ARCHIBASE:A CITY-SCALE SPATIAL DATABASE FOR ARCHITECTURAL RESEARCH

YICHEN MO[1], BIAO LI[2], JIAQIAN WU[3] and PENG TANG[4]
[1,2,3,4]*Southeast University*
[1,2,4]*{moyichen|jz_studio|tangpeng}@seu.edu.cn*
[3]*wuchiachien@163.com*

Abstract. The explosion of geolocation data and data-based algorithms has the potential to analyze sophisticated urban areas and foster a more robust urban model. To better collect and organize the city data, this paper introduces a city-scale spatial database called ArchiBase, built upon Java and web APIs of open source databases. With hierarchical, layered, and regularly-updated spatial data defined by relation table, ArchiBase allows indexing and geometric searching of the entire city and supports applications and extensions for different cities. This research is from a graduate urban design course aiming to renew Prato, an industrial city in Italy. ArchiBase first creates the base version of Prato from multiple data sources, then illustrates the usability and expandability through three simple applications. The use of ArchiBase can better interpret future cities and demonstrate the unparalleled opportunities of collaboration and remote work for urban researchers and designers.

Keywords. Spatial Database; Data Model; Urban Design; Design Support Tools.

1. Introduction

The digitalization of urban life embeds the spread of ubiquitous computing. State-of-the-art machine learning technologies are improving quickly both on efficiency and capability based on text or image. Yet the representation of cities is not simply text or images, but geolocation spatial data. This poses the challenge for machine learning and urban analysis.

In order to organize, retrieve, and index city data, this paper introduces a novel city-scale spatial database called ArchiBase, which builds upon Java and web APIs to utilize open data sources. ArchiBase allows indexing and geometric searching of the entire city. The spatial data defined by the relation table are both hierarchical and layered. After the first setup, ArchiBase regularly updates itself with the latest data on the Internet, eliminating the need to retrieve the entire piece of data each time. The structure of the data table is flexible for applications and extensions for different cities. ArchiBase provides a forum in which researchers, architects, and technologists test the potentials of data-based

PROJECTIONS, Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2021, Paper 135 (Preprint).

algorithms in addressing contemporary urban challenges. The project and related experiments are public on the open-source platform.

The rest of the paper organizes as follows: Section 2 shows that ArchiBase is a city-scale and user-friendly spatial database, providing a collaboration platform for urban planning decision making. In Section 4, a few analyses on Prato show that ArchiBase example a framework for data-based urban design and site analysis for all cities. The construction of ArchiBase is beyond traditional data collection method; Section 3 describes how ArchiBase is built automatically with Java.

2. ArchiBase and Related Works

2.1. RELATED WORKS

Existing research has already tried to propose theoretical databases for specific urban design project or format data representation for the entire city. Recently, more and more data-based algorithms and visualizations reflect the value of this research.

Database: With precise design and modeling techniques, the database supports a holistic approach to urban design practice. Gil et al. (2011) implement a spatial data model that serves as a backbone of the City Information Model, including the urban environment and design process domain. Coorey and Jupp (2013) give a parametric-based schema to extract geometric and topological spatial data from the 3D building model with MySQL. Xu and Li (2019) create the city case base for searching and retrieving similar sites. These works perform data abstraction of the architecture of the city, albeit the scale is usually limited to a specific project and lacks the procedural of data acquisition and integration.

Data format: For interoperability and compatibility, the database uses standard specifications for the query language. OpenGIS Simple Features defines mostly 2-d geometries (such as point, line, polygon, multi-point, multi-line) used in the field of geographic information systems (de La Beaujardiere, 2006). There also exists a data model and exchange format to store city data, like CityGML(Kolbe et al. 2005), standardized by Open Geospatial Consortium(OGC), using Simple Features to present its geometry attributes. In recent works, CityJSON presented (Ledoux et al., 2019) a JSON encoding and recording the topological information in a compact format.

Data-based algorithms: In recent years, with the spread of deep learning, researchers are shifting to data-based algorithms to exploit abundant data generated by cities' daily activities. By measuring street network from data with statics, the orientation and configuration define city spatial logic and order (Boeing, 2019). While urban morphology meets deep learning, millions of cities could compare and evaluate in the same urban model (Moosavi, 2017). With thousands of geotagged satellites and perspective images, machine learning portrays a city's personal view without any previous knowledge about the cities (Alvarez-Marin and Ochoa, 2020). To depict the power of visualization, MorphoCode and Senseable City Lab uses detailed datasets such as NYC Open Data, revealing the latent information from data. However, not all cities have such organized data. ArchiBase, therefore, builds a database on general city comparatively without sufficient data in the current stage.

Above all, ArchiBase aims to be the basis of data-based algorithms, visualization, and analysis. Derived from geospatial data, it takes architectural needs under consideration, reorganizes, and integrates into a hierarchy and layered relation database.

2.2. FEATURES OF ARCHIBASE

The collection and preparation phase are typical and effortful due to the variety of data formats and design tasks. ArchiBase partly automate these tedious tasks, aggregating to a *city-scale spatial* database, which is *versional* and *collaborative*.

City Scale: ArchiBase builds upon the entire city, manipulating built environments containing roads, buildings, urban spaces, functions. Such a choice is for three purposes: (1) city-scale data model provides an emphasis on typology in urban planning and architecture (2) build a database on the city is a challenging task, while it is unnecessary to do this when research on a small city block (3) in the practice of urban design, projects in the same city usually share some common properties. (see Sec.3.3)

Spatial Query: ArchiBase built upon the backbone of open source software PostgreSQL and PostGIS. The latter adds support for geographic objects to PostgreSQL, including spatial predicates and operators provided by the GEOS library. It supports Geospatial measurements like area, distance, length; set operations like union, difference, buffers; 3x3 DE-9IM for determining geometries' interaction. Besides, ArchiBase also provides a few functions and examples to retrieve and index the city for convenience.

Version Control: Most data-based algorithms are offline. Cities are deterministic and static: algorithms concentrate on the cities per see. However, the spatial data online changes each day. ArchiBase could set up regular updating tasks with upstream web mapping services and keep the historical data version if needed.

Coworking: ArchiBase is an interactive and participative platform for researchers to develop data-based algorithms conveniently. Its infrastructure deploys on a database server for sharing and synchronization, providing high-speed connection and integration. Collaborative working allows researchers to extend the content of the database, which increases the ability and knowledge of ArchiBase.

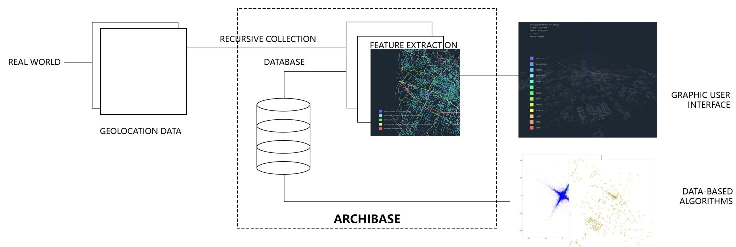

Figure 1. Overview workflow of ArchiBase.

3. Building ArchiBase

As an ambitious project, ArchiBase is aiming to automatically generate the whole city database from the data on the Internet. After parsing data with the specific data format, the author preprocess, extract practical information, then enter into the database with its structure and relation (Figure 1).

3.1. DATA ACQUISITION AND MAINTAINING

Nowadays cities are producing abundant data spontaneously from geotagged social media and web mapping (Alvarez-Marin and Moosavi, 2017). Open Street Map, as a collaborative web map, has a relatively high-quality data set covering the whole planet (Haklay, 2010). ArchiBase mainly chooses OSM as the data source; data from other sources need to translate into the same geographic coordinate system following a similar acquisition methodology. With the input range of the city, the data acquisition procedure divides the bounding box of the map according to the HTTP status code. Our algorithms recursively send requests for collecting data adaptive to the data density.

Once the ArchiBase is built, the maintaining procedure also implements by operating the web API. First, ArchiBase gets the ChangeSet ID for a certain period; Then the detailed changing of data is requested. There are three strategies to integrated information into the database: adding, modifying, and deleting. Adding clean the data and place them into the database; modifying select the specific data to change; deleting choose the data and tag them with deprecated.

3.2. INTERACTION AND REPROJECTION

Construction of ArchiBase is working synergistically with visualization of data to confirm the correctness. To understand and manipulate data, ArchiBase provides the user interface and necessary visualizations in Java, including a reprojection from geographic coordinate system to model coordination and a color toolbox to color geometric data with its specific labels.

3.3. FEATURE EXTRACTION FOR MIGRATION

Before migrating from the online data sources to ArchiBase, it is critical to extract information from geospatial data. The Web API provided by the Internet usually uses an elaborated tag system. According to tags' characters, they are simplified to a few particular categories. This part uses the theoretical knowledge of architecture and city morphology to ensure the expertise of feature extraction.

Functions: ArchiBase uses POIs to describe the functional characteristics of a zone. The labels are grouped into 13 particular functions by its similarity. The original attributes - like name and rating of the place - is recorded for later usage.

Roads: The hierarchical structure of city is related to the transportation network. By grouping the data labels, the LineString of road is divided into five levels: R1, R2, S1, S2, and S3 (Beirão et al., 2009).

Blocks: After removing of the maximum level R1 and minimum level S3 of the roads, the LineString collection are divided into the road-defined polygons, which is used roughly as city blocks.

Buildings: Building is the main body of the city, useful to create the graphic-ground relation imaging of a city and weaving the texture of a field. OSM taginfo shows building=yes as the most used tag.

Imagery: The landmarks or artifacts are essential to shaping common memories of local people (Rossi et al., 1982). Plazas, education, and cultural, industrial buildings are identified from massive data. Natural features, like the water and greenfield, is also filtered by this part.

3.4. DATABASE SCHEMAS

Technically, ArchiBase uses JDBC to access a PostgreSQL database with extension plugins including PostGIS and HStore. PostGIS enable the spatial query with its stored geometry information in well-known text form. HStore uses a data type able to record the original key-value type for further usage. The database schemas which structure described the knowledge of ArchiBase, is divided in to database tables in a spatial relational database. Current data tables: functions, roads, blocks, buildings, and urban spaces correspond to the five building elements of 3.3. For scalability and adaptability, the users can alter table, add columns according their requirements.

4. Applications: A case study of Prato

In this section, the author uses a case study of Prato to show the application of ArchiBase. Prato is the third-largest city in central Italy and the second largest in Tuscany. As an industrial city, Prato focused its urban renew on improving abandoned factory areas and finding solutions to cultural conflicts. ArchiBase created the environment for the entire design project. The database has been deployed on a server that can be accessed through the network. Therefore, students can execute Structured Query Language to obtain the required data and use it for subsequent analysis using their programming. By analyzing and abstracting universal urban elements, research conducted on Prato implements robust algorithms applicable and scalable for most cities.

4.1. CITY DATA OF PRATO

The first data setup of Prato includes 31273 buildings, 8756 roads and 1228 blocks; 10 other types of urban space are also identified from tag info(Figure 2). The general information about the city, including the city name, latitude and longitude coordinate ranges, and geographic coordinate system, is recorded in the database tables. In order to extend the characteristics of the general city information, the author implement algorithems for Prato to calculate the city orientation and shape index.

City orientation: Pinpointing the city orientation reveals the basic structure in the city building data. In ArchiBase, building data is represented as the closed LineString. They are divided into segments and defined them as vectors (plot as blue points in Figure 3). By finding optimal components, principal component analysis (PCA) reduces the dimension from (102267, 2) into (2, 2), offering the orientation of the city.

Shape Index: To index similar shape of the city, ArchiBase implements Boyce-Clark shape index algorithms (Figure 4), which is able to show the similarity of shape (Boyce and Clark, 1964).

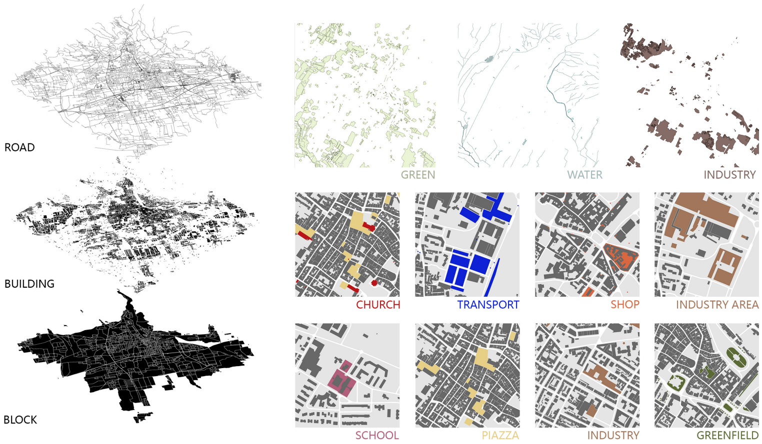

Figure 2. City data of Prato.

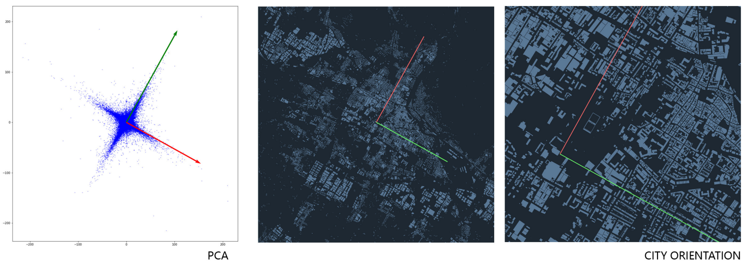

Figure 3. A PCA approach to city orientation.

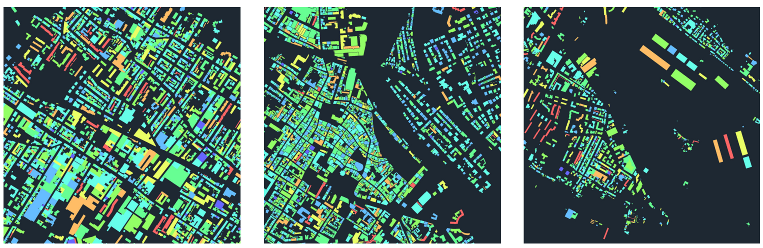

Figure 4. Boyce-Clark index of building shapes.

4.2. FUNCTIONAL ZONES OF PRATO

In light of the Internet and mobile devices, POI data of map services dynamically implies the activities of the city and the functions of an urban area. Google Places API provides 23925 POI data, which demonstrate the functional zones of Prato. The 100 detailed categories of POI are divided into 13 broad types as Sec. 3.3 indicates. ArchiBase provides a simple grid viewer for showing POI data. By selecting the type and scope of statistics, the viewer displays the distribution of POI points interactively (Figure 5). The algorithm counts numbers of POI points in each grid and then fills each cell by its quantity. As shown in left part of Figure 5, 150x150m grid has the best display effect.

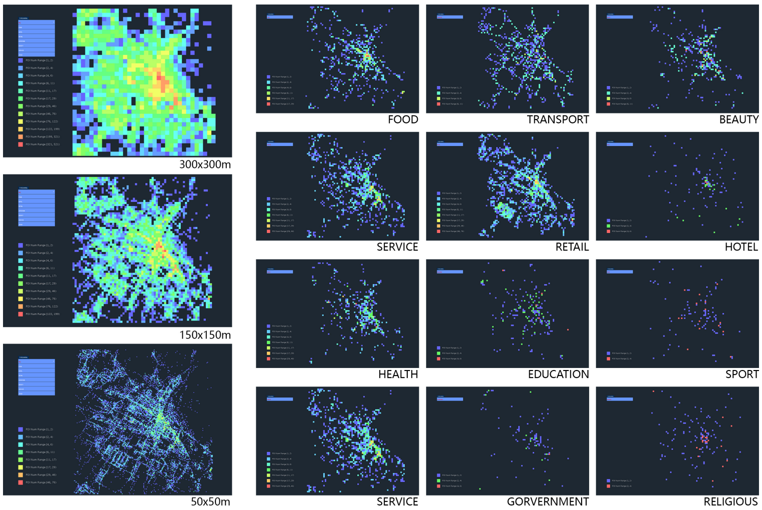

Figure 5. Viewer screenshots. Showing the type and scope of statistics results of POI data.

In terms of revealing the connectivity and magnitude of regions, the DBSCAN algorithm groups the points that are closely packed together into 36 clusters. The result of the spatial cluster algorithm is different from counting POI numbers in road-defined blocks (see Figure 6).

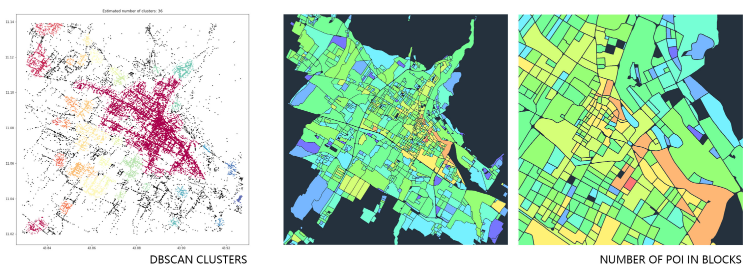

Figure 6. DBSCAN results in 36 clusters, including more information than counting numbers of POI in blocks.

4.3. SPACE AND TIME IN CHINESE IMMIGRANT POPULATION

Language identification depicts Prato’s distinctive urban structure and adds data columns to the database. As the second-largest Chinese immigrant population, Prato is one of the few cities in Italy with a growing economy. From the official website, the percentage of the Chinese population in the total population increased from 0.9% to 11.8% between 1995 and 2018. Another evidence is the spatial occupation from POI data. Google Cloud Translate identifies the language of the POI name and finally found 866 Chinese-related points from 23925 data. These identified Chinese names mainly use pinyin for its English name, which is a unique phenomenon in Chinese culture. Figure 7 shows that two areas with the highest density are none other than Macrolotto Zero and the factory-strewn Macrolotto I.

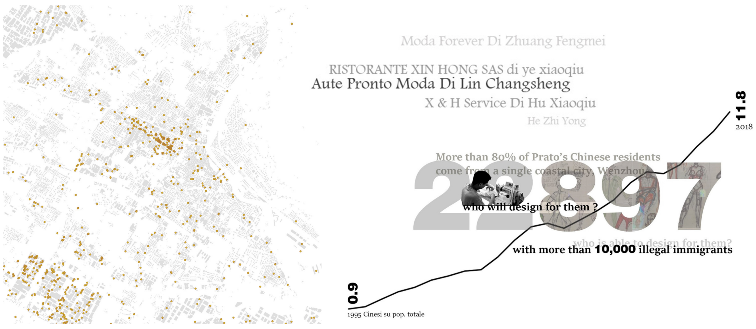

Figure 7. the distribution of Chinese POI and the proportion of Chinese population between 1995 and 2018.

4.4. CITY SENSATION OF STREET BLOCKS

Using street view images of Prato, deep learning demonstrates methods for measuring the built environment of the city with a subjective sensation. By transfer learning on ResNet50 (He et al. 2015) with Place Pulse 2.0 dataset (Salesses et al. 2013), the model evaluates a street view image into 6 perspectives including safety, lively, depressing, boring, beautiful, and wealthy. The author uniformly sampled points on city streets and collected street view images of the points. The evaluation results show the consistency of the discrete samples (Figure 8). After computing with the model, the score of the points contributes to the nearby city blocks. The sensation maps (Figure 9) depicts a significant correlation with the distance to the city center (result from Sec. 4.2).

Figure 8. ArchiBase indexes the best 5 places and vice versa, showing their street view and 6 scores.

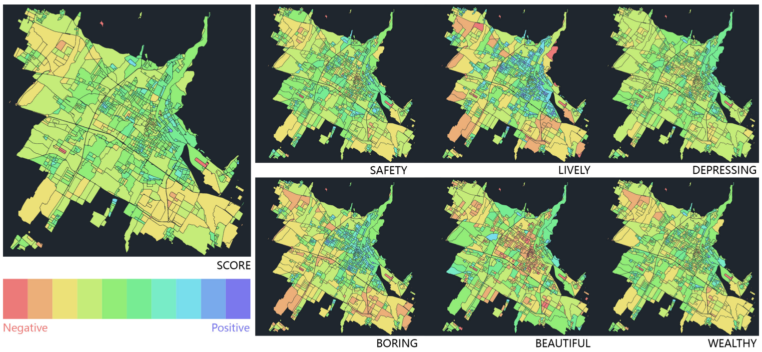

Figure 9. The sensation score of city blocks.

5. Conclusion and Future Works

This paper mainly focuses on the framework for collaboration and representation of city data extracted from the Internet. The value of ArchiBase lies in:

1. The abstraction of city data and its representation in a spatial database.
2. The automated construction and updating tool reducing the burden of data acquisition.
3. An attempt to realize a universally applicable urban analysis algorithm.

Current limitations of ArchiBase include the insufficient variety of algorithms and feature representations. Therefore, in the future: (1) more algorithms could test and evaluate through exploiting ArchiBase (2) feature extraction of data requires theories and experiments to meet the challenge of urban design tasks.

The use of open-source geographic tools in this research demonstrates that programming can partially remove professional barriers. Today, programming is also no longer a specialized skill for professionals. Through the understanding of data representation, it is possible to analyze cities with lightweight programs. The data definitions and structures of ArchiBase can better interpret future cities and offer unparalleled opportunities for researchers and designers in computational research and design.

References

Alvarez-Marin, D. and Moosavi, V.: 2017, Inverting normative city theories and computational urban models: Towards a coexistence with urban data streams, *CoRR*, **abs/1707.03678**, http://arxiv.org/abs/1707.03678..

Alvarez-Marin, D. and Saldana Ochoa, K.: 2020, Indexical Cities: Articulating Personal Models of Urban Preference with Geotagged Data, *CoRR*, **abs/2001.10615**, https://arxiv.org/abs/2001.10615.

Beirão, J., Montenegro, N., Gil, J., Duarte, J.P. and Stouffs, R.: 2009, The city as a street system: A street description for a city ontology, *Proceedings of the 13th Congress of the Iberoamerican Society of Digital Graphics*.

Boeing, G.: 2019, Urban spatial order: Street network orientation, configuration, and entropy, *Applied Network Science*, **4**(1), 67.

Boyce, R.R. and Clark, W.A.V.: 1964, The Concept of Shape in Geography, *Geographical Review*, **54**(4), 561-572.

Gil, J., Almeida, J. and Duarte, J.: 2011, The backbone of a City Information Model (CIM): Implementing a spatial data model for urban design, *Proceedings of 29th eCAADe Conference*, Ljubljana, Slovenia.

Haklay, M.: 2010, How Good is Volunteered Geographical Information? A Comparative Study of OpenStreetMap and Ordnance Survey Datasets, *Environment and Planning B: Planning and Design*, **37**(4), 682-703.

He, K., Zhang, X., Ren, S. and Sun, J.: 2016, Deep residual learning for image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Moosavi, V.: 2015, Computational Urban Modeling: From Mainframes to Data Streams, *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Moosavi, V.: 2017, Urban morphology meets deep learning: Exploring urban forms in one million cities, town and villages across the planet, *CoRR*, **abs/1709.02939**, http://arxiv.org/abs/1709.02939.

OpenStreetMap, C.: 2017, "Planet dump retrieved from https://planet.osm.org" . Available from <https://www.openstreetmap.org/>.

Rossi, A. and Eisenman, P.: 1982, *The architecture of the city*, MIT press Cambridge, MA.

Salesses, P., Schechtner, K. and Hidalgo, C.: 2013, The collaborative image of the city: mapping the inequality of urban perception, *PloS one*, **8**, 7.

Xu, J. and Li, B.: 2019, Application of Case-Based Methods and Information Technology in Urban Design-The Renewal Design of the urban region around Roma Railway Station, *Intelligent & Informed - Proceedings of the 24th CAADRIA Conference*, Wellington, New Zealand, 625-634.