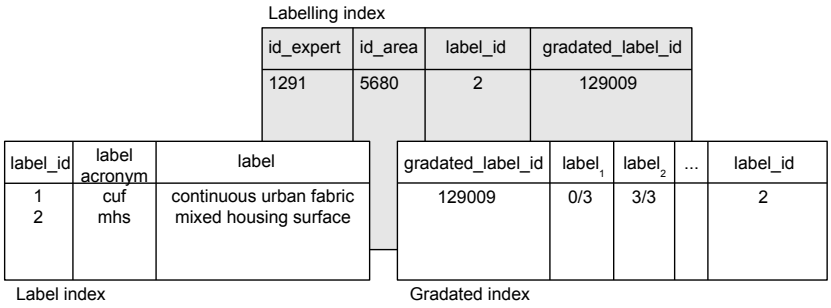


Fig. 4 Schema of the labelling database.

Moreover, the labelling process can be a one label per area labelling or a several labels per area labelling. It depends on the labels definition. Indeed, they can represent exclusive classes or overlapping classes. In our case study, labels can not be exclusive because an area can contain *housing surface* and *specific urban surface* at a time. Our problem turned out to be an exclusive labels case, if an additional specific one *(mixed urban surface)* is defined. Let us emphasize that the labelling structure can manage the 2 solutions. The only change is the function used to determine the *label_id* column of the gradated index (max of the $label_i$ columns for instance).

2.5 Identifying the minimum background

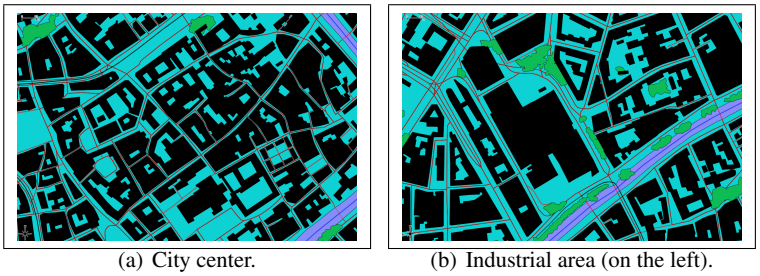
Additionally, the module provides different manners to choose areas to be labeled. Indeed, in a learning perspective, areas have got to be well chosen in order to correspond to a representative set. Several solutions are provided (randomly-based, user-based, active learning).

Once input data and classes are correctly defined, we can experiment several ways of labelling in order to find the best method requiring the minimum information to display, providing however sufficient information to experts and identifying the adequate retrieval of information necessary for the learning task.

Three visualizations are proposed in our platform (figure 6), providing different widths of spatial context:

1. the area to label only (with its buildings),
2. the area to label and its surrounding areas (with their micro objects)
3. the entire map of the city.

In our case study, experts claim that the first visualization is too poor and there is not enough information to make a decision. For example, it is difficult to distinguish a city center area from a specific urban one (industrial area for instance) without the direct neighborhood and their scale (figure 5).



(a) City center. (b) Industrial area (on the left).

Fig. 5 Confusing areas difficult to label.

The two other visualizations provide this additional contextual information. However, there is a problem of *over-contextualisation* for the third one because the visualization of the entire map causes the expert to use his background knowledge of the geographic area (experts could then recognize the displayed city) instead of the conformation of objects in areas.

3 Data processing

This section presents module facilities concerning the automation of the labelling procedure: data acquisition, analysis and visualization, and its use in a learning process.

3.1 Data Acquisition

The problem depicted here concerns the manner areas to label are chosen. Several solutions are proposed on the platform. One solution is to visualize a random area not yet labeled (figure 6(a) and 6(b)). An alternative solution provided consists in allowing the expert to choose the areas to label (figure 6(c)).

Actually the aim of the platform is to include a learning process, able to label the majority of areas from a few examples. With the second solution, experts have to carefully choose the areas. They must not select the ones easy to label only. If the target of the learning process is to label every area, it musts be trained from typical and also fuzzier / tricky / complex areas. Another solution would be to assign this areas selection to the learning process, thanks to active learning methods [5].

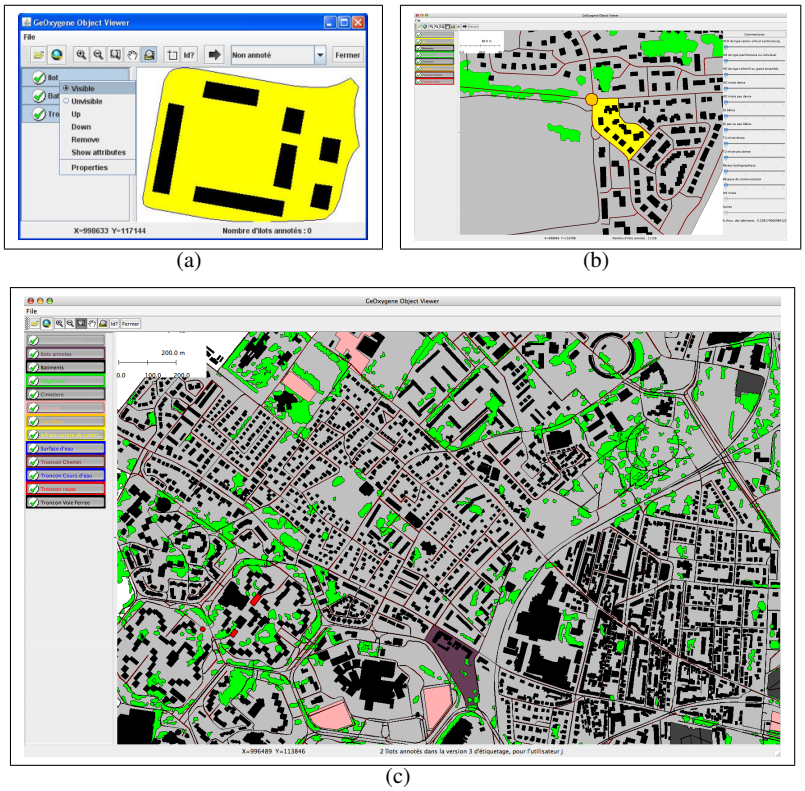


Fig. 6 Three visualization widths offered by the module.

3.2 Analysis and visualization of labellings

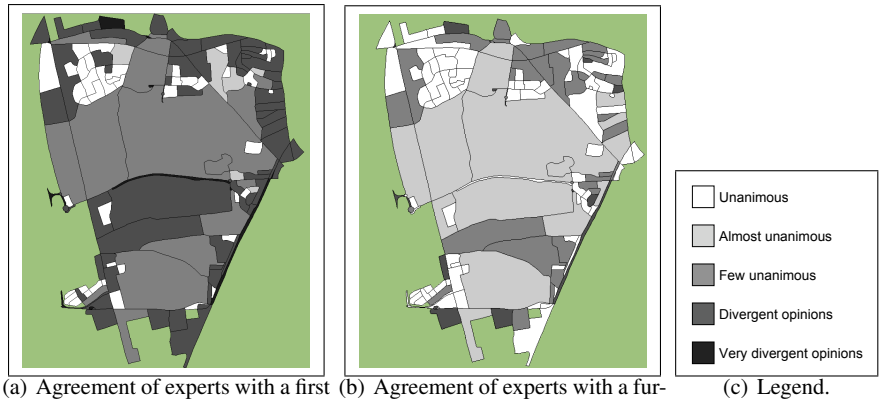
This section presents additional functionalities in order to compare labellings between experts. Indeed, it is important to check whether different experts agree on the chosen labelling process: labels and sampling.

A particular attention is made on the unanimity of labelling between them. First of all, a progress report visualization is provided for each expert, for each part of the city, in order to indicate to label in priority the areas that have already been labelled by other experts. Then an agreement visualization (figure 7) allows to see which areas are the most conflicting ones. The agreement measure is made thanks to the equation 1.

$$U = \frac{\sqrt{\Sigma_{c=1}^{n}(m_c - e_1(c))^2} + \sqrt{\Sigma_{c=1}^{n}(m_c - e_2(c))^2} + \ldots}{nb_e} = \frac{\Sigma_{e=1}^{m}\sqrt{\Sigma_{c=1}^{n}(m_c - e(c))^2}}{m} \quad (1)$$

- where
  - e denotes the experts (their cardinality is m)

  - c denotes the classes (their cardinality is n)
  - m_c denotes the average of values for the c class
  - $e_1(c)$ denotes the value chosen by the first expert for the c class
- A greater value of U denotes a stronger disagreement of the experts.



(a) Agreement of experts with a first experiment. (b) Agreement of experts with a further experiment. (c) Legend.

Fig. 7 Agreement variation along the experiments.

The figure 7 shows a clear improvement between the first and the latter labelling process. These kinds of analysis allow the experts to choose the best way of labelling.

In order to eliminate residual conflicts, we created a functionality allowing experts to add comments (figure 1 (B)) to their labelling and to visualize areas by comment (not-well formed area, city border area ...).

3.3 Towards automating the labelling process

Finally, export functionalities allow to use labellings results in learning programs. In particular, exports in arff format for Weka [7], an Open Source learning toolkit, and in 1BC format for 1BC, an ILP[2] learning tool [8] are provided.This is a first step to automatize the labelling from a training set. The figure 8 shows an excerpt of automatically labeled map, produced thanks to a SVM learning algorithm applied on experts training sets from the urban labelling procedure.

A part of the Weka library will be integrated in the module in order to envisage a semi-automatic learning process based on a supervised classification. Moreover, this integration allows us to imagine active learning functionalities, that will improve

[2] Inductive Logic Programming.

the labelling by choosing the most appropriate areas to label for an efficient learning process.

Fig. 8 Example of learned labels thanks to a classifier.

4 Conclusion

This paper presents the labelling functionalities of a Geoxygene extension, within the framework of a urban classification. The module aims at being generic in order to perform similar classifications for other geographical layers. We investigated pertinent facilities for a labelling module, like the labels management, the confidence level capability, the exclusive or overlapping classes choice, the several visualizations, and the analysis and use of labelling results.

Future works concern experiments of the integrated learning facility, in order to classify the whole dataset and the active learning option allowing the machine to choose the best training examples, i.e. to speed up the collect of training data.

References

1. T. Badard and A. Braun. Oxygene - d'une plate-forme interopérable au déploiement de services web géographiques. *Revue internationale de géomatique*, 3(13):411–430, 2003.
2. J. X. Barros. Simulating urban dynamics in latin american cities. In *proceedings. of the 7th International Conference on GeoComputation*, University of Southampton, United Kingdom, September 2003.
3. M. Batty. *Cities and complexity: understanding cities with cellular automata, agent-based models, and fractals*. MIT Press, Cambridge MA, USA, October 2005.
4. I. Benenson and J. Portugali. Agent-based simulations of a city dynamics in a gis environment. In *COSIT '97: Proceedings of the International Conference on Spatial Information Theory*, pages 501–502, London, UK, 1997. Springer-Verlag.
5. A. Bondu and V. Lemaire. Etat de l'art sur les méthodes statistiques d'apprentissage actif. *RNTI, Numéro spécial sur l'apprentissage et la fouille de données*, 2007.
6. Y. Hammam, A. Moore, and P. A. Whigham. The dynamic geometry of geographical vector agents. *Computers, Environment and Urban Systems*, 31(5):502–519, 2007.
7. G. Holmes, A. Donkin, and I. Witten. Weka: a machine learning workbench. *Intelligent Information Systems,1994. Proceedings of the 1994 Second Australian and New Zealand Conference on*, pages 357–361, Nov-2 Dec 1994.
8. N. Lachiche and P. Flach. 1BC2: a true first-order bayesian classifier. In Springer-Verlag, editor, *12th International Conference on Inductive Logic Programming*, pages 133–148, 2002.
9. D. O'Sullivan. Graph-cellular automata: a generalised discrete urban and regional model. *Environment and Planning B : Planning and Design*, 28:687–705, 2001.