A Framework for Integrating and Reasoning about Geospatial Data

Shubham Gupta and Craig A. Knoblock

University of Southern California,
Information Sciences Institute,
Marina del Rey, CA 90292, USA,
Email: {shubhamg, knoblock}@isi.edu

1. Introduction

The amount of geospatial data continues to proliferate with recent advancements in information technology. Available geospatial data sources include mapping services (GoogleMaps, YahooMaps, etc.), Web2.0 based collaborative projects (OpenStreetMaps and WikiMapia), traditional geospatial data sources (raster maps, KML vector layers, etc.) and non-traditional geospatial data sources (phonebooks and property-records). To fully exploit these diverse geospatial data sources, we are developing an integrated approach to extraction and fusion of these sources within a unified framework.

A geospatial fusion framework needs to support both the integration and reasoning of heterogeneous geospatial data. The data integration tasks involve gathering the available geospatial data from a wide variety of sources, such as those listed above. The geospatial reasoning processes can infer new and useful knowledge about a region by applying various reasoning methods over the integrated data. Some common geospatial reasoning processes are creating 3D models of buildings from LIDAR (Zhou & Neumann, 2008), identifying streets from raster maps (Chiang et al., 2008), automatic conflating road vector data with orthoimagery (Chen et al., 2007), and so on. Figure 1 shows an example screenshot where a variety of data sources and reasoning capabilities have been integrated into a single integrated framework. In this figure, the fusion of the datasets and reasoning processing make it possible to identify the locations of the buildings, the names of the streets, and the businesses associated with each of the buildings.

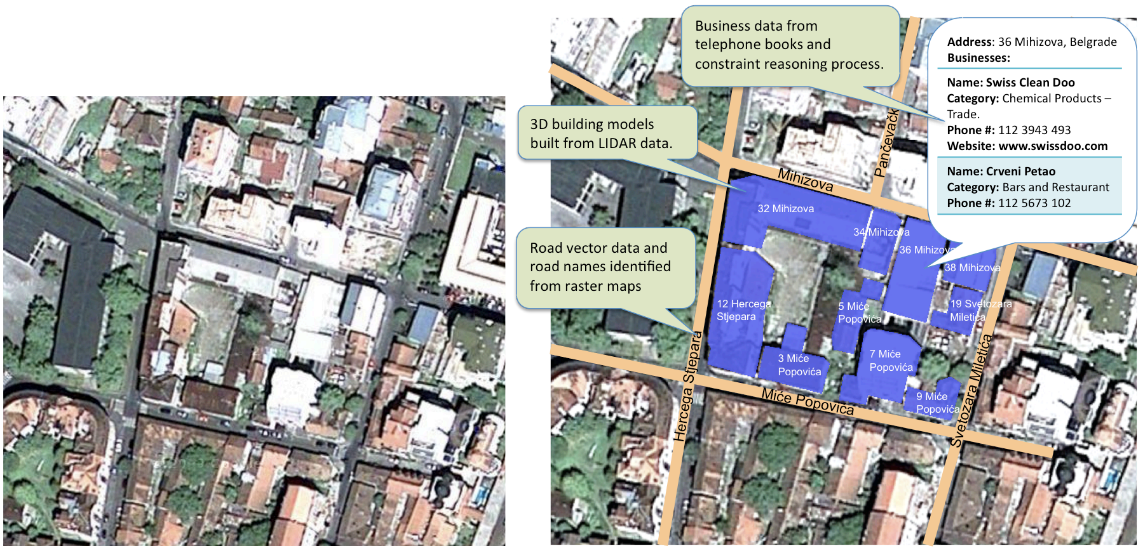

Figure 1: Area in Belgrade before and after the geospatial fusion process.

To provide a common platform for such data integration and geospatial reasoning tasks, we have developed an integration framework that allows the user to interactively fuse different kinds of geospatial data sources and exploit the integrated data to carry out various geospatial reasoning processes. In the next section, we describe our approach and its implementation in a system called InfoFuse.

2. A Framework for Geospatial Data Integration and Reasoning

We are developing a framework for integrating and reasoning about geospatial data. The various geospatial layers are integrated on top of a base layer, such as the satellite imagery for a given area. The system imports other data into the system and converts them into a uniform KML format. The reasoning processes then operate on the data layers that are available and either generates new layers or associates the results of the reasoning with the existing layers. This uniform approach to representing and reasoning about the data hides the heterogeneity present in the input data formats from the geospatial reasoning processes, thus allowing the reasoning processes to focus on the logic. This heterogeneity has been a major hurdle for achieving semantic interoperability of geospatial data sources (Sboui et al. 2007). The reasoning methods are able to exploit the integrated data and present the results on a map or image using this framework.

To illustrate the underlying approach of our framework, we describe an example task of identifying the buildings visible in the aerial imagery. Figure 2 shows the interface for InfoFuse, which is implemented on top of GoogleMaps. The right column shows the various operations available to import and reason about the data. This figure

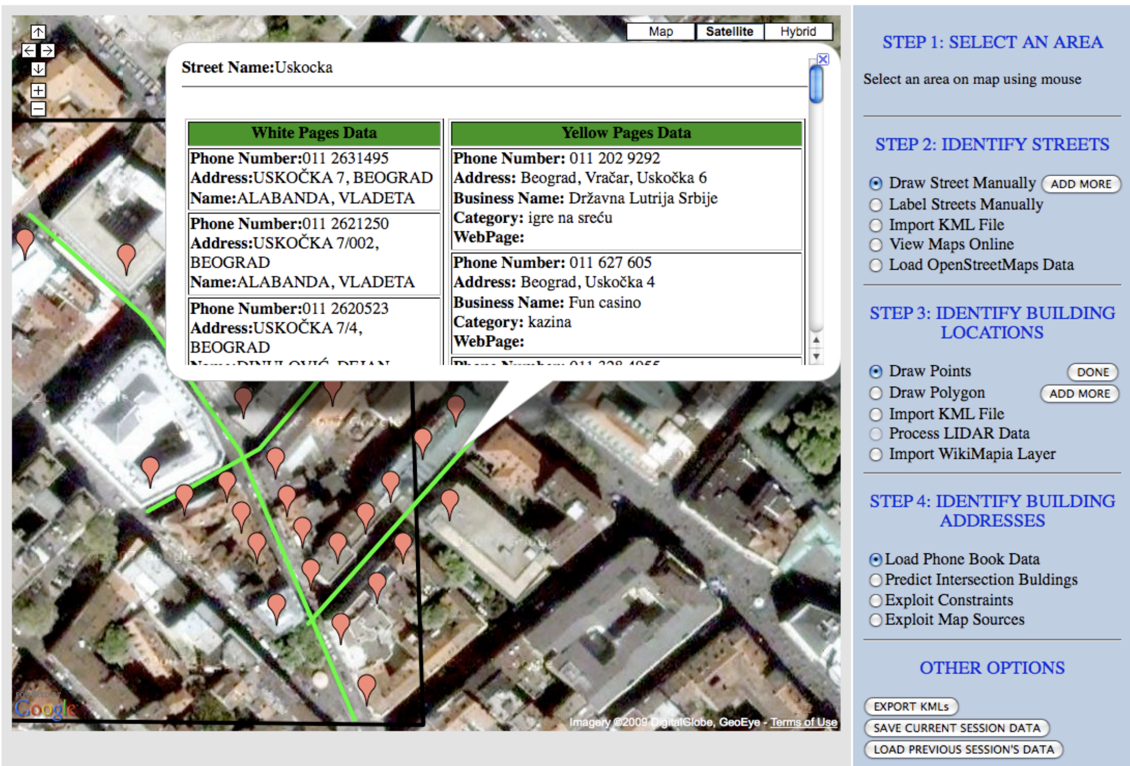

Figure 2: InfoFuse: A system for integrating and reasoning about geospatial data

shows the streets and buildings for a given region in Belgrade. At this point the system has imported the data for each of the streets shown from the white pages and yellow pages for Belgrade. The next task is to apply an information reasoning process to determine which address is associated with which building.

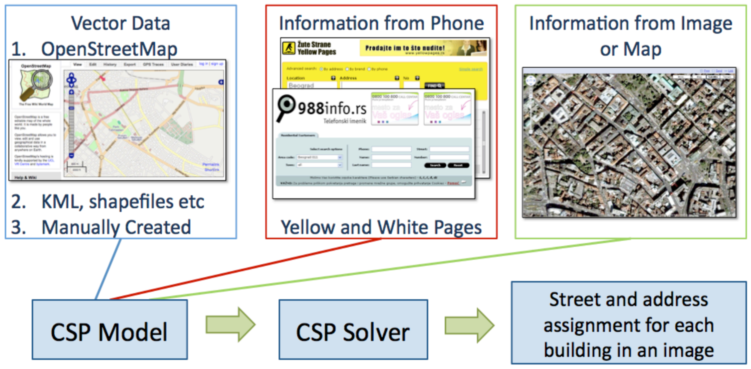

Figure 3: Building identification as constraint satisfaction.

In order to determine how to map the telephone book data to the individual buildings, the system turns the problem into a constraint satisfaction problem (CSP) (Bayer et al., 2007; Michalowski & Knoblock, 2005). The CSP formulation (Figure 3) integrates the vector data that defines the layout of the streets in a city, the building locations along the street, the addresses obtained from online phonebooks, and the addressing patterns used in the given city.

The reasoning task consists of various integration and geospatial reasoning steps such as identifying streets, identifying building locations, acquiring phonebook data, etc. for the region under consideration. We now describe the steps in integrating the various geospatial layers and applying the CSP reasoning process on the integrated data to generate the mapping of the addresses to the buildings.

Identifying Streets in an Image

In order to identify the buildings in an image, we first need to identify the street locations and names for the area of interest. InfoFuse provides several alternative methods to perform this task. For example, the user can automatically import the OpenStreetMap data available for the selected region if it is available. To automate this process, InfoFuse executes a software wrapper for the OpenStreetMap website to download the data and overlay it on the image (Figure 4).

The user can also import an existing KML road vector layer that is available. If no existing data can be found for the selected region, the user can create the road vector layer by interactively drawing the road lines on the image and labeling the roads using

Figure 4: Streets identified with OpenStreetMap data.

another source such as a paper map.

Identifying Building Locations

The next step is to identify the locations of the buildings in an image. InfoFuse provides several different ways to perform this task. This can be done by manually drawing points or polygons to represent the buildings (Figure 5), loading in an existing KML layer for the building locations, or extracting data from another source, such as WikiMapia, which contains a lot of user-generated information about a location (such as houses, businesses, airports, etc.).

Figure 5: Building locations manually identified as polygons.

Identifying Building Addresses and Linking Business Data

To gather current information about people and businesses for a region, InfoFuse uses wrappers to extract data from the Yellow Page and White Page websites. InfoFuse executes the wrappers to extract the data for each street, links the extracted data with the road vector data, and makes it available for viewing. Figure 2 shows the businesses listing and phonebook data in the popup for the street Uskočka of Belgrade City.

The CSP reasoner combines the road vector data, the building location data, and the phone book data in a reasoning process to map the addresses to the individual buildings. This reasoning process takes the phone book data associated with the roads vectors, performs the reasoning over data, and links the resulting data to the individual buildings (Bayer et al., 2007; Michalowski & Knoblock, 2005).

In the interface, the user invokes the constraint reasoning process using the “Exploit Constraints” option. InfoFuse displays the generated mapping with color-coded placemarks (Figure 6). Instances of building variables that are mapped to a single address are depicted with green placemarks and instances mapped to multiple addresses are depicted with red placemarks. The ambiguity of multiple possible addresses mapped to a single location is due to the uncertainty that may be present in the input data, such as missing addresses in the phonebook.

3. Discussion

The work on InfoFuse provides a proof-of-concept of our approach to building a integration and reasoning framework for geospatial data. In this initial work we focused on a specific integration task to solve the problem of identifying buildings in imagery by combining highly diverse types of sources (road vector data, building locations, and telephone books). The approach described is implemented and has been applied to the data that is available online for the city of Belgrade.

In future work, we plan to generalize this work such that we can dynamically fuse a much broader set of sources without having to engineer the integration of those sources in advance. In order to make that possible, we plan to develop a general information fusion framework that can be applied to a variety of geospatial integration

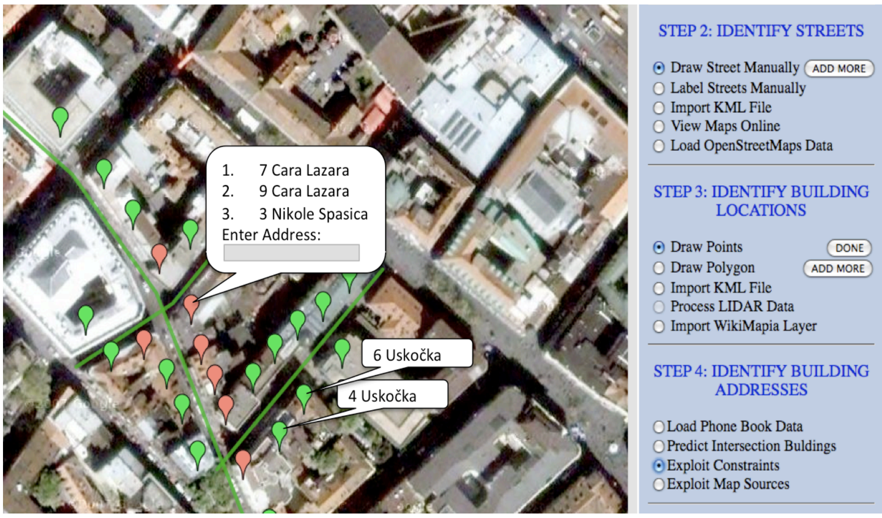

Figure 6: InfoFuse displays the resulting mapping

and reasoning tasks. This requires extending the framework to support the rapid integration of new data sources and new reasoning methods. In addition, we will develop the tools for importing a range of diverse sources as well as a library of reasoning algorithms that can operate on the available data.

References

C.-C. Chen, C. A. Knoblock, and C. Shahabi ,2006, Automatically conflating road vector data with orthoimagery. *GeoInformatica*, 10(4):495–530.

Y.-Y. Chiang, C. A. Knoblock, C. Shahabi, and C.-C. Chen ,2008, Automatic and accurate extraction of road intersections from raster maps. *GeoInformatica*, 13(2):121–157.

K.M Bayer, M. Michalowski, B.Y. Choueiry, and C.A. Knoblock, 2007, Reformulating CSPs for Scalability with Application to Geospatial Reasoning. Proceedings of CP 2007, Lecture Notes in Computer Science 4741, 164-179.

M. Michalowski, and C.A. Knoblock, 2005, A Constraint Satisfaction Approach to Geospatial Reasoning. Proceedings of the Twentieth National Conference on Artificial Intelligence.

T. Sboui, Y. Bédard, J. Brodeur, and T. Badard, 2007, A Conceptual Framework to Support Interoperability of Geospatial DataCubes. Lecture Notes in Computer Science 4802, 378–387.

Q. Zhou and U. Neumann, 2008, Fast and Extensible Building Modeling from Airborne LiDAR Data. Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems.