Procedural City Generation Beyond Game Development

Joon-Seok Kim[1], Hamdi Kavak[1], Andrew Crooks[1,2]
[1]Department of Geography and Geoinformation Science, George Mason University, USA
[2]Department of Computational and Data Sciences, George Mason University, USA
{jkim258,hkavak,acrooks2}@gmu.edu

Abstract

The common trend in the scientific inquiry of urban areas and their populations is to use real-world geographic and population data to understand, explain, and predict urban phenomena. We argue that this trend limits our understanding of urban areas as dealing with arbitrarily collected geographic data requires technical expertise to process; moreover, population data is often aggregated, sparsified, or anonymized for privacy reasons. We believe synthetic urban areas generated via procedural city generation, which is a technique mostly used in the gaming area, could help improve the state-of-the-art in many disciplines which study urban areas. In this paper, we describe a selection of research areas that could benefit from such synthetic urban data and show that the current research in procedurally generated cities needs to address specific issues (e.g., plausibility) to sufficiently capture real-world cities and thus take such data beyond gaming.

1 Introduction

Urban areas are complex systems composed of densely-situated populations which are mobile and interact with each other. It is the structure (i.e., form) and function (i.e., how people use areas) of urban areas that impact how people use, extend, and manipulate such environments. Many disciplines such as geography, data science, and the social sciences more generally study urban areas and its populations. The intention of these disciplines is often to understand, explain, and predict various urban phenomena ranging from gentrification [3] to traffic jams [36]. A common approach followed by these disciplines is to *inquire about a specific scientific question*, *capture or obtain empirical data related to the question*, and *use or create a data-driven model that advances the current body of knowledge*.

Such empirical data can be placed into one of two groups: *geographic data* and *population data*. Here we refer to geographic data to include maps of administrative areas, land use, location footprints, point-of-interests, various levels of road networks, and satellite images. Geographic data is often publicly available (especially in developing and developed countries). While population data includes socioeconomic data such as census information (i.e., general population characteristics) and mobility data in the form of check-ins, travel diaries, public transportation information, and traffic sensors to name but a few. However, unlike geographic data, population data is often at times restricted and aggregated as it contains sensitive information of people or sometimes not available at all (as is the case in less developed countries [40]).

Although both geographic and population data are frequently used in research, their use poses several challenges. The recent emergence of volunteered geographic information (VGI, [19]) makes large-scale geographic data possible via initiatives like OpenStreetMap; but due to its relaxed contribution rules, such spatial network data is flexible but has no guarantee of correctness. Moreover, vandalism is one of the latent challenges VGI is facing [43]. Common problems in terms of quality include missing/wrong tags (i.e., misclassification of features), digitizing error (e.g., overshoot/undershoot), topologically inconsistent data (e.g., spaghetti model) and so on. These require costly post-processing (e.g., cleaning data) when being utilized for analyzing urban areas. While tools for and mechanism of quality assurance and quality control (QA/QC) have been developed to improve quality of real datasets, they require technical expertise in data processing which is often lacking in many fields exploring urban areas. Population data pertaining to individuals' movement is mostly aggregated, sparsified, or anonymized to preserve the privacy of individuals. Many nontechnical scientists face one or more of these challenges when using data in each urban area they focus.

We would argue that, while focusing on single urban areas is a necessity for specific applications (e.g., for urban planning), in other instances it might be more desirable to work on standardized synthetic urban areas should they have sufficient details for the question at hand. For instance, the self-driving car technology aims to improve the safety of the real-world roads and reduce fatal accidents; it is quite possible to test self-driving algorithms and their safety on an entirely synthetic simulated urban area that resembles the real-world urban areas. Moreover, this standardized synthetic urban area could be used to benchmark different algorithms by different companies in the self-driving marketplace. For such a synthetic dataset to be created and used, urban areas needs to be roughly characterized with respect to their form (e.g., mono-centric, poly-centric, road distributions) and characteristics of the inhabitants (e.g., density, distribution, social characteristics, etc.) so that they can resemble in a synthetic form. To our knowledge, there is only a handful of studies that partly tackle creating such synthetic urban areas for a broader scientific community [33, 34].

This is where the *procedural city generation* (PCG) techniques become useful. Unlike manual data generation that needs substantial effort, procedural generation is performed by a procedure to automatically generate content and data. Currently, many existing PCG approaches focus on the game industry and its requirements [52, 57]. We believe that synthetic urban areas generated through PCG techniques can provide great opportunities for the scientific community at large and help to advance the *state-of-the-art* in many disciplines by providing a standard dataset to test ideas, hypotheses, and theories about urban phenomena. Furthermore, with an interdisciplinary contribution (especially from the social sciences), the impact can be even greater. In Section 2, we present such application areas that we believe could benefit from such synthetic urban areas. In Section 3, we survey the current state in PCG and express the gaps in the literature. We conclude by providing some future research directions in Section 4.

2 Application Areas: From Social Simulation to Urban Testbeds

We identify two broad and related application areas that PCG techniques could make a great impact with regards to studying urban areas. The first and perhaps the most important one is the *social simulation*. Social simulation is a modeling paradigm that allows exploring social systems from an individualistic angle (i.e., from the bottom-up via agent-based models). The second one is *urban testbeds*, a software technology to conduct costly experiments in a synthetic environment. Below we describe each of these areas (Sections 2.1 and 2.2 respectively), their inter-relations, and how PCG could impact them.

2.1 Social Simulation

Social simulation, or sometimes called Agent-Based Simulation, is a relatively new modeling paradigm that allows representing and inquiring social systems from a bottom-up perspective [17]. That is, social system entities (e.g., humans, firms, organizations) are individually represented by their own decision making logic and simulated to understand emerging aggregated patterns. Social scientists are increasingly using spatial networks and other geographical data in their simulations to develop empirically-grounded models [10]. To this end, even theoretical models (e.g., segregation model of Schelling [50]) have been supported with geographical data to elicit new insights into the process of segregation [11]. While this exciting adoption of geospatial technology has created new opportunities for studying cities and smaller or larger geographies, it comes with several challenges that need to be addressed.

For instance, geographical data is often crowd-sourced via VGI or collected without following a strict guideline. As a result, many open source geographical data are messy with missing/wrong tags etc. as discussed in Section 1. For instance, the Topologically Integrated Geographic Encoding and Referencing (TIGER) data are widely used in the SIGSPATIAL community for experiments [47]. But even today, the quality of the data is in question [62]. Often using such data leaves the (nontechnical) social scientist with exhaustive work of data cleaning and pre-processing in order to incorporate such geographical data into their social simulation models. Even worse is the case when the same model is used to study another geographic area which requires the modeler to make sure that new area data is properly prepared.

The spatial data community could help to address the aforementioned challenges and help advance the *state-of-the-art* in social simulations. Especially the main contribution could be creating synthetic urban areas that would help the social science modeler to generate standardized geographical datasets with plausible characteristics of cities. For instance, it would be desirable to generate synthetic cities with an arbitrary number diverse of inhabitants and plausible urban geometry [5] not only for the spatial network (e.g., roads) but also other environmental pieces like the point of interests, etc.

Having such advanced geospatial data generators would help achieve scientific impact what is way wider than what the spatial data community often deals with. Social simulation provides a virtual laboratory for testing existing social theories and create new ones [14]. Synthetic geospatial data, when generated according to *stylized facts* [20] about urban areas, could aid theory testing and the creation activities in three main points. (1) Examining the impact of geography on the robustness of a theory. For instance, how do physical obstacles affect the spread of ideas or innovation? (2) Facilitating the means for comparing and aligning different theories (i.e., models) more objectively. Which theory better predicts the spread of ideas or innovation under the same environmental conditions. (3) Standardizing the structure and naming of geographic and population data thus saving time and effort.

2.2 Urban Testbeds

We define an urban testbed as a synthetic software system that has the ability to represent and simulate an urban area in sufficient detail with the goal of providing rigorous and replicable testing platform for various application areas. Urban testbeds have two main component: *the urban environment* that is generated using PCG techniques and *the urban population* that is created simulated based on the principles of agent-based modeling. Depending on the test in hand, the abstraction level of the representation of the city and urban population may change. In the era of smart cities, urban testbeds could play a critical role in future urban developments. Below, we identify up and coming areas that could benefit from urban testbeds created with PCG techniques.

Self-driving cars and transportation: Self-driving cars have been a long dream not only for car makers but also drivers [59]. In the last few years, this dream is coming near to reality due to initiatives from technology companies, start-ups, and car makers that develop and deploy artificial intelligence techniques into self-driving cars (e.g., Google's Waymo, Tesla's Autopilot). Due to unforeseen fatal accidents occurring despite all efforts, self-driving car technologies need rigorous testing platforms in an isolated, synthetic environment which is a great application area for urban testbeds. Such a testbed could help such self-driving algorithms adapt and get validated in different urban settings while at the same time within the safe environment of a computer. In a more broader perspective, new transportation systems or additions to existing transportation systems (e.g., underground, sea, or air) could also be a good case for urban testbeds.

Utility infrastructure and services: Utility services in the urban setting are as critical as, if not more than, the transportation systems. Urban areas in the world have services including electricity, gas, water, cable, and garbage collection. Major changes to such services or the impact of natural disasters need rigorous testings

[38]. Urban testbeds with proper utility service infrastructure implementation could serve as an objective way of testing such changes [41]. While urban testbeds might not be suitable for real-world testing, for example changes in a utility service for a specific city, it could potentially fill an important gap when it comes to testing changes at the conceptual level. For instance, one could test the potential of delivering electricity via cables vs. wirelessly on an urban testbed which is currently only a futuristic idea and thus exploring general notions of adoption and coverage needs of such technological innovations.

3 Procedural City Generation

In this section, we review work related to PCG from several perspectives: *goals*, *inputs*, *outputs* and *methods*. All procedural city generators (e.g., [12, 21, 23, 24, 25, 45, 53]) are subject to specific **goals** such as a realistic scene in a movie or game (e.g., [63]). For this reason, game environment generation or content generation [12, 21] tends to focus on computer graphics including generating 3D meshes, textures, and animation effects that look realistic. Due to the physical extent and vertical dimensions (in terms of both the natural and built environment) of real-world cities, the majority of content may be automatically created by generators; yet, *user interaction* is a necessary feature to enhance and refine specific details and to obtain the required level of detail data needed to meet the goal of the application [25]. Moving from the movie and game industries to urban planning and analysis, often the goal concerning city generation entails simulations to evaluate potential renderings of conceived plans such as new city developments [26]. In which case, real-world datasets are likely to be considered as an input to PCG. To harmonize synthetic datasets with real-world datasets, data formats for interoperability such as Open Geospatial Consortium (OGC) standards (e.g. CityGML, IndoorGML [27], Common DataBase (CDB)[49], GeoPackage, etc.) need to be employed [32]. As discussed in Section 2.1, social simulation needs geographic and population datasets that are *plausible* whether real-world data is used as input or not. By plausible, we mean that characteristics of the generated city should fall within the properties of real cities (e.g., topological characteristics of the road network).

Existing procedural city generators tend to create one or more of the following **outputs**: geographic environments (e.g., terrain [6, 52], water bodies [48], and vegetation [13]), urban components (e.g., road networks [7], traffic signs [56], land uses [34], population [39], and social networks [1]), buildings [8] (e.g., building layout [42], interiors [57, 58], and furniture arrangement [16]) and textures [35]. Figure 1 shows city generators with relationships among them, where each solid box represents a generator and each directional edge represents an input-output relationship between them. Depending on generators, an input/output relationship can be represented as a bidirectional edge as shown in Figure 1. For instance, spatial networks can be utilized to define city layouts and vice versa. Because each component has different characteristics, generation techniques used for each vary.

Generation **methods** for procedural cities can be categorized as follows: generative grammar, simulation-based, tensor field, stochastic, data-driven, and inverse procedural generation. Urban components including natural environments can be described as a fractal and hierarchical structure [4, 6] and such a structure is often implemented by generative grammars, one of the most popular methods to generate artificial patterns [53]. Since Lindenmayer [37] first introduced the L-system in biology, many variations including stochastic L-system [15] and radial L-system [51] have been developed. To overcome some of the limitations (e.g., lack of multi-dimensionality) of the L-system, other generative grammars such as shape [54], split[64], and generalized grammars [29] have been developed.

Independently of the grammars above, tensor fields have been developed for PCG which can smooth road networks along with geographic environments such as terrain and water bodies [9]. *Simulation-based* generation employs simulation techniques such as agent-based modeling [33, 34] to generate plausible data. For instance, iterative generation proposed in [7] simulates road traffic to prescribe expanding road networks to accommodate more population. *Stochastic* approaches including Perlin noise [46] are widely used to generate terrain and

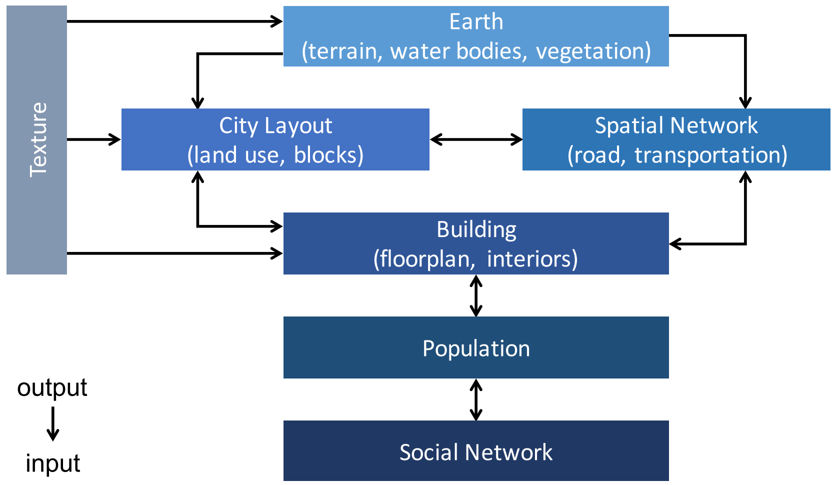

Figure 1: Procedural city generation and input-output relationships

textures. The main idea of *data-driven* generation is to weave predefined or existing data including templates [55], patterns (e.g., population-based, radial, raster, mixed) or real examples [44] into unified data. Inverse procedural generation [65] strives to understand real data in a reverse-engineering manner and take advantage of other generation techniques such as generative grammar.

In what follows, we discuss about **inputs** of generation and factors to shape cities. For creators who have control over generation and interaction with real data, inputs are considered a vital factor. Since urban components are deeply relevant each other, real datasets or outputs of one generation become inputs of other component generators. For example, CityEngine [45] employs a set of statistical and geographical input data. Natural environments are most likely to be the first-order influencer to form urban components unless humankind is involved in reconstructing (e.g., deforesting and reclaiming) nature structures. Population maps are used to control the size and the shape of urban structures [45, 60]. Historical events are also a candidate of inputs to manipulate cities over time [28].

From a different perspective, urban scientists have studied various factors of urban structures and city growth [30, 31]. Such factors can be roughly divided into three [31]: (1) natural environment, (2) human activities, including movement and occupation of land, (3) the physical productions of transformation, including both built and planted features. We would also add additional factors to complement these including (a) benefits to individuals and society, (b) economy, and (c) technologies. Geographic environments such as a river and a forest can provide benefits that attract population. Location, available natural resources, and climates are also a factor to determine a type of city: resource city, processing city, market city, and others [22]. Activities and events shape the city as well [28]. The size and population of the city are affected by those benefits. Also, the economy has played a role in city growth [2]. Technologies transform shapes of cities in many ways. Especially, road networks are affected by transportation [18, 61]. Even with the same technology, a paradigm in society can build a different transportation system such as bicycle sharing and bus rapid transit (BRT).

4 Future Direction

From our review in the previous section, in this section, we address open issues of PCG for a wide range of users.

- **Plausibility**: It is an intrinsic requirement of data generators. However, there exist thousands of cities in our world, each with different shapes. Plausibility does not mean a synthetic city should simply resemble one of them, but a plausible procedural generator should allow creators to create a wide variety of intended cities.

- **Diversity vs. controllability**: To achieve diversity of data, stochastic elements in the procedural generation are inevitable. Such randomness enables us to create massive amounts of data with diversity. However, they have difficulty in controlling their outputs due to randomness. Thus, controllability with diversity will help advanced users to achieve the results they want. For instance, it would allow users to opt for various urban features such as spatial network from small scale to large scale, from monocentric to polycentric, and from organic to planned cities.

- **Interoperability**: Many different solutions for PCG have been studied. While some of them are standalone PCG to create most of the content ranging from terrain to buildings [45], many of them focus on specific features such as terrain, building, and road networks. For one type of features, even different PCG techniques can be used, e.g., tensor field [9] and L-system[45] for spatial networks. There is no best solution that fits all. Therefore, a unified solution consisting of different implementations can complement each other. If they can interface with others through standardized formats as we discussed in Section 3, we expect integration can be resolved.

- **Level of detail**: Not every application requires high-quality data (i.e., high level of detail) as seen in computer games. While some application may want 3D buildings with polygonal roads with high-quality rendering in a 3D virtual world, some simulations may need just a graph of a road network with 2D footprints of buildings for simulating commutes to work.

- **Ease of use**: Since user inputs determine a shape of the city among numerous cases, PCG may require many parameters and their combination. Most of the users simply want plausible datasets without complex configurations. A gallery of synthetic cities with predefined parameters will be helpful for users.

- **Cost**: Cost is one of elements to hinder use of real datasets. Similarly, it will discourage use of procedural generators if users have to pay the same amount of cost including time and effort. A publicly available procedural city generator is needed if we are going to advance their use in social simulation and as a testbed for urban issues (Section 2).

A city is an artifact of numerous interactions between people who currently live or did live in them, cities are not just created today but are shaped by past decisions and actions of others. An ultimate city generator should be a simulator taking all the factors that shape and potentially will shape future cities into account so that it can generate synthetic cities that resemble real cities, even capable of drawing future cities. To make that happen, several things need to be done. First and foremost, it is required to develop a method to measure *similarity* between a synthetic city and a real city. Without adequate measurements, we cannot guarantee outputs of PCG are plausible. Secondly, *across-the-board parameters* of PCG that capture characteristics of a city and all features in it need to be defined (e.g., dimensions [6]). Lastly, modeling technologies that affect society and form a city is needed. A plug-and-play model would allow users to conduct meaningful experiments (e.g., how autonomous vehicles or drones can existing transportation networks) but at the same time provide a synthetic city to bench mark new algorithms or models.

References

[1] M. Alizadeh, C. Cioffi-Revilla, and A. Crooks. Generating and analyzing spatial social networks. *Computational and Mathematical Organization Theory*, 23(3):362–390, 2017.

[2] A. Anas, R. Arnott, and K. A. Small. Urban spatial structure. *Journal of economic literature*, 36(3):1426–1464, 1998.

[3] C. Atkinson, R. Atkinson, and G. Bridge. *Gentrification in a Global Context*. Housing and Society Series. Taylor & Francis, 2004.

[4] M. Batty. *Cities and complexity: understanding cities with cellular automata, agent-based models, and fractals*. The MIT press, 2007.

[5] M. Batty. The size, scale, and shape of cities. *Science*, 319(5864):769–771, 2008.

[6] M. Batty and P. A. Longley. *Fractal cities: a geometry of form and function*. Academic press, 1994.

[7] J. Beneš, A. Wilkie, and J. Křivánek. Procedural modelling of urban road networks. In *Computer Graphics Forum*, volume 33, pages 132–142. Wiley Online Library, 2014.

[8] F. Biljecki, H. Ledoux, and J. Stoter. Generation of multi-LOD 3D city models in CityGML with the procedural modelling engine Random3Dcity. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, pages 51–59, 2016.

[9] G. Chen, G. Esch, P. Wonka, P. Müller, and E. Zhang. Interactive procedural street modeling. In *ACM transactions on graphics (TOG)*, volume 27, page 103. ACM, 2008.

[10] A. Crooks, C. Castle, and M. Batty. Key challenges in agent-based modelling for geo-spatial simulation. *Computers, Environment and Urban Systems*, 32(6):417 – 430, 2008. GeoComputation: Modeling with spatial agents.

[11] A. T. Crooks. Constructing and implementing an agent-based model of residential segregation through vector gis. *International Journal of Geographical Information Science*, 24(5):661–675, 2010.

[12] D. M. De Carli, F. Bevilacqua, C. T. Pozzer, and M. Cordeiro dOrnellas. A survey of procedural content generation techniques suitable to game development. In *Games and Digital Entertainment (SBGAMES), 2011 Brazilian Symposium on*, pages 26–35. IEEE, 2011.

[13] O. Deussen, P. Hanrahan, B. Lintermann, R. Měch, M. Pharr, and P. Prusinkiewicz. Realistic modeling and rendering of plant ecosystems. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 275–286. ACM, 1998.

[14] S. Y. Diallo, J. J. Padilla, I. Bozkurt, and A. Tolk. Modeling and simulation as a theory building paradigm. In *Ontology, Epistemology, and Teleology for Modeling and Simulation*, pages 193–206. Springer, 2013.

[15] P. Eichhorst and W. J. Savitch. Growth functions of stochastic lindenmayer systems. *Information and Control*, 45(3):217–228, 1980.

[16] T. Germer and M. Schwarz. Procedural arrangement of furniture for real-time walkthroughs. In *Computer Graphics Forum*, volume 28, pages 2068–2078. Wiley Online Library, 2009.

[17] N. Gilbert, P. Gilbert, and S. Publications. *Agent-Based Models*. Number no. 153 in Agent-based Models. SAGE Publications, 2008.

[18] M. F. Goodchild. Gis and transportation: status and challenges. *GeoInformatica*, 4(2):127–139, 2000.

[19] M. F. Goodchild. Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69(4):211–221, 2007.

[20] B.-O. Heine, M. Meyer, and O. Strangfeld. Stylised facts and the contribution of simulation to the economic analysis of budgeting. *Journal of Artificial Societies and Social Simulation*, 8(4), 2005.

[21] M. Hendrikx, S. Meijer, J. Van Der Velden, and A. Iosup. Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 9(1):1, 2013.

[22] J. H. Johnson. *Urban geography: an introductory analysis*. Elsevier, 2013.

[23] G. Kelly and H. McCabe. A survey of procedural techniques for city generation. *The ITB Journal*, 7(2):5, 2006.

[24] G. Kelly and H. McCabe. Citygen: An interactive system for procedural city generation. In *Fifth International Conference on Game Design and Technology*, pages 8–16, 2007.

[25] G. Kelly and H. McCABE. An interactive system for procedural city generation. *Institute of Technology Blanchardstown*, page 25, 2008.

[26] B. Kim, A. Jarandikar, J. Shum, S. Shiraishi, and M. Yamaura. The smt-based automatic road network generation in vehicle simulation environment. In *Embedded Software (EMSOFT), 2016 International Conference on*, pages 1–10. IEEE, 2016.

[27] J.-S. Kim, S.-J. Yoo, and K.-J. Li. Integrating indoorgml and citygml for indoor space. In *International Symposium on Web and Wireless Geographical Information Systems*, pages 184–196. Springer, 2014.

[28] L. Krecklau, C. Manthei, and L. Kobbelt. Procedural interpolation of historical city maps. In *Computer Graphics Forum*, volume 31, pages 691–700. Wiley Online Library, 2012.

[29] L. Krecklau, D. Pavic, and L. Kobbelt. Generalized use of non-terminal symbols for procedural modeling. In *Computer Graphics Forum*, volume 29, pages 2291–2303. Wiley Online Library, 2010.

[30] K. Kropf. Aspects of urban form. *Urban Morphology*, 13(2):105, 2009.

[31] K. Kropf. *The Handbook Of Urban Morphology*. John Wiley & Sons Ltd, 2017.

[32] M. Krückhans. Iso and ogc compliant database technology for the development of simulation object databases. In *Simulation Conference (WSC), Proceedings of the 2012 Winter*, pages 1–9. IEEE, 2012.

[33] T. Lechner, B. Watson, and U. Wilensky. Procedural city modeling. In *In 1st Midwestern Graphics Conference*, 2003.

[34] T. Lechner, B. Watson, U. Wilensky, S. Tisue, M. Felsen, A. Moddrell, P. Ren, and C. Brozefsky. Procedural modeling of urban land use. Technical report, North Carolina State University. Dept. of Computer Science, 2007.

[35] J. Legakis, J. Dorsey, and S. Gortler. Feature-based cellular texturing for architectural models. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 309–316. ACM, 2001.

[36] Y. Li and C. Shahabi. A brief overview of machine learning methods for short-term traffic forecasting and future directions. *SIGSPATIAL Special*, 10(1):3–9, 2018.

[37] A. Lindenmayer. Mathematical models for cellular interactions in development. *Journal of theoretical biology Parts I and II*, 18(3):280–299, 1968.

[38] R. A. Loggins, W. A. Wallace, and B. Cavdaroglu. Municipal: A decision technology for the restoration of critical infrastructures. In *Proceedings of the 2013 Industrial and Systems Engineering Research Conference*, pages 1767–1776. Institute of Industrial and Systems Engineers (IISE), 2013.

[39] X. Lyu, Q. Han, and B. De Vries. Procedural urban modeling of population, road network and land use. *Transportation Research Procedia*, 10:327–334, 2015.

[40] R. Mahabir, A. Croitoru, A. T. Crooks, P. Agouris, and A. Stefanidis. A critical review of high and very high-resolution remote sensing approaches for detecting and mapping slums: Trends, challenges and emerging opportunities. *Urban Science*, 2(1):8, 2018.

[41] D. Mendonça, W. A. Wallace, B. Cutler, and J. Brooks. Synthetic environments for investigating collaborative information seeking: An application in emergency restoration of critical infrastructures. *Journal of Homeland Security and Emergency Management*, 12(3):763–784, 2015.

[42] P. Merrell, E. Schkufza, and V. Koltun. Computer-generated residential building layouts. In *ACM Transactions on Graphics (TOG)*, volume 29, page 181. ACM, 2010.

[43] P. Neis, M. Goetz, and A. Zipf. Towards automatic vandalism detection in openstreetmap. *ISPRS International Journal of Geo-Information*, 1(3):315–332, 2012.

[44] G. Nishida, I. Garcia-Dorado, and D. G. Aliaga. Example-driven procedural urban roads. In *Computer Graphics Forum*, volume 35, pages 5–17. Wiley Online Library, 2016.

[45] Y. I. Parish and P. Müller. Procedural modeling of cities. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 301–308. ACM, 2001.

[46] K. Perlin. An image synthesizer. *SIGGRAPH Comput. Graph.*, 19(3):287–296, 1985.

[47] D. Pfoser and C. S. Jensen. Indexing of network constrained moving objects. In *Proceedings of the 11th ACM International Symposium on Advances in Geographic Information Systems*, GIS '03, pages 25–32, New York, NY, USA, 2003. ACM.

[48] P. Prusinkiewicz and M. Hammel. A fractal model of mountains with rivers. In *IN PROCEEDINGS OF GRAPHICS INTERFACE '93*, pages 174–180, 1993.

[49] S. Saeedi, S. Liang, D. Graham, M. F. Lokuta, and M. A. Mostafavi. Overview of the ogc cdb standard for 3d synthetic environment modeling and simulation. *ISPRS International Journal of Geo-Information*, 6(10):306, 2017.

[50] T. C. Schelling. Dynamic models of segregation. *Journal of mathematical sociology*, 1(2):143–186, 1971.

[51] G. Siromoney and R. Siromoney. Radial grammars and radial l-systems. *Computer Graphics and Image Processing*, 4(4):361–374, 1975.

[52] R. M. Smelik, K. J. de Kraker, S. A. Groenewegen, T. Tutenel, and R. Bidarra. A survey of procedural methods for terrain modelling. In *Proceedings of the CASA Workshop on 3D Advanced Media In Gaming And Simulation*, pages 25–34, 2009.

[53] R. M. Smelik, T. Tutenel, R. Bidarra, and B. Benes. A survey on procedural modelling for virtual worlds. In *Computer Graphics Forum*, volume 33, pages 31–50. Wiley Online Library, 2014.

[54] G. Stiny and J. Gips. Shape grammars and the generative specification of painting and sculpture. pages 1460–1465, 1972.

[55] J. Sun, X. Yu, G. Baciu, and M. Green. Template-based generation of road networks for virtual city modeling. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 33–40. ACM, 2002.

[56] F. Taal and R. Bidarra. Procedural generation of traffic signs. In *Proceedings of the Eurographics Workshop on Urban Data Modelling and Visualisation*, pages 17–23. Eurographics Association, 2016.

[57] T. Tutenel, R. Bidarra, R. M. Smelik, and K. J. De Kraker. Rule-based layout solving and its application to procedural interior generation. In *CASA Workshop on 3D Advanced Media In Gaming And Simulation*, 2009.

[58] T. Tutenel, R. M. Smelik, R. Lopes, K. J. De Kraker, and R. Bidarra. Generating consistent buildings: a semantic approach for integrating procedural techniques. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3):274–288, 2011.

[59] C. Urmson and W. . Whittaker. Self-driving cars and the urban challenge. *IEEE Intelligent Systems*, 23(2):66–68, March 2008.

[60] C. A. Vanegas, D. G. Aliaga, B. Benes, and P. A. Waddell. Interactive design of urban spaces using geometrical and behavioral modeling. In *ACM Transactions on Graphics (TOG)*, volume 28, page 111. ACM, 2009.

[61] N. Waters. Transportation gis: Gis-t. Wiley, 2005.

[62] O. Wiki. Tiger fixup. `https://wiki.openstreetmap.org/wiki/TIGER_fixup/`, Oct. 2018.

[63] Wikipedia. Need for speed. `https://en.wikipedia.org/wiki/Need_for_Speed`, Oct. 2018.

[64] P. Wonka, M. Wimmer, F. Sillion, and W. Ribarsky. *Instant architecture*, volume 22. ACM, 2003.

[65] F. Wu, D.-M. Yan, W. Dong, X. Zhang, and P. Wonka. Inverse procedural modeling of facade layouts. *ACM Transactions on Graphics (TOG)*, 33(4):121, 2014.