

Visualization systems in Urban Planning related tasks

Shi Yin

Electronic Visualization Laboratory
Department of Computer Science
University of Illinois at Chicago

Abstract

In this report, we survey a variety of visualization systems in different domains related to urban planning, including Geographical Information Systems, Urban Search & Rescue systems, and urban simulation systems.

1. Introduction

Urban planning is a technical and political process concerned with the use of land and design of the urban environment, including economy, environment, transportation and distribution networks, etc.

The modern origins of urban planning lie in the movement for urban reform that arose as a reaction against the disorder of the industrial city in the mid-19th century. Urban planning can include urban renewal, by adapting urban planning methods to existing cities suffering from decline. Alternatively, it can concern the massive challenges associated with urban growth, particularly in the Global South.[2]

Visualization has always been of significant importance in geographic related fields [AK99]. Good visualization makes analyzing and investigating collected geospatial data simpler and more efficiently thus enabling a deeper and more organized understanding and more effective application into practice.

In this report, we survey the state-of-the-art in visualization systems in a variety of urban planning related fields. In Section 2, we discuss Geographical Information Systems and online Mapping APIs. In Section 3, we introduce an application of visualization in Urban Search & Rescue Mission planning. In Section 4, we take a look at an urban visualization system built upon urban simulation software. In Section 5, we include two interviews of domain scientists. In Section 6, we conclude the report.

2. Geographic Information Systems Using Google Maps API

Geographic Information Systems (GIS) is probably the most popular and the most famous visualization system that are

used by urban planning researchers, professionals and students to process, manipulate, manage, and visualize geographical information data, and known by researchers from other research areas.

Online mapping APIs are new tools that emerged only in recent years. The advantages of them are: 1. **free for use**. Most APIs are provided to any user for free, while some are not, or have limited functionality in free version. For example, ArcGIS has a Javascript API, which is not as powerful as its desktop mapping software ArcGIS for Desktop and its mapping servicer software ArcGIS for Server, but includes most of the frequently used functionality. 2. **easy-to-use**. Most APIs are well documented and organized. 3. **scalability**. Some of online mapping APIs, such as Google Maps API, Bing Maps API, are providing limited version for free use, and provided also non-free versions for users with special needs.

Each API is actually with significant difference from each other in terms of weight (lightweight/heavyweight), scalability (small amount of data/large amount of data), functionality (simple-and-frequent-tasks-only/all-in-one) while at the same time keeps the similarity in that 1) the supported data is similar and 2) task sets are similar.

Here we list some interesting work that facilitates GIS and/or Google Maps API to visualize geographical urban data to give readers an understanding of the range of applications.

2.1. logistics network system

In [FWXL10], the authors introduced a logistics network system built upon Google Maps API. The major features of Google Maps API utilized by this system include: basics

functions of mapping, including the map layers and UI controls; event-driven asynchronous HTTP communication that interact with Google Maps API, and geocoding service that allows converting address to coordinates.

The system consists of 5 modules. **Registration module** and **Information Publishing module** are for administrative purposes and we omit them here. **Transaction module** and **Information Query and Matching module** are to match the vehicle and cargo based on a series of parameters such as the vehicle's location, cargo types, cargo weight and route, and then saves the transaction information into the database, waiting for monitoring and other queries.

The last module, **Map Service module** is the major module of the system. This module itself includes 4 sub-modules. **Querying sub-module**, which provides functionality for the vehicle owner or the cargo owner to conduct historical information management, and **Navigating sub-module**, whose primary function is to supply the vehicle owners and the drivers with map navigation, are with no significant interest.

The more important sub-module of this system as a logistics management tool, is the **Monitoring sub-module** that is able to show real time location of vehicles and goods dynamically, as well as several previous checkpoints through the route. In the backend, this module receives GPS positioning data with real-time communication, processes it and stores it in the database awaiting for user's call.

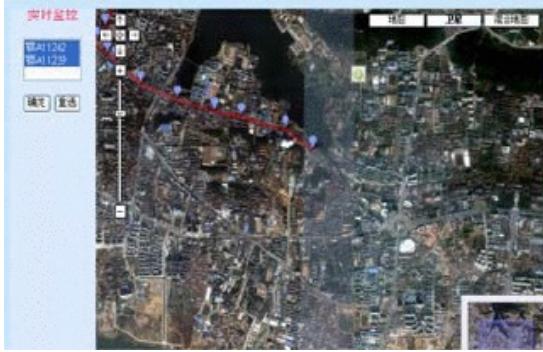


Figure 1: Monitoring vehicles on Google Maps

The fourth sub-module of Map Service module is **POI sub-module**, where Points of Interest associated with the logistics, such as large-scale logistics park, logistics companies, and distribution centers, are stored and ready to be displayed. Served as a auxiliary companion of **Monitoring sub-module**, POI sub-module enhances the usability of this system.

2.2. tourism information systems

In [WLL13], the author introduced their design and implementation of a tourism information system based on

Google Maps API. The traditional way that a tourist collects tourism-related information, such as restaurants, hotels, attractions, transportation and entertainment, is by searching on different websites. Three deficiencies are within this traditional way of information collecting: 1) tourism information is scattered; 2) tourism information is unconfirmed; 3) tourism information is flat. This work is an attempt to solve these issues by building an integrated tourism information platform where tourists can conveniently obtain more references for their travel.

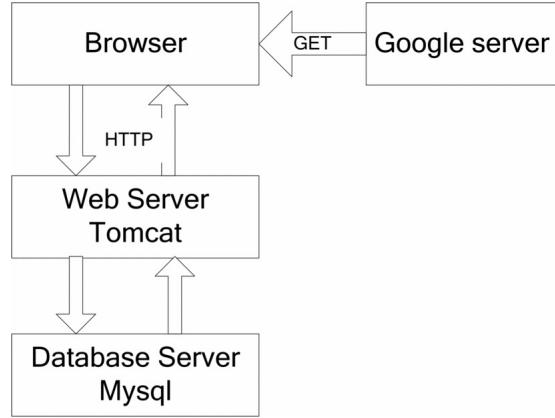


Figure 2: Architecture diagram.

A similar system for tourists is presented in [PPP09]. This system makes use of services made available through Google Maps API such as Geocoding which from given address returns a geolocation, Directions which provides routing instructions from a start point to a destination point, Streetview which shows virtual picture of locations, Google-Bar for local search of the map, and so on. The difference is that this platform not only provides basic mapping functionality but also integrated an expert system that can provide customized information to users.

2.3. land use information release system

In 2010, Zhang et al. [ZLC*10] introduced a land use information release system. It uses Google Maps API for map service, XML for data storage. It has a three-layer architecture (data layer, business logic layer and presentation layer). It allows user to query land use information by selecting filters (text query), keyword search (local search), geocoding search (reverse address resolution). It also allows user to send feedback about each data feature on the map.

2.4. railroad settlement information management system

Also in 2010, Wang and Zhou [WZ10] proposed a railroad settlement information management system that makes use

of similar techniques as [ZLC^{*}10], such as Google Map Service, XML, Ajax, etc. The major difference is they provided a better front-end, which visualized data using traditional charts such as line charts and bar charts to perform analysis on the data.

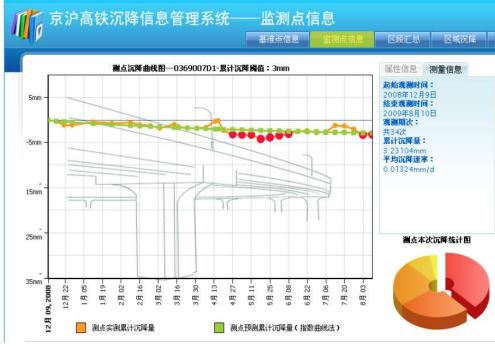


Figure 3: Search, warning and statistics of monitor point's settlement information



Figure 4: Statistical analysis of railway profile's cumulative settlement information

3. Urban Search & Rescue Mission Planning system

In an *Urban Search & Rescue* (USAR) operation, victims' survivability is mostly determined by the time-to-rescue. It is critical that medical attention or extraction be provided as fast as possible. As structural damage caused by disasters makes available floor plans outdated, it is very important that the *Incident Commander* (IC) can analyze the available information and coordinate multiple rescue responders. In recent years, the traditional way of assessment where 2D maps of the collapsed buildings are hand-drawn based on the descriptions of rescue responders on site is replaced by unmanned robots equipped with sensors to detect victims, gather information about hazardous area, and perform scans

of rooms to create a 3D point cloud of the building's interior.

In 2014, Bock, A. et al. [BKLR14] proposed a visualization system for IC to use in USAR scenarios. The system creates an interactive 3D rendering that increases the commander's spatial awareness of the building (Figure 5 (a-c) and Figure 7) and supports the planning and analysis of available access paths that are automatically generated by the system based on varying risk factors (Figure 8). Then IC uses this information to instruct rescue responders to reach *Points of Interest* (POI). At the same time the IC is able to annotate the visualization with new information provided by the on-site responders and thus shift the decision making process from opportunistic to being strategic.

This is the first system facing post-disaster USAR scenarios that uses 3D point cloud for visualization. Before this system, most of visualization-oriented work published in the field is concerned with pre-disaster evacuation planning. Reddy et al. [RHDW12] performed notable work based on analyzing possible bottlenecks of escape routes. Ribarsky et al. presented a system organizing first responders in intact structures [RSL^{*}10]. Kim et al. developed a system enhancing the situational awareness of responders using a mobile visual analytics tool [KMO^{*}08]. As for post-disaster USAR planning, existing systems are mostly based on 2D representations [KD09] [PP10] [WP07] [DK13].

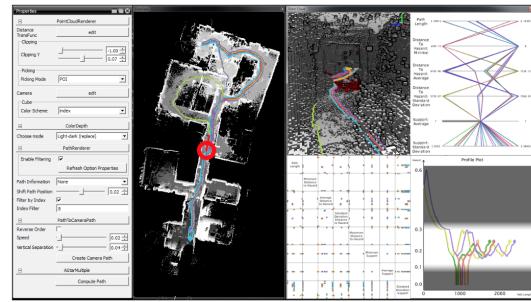


Figure 6: A screenshot showing the system for a typical scenario. each view can be maximized to fill the entire screen for in-depth inspection. An overview (left) shows a top-down view of the building to provide context, a multi-view (right) shows the different components of our system.

3.1. side note about point cloud

It is worth mentioning that there are not many applications we found in urban planning context that utilizes point cloud for 3D rendering. There are great potential in point cloud visualization in urban planning related tasks though, as large amount of fundamental research has been conducted, making point cloud ready-to-use.

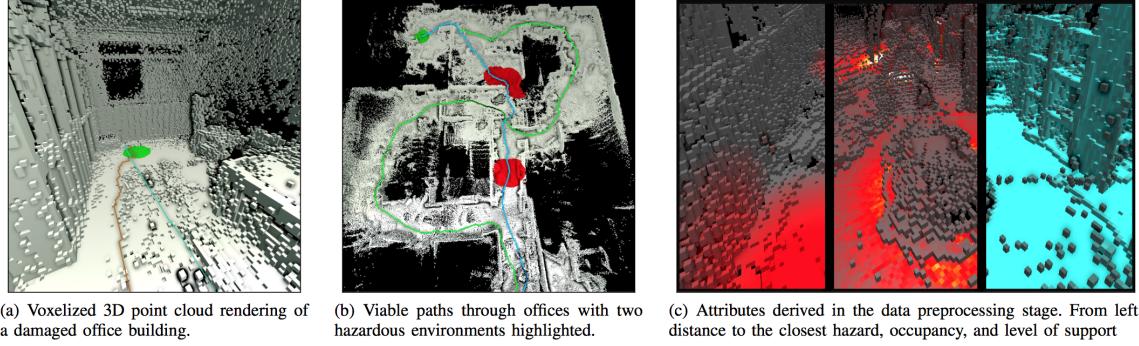


Figure 5: The system applied to a building at Tohoku University. Different views support a comprehensive understanding, allowing the IC to select and inspect paths that reach a Point of Interest, which are integrated into the rendering. Inspection and path computation is based on a set of derived attributes (c).

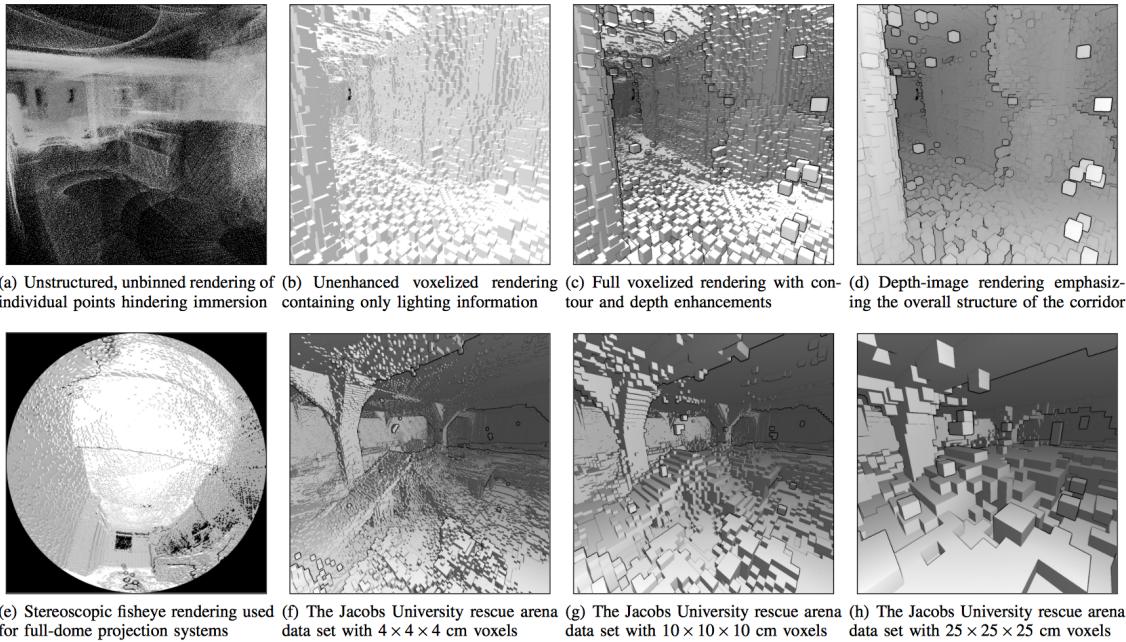


Figure 7: These images show different rendering techniques of the same location in the Tohoku dataset (a-d). Rendering the individual voxels as points does not allow for an immersive rendering as depth-cues are missing (a). Binning the point cloud and representing each voxel as a axis-aligned box solves this problem (b). In order to enhance the contours of the scene and produce a better immersion, image-space enhancements were performed (c). Alternatively, the IC can choose a rendering method imitating the output of range imaging cameras (d). It is possible to render the point cloud in stereoscopic fisheye that can be used for dome surfaces or VR glasses (e). The voxel size's effect during binning is shown for the rescue arena dataset (f-h).

There has been work by Richter et al. using a level-of-detail structure to render massive point clouds at high frame rates [RD10]. Xu et al. showed that non-photorealistic rendering techniques can be applied to point cloud data [XC04]. It is the contour lines in their rendering that inspired

xx's rendering algorithm. More recently, Pintus et al. presented a rendering algorithm that enhances features of unstructured point clouds in real-time without preprocessing [PGA11]. Some other work in this area that converts point cloud to surfaces, which is essential for urban modeling, in-

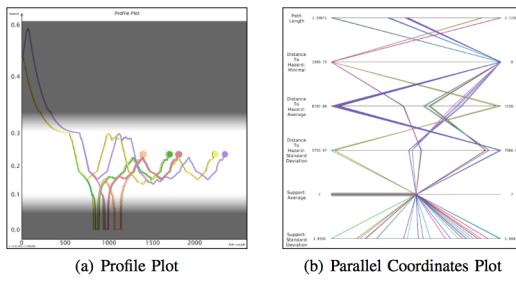


Figure 8: Views supporting comparative path analysis. (a) the profile plot presents the change of an attribute along paths; here the distance to the closest hazard. (b) parallel coordinates plot showing correlations between attributes.

clude [F*03], [HRD*12] and [NOS15]. Last but not least, the widely used Point Cloud Library (PCL) [RC11] is also making major contribution to the increasing popularity of point cloud.

4. Urban Modeling and Urban Simulation

Another field in urban planning context where visualizing data plays a central role in research activity is urban simulation. The term **urban simulation** has two definitions in urban planning context. One common use of this term by domain scientists in urban planning is to describe 3D rendering of urban landscapes, like in [FZ03] [Mer07] [MWH*06] [MZWVG07] [PM01] [RWF02]. Here in our discussion, we focus on the second definition though, where urban simulation is a behavioral or process modeling of the dynamic changes in urban activities and landscapes.

Urban simulation models and the visualization of their results have been increasingly used by planning agencies in city, county, and regional scopes to assess alternative transportation investments, land use regulations, and environmental protection policies.

There are two major challenges in urban simulation. The first and fundamental one is to adequately model and predict the complex socio-economic interactions that determine the growth of an urban space in order to simulate the change of this urban space over time. The future structure of a city is governed by both deterministic rules (e.g. population capacity of the city must grow) and organic rules (e.g. social, cultural, and economic interactions strongly influence how a city grows). Because of the intricate nature in urban simulation, a simulation model will usually generate an overwhelming amount of data when running over a long period of forecasting time in a large geographical scale (e.g., one or more decades for a large city with a several million population). This makes it difficult for researchers, planners, or policy-makers to interpret the data.

This is where the second challenge in urban simulation lies. An effective and intuitive visualization is essential to extract useful information from the large mass of data generated by such simulations. In fact, visualization has played an integral part in the development and use of urban simulations of different types for a while. Batty introduced various approaches that relate urban modeling, GIS, and computer graphics in [Bat93] and described the impact of virtual reality and 3D visualization to GIS and demonstrated it in a variety of examples [Bat97]. Cartograms, which use map shape warping to visualize relationships and values of urban and geospatial datasets, is widely used [DBN06] [HJO*01] [PSKN06] [PDBB00]. Dykes and Brunsdon [DB07] introduced geowigs, a series of geographically weighted interactive graphics, to provide large-scale geographical environment visualization. Visualization has also been used for the purposes of education, exploration, explanation, and engagement [BSX06]. Some approaches also make use of choroplethic maps, generated by exporting simulation results to GIS for rendering.

Recently, Vanegas et al. [VABW09] introduced a method that automatically and interactively infers geometry and image content of urban layouts using the input data and output data of an urban simulation, pushing urban simulation and urban visualization one more step forward. The major contributions of their work are:

1. A methodology for enhancing the visualization of the result of urban simulations by using inferred higher-level structural information.
2. A set of automatic and interactive algorithms for generating a visually plausible urban layout from the data generated by an urban simulation.

The input of this method consists of three components: the first one is a set of parcel and street geometries obtained from shapefiles from ArcGIS; the second is the high-resolution geo-referenced aerial imagery of the targeted area, in this case GeoTIFF files containing coordinates information; and the last is the socio-economic data of this area, which is used as input to an urban simulation engine, this case UrbanSim, that predicts the future value of several state variables of the urban space. These intermediate results of the simulation will then be used by the visualization tool to generate detailed urban layouts for any point in the simulation. The overall process of the pipeline is illustrated in Figure 9.

4.1. simulation of urban spaces

4.2. layout generation

Spatial transformations of parcels and of the buildings that occupy them can be inferred from the state variables of the urban simulation system. The transformations that frequently take place in the layout include the replacement of buildings with newer and/or larger structures, the division of a parcel into smaller parcels, and the creation of new streets.

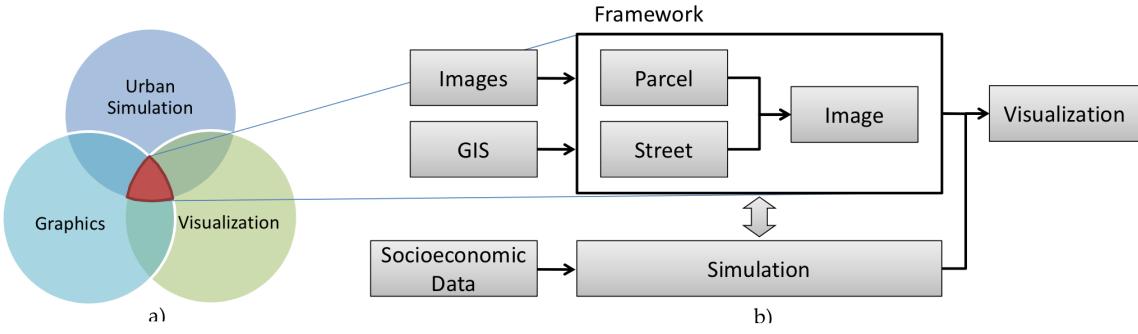


Figure 9: (a) This method is built upon the combination of urban simulation, urban visualization and computer graphics. (b) the processing pipeline completes an urban simulation with visualization. Images and GIS data, together with simulation results data, are fed into the framework to generate parcels and streets, which are then used as basis for generating urban layout images.

The visualization system can detect changes in user-defined state variables and translates such changes into transformations to the initial street and parcel information, and also to the fragments of aerial imagery associated to each transformed parcel.

This part is where the core visualization algorithm of the system lies. It can be divided into three steps: parcel generation, street generation and parcel-content generation.

4.2.1. parcel generation

The first step is to generate new parcels based on simulation results. How a parcel is partitioned is built upon some desired properties: (i) parcels generally have egress, (ii) city blocks are usually formed by rows of one or two parcels, and (iii) the contour of a parcel is most often a quadrilateral and often nearly rectangular.

Based on these properties, the system uses a recursive algorithm to partition an existing city block or a large parcel. Firstly, parcels whose population has changed (based on simulation data) are marked as candidate for subdivision. Then subdivision attempts start from the largest candidate parcel. A block-sized parcel is split into two parcels, each of which is recursively subdivided until reaching either the number of households specified by the simulation or a minimum parcel area, which is computed automatically as the mean parcel area of all parcels within a chosen distance threshold. If egress is desired (first property), a split is not made if resulting parcels will have no access to streets. This also makes sure that up to two rows of parcels are generated (second property). Since this algorithm works recursively, after a few subdivisions, most new parcels will be nearly rectangular (third property), even if the initial parcel is with complex contour.

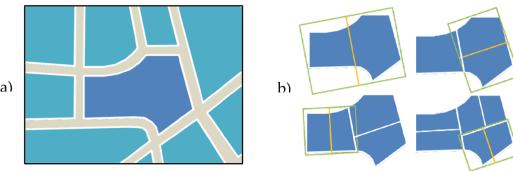


Figure 10: (a) A candidate large parcel corresponding to a city-block is divided into smaller parcels (beige=street, light blue=parcels/city-blocks, dark-blue=parcel to subdivide). (b) First four subdivisions of the original parcel yield to parcels with egress (green=oriented bounding box)

4.2.2. street generation

The second step is to generate streets. Parcel subdivision generates a subdivision line in each step. For street generation, decisions about whether the subdivision line corresponds to a new street segment or a new parcel boundary segment are made. To make the decision, the system takes the number of parcels per city-block, a value whether from simulation or specified by user, into consideration. When the subdivision line is selected to be a street segment, the system will make sure newly-generated city blocks has similar aspect ratio to that of nearby city blocks.

The geometry of a new street is obtained by perturbing the initial subdivision line segment, which is in fact a poly-line with a certain number of inner control points. These control points are used to make sure the new street segment is similar to the streets in nearby area. In this step, each initial large parcel is subdivided independently, and further processing is conducted to prevent too many deadends or T-intersections. When endpoints of two street segments are close and the turning angle is not too sharp, they are joined to form a single longer street segment.

This step uses the same stop condition as the first step. The result of this process is a new network of parcels and streets that can be output in vector format. Figure 5.3 shows an example of the recursive process, where the target number of additional parcels is twelve and the chosen approximate number of parcels per city block is four.

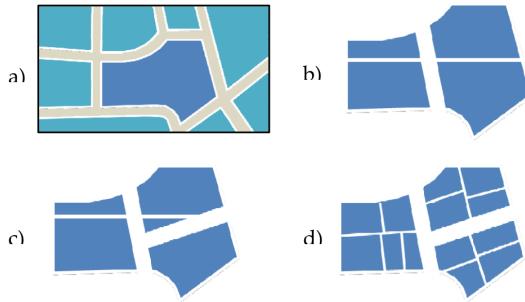


Figure 11: (a) An initial parcel (from Figure 4). (b-c) First two subdivisions are chosen to be streets. (d) In further iterations, only more parcels per city-block are produced.

4.2.3. parcel-content generation

The last step of layout generation is to generate plausible image content for each of the newly generated parcels. The way to do this is to reuse existing aerial image fragments of parcels with similar characteristics. The similarity metric is defined as the weighted sum of the similarity of the simulation state values (e.g. households per parcel, year of the construction for buildings, zoning classification, etc) and similarity of geometric shape (e.g. area, aspect ratio, etc).

Since the number of vertices of the source and destination parcels does not necessarily match, a image-wrapping algorithm is used to warp an a -sided polygon to b -sided polygon. The selected source image fragment is texture-mapped onto the destination parcel geometry and rendered together with the rest of the layout. Oriented-bounding box of both source parcel and destination parcel are partitioned into scan lines perpendicular to their longest axis. A fixed number of points is regularly spaced and the points are generated along both scan lines and culled to be inside each respective polygon. These points correspond one-to-one. The process is repeated for each scan line. The destination parcel is rendered using (i) the coordinates of the points to define a quadrilateral mesh and (ii) the coordinates of the corresponding source parcel quadrilateral mesh to make texture coordinates.

4.3. rendering and user interaction

The system enables the visualization of the simulation results using a variety of rendering methods, such as rainbow maps and choroplethic maps, to display one of several simulation variables, such as house holds, buildings, distance to



Figure 12: (a) An original urban area from where parcel image fragments are extracted; (b) The newly generated area, with some of reused image content highlighted.

large streets, distance to highways, land value, and year built of the constructions, etc. (See Figure 14)

The system loads the data within the current field-of-view dynamically, allowing user to explore the dataset freely. The whole process is fully automatic, with only minimal selection of the simulation variables and visualization methods. This simplifies the visualization task and makes a web-based viewing program possible.

4.4. side note about urban simulation

For more information about urban simulation modeling, please refer to the excellent overview by Waddell [WU04].

The urban simulation tool used by [VABW09] is UrbanSim, a widely used software-based simulation system for supporting planning and analysis of urban development, incorporating the interactions between land use, transportation, the economy, and the environment. Overviews about UrbanSim can be found at [Wad02] and [Wad11].

There are multiple research projects ongoing that utilizes or extends UrbanSim, here we list some of such projects that are of particular importance:

- **SustainCity Project** (<http://www.sustaincity.org>). This

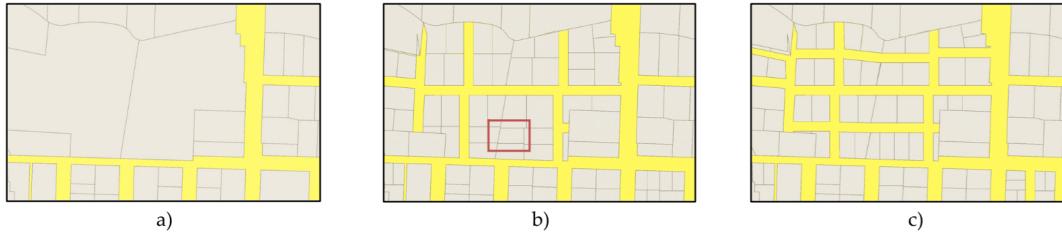


Figure 13: *Geometry generation.* (a) Original area. (b) Automatic subdivision of the area. (c) Additional streets are generated in order to enforce egress rule.

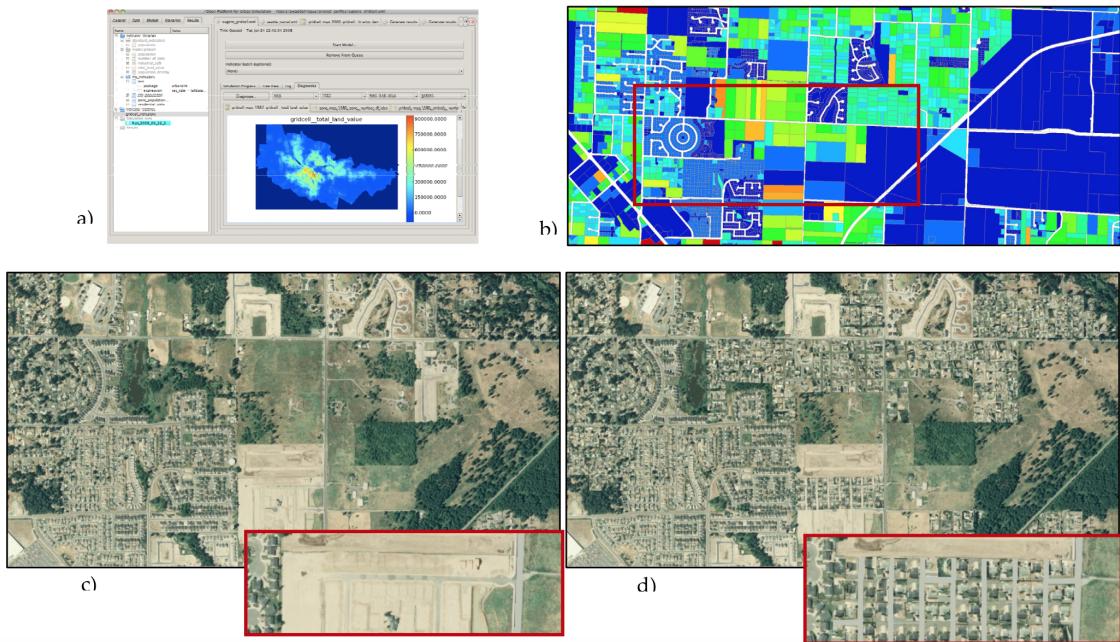


Figure 14: *Exemplary choroplethic visualizations of urban simulation data and views.* (a) A screen snapshot from the Indicator system supported by UrbanSim. (b) A rainbow color map used to represent changes in households/buildings for the initial set of parcels; blue corresponds to no change and green/yellow/red corresponds to a small/medium/large increase in the number of households. (c) The original layout of an urban area before the simulation and (d) generated layout after the simulation.

project is funded by the European Research Council and is seeking to develop an European-adapted version of UrbanSim (called UrbanSim-E) to make it more suitable for the context of European cities. The system was then tested in three European case studies: Brussels, Paris and Zurich.

- **FHWA: Modeling the Urban Continuum** (<http://urbanmodel.asu.edu/intmod.html>). This project is funded by the Federal Highway Administration, and focuses on 'Modeling the Urban Continuum in an Integrated Framework: Location Choice, Activity-Travel Behavior, and Dynamic Traffic Patterns'.
- **NSF: Robust Intelligence Project.** This project focuses

on melding artificial intelligence techniques with discrete choice econometric methods to develop dynamic models of activity and routing behaviors. UC ITS Multi-Campus Research Program on Sustainable Transportation.

These projects, among some others, are part of a initiative that is intended to develop common data structures for urban modeling and 3D visualization.

5. Interviews

We interviewed two domain scientists to get first-hand information about their everyday research work and how visualization helps.

5.1. Interview 1

Interviewee: Dr. Nebiyou Tilahun, Assistant Professor, Department of Urban Planning and Policy, University of Illinois at Chicago

What are the main research project(s) you work on?

I work on questions of travel behavior, access, and the economic outcomes for people that are linked with the provision of transportation.

What would be an ideal result from the project(s)?

The ideal would be to be able to influence transportation policy that is consistent with travel behavior and decisions that allow for equitable access to opportunities.

What kind of data do you work with most often in your research?

I work with Census data, travel behavior data (such as Travel Tracker collected by CMAP, or Travel Behavior Inventory collected by Metro Council (Minnesota)), LEHD, Census CTPP. I have also collected data through surveys when needed.

How do you gather or generate this data?

Most are available through online portals by the collecting agency.

How is this data used/analyzed?

The data is used to do empirical work. Questions of choice are analyzed as a function of cost, service availability, land use, and other variables.

What visualization tools/techniques do you use to help make sense of this data?

Mostly statistical software to do plots and graphs. Another example is what we did for the accessibility project.

What visualization tools/techniques do you use to display the data and/or communicate with other experts in your field?

Mostly figures and plots done in R. On occasion made in Excel. They are presented using Powerpoint or Keynote.

5.2. Interview 2

Interviewee: Dr. Ning Ai, Assistant Professor, Department of Urban Planning and Policy, University of Illinois at Chicago

What are the main research project(s) you work on?

In general, my research focuses on **urban environmental planning**. I have three ongoing projects: (1) modeling the sustainability, resilience, and stability of urban systems in a metabolism framework (e.g., categorizing inputs and outputs of urban systems); (2) modeling food waste generation volume and recovery/reuse opportunities in urban neighborhoods; (3) investigating how healthcare sector can reduce greenhouse emissions through sustainable transportation programs.

What would be an ideal result from the project(s)?

Ideal outcomes would be deliverables that can benefit researchers, practitioners, and general public. That means the results should be publishable, transferable and replicable in other studies and regions, and accessible to the public.

What kind of data do you work with most often in your research?

Most often we use data on demography, land use, economy, ecology, and environment at various geographic and temporal scales.

How do you gather or generate this data?

We collect data from various government agencies, industry/market reports, as well as research publications, through online data/ literature reviews, formal surveys, and personal communications.

How is this data used/analyzed?

Data may be aggregated or disaggregated using spatial and statistical analyses. Advanced models may also need to be developed for environmental and socioeconomic impact analyses.

What visualization tools/techniques do you use to help make sense of this data?

ArcGIS Spatial Analyst/Network Analysis, MS Excel and Access.

What visualization tools/techniques do you use to display the data and/or communicate with other experts in your field?

ArcGIS, Photoshop, Adobe suites.

6. Conclusion

There are many other fields in urban planning, including urban transportation, urban accessibility, urban modeling and so on. Much exciting work has been done in these directions. Due to the limit in space, we have to omit them. We hope in the future, we can conduct a more comprehensive survey of the state-of-the-art of more fields in urban planning.

References

- [AK99] AL-KODMANY K.: Using visualization techniques for enhancing public participation in planning and design: process,

- implementation, and evaluation. *Landscape and urban planning* 45, 1 (1999), 37–45. 1
- [Bat93] BATTY M.: Urban modeling in computer-graphic and geographic information system environments. *Environment and Planning B* 19 (1993), 663–663. 5
- [Bat97] BATTY M.: Virtual geography. *Futures* 29, 4 (1997), 337–352. 5
- [BKLR14] BOCK A., KLEINER A., LUNDBERG J., ROPINSKI T.: An interactive visualization system for urban search & rescue mission planning. In *Safety, Security, and Rescue Robotics (SSRR), 2014 IEEE International Symposium on* (2014), IEEE, pp. 1–7. 3
- [BSX06] BATTY M., STEADMAN P., XIE Y.: Visualization in spatial modeling. In *Complex artificial environments*. Springer, 2006, pp. 49–70. 5
- [DB07] DYKES J., BRUNSDON C.: Geographically weighted visualization: interactive graphics for scale-varying exploratory analysis. *Visualization and Computer Graphics, IEEE Transactions on* 13, 6 (2007), 1161–1168. 5
- [DBN06] DORLING D., BARFORD A., NEWMAN M.: Worldmapper: the world as you've never seen it before. *Visualization and Computer Graphics, IEEE Transactions on* 12, 5 (2006), 757–764. 5
- [DK13] DORNHEGE C., KLEINER A.: A frontier-void-based approach for autonomous exploration in 3d. *Advanced Robotics* 27, 6 (2013), 459–468. 3
- [F*03] FABIO R., ET AL.: From point cloud to surface: the modeling and visualization problem. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 34, 5 (2003), W10. 5
- [FWXL10] FU C., WANG Y., XU Y., LI Q.: The logistics network system based on the google maps api. In *Logistics Systems and Intelligent Management, 2010 International Conference on* (2010), vol. 3, IEEE, pp. 1486–1489. 1
- [FZ03] FRUH C., ZAKHOR A.: Constructing 3d city models by merging aerial and ground views. *Computer Graphics and Applications, IEEE* 23, 6 (2003), 52–61. 5
- [HJO*01] HERTZMANN A., JACOBS C. E., OLIVER N., CURLESS B., SALESIN D. H.: Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), ACM, pp. 327–340. 5
- [HRD*12] HOLZER S., RUSU R. B., DIXON M., GEDIKLI S., NAVAB N.: Adaptive neighborhood selection for real-time surface normal estimation from organized point cloud data using integral images. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (2012), IEEE, pp. 2684–2689. 5
- [KD09] KLEINER A., DORNHEGE C.: Operator-assistive mapping in harsh environments. In *Safety, Security & Rescue Robotics (SSRR), 2009 IEEE International Workshop on* (2009), IEEE, pp. 1–6. 3
- [KMO*08] KIM S., MACIEJEWSKI R., OSTMO K., DELP E. J., COLLINS T. F., EBERT D. S.: Mobile analytics for emergency response and training. *Information Visualization* 7, 1 (2008), 77–88. 3
- [Mer07] MERRELL P.: Example-based model synthesis. In *Proceedings of the 2007 symposium on Interactive 3D graphics and games* (2007), ACM, pp. 105–112. 5
- [MWH*06] MÜLLER P., WONKA P., HAEGLER S., ULMER A., VAN GOOL L.: *Procedural modeling of buildings*, vol. 25. ACM, 2006. 5
- [MZWVG07] MÜLLER P., ZENG G., WONKA P., VAN GOOL L.: Image-based procedural modeling of facades. In *ACM Transactions on Graphics (TOG)* (2007), vol. 26, ACM, p. 85. 5
- [NOS15] NAGAI Y., OHTAKE Y., SUZUKI H.: Tomographic surface reconstruction from point cloud. *Computers & Graphics* 46 (2015), 55–63. 5
- [PDBB00] PINNEL L. D., DOCKREY M., BRUSH A. B., BORNING A.: *Design of visualizations for urban modeling*. Springer, 2000. 5
- [PGA11] PINTUS R., GOBBETTI E., AGUS M.: Real-time rendering of massive unstructured raw point clouds using screen-space operators. In *Proceedings of the 12th International Conference on Virtual Reality, Archaeology and Cultural Heritage* (Aire-la-Ville, Switzerland, Switzerland, 2011), VAST'11, Eurographics Association, pp. 105–112. URL: <http://dx.doi.org/10.2312/VAST/VAST11/105-112>, doi:10.2312/VAST/VAST11/105-112. 4
- [PM01] PARISH Y. I., MÜLLER P.: Procedural modeling of cities. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), ACM, pp. 301–308. 5
- [PP10] PELLENZ J., PAULUS D.: Stable mapping using a hyper particle filter. In *RoboCup 2009: Robot Soccer World Cup XIII*. Springer, 2010, pp. 252–263. 3
- [PPP09] PEJIC A., PLETI S., PEJIC B.: An expert system for tourists using google maps api. In *Intelligent Systems and Informatics, 2009. SISY'09. 7th International Symposium on* (2009), IEEE, pp. 317–322. 2
- [PSKN06] PANSE C., SIPS M., KEIM D. A., NORTH S. C.: Visualization of geo-spatial point sets via global shape transformation and local pixel placement. *Visualization and Computer Graphics, IEEE Transactions on* 12, 5 (2006), 749–756. 5
- [RC11] RUSU R. B., COUSINS S.: 3d is here: Point cloud library (pcl). In *Robotics and Automation (ICRA), 2011 IEEE International Conference on* (2011), IEEE, pp. 1–4. 5
- [RD10] RICHTER R., DÖLLNER J.: Out-of-core real-time visualization of massive 3d point clouds. In *Proceedings of the 7th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa* (2010), ACM, pp. 121–128. 4
- [RHDW12] REDDY R., HÖFERLIN M., DAMBIER M., WEISKOPF D.: Visual analytics for dynamic evacuation planning. In *Proceedings of EuroVA 2012: International Workshop on Visual Analytics* (2012), pp. 13–17. 3
- [RSL*10] RIBARSKY W., SUBRAMANIAN K. R., LIU J., OBIRIEZE O., GUEST J.: Mobile application for first response and emergency evacuation in urban settings. In *Proceedings of the 1st International Conference and Exhibition on Computing for Geospatial Research & Application* (2010), ACM, p. 62. 3
- [RWF02] RIBARSKY W., WASILEWSKI T., FAUST N.: From urban terrain models to visible cities. *Computer Graphics and Applications, IEEE* 22, 4 (2002), 10–15. 5
- [VABW09] VANEGAS C. A., ALIAGA D. G., BENES B., WADDELL P.: Visualization of simulated urban spaces: Inferring parameterized generation of streets, parcels, and aerial imagery. *Visualization and Computer Graphics, IEEE Transactions on* 15, 3 (2009), 424–435. 5, 7
- [Wad02] WADDELL P.: Urbansim: Modeling urban development for land use, transportation, and environmental planning. *Journal of the American Planning Association* 68, 3 (2002), 297–314. 7
- [Wad11] WADDELL P.: Integrated land use and transportation

- planning and modelling: addressing challenges in research and practice. *Transport Reviews* 31, 2 (2011), 209–229. [7](#)
- [WLL13] WU Y., LIANG Z., LIU L.: Design and implementation of tourism information system based on google maps api. In *Geoinformatics (GEOINFORMATICS), 2013 21st International Conference on* (2013), IEEE, pp. 1–4. [2](#)
- [WP07] WIRTH S., PELLENZ J.: Exploration transform: A stable exploring algorithm for robots in rescue environments. In *Safety, Security and Rescue Robotics, 2007. SSRR 2007. IEEE International Workshop on* (2007), IEEE, pp. 1–5. [3](#)
- [WU04] WADDELL P., ULFARSSON G. F.: Introduction to urban simulation: Design and development of operational. [7](#)
- [WZ10] WANG Y., ZHOU W.: Settlement information management system of beijing-shanghai high-speed railway based on google map service. In *Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International* (2010), IEEE, pp. 2326–2329. [2](#)
- [XC04] XU H., CHEN B.: Stylized rendering of 3d scanned real world environments. In *Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering* (2004), ACM, pp. 25–34. [4](#)
- [ZLC*10] ZHANG H., LI M., CHEN Z., BAO Z., HUANG Q., CAI D.: Land use information release system based on google maps api and xml. In *Geoinformatics, 2010 18th International Conference on* (2010), IEEE, pp. 1–4. [2](#), [3](#)