The SIGSPATIAL Special

Newsletter of the Association for Computing Machinery
Special Interest Group on Spatial Information

Volume 10 Number 2 July 2018

The SIGSPATIAL Special

The SIGSPATIAL Special is the newsletter of the Association for Computing Machinery (ACM) Special Interest Group on Spatial Information (SIGSPATIAL).

ACM SIGSPATIAL addresses issues related to the acquisition, management, and processing of spatially-related information with a focus on algorithmic, geometric, and visual considerations. The scope includes, but is not limited to, geographic information systems.

For more details and membership information for ACM SIGSPATIAL as well as for accessing the newsletters please visit http://www.sigspatial.org.

The SIGSPATIAL Special serves the community by publishing short contributions such as SIGSPATIAL conferences' highlights, calls and announcements for conferences and journals that are of interest to the community, as well as short technical notes on current topics. The newsletter has three issues every year, i.e., March, July, and November. For more detailed information regarding the newsletter or suggestions please contact the editor via email at azufle@gmu.edu.

The SIGSPATIAL Special (ISSN 1946-7729) Volume 10, Number 2, July 2018.

Table of Contents

Message from the Editor

Andreas Züfle
Department of Geography and GeoInformation Science, George Mason University, USA
Email: azufle@gmu.edu

Dear SIGSPATIAL Community,

The newsletter serves the community by publishing short contributions such as SIGSPATIAL conferences' highlights, calls and announcements for conferences and journals that are of interest to the community, as well as short technical notes on current topics. This July 2018 continues to feature a special topic of "Urban Analytics and Mobility". The choice for this topic follows the rapid trend of the last years: The UrbanGIS'17 workshop at SIGSPATIAL 2017 was one of the largest workshops, many papers and research sessions focused on related topics, and the SIGSPATIAL 2017 keynote by Bryan Mistele, Founder & CEO of INRIX, discussed many future challenges in urban environments.

I want to sincerely thank all authors of for their generous contributions of time and effort that made this issue possible. I hope that you will find the newsletters interesting and informative and that you will enjoy this issue.

You can download all Special issues from:

`http://www.sigspatial.org/sigspatial-special`

Yours sincerely,
Andreas Züfle
SIGSPATIAL Newsletter Editor

The SIGSPATIAL Special

Special Issue on Urban Analytics and Mobility (Part 2)

ACM SIGSPATIAL
http://www.sigspatial.org

Introduction to this Special Issue: Urban Analytics and Mobility (Part 2)

Andreas Züfle
Department of Geography and GeoInformation Science, George Mason University
Email: azufle@gmu.edu

According to a US Census report [2], the daytime population of cities like Washington D.C. nearly doubles the nighttime population, coining the notion of "Mega Commuting". To understand, explain, and predict urban mobility, our current data-centered era provides a plethora of rich data sources. These data sources capture mobility on the road, including GPS trajectories, metro, bus and taxi origin-destination data, indoor navigation data and many more types and sources of data.

These rich data sources present challenges and opportunities to develop new spatial and spatio-temporal data management systems, as well as novel geographic information systems. Broader impacts of this research directly affect urban life, such as a reduction of the 11 billion liters of fuel wasted traffic each year in the the United States [1]. This special issue of the SIGSPATIAL Special Newsletter contains five articles which present visions, challenges, and solutions to improve transportation issues in urban environments.

1. In the first article, we hit the road: Eftelioglu surveys the challenge and future research directions of finding "avoidance patterns" using GPS trajectory data. Therefore, the challenge is to automatically identify areas of a road network that users are avoiding, for reasons such as potholes and crime,
2. the second article takes us indoors: Cheema gives an overview of challenges and opportunities using indoor location-based services towards making them as ubiquitous as their outdoor counterpart,
3. for the third article, we use peer-to-peer ride-sharing services: Tong and Zhou describe the challenge of dynamically and efficiently assigning tasks for spatial crowdsourcing platforms, such as ride-sharing services, to minimize the overhead on the road,
4. for the fourth article, we take the bus: Fei and Gkountouna propose to analyze bus data, including GPS and odometer readings (distance traveled), to find spatio-tempral patterns of congested areas. These patterns will be paramount towards future research on more efficient public transportation,
5. In the fifth and final article, we visit alternate worlds: Kim, Kavak and Crook propose urban simulation as a paradigm to generate, simulate, explain and predict urban population and mobility. They propose the challenge of creating socially plausibly simulations that capture the complexity of real-world cities, thus providing unlimited and perfect data of all aforementioned urban mobility data types.

I would like to thank the authors for their contributions, and I hope the readers will enjoy this issue and find it useful in their research work.

References

[1] D. Schrank, B. Eisele, T. Lomax, and J. Bak. Urban Mobility Scorecard. The Texas A&M Transportation Institute and INRIX, 2015.

[2] U.S. Census Bureau. U.S. Department of Commerce. Economics and Statistics Administration. Measuring America: An Overview to Commuting and Related Statistics https://www.census.gov/content/dam/Census/data/training-workshops/recorded-webinars/commuting-nov2014.pdf.

Urban Analytics in the Context of Public Safety: The Case of Avoidance Patterns

Emre Eftelioglu, Cargill Inc., USA

Abstract

Given a collection of geolocated activities, the goal of urban analytics in the context of public safety is to discover the underlying motives of people that affect their movement/activity patterns in space and time. Understanding the spatial patterns from urban mobility/activity datasets is an important task in public safety, city planning and sociology since these may reveal the underlying causes of crimes and safety issues, as well as behavior changes of individuals. Avoidance patterns are a type of behavioral change characterized by a lack of movement contrary to expectation. Avoidance pattern detection is a challenging task due to the lack of observations (e.g. lack of movement), defining the expected "normal" movement and large datasets (i.e. high number of GPS trajectories which are spread across the study area and large road network graphs). In addition, these challenges are exacerbated by the complicated and often hidden drivers of human activities and the complex relationships and dependencies between the spatially associated features.

In this paper, we will provide a brief overview of the state-of-the-art spatial data science approaches in the context of avoidance patterns. First, we introduce the background from the domain (i.e. public safety) perspective, followed by an overview of the current state-of-the-art work. Then we will discuss possible future directions that may help shape future research on the topic.

1 Introduction and Motivation

With the increasing availability of geolocated data collected from a variety of sources, there is a tremendous opportunity to understand the movement and activity patterns of people [3]. These patterns are influenced by many motives which include the underlying demographics, goals (e.g. sightseeing, shopping, work-home commute, etc.), road conditions, etc. One of such patterns is avoidance. Avoidance patterns are the locations where drivers/pedestrians try to bypass when commuting. These are the result of a variety of driver concerns including rush hour traffic when there is congestion, road imperfections (e.g. potholes, etc.), safety of a neighborhood as well as hiding from detection (e.g. criminals' avoidance behavior).

Avoidance patterns are overlooked compared to other work on urban mobility analytics despite their importance to understand human behavior. This is due to the ease of focus on the observable phenomena rather than the lack of phenomena in urban analytics.

One way to define the avoidance pattern is as the area between the shortest path and the taken path by the driver [7]. Another definition may be the segment of the shortest path that is different than the taken path. These different definitions of avoidance patterns can be generalized as discovering a region (e.g. polygon, road segment, etc.) which lacks movement when it is expected.

Figure 1 shows an illustration of an avoidance. Suppose the blue line represents a shortest path between green-tagged start and end locations; and the red line represents the actual path taken by a driver. Depending on

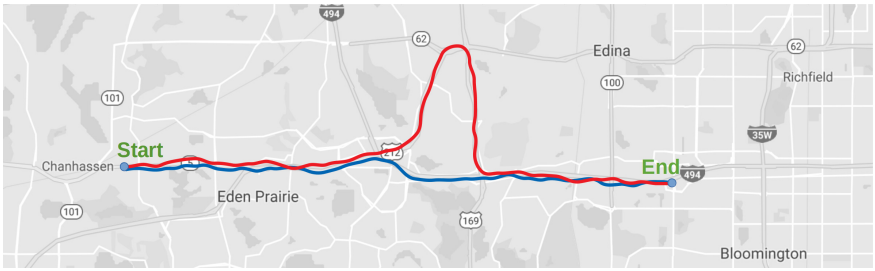


Figure 1: An illustration of an avoidance on a road network.

the definition of an avoidance, the output can be the region(s) enclosed by these two paths or a segment of the shortest (or expected) path that is different than the taken path.

In the following sections, we will provide some example application domains of avoidance pattern discovery as well as how these may help improve domain officials' work.

1.1 Avoidance Patterns in City Planning and Sociology

"Safety" or "Feeling Safe" is an often overlooked aspect that affects movement and activity patterns.

Feeling unsafe may affect the socialisation and activity patterns of individuals [8]. Thus, sociologists and city officials often investigate the neighborhoods by their demographic structures and take the necessary measures to mitigate the severity of feeling unsafe and preventing economic loss caused by the stigmatization. For example, some larger cities have distressed neighborhoods that are known to be riskier. These neighborhoods are not always spoken publicly but locals often know and avoid them. Such neighborhoods may not have been dangerous or risky before, but these characteristics may emerge gradually. Since, it is harder to do surveys/sociological analysis frequently, they may be undetected until it is too late. However, certain occupations are more sensitive to these changes and this fact can be leveraged instead of relying on the relatively sparse collection of surveys. One such occupation is taxi driving, where taxi drivers learn these neighborhoods through the experiences of each other and avoid entering them. For example, in the Chinese city of Kunming, taxi drivers try to not take customers from the regions where marginalized people are thought to be living [17]. Similarly, as shown in Figure 2, crowdsourced mobile applications such as Waze [1], allow users to flag some regions as dangerous, and let their users plan their routes accordingly.

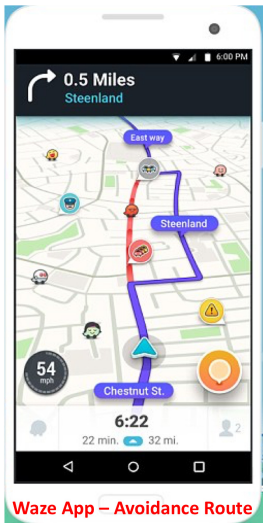


Figure 2: Screenshot of Waze app [1]

Nowadays, taxis are equipped with GPS devices due to their cheap availability and legal reasons to prevent conflicts with customers. One may leverage the GPS trajectory data collected from those devices [9] to identify the regions where taxi drivers avoid. In addition, since these datasets are collected in real time, the emergence of avoidance regions will be noticed much quicker than traditional surveys. Thus, the risk of a neighborhood being stigmatized can be prevented before the word of "unsafe" is widespread to all residents. City officials may mitigate the negative public opinion by updating their policies and planning more investments.

1.2 Avoidance Patterns in Law Enforcement

In the previous section, we provided a wide-lens perspective on the use cases of avoidance patterns for law-abiding citizens. However, sometimes it is not enough to solve the neighborhoods' sociological problems with-

out getting into the criminal mind. Thus, in this section, we are going to introduce the avoidance patterns from a criminal mind's perspective.

A fundamental task in criminology is the analysis of crime locations and locating the criminal to prevent more crimes [4]. In the past, these analysis were done manually using paper maps and pins, but nowadays modern law enforcement agencies around the globe use specialized tools and spatial analysis techniques to automate this process as well as improve their accuracy and efficiency (e.g. CrimeStat [15]).

There is one particular problem with these tools: To successfully use them, multiple crimes should occur and this will cause a delay in the detection of a criminal behavior as well as cause more harm to society. In addition, even if a criminals' location is estimated by these tools, these will neither pinpoint the exact location of the criminal nor find their mobility behavior causing an extensive search by the security officials in field.

To overcome these issues, the mobility behavior of vehicle trajectories can be analyzed. Criminals often avoid some locations where there are security cameras or law enforcement checkpoints since these may lead to their arrest or identification [14]. Thus, given a set of trajectories and security checkpoints/cameras, suspicious behavior of a trajectory can be identified by its difference from the expected path.

Another preventive measure may be to identify loops in trajectories. These will also cause specific regions to be flagged as avoidance regions. However, the pattern may not have an intention of avoidance, but a result of surveillance. For example, criminals may do a surveillance around a target crime site (e.g. a bank for robbery). Such looping/circling behavior by an individual may be for surveillance. Thus, identifying such trajectories and the individuals who created them may help public security officials prevent crimes before they occur.

1.3 Avoidance Patterns in Transportation Planning

Transportation planners often deal with multiple data sources including cameras, road sensors, loop sensors, accident data, etc. to understand the flow of traffic throughout the day. These datasets are then used to improve design and synchronization of traffic lights, fix the flawed road segments, and plan new roads or increase the capacity of existing ones. One particular need of transportation planners is to understand the driver behavior under different road conditions [2].

For example, long term residents of cities know where and when the traffic congestion happens and avoid these locations even though this may result with a longer route to destination. Similarly, when there are structural problems (e.g. potholes, cracks, etc.) or there are construction zones on the road, locals know and avoid these areas. For example, the magnitude of potholes in Figure 3 may cause drivers' to use another road instead of a shorter one. Using the GPS trajectory data collected from location based applications (e.g. Google Maps, Apple Maps, Waze, etc.), transportation planners may better understand the driver behavior and the underlying causes.

Figure 3: Potholes that may cause drivers' avoidance behavior [19]

2 Related Work

There have been several attempts to use mobility datasets (i.e. trajectory datasets) to identify interesting patterns that can be further analyzed by domain scientists. Mobility datasets (i.e. GPS trajectories) are large sets of points with ordered timestamps (Figure 5(a)). Due to the imperfection of GPS devices as well as minor variability caused by driver behavior (e.g. lane changes, speed differences), it is hard to do analysis on these datasets without any pre-processing.

To overcome such difficulties as well as reduce the computational cost, some studies discretize the space into grids, and use grid cells to represent trajectories. For example, several works analyze anomalous trajectories to identify the taxi drivers who were using longer paths for their customers to increase the bill [24, 5]. This is done by representing the trajectories as grid cells and comparing these sets of cells with the set of cells that were representing the appropriate route for a source and destination pair. Grid based mobility analysis is used for other patterns as well. Some example work includes identifying the outlying trajectories by using their grid cell representations, and clustering these to understand the movement behaviors as well as the transportation modes [11, 27]. However, the output of these studies are sensitive to the selected grid cell sizes. Selecting a large grid cell size causes large areas to be flagged as outliers or miss them entirely. Also, selecting a too small cell size may make the comparison impossible. In addition, since the graph notion of the road network is not taken into account, the outputs can be unrealistic especially for vehicle trajectories.

Hence, there are other works that use the GPS points of trajectories, instead of their grid cell representations, to classify trajectories by their transportation modes (e.g. walking, cycling, driving, etc.) [28, 26, 21, 25], to infer the Points of Interests (POI), and to understand the public transportation behavior, i.e. finding preferred paths instead of shortest paths [6, 12, 23, 18, 22, 10].

However, the aforementioned approaches lack three important considerations. First, these works consider GPS trajectories as either a set of points or cells instead of a single trajectory entity. Second, they do not account for the underlying road network structure. However, most human mobility patterns are dependent on the roads and those roads affect mobility behavior. Third, they focus on the presence of mobility but sometimes the lack of movement, when it is expected, may be more interesting.

3 Avoidance Pattern Discovery

Avoidance patterns may be observed in different applications domains. Some avoidance patterns such as aircrafts avoiding extreme weather events, or a predator species avoiding another's territory [20] may occur in Euclidean space but most human activities on land occur on road networks. In addition, the minor perturbations as well as the driving behaviors (e.g. frequently changing to left, middle or right lanes) can be compensated by the help of the road network. Given a GPS trajectory with $tr = [p_1 \rightarrow p_2 ... \rightarrow p_n]$ where each point $p_i = (x_i, y_i, t_i) \in tr$ and a spatial network graph $G = (V, E)$ where each road intersection is represented by vertices ($v \in V$) and each street segment is represented by edges ($e \in E$), it is possible to match the trajectories on the road network to represent them by a collection of nodes and edges [16, 13] as shown in Figure 5(b).

Once trajectories are map-matched, the idea behind the Avoidance Region Discovery [7] is to compare them with a path which should be used by the normal drivers. Normal behavior can be defined as a shortest path between the source and destination of the trajectory. Thus, once a trajectory tr_i and a shortest path sp_i are compared, the edges and the nodes are used to create a set of polygons which represent an avoidance polygon set for that trajectory. Figure 5(d) shows an example avoidance polygon in red. When the road network graph is bigger and the trajectory-shortest path differences are in multiple locations, these polygons will create an avoidance polygon set for that pair.

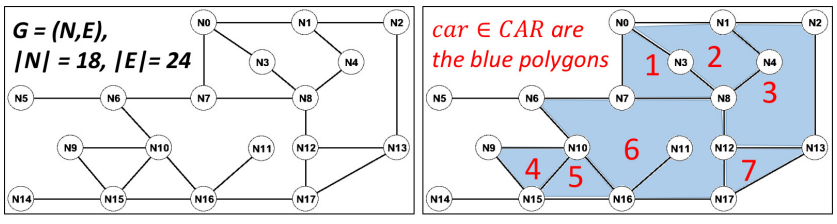
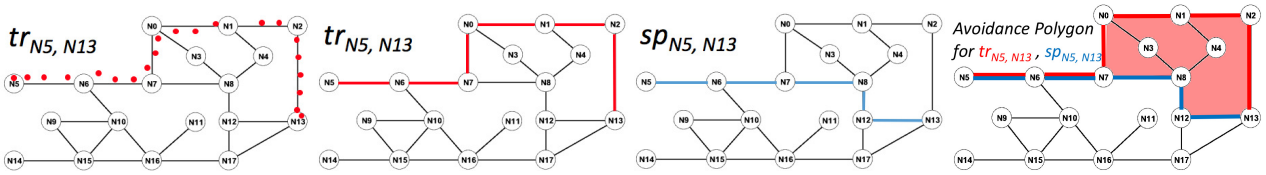


Figure 4: Illustration candidate avoidance regions (graph faces). Figure is excerpted from [7].

Avoidance polygons for each trajectory may be in different locations throughout the space (i.e. each trajectory may have different source and destination as well as shortest path). However, to evaluate the regions which are avoided by more than one trajectory, one will need the provide a consistent set of candidate avoidance regions. To overcome this issue, the space can be discretized to smaller polygons which are represented by the faces of a graph. Using Euler's theorem for planar graphs, the number of candidate avoidance patterns (CAR) on road network will be $|CAR| = |E| - |N| + 2$.

Using the count of the avoidance (denoted as c) for a region may be misleading. For example, close to a city center, there may be many candidate avoidance patterns that are avoided due to the higher number of trajectories intersecting them. However, in a rural area this number will plummet because of the fact that the number of trajectories in that location will be lower. Therefore, for each candidate avoidance region, the expectation of non-avoidance count should be known as well. To do this, the number of shortest paths that intersect avoidance polygons are counted (denoted as nc) and these counts are propagated to the candidate avoidance regions that are covered by those avoidance polygons.



(a) Trajectory Points (b) Map-matched Trajectory (c) Shortest Path (d) Avoidance Polygon

Figure 5: An example trajectory (5(a)), its map-matched edge representation(5(b)), corresponding shortest path(5(c)) and this pair's avoidance polygon(5(d)) (excerpted from [7]).

Finally, by defining a metric, i.e. interestingness ratio ($I = \left(\frac{c}{c+nc}\right) \times c$), which takes both avoidance and non-avoidance counts into account, is used to compute an interestingness score for each of the candidate avoidance polygons. The ones which exceed a specified threshold (λ) on this metric are flagged as interesting avoidance patterns.

This overall process is a challenging task due to the large number of candidate avoidance regions (CAR) which is related to the size of a road network graph (e.g. 10^6 edges in a road graph) and the large number of trajectories that can be collected from GPS devices (e.g. 10^6 trajectories per year for a large city's taxis). To overcome this challenge, [7] proposed an avoidance region miner algorithm that creates a road network subgraph that includes the road segments which were used by the trajectories instead of the whole road network graph. In addition, the authors proposed a pruning algorithm that eliminates the computation of the metric when it is proved to not exceed the specified threshold. Nevertheless, the proposed algorithm's pruning methods do no reduce the worst case complexity.

4 Discussion and Future Directions

The example work on Avoidance Region Discovery in the context of public safety is a starting point but there are still opportunities for improvement.

Identifying and Comparing with the Non-Shortest Paths: Although shortest paths are a logical choice for most of the drivers, sometimes non-shortest paths are more convenient due to the travel times, speed limits, road conditions (many turns vs. straight driving), rush hour, etc. Therefore, one may argue that the shortest path assumption for a comparison with trajectories may not be valid. In those cases, the preferred paths can be used.

Minor vs. Major Deviations of Trajectories: The current state of the art doesn't distinguish between minor and major deviation between shortest paths and trajectories. However, this may be particularly important depending on the use case. For example, the minor deviations may be used in the public safety domain since

in case of avoiding security cameras/checkpoints the deviations may be minor but when there is a rush hour congestion the deviation may be greater (e.g. avoiding the city center at rush hour).

Privacy Concerns: One of the key things needed in the context of urban mobility analytics is datasets. Due to the privacy concerns, this is not usually possible unless Volunteered Geographic Information (VGI) sources are used. However, since the people with suspicious behavior may not be willing to share their locations, it is hard to collect these. In addition, tagging each trajectory with the driving individual's ID may violate privacy rights because these will point to the source and destination, consequently the locations where people live. Thus, some of the capabilities of the avoidance pattern discovery may be limited such as distinguishing between the population and individual avoidance behaviors.

Statistical Significance: In [7], the avoidance and non-avoidance counts were used but the interestingness ratio metric does not provide a statistical significance value for the output. Thus, spurious/chance patterns may exist in the output. One approach may be to understand the distribution of trajectories over the study area and using this distribution in to provide meaninful significance values for the output.

Emerging Avoidance Regions: One of the most interesting applications of avoidance region discovery is the detection of emerging such regions. For example, a structural damage to a road segment may occur over time by the traffic and/or weather conditions. Thus, the temporal information related to the trajectory datasets can be used to find some long term (e.g. structural damage) and short term (e.g. accidents causing congestion) emerging avoidance regions.

Acknowledgments

This work is self supported and is not affiliated with any institution. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the affiliations of the authors. Author(s) would like to thank to the Data Science team at Cargill as well as Spatial Computing Research Group at the University of Minnesota for their constructive feedbacks.

References

[1] W. Application. Waze map safety re-route, 2017. http://www.cbsnews.com/waze-map-app-suicide-straight-safety-reroute-high-crime-neighborhoods/.

[2] R. Arnott and K. Small. The economics of traffic congestion. *American scientist*, pages 446–455, 1994.

[3] S. Bouton, S. M. Knupfer, I. Mihov, and S. Swartz. Urban mobility at a tipping point. *McKinsey and Company*, 2015.

[4] P. J. Brantingham, P. L. Brantingham, et al. *Environmental criminology*. Sage Publications Beverly Hills, CA, 1981.

[5] C. Chen et al. Real-time detection of anomalous taxi trajectories from gps traces. In *Int. Conf. on Mobile and Ubiquitous Systems*, pages 63–74. Springer, 2011.

[6] C. Cheng et al. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*, volume 13, pages 2605–2611, 2013.

[7] E. Eftelioglu, X. Tang, and S. Shekhar. Avoidance region discovery: A summary of results. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pages 585–593. Society for Industrial and Applied Mathematics, 2018.

[8] K. F. Ferraro. *Fear of crime: Interpreting victimization risk*. SUNY press, 1995.

[9] N. Ferreira, J. Poco, H. T. Vo, J. Freire, and C. T. Silva. Visual exploration of big spatio-temporal urban

data: A study of new york city taxi trips. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2149–2158, 2013.

[10] Y. Fu et al. Real estate ranking via mixed land-use latent models. In *In Proc. of 21th SIGKDD*, pages 299–308. ACM, 2015.

[11] Y. Ge et al. Top-eye: Top-k evolving trajectory outlier detection. In *Proc. of 19th ACM Int. Conf. of CIKM*, pages 1733–1736. ACM, 2010.

[12] F. Giannotti et al. Trajectory pattern mining. In *In Proc. of 13th SIGKDD*, pages 330–339. ACM, 2007.

[13] P. Karich and S. Schröder. Graphhopper. *http://www. graphhopper. com*, 2014.

[14] S. Leman-Langlois. The local impact of police videosurveillance on the social construction of security. *Technocrime: Technology, Crime and Social Control*, pages 27–45, 2008.

[15] N. Levine et al. Crimestat iii: a spatial statistics program for the analysis of crime incident locations (version 3.0). *Houston (TX): Ned Levine & Associates/Washington, DC: National Institute of Justice*, 2004.

[16] P. Newson and J. Krumm. Hidden markov map matching through noise and sparseness. In *Proc. of the 17th SIGSPATIAL Conf.*, pages 336–343. ACM, 2009.

[17] B. E. Notar. 10 off limits and out of bounds taxi driver perceptions of dangerous people and places in kunming, china. *Rethinking Global Urbanism*, pages 190–207, 2012.

[18] P. Wang et al. Human mobility synchronization and trip purpose detection with mixture of hawkes processes. In *In Proc. of 23th SIGKDD*, pages 495–503. ACM, 2017.

[19] Wikimedia commons, Max Ronnersj. Pothole — Wikipedia, the free encyclopedia. [Online; accessed 10-23-2018].

[20] H. Wolf. Odometry and insect navigation. *Journal of Experimental Biology*, 214(10):1629–1641, 2011.

[21] C. Xu et al. Identifying travel mode from gps trajectories through fuzzy pattern recognition. In *Int. Conf. on Fuzzy Systems and Knowledge Discovery*, volume 2, pages 889–893. IEEE, 2010.

[22] J. Yuan et al. Discovering regions of different functions in a city using human mobility and pois. In *In Proc. of 18th SIGKDD*, pages 186–194. ACM, 2012.

[23] Q. Yuan et al. Time-aware point-of-interest recommendation. In *Proc. of 36th Int. SIGIR Conf. on Research and Development in Information Retrieval*, pages 363–372. ACM, 2013.

[24] D. Zhang et al. ibat: detecting anomalous taxi trajectories from gps traces. In *Proc. of 13th Int. Conf. on Ubiquitous Comp.*, pages 99–108. ACM, 2011.

[25] L. Zhang, S. Dalyot, D. Eggert, and M. Sester. Multi-stage approach to travel-mode segmentation and classification of gps traces. *Int. Arch. of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(W25):87–93, 2011.

[26] Y. Zheng et al. Understanding mobility based on gps data. In *Proc. of 10th Int. Conf. on Ubiquitous Comp.*, pages 312–321. ACM, 2008.

[27] Y. Zheng et al. Mining interesting locations and travel sequences from gps trajectories. In *Proc. of 18th Int. Conf. on WWW*, pages 791–800. ACM, 2009.

[28] Y. Zheng et al. Understanding transportation modes based on gps data for web applications. *Transactions on the Web (TWEB)*, 4(1):1, 2010.

Indoor location-based services: Challenges and Opportunities

Muhammad Aamir Cheema
Faculty of Information Technology, Monash University, Australia
aamir.cheema@monash.edu

Abstract

Billions of smartphone users throughout the world have come to expect, and rely upon, intuitive, reliable and accurate maps, directions, turn-by-turn navigation and other location-based services (LBSs). Those same users will over the next few years come to expect and then demand the same experience and services when they enter any large building or facility in the world whether that be a hospital, airport, shopping mall or university campus. Based on various reports and surveys, it is reported that indoor LBSs are expected to have an even bigger impact than outdoor LBSs mainly because indoor is where we spend our time and money, meet friends, and where business happens. Indoor LBSs have numerous applications including navigation, location based social networking, emergency services, location-based marketing, mobile games, asset tracking, and workforce location. In this paper, we describe the challenges that need to be solved in order to make indoor LBSs as ubiquitous as their outdoor counterpart and discuss the opportunity this provides.

1 Introduction

Location-based services (LBSs) are the services that take into account geographical locations of users and other entities. Some applications of LBSs include car navigation systems, emergency services, travel planning, asset management, location-based recommendation, and geosocial networking. LBSs have become ubiquitous because of the surge in adoption of smartphones and the availability of cheap wireless networks. The Australian Communication and Media Authority (ACMA) reported [1] that 72% of Australians accessing the internet via their mobile phone use a LBS at least once a week. ACMA concluded that: *"Location services exhibit their potential in countless situations, which generally fall within the government, business, and consumer domains. The uses encompass emergency management and government applications, business solutions and consumer applications'*.

Although we spend more than 85% of our time indoors (30% at indoor venues other than homes, often in unfamiliar places [29]), nearly all of the existing LBSs focus on outdoor space and ignore indoor space altogether. The indoor LBSs promise huge potential for research organisations, government agencies, technology giants, and enterprising start-ups – to adapt to the indoor applications such emergency services, assisted healthcare systems, indoor asset tracking, and event planning. For example, indoor LBSs can be used to help visually impaired people navigate indoor venues, directing people to safe exits during emergency evacuations, tracking staff, patients and equipment in hospitals [2] and providing location-based shopping assistance for customers.

Realising the potential of indoor LBSs, major technology companies, government and research organizations and start-ups are investing heavily in indoor technology. For example, the US Federal Communications Commission is exploring indoor positioning for more timely and effective emergency services [3]. In October

2014, Apple allowed [4] businesses to use its indoor location capabilities – but the service was soon completely overwhelmed by pent-up demand, forcing Apple to limit it to venues with over 1 million visitors a year. Based on such reports of its immense popularity, Forbes reported that indoor venues are the next frontier for LBSs [5] and indoor LBSs are expected to have an even bigger impact than their outdoor counterpart [14].

Despite the huge need of indoor LBSs, they are still not widely available due to some major challenges that hinder its ubiquitous availability. In the next section, we present the details of two such major challenges. Then, in Section 3, we present some important research directions that address the two challenges and pave the way for ubiquitous indoor LBSs. We remark that this paper does not aim to provide a *comprehensive* list of important research directions in this area. Although some of these research directions have already received some research attention [35, 39, 31, 27], we believe that these research areas demand significant more work from the research community. The goal of this paper is to highlight the importance of indoor LBSs and to provide *some* of the important research directions, thus encouraging research in these areas.

2 Challenges

In this paper, we discuss two major challenges that need to be addressed before the indoor LBSs become as ubiquitous as outdoor LBSs: 1) there does not exist any ubiquitous indoor positioning technology to identify a user's location; 2) in order to provide indoor location-based services at a global-scale, efficient and effective data management and analytics techniques are required to handle indoor venues and indoor data. Below, we provide the details.

2.1 Ubiquitous Indoor Positioning System (IPS)

Global positioning system (GPS) is a ubiquitous technology that identifies the location of a user carrying a GPS-enabled device such as a smartphone. Unfortunately, GPS does not work in indoors and we are still far from a ubiquitous technology for indoor environments. Some indoor positioning technologies require installation of special hardware (e.g., RFID readers, bluetooth beacons) in the indoor venue that makes them infeasible for global deployment. WiFi based positioning technologies [18, 30] provide a better option due to the ubiquitous availability of WiFi in indoor venues. However, most existing technologies require *fingerprinting* – manually mapping signal strengths at different indoor locations – which is time consuming and labor intensive. This is a major hurdle in the worldwide deployment of such a technology also because fingerprinting is sensitive to indoor environments and becomes invalid with time due to the changes in indoor environment. Some systems have been proposed to reduce the fingerprinting overhead of WiFi-based localization systems. However, these systems depend on installing special hardware to monitor changes in the signal strength [30], crowd-sourcing [34] which requires active feedback from users, and/or theoretical modelling tools that rely on detailed information of the indoor venues such as material of walls and doors etc. [22, 28]. Due to the manual efforts involved, none of these approaches is suitable for a ubiquitous indoor positioning system that can locate indoor users in *any* WiFi-enabled building with minimum overhead.

2.2 Indoor Data Management and Analytics

Another largely unmet challenge is how to effectively manage and analyze indoor location data. Current indoor indexing and query processing technology is in its infancy and falls short in managing different types of indoor data critical for a variety of location-based services. Some limitations of the existing techniques are: 1) rich textual information associated with indoor locations is not utilized; 2) uncertainty in the data is not adequately handled, leading to poor or incorrect results; 3) indoor trajectory data, which can be very useful in providing insights, have not been exploited; and 4) outdoor space is not integrated with indoor space, ruling out a large class of applications that involve both outdoor and indoor space.

Outdoor techniques [15] cannot address the above limitations due to the specific characteristics of indoor settings. For example, we need to not only represent the spaces (airport, hospital) in proper data model but also manage all the indoor features (lifts, escalators, stairs) and locations of interest (boarding gates, exit doors, counters) such that search can be conducted efficiently. Indoor spaces are characterized by indoor entities such as walls, doors, rooms, hallways, etc. Such entities constrain as well as enable indoor movements, resulting in unique indoor topologies. Therefore, outdoor techniques cannot be directly applied to indoor venues. One possible approach for indoor data management is to first model the indoor space to a graph using existing indoor data modelling techniques [26] and then applying existing graph algorithms to process queries on the indoor graph. However, this approach is inefficient because the techniques fail to exploit the properties specific to indoor space. For example, it was recently shown [36] that the state-of-the-art outdoor algorithm [41] takes over one second to answer a single shortest distance query between two indoor points in the Clayton campus of Monash University. A world-scale indoor service provider using the outdoor techniques would have a low throughput and would be unable to meet the high query workload, e.g., Google Maps is adding indoor venues and may provide spatial queries involving indoor spaces in the near future. The query workload is expected to be quite high and the existing techniques would not be able to meet the demand. In contrast, the techniques that exploit the properties specific to indoor space [36] can answer a shortest distance query in around 0.01 milliseconds on the same dataset, a 10^5 times improvement. To support a large number of indoor queries in real time, there is a need to develop techniques for indoor location data that address the limitations mentioned above and carefully exploit the properties specific to indoor venues to provide efficient query processing capabilities.

3 Research Directions and Opportunities

Indoor LBSs exhibit their potential in countless situations, which generally fall within the government, business and consumer domains. The uses encompass emergency management and government applications, business solutions and consumer applications. Below, we briefly describe some representative applications of indoor LBSs in each of the three domains:

- *Government.* Indoor LBSs are critical in areas such as public safety, emergency services, and healthcare. The U.S. Federal Communications Commission has a strong interest in improving emergency services using indoor positioning technology [3]. LBSs are also used in hospitals for indoor navigation, tracking staff and patients, location-based messaging, asset management, location analytics, and in integrating with other clinical systems. The global LBS market in the healthcare sector was predicted [2] to grow at a compound annual growth rate (CAGR) of 31.23% from 2015 to 2019.
- *Individuals.* Indoor LBSs have many applications for individuals such as navigation, in-store guidance, guided tours, and location-based social networking. For example, Google reported [6] that 84% of the smartphone shoppers use their mobile to help shop while in-store and 1 in 3 shoppers use their smartphones to find information instead of asking store employees. Indoor LBSs will also benefit visually impaired people and autonomous machines such as robots. Analyst firm ABI research estimated that, by 2018, over 800 million mobile devices will be using indoor LBSs [7].
- *Businesses.* Commercial applications of Indoor LBSs include location-based marketing, asset management, and workforce allocation. Indoor location- and place-based marketing is expected to surpass 10 billion dollars by 2018 [8]. Also, Forbes stated that the location-based services are a bonanza for start-ups due to their immense popularity and low entry barrier [9].

Realising the potential of indoor location-based services, major companies and research organizations have started investing heavily in this area. For example, Google offers more than 10,000 indoor maps of U.S. and international facilities in Google Maps [10], Microsoft and Nokia have partnered to provide indoor services on more than 3,000 facilities including U.S. airports and convention centers, and Apple acquired WiFiSLAM, a company providing indoor positioning services [11]. Huge demand of indoor LBSs and increasing availability

of indoor maps have created a huge opportunity for research and development in indoor LBSs. In this section, we briefly describe several important and promising research directions and opportunities.

3.1 Developing a Ubiquitous Indoor Positioning System

There is a large body of work on indoor positioning systems (e.g., see [17, 32, 38]). However, almost all existing techniques either require installation of special hardware or require extensive manual calibration that makes them infeasible for global deployment. Although the accuracy of these positioning systems has improved a lot in the past few years, such indoor positioning systems are still far from being as ubiquitous as GPS is for outdoor spaces. This is a major hindrance in the deployment of indoor LBSs at a global scale. Thus, there is a need to develop an indoor positioning system that relies on the ubiquitous availability of existing equipment (e.g., WiFi access points or light sources) and does not rely on manual calibration (e.g., fingerprinting). Some recent research [23] have started working towards addressing this need. However, such efforts must be continued and more work is needed before the vision of global deployment of such systems is realized.

3.2 Indexing and Querying Textual Indoor Location Data

In the present Web 2.0 era, spatial data are increasingly annotated, whether manually or algorithmically. This results in a rich body of information associated with objects. For instance, products in a supermarket may be tagged with price, ingredients, nutritional information and use-by date. Similarly, medical instruments in a hospital are tagged with textual information such as name, category and department.

Despite the popularity of keyword search, the current indoor query processing systems only deal with the spatial dimension of the data and *cannot* support keyword search on spatial data (called *spatial keyword search*). In a spatial keyword query, the objects are returned not only based on their distances from the query location but also based on their keyword similarity to the query keywords. A user may issue a query with the keyword string "low fat milk" to find nearby shops that sell low fat milk. Or a library user may want to navigate to the location of a book, and use its title as keywords. Existing systems that answer spatial keyword queries in outdoor space rely on specialized indexes [19] (e.g., IR-tree, KR*-tree, S2I etc.) that are only applicable for outdoor venues. They do not extend efficiently to ontologies that are typical of indoor domains.

There is a need to develop efficient indexing and query processing techniques for spatial keyword queries that allow to search for indoor objects based not only on their distances from the query location, but also on how well they match query terms. For example, a user may issue a query to find the nearest defibrillator in an emergency situation. Queries may be ambiguous (when several objects match a query), inaccurate (when there are no objects that match all the requested attributes, e.g., "a *cheap* food place *nearby*"); and if spoken, some words may be mis-heard by an Automatic Speech Recognizer. In addition, the data may be dynamic, e.g., locations or terms associated with objects may change, medical instruments may be moved, the user may be walking, or the price of a product may change.

3.3 Handling Uncertainty in Indoor Location Data

Real-world data are noisy, and location-based data are even more so [16]. Reasons include built-in inaccuracy of the positioning technology (for GPS, IPS, etc.), transmission delay, and deliberately added noise to protect privacy [21]. This is worsened by the fact that data are increasingly user-created, or automatically annotated by spatial data-mining algorithms [37]. More and more queries are sent from mobile devices with misspelt or otherwise defective keywords. There could be serious consequences of ignoring such uncertainties in data. Notoriously, there are news reports on how errors in Google Maps have led to unwanted traffic, wrong destinations or itineraries [12], and even international conflicts [13].

Location inaccuracy in indoor space is even more of a concern. A minor discrepancy in reported location may render results worse than useless. A location error of just a metre or two in Euclidean distance may indicate a different room, or even the wrong floor, and a wildly incorrect estimate of indoor walking distance that could be catastrophic in an emergency.

A fundamental challenge is to model the uncertainties for different types of data, and to design efficient techniques for answering probabilistic queries regarding uncertain indoor data, such as probabilistic k nearest neighbors and probabilistic range queries. In general, uncertainty significantly increases the complexity of query processing, e.g.,the complexity of evaluating conjunctive queries over uncertain data is #P-complete [20]. When uncertainty is considered together with the characteristics of indoor settings, the queries are even harder to process.

3.4 Indoor Trajectory Management and Analytics

Just as a user's web browsing history (e.g.,clickstream) in an online world provides insights about the user, a user's trajectory gives insights about him/her in the physical world [33]. For example, the trajectories of indoor users may be used to learn how people flow through an indoor venue. These insights may be valuable for users, government agencies and venue owners, and scenarios such as optimizing the layout of a venue, planning emergency evacuations, flow analysis, and congestion prediction. Due to the different topology (indoor vs outdoor space), different positioning systems used (GPS vs IPS) and different user behaviours (driving vs walking), indoor trajectories have different characteristics from outdoor trajectories [27]. Thus, there is a need to develop new indexing, retrieval and analytics techniques to exploit the potential of the indoor trajectories.

3.5 Integrating Outdoor and Indoor Space

Almost all existing query processing techniques are designed either for outdoor space or for indoor space. However, a lot of real-world applications encompass both − for example, navigation from a multi-level car park to an office on a university campus. Hence, it is important to seamlessly integrate outdoor and indoor space (OI-space, together) and propose a unified indexing scheme to support a wide range of applications in OI-space that are not supported by the current systems. This is non-trivial mainly due to the inherent differences between outdoor and indoor space.

Concern for integrating indoor and outdoor space (OI-space) has prompted research in the past few years. This includes seamless positioning handover between indoor and outdoor [25], data models for OI-space [24], and ontologies for OI-space [40]. However, there is no work on a unified index to allow efficient processing of spatial queries in OI-space. Thus, there is a need to effectively and seamlessly integrate outdoor and indoor space in a unifying index, to support efficient and scalable processing of queries in OI-space. Given inherent differences between the ontologies appropriate to indoor and outdoor space, the techniques and indexing schemes designed for one do not work well for the other.

4 Conclusions

We spend a large part of our lives in indoor environment. However, almost all existing location-based services (LBSs) focus on outdoor spaces. To meet the growing demand and popularity of indoor LBSs, several challenges must be addressed that hinder the ubiquitous availability of indoor LBSs. In this paper, we first present an overview of two major challenges and then provide some important and promising research directions that will support and enhance a wide range of indoor applications, such as emergency services, assisted healthcare systems, indoor asset tracking and event planning, thereby improving the stakeholders' experience.

Acknowledgments

The author is supported by Australian Research Council Future Fellowship FT180100140 and Discovery Project DP180103411.

References

[1] *http://acma.gov.au/theACMA/Library/researchacma/Research-reports/here-there-and-everywhere-consumer-behaviour-and-location-services*.

[2] *http://www.researchandmarkets.com/research/lvgkh8/global_lbs_market*.

[3] *https://apps.fcc.gov/edocs_public/attachmatch/FCC-14-13A1.pdf*.

[4] *gpsbusinessnews.com/Apple-Has-Difficulties-to-Keep-Up-with-Indoor-Map-Interest-from-Venue-Owners_a5128.html*.

[5] *www.forbes.com/sites/forrester/2013/01/23/indoor-venues-are-the-next-frontier-for-location-based-services*.

[6] *http://www.marcresearch.com/pdf/Mobile_InStore_Research_Study.pdf*.

[7] *https://www.abiresearch.com/press/over-800-million-smartphones-using-indoor-location*.

[8] *http://opusresearch.net/wordpress/pdfreports/OpusIndoorReport_Jan2014_Leadup.pdf*.

[9] *http://www.forbes.com/sites/martinzwilling/2011/01/31/location-based-services-are-a-bonanza-for-startups*.

[10] *https://www.google.com/maps/about/partners/indoormaps/*.

[11] *https://www.wired.com/insights/2013/06/the-next-frontier-of-navigation-in-location-positioning/*.

[12] *consumerist.com/2011/08/google-maps-no-longer-confuses-womans-driveway-with-state-park-entrance.html*.

[13] *http://www.foxnews.com/tech/2010/11/08/oops-google-sparks-invasion/*.

[14] Silicon valley VCs predict 2013 trends: Space, robots, self-driving cars. *http://venturebeat.com/2012/12/31/trends/*.

[15] T. Abeywickrama, M. A. Cheema, and D. Taniar. k-nearest neighbors on road networks: A journey in experimentation and in-memory implementation. *PVLDB*, 9(6):492–503, 2016.

[16] C. C. Aggarwal. Managing and mining uncertain data. *Springer*, 2009.

[17] A. Y. Al-Dubai, Y. Nasser, M. Awad, R. Liu, C. Yuen, R. Raulefs, and E. Aboutanios. Recent advances in indoor localization: A survey on theoretical approaches and applications. *IEEE Communications Surveys and Tutorials*, 19(2):1327–1346, 2017.

[18] P. Bahl and V. N. Padmanabhan. RADAR: an in-building RF-based user location and tracking system. In *IEEE INFOCOM*, 2000.

[19] L. Chen, G. Cong, C. S. Jensen, and D. Wu. Spatial keyword query processing: An experimental evaluation. *PVLDB*, 2013.

[20] N. N. Dalvi and D. Suciu. Management of probabilistic data: foundations and challenges. In *PODS*, 2007.

[21] M. L. Damiani, C. Silvestri, and E. Bertino. Fine-grained cloaking of sensitive positions in location-sharing applications. *Pervasive Computing*, 2011.

[22] K. El-Kafrawy, M. Youssef, A. El-Keyi, and A. F. Naguib. Propagation modeling for accurate indoor WLAN rss-based localization. In *Proceedings of the 72nd IEEE Vehicular Technology Conference, VTC Fall 2010, 6-9 September 2010, Ottawa, Canada*, pages 1–5, 2010.

[23] M. Elhamshary and M. Youssef. Towards ubiquitous indoor spatial awareness on a worldwide scale. *SIGSPATIAL Special*, 9(2):36–43, 2017.

[24] N. A. Giudice, L. Walton, and M. F. Worboys. The informatics of indoor and outdoor space: a research agenda. In *ISA*, pages 47–53, 2010.

[25] R. Hansen, R. Wind, C. S. Jensen, and B. Thomsen. Seamless indoor/outdoor positioning handover for location-based services in streamspin. In *MDM*, 2009.

[26] C. S. Jensen, H. Lu, and B. Yang. Graph model based indoor tracking. In *Mobile Data Management*, pages 122–131, 2009.

[27] C. S. Jensen, H. Lu, and B. Yang. Indexing the trajectories of moving objects in symbolic indoor space. In *SSTD*, 2009.

[28] Y. Ji, S. Biaz, S. Pandey, and P. Agrawal. ARIADNE: a dynamic indoor signal map construction and localization system. In *MobiSys*, 2006.

[29] N. E. Klepeis, W. C. Nelson, W. R. Ott, J. P. Robinson, A. M. Tsang, et al. The national human activity pattern survey (NHAPS): a resource for assessing exposure to environmental pollutants. *Journal of exposure analysis and environmental epidemiology*, 2001.

[30] P. Krishnan, A. S. Krishnakumar, W. Ju, C. L. Mallows, and S. Ganu. A system for LEASE: location estimation assisted by stationary emitters for indoor RF wireless networks. In *IEEE INFOCOM*, 2004.

[31] H. Lu, B. Yang, and C. S. Jensen. Spatio-temporal joins on symbolic indoor tracking data. In *ICDE*, 2011.

[32] J. Luo, L. Fan, and H. Li. Indoor positioning systems based on visible light communication: State of the art. *IEEE Communications Surveys and Tutorials*, 19(4):2871–2893, 2017.

[33] R. Nandakumar, S. Rallapalli, K. Chintalapudi, V. Padmanabhan, L. Qiu, A. Ganesan, S. Guha, D. Aggarwal, and A. Goenka. Physical analytics: A new frontier for (indoor) location research. Technical report, Microsoft Research Technical Report, October 2013.

[34] J. Park, B. Charrow, D. Curtis, J. Battat, E. Minkov, et al. Growing an organic indoor location system. In *MobiSys*, 2010.

[35] Z. Shao, M. A. Cheema, and D. Taniar. Trip planning queries in indoor venues. *The Computer Journal*, 61:1–18, 2017.

[36] Z. Shao, M. A. Cheema, D. Taniar, and H. Lu. VIP-tree: An effective index for indoor spatial queries. *PVLDB*, 10(4):325–336, 2016.

[37] V. Singh, S. Venkatesha, and A. K. Singh. Geo-clustering of images with missing geotags. In *GrC*, 2010.

[38] J. Xiao, Z. Zhou, Y. Yi, and L. M. Ni. A survey on wireless indoor localization from the device perspective. *ACM Comput. Surv.*, 49(2):25:1–25:31, 2016.

[39] B. Yang, H. Lu, and C. S. Jensen. Probabilistic threshold k nearest neighbor queries over moving objects in symbolic indoor space. In *EDBT*, 2010.

[40] L. Yang and M. F. Worboys. A navigation ontology for outdoor-indoor space: (work-in-progress). In *ISA*, 2011.

[41] R. Zhong, G. Li, K. Tan, L. Zhou, and Z. Gong. G-tree: An efficient and scalable index for spatial search on road networks. *TKDE*, 2015.

Dynamic Task Assignment in Spatial Crowdsourcing

Yongxin Tong[1], Zimu Zhou[2]
[1] BDBC and SKLSDE Lab, Beihang University, Beijing, China
[2] TIK, ETH Zurich, Zurich, Switzerland
[1]yxtong@buaa.edu.cn, [2] zzhou@tik.ee.ethz.ch

Abstract

Spatial crowdsourcing is a crowdsourcing paradigm featured with spatiotemporal information of tasks and workers. It has been widely adopted in mobile computing applications and urban services such as citizen sensing, P2P ride-sharing and Online-To-Offline services. One fundamental and unique issue in spatial crowdsourcing is dynamic task assignment (DTA), where tasks and workers appear dynamically and need to be assigned under spatiotemporal constraints. In this paper, we aim to provide a brief overview on the basics and frontiers of DTA research. We define the generic DTA problem and introduce the evaluation metrics to its solutions. Then we review mainstream solutions to the DTA problem. Finally we point out open questions and opportunities in DTA research.

1 Introduction

Crowdsourcing is a computing paradigm where humans actively participate in the procedure of computing, especially for the tasks that are intrinsically easier for humans than for computers. There has been active research on crowdsourcing [3, 11, 16, 22, 23] using web-based crowdsourcing platforms such as Amazon Mechanical Turks (AMT) and oDesk. The development of mobile Internet and sharing economy has triggered the shift from web-based crowdsourcing to spatial crowdsourcing (*a.k.a* mobile crowdsourcing) [4], where *(i)* each worker is considered as a mobile computing unit to complete tasks using their mobile devices [18] and *(ii)* spatial information such as location, mobility and the associated contexts plays a crucial role. Applications of spatial crowdsourcing have deeply penetrated into everyday life. Some of the most representative applications include real-time taxi-calling services (*e.g.*, Uber and DiDi), product placement checking services in supermarkets (*e.g.*, Gigwalk and TaskRabbit) on-wheel meal-ordering services (*e.g.*, GrubHub and Instacart), and citizen sensing services (*e.g.*, Waze and OpenStreetMap).

As with web-based crowdsourcing, a central issue in spatial crowdsourcing is task assignment, which aims to assign tasks to suitable workers such that the total weighted value of the assigned pairs of tasks and workers is maximized or the total moving distance of the workers is minimized [13, 26, 25, 28, 19, 27, 29, 30]. Different from task assignment in web-based crowdsourcing, the unique spatiotemporal dynamics in spatial crowdsourcing calls for new designs in task assignment theories and methods. Particularly, the tasks and works in spatial crowdsourcing may appear dynamically and task assignment needs to be performed immediately or in a short period, *a.k.a.* dynamic task assignment (DTA).

The DTA problem is challenging because *(i)* assignments are made under incomplete information; *(ii)* assignments usually cannot be revoked; and *(iii)* assignments need to be performed computationally efficient to meet the real-time requirements on large datasets. We formulate the generic DTA problem in Sec. 2 and review representative solutions to the DTA problem in Sec. 3. We finally point out open questions and opportunities for future research on DTA in Sec. 4.

2 Dynamic Task Assignment Problem

2.1 Problem Statement

For a spatial crowdsourcing platform ("platform" for short), the generic dynamic task assignment problem can be formulated based on the following definitions.

Definition 1 (Task): A task, denoted by $t =< \boldsymbol{l}_t, a_t, d_t, c_t >$, at the location $\boldsymbol{l}_t$ in the 2D space is posted on the platform at time a_t and is either allocated to a worker who arrives on the platform before the response deadline d_t or cannot be allocated thereafter. No more than c_t worker are required to perform the task.

Definition 2 (Worker): A worker, denoted by $w =< \boldsymbol{l}_w, a_w, d_w, c_w >$, arrives at the platform with an initial location $\boldsymbol{l}_w$ in the 2D space at time a_w and either performs a task which arrives at the platform before its response deadline d_w or does not conduct any task. Once a worker finishes a task, s/he can be viewed as a new worker if s/he is willing to be assigned other tasks. A worker is able to perform c_w tasks at most.

Definition 3 (Constraint Function): A constraint function $f_c(t, w)$ is used to indicate whether t can be assigned to w. Generally speaking, the constraint function is related to some spatiotemporal requirements, such as whether t is in the service range of w, or whether w can arrive at the position of t before its deadline.

Definition 4 (Utility Function): A utility function $f_u(t, w)$ is used to measure the utility of assigning t to w. It can be the payoff of the task or the payoff times the probability that t can be finished successfully.

Definition 5 (Distance Function): A moving cost function $f_d(t, w)$ is used to measure the cost of w if s/he moves to the location of t to perform the task. In practice the distance function can be the Euclidean distance or road network distance between t and w.

Definition 6 (DTA Problem): Assume a set of tasks T, a set of workers W, a constraint function $f_c(\cdot, \cdot)$, a utility function $f_u(\cdot, \cdot)$ and a distance function $f_d(\cdot, \cdot)$ on a spatial crowdsourcing platform. Suppose initially there is no task or worker on the platform. Workers and tasks then arrive dynamically at any time. The DTA problem is to find an assignment M among the tasks and the workers for different objectives, which can be either maximizing the total utility $U = \sum_{(t,w)\in M} f_u(t, w)$ or minimizing the total moving cost $C = \sum_{(t,w)\in M} f_d(t, w)$ of the assignment pairs, such that the following constraints are satisfied:

- Spatiotemporal constraint: $\forall (t, w) \in M, f_c(t, w) = 1$, which means that t can be assigned to w.
- Invariable constraint: once a task t is assigned to a worker w, the allocation of (t, w) cannot be changed.

As opposed to static task assignment, where the spatiotemporal information of all the workers and tasks is known, the DTA problem needs to make an effective assignment with partial information about the workers and tasks. We illustrate the DTA problem for maximizing the total utility using the following example.

Example 1: Suppose we have five tasks t_1-t_5 and three workers w_1-w_3 on a spatial crowdsourcing platform, whose initial locations are shown in a 2D space in Fig. 1. Each worker has a spatial restricted activity range, indicating that the worker can only conduct tasks that locate within the range, which is shown as a dotted circle in Fig. 1. Each user also has a capacity (*i.e.*, c_w), which is the maximum number of tasks that can be assigned to him/her. In this example, the capacity of each worker is 2 and the capacity of each task is 1. Fig. 2 presents the utility values for each pair of task and worker, which is marked on the edge between the worker and the task. In the static scenario, the total utility of the optimal task assignment is 10 (marked in red in Fig. 2). However, in the dynamic scenario, the offline solutions are not always applicable since both the workers and the tasks arrive dynamically at the platform. This is the main challenge for dynamic task assignment.

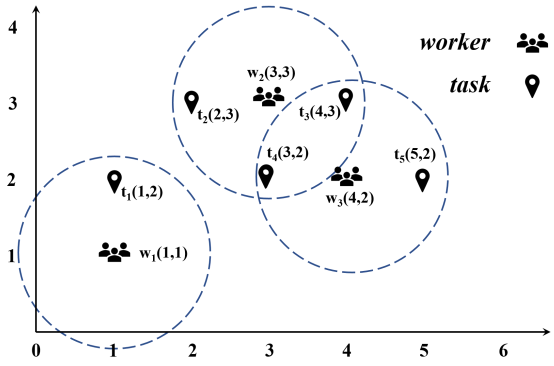


Figure 1: An instance of dynamic task assignment.

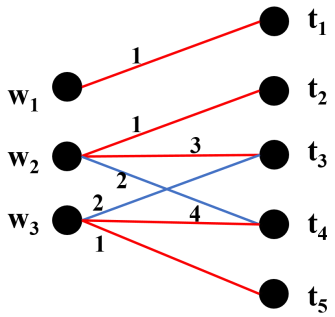


Figure 2: The utility between workers and tasks.

2.2 Evaluation Metric for DTA Algorithms

The solutions to the DTA problem are usually online algorithms [2].Different from traditional approximation algorithms for which approximation ratios are utilized to measure the approximation quality, for online algorithms, *competitive ratios (CR)* are used to evaluate their performance. In particular, the competitive ratio measures how good an online algorithm is compared with the optimal result of the offline model where all the information is provided. Based on different assumptions on the arrival order of the tasks and workers, typical online models include the adversarial model, random order model and i.i.d model. Take the goal of maximizing total utility as examples. The corresponding competitive ratios of the three types of online models are defined as follows.

Definition 7 (CR in the Adversarial Model [17]): The competitive ratio in the the adversarial model of a specific online algorithm for the DTA problem is the following minimum ratio between the result of the online algorithm and the optimal result over all possible arrival orders of the tasks and the workers,

$$CR_A = min_{\forall G(T,W,U) \text{ and } \forall v \in V} \frac{\text{Performance of } M}{\text{Performance of } OPT} \tag{1}$$

where $G(T, W, U)$ is an arbitrary input of tasks, workers and their utilities, V is the set of all possible input orders, and v is one order in V.

Definition 8 (CR in the Random Order Model [17]): The competitive ratio in the the random order model of a specific online algorithm for the DTA problem is the following ratio,

$$CR_{RO} = min_{\forall G(T,W,U)} \frac{E[\text{Performance of } M]}{\text{Performance of } OPT} \tag{2}$$

where $G(T, W, U)$ is an arbitrary input of tasks, workers and their utilities, $\frac{E[MaxSum(M)]}{MaxSum(OPT)}$ is the expectation of the ratio of the total utility produced by the online algorithm and the optimal total utility of the offline scenario over all possible arrival orders.

Definition 9 (CR in the i.i.d Model [10]): The competitive ratio in the i.i.d model of a specific online algorithm for the DTA problem is the minimum ratio of the result of the online algorithm over the optimal result under all possible arrival orders generated by the spatiotemporal distributions of the tasks and the workers $\mathcal{D}_{\mathcal{R}}$ and $\mathcal{D}_{\mathcal{W}}$,

$$CR_{i.i.d} = min_{\forall G(T,W,U) \text{ and } \forall v \in V \text{ follows } \mathcal{D}_{\mathcal{R}} \text{ and } \mathcal{D}_{\mathcal{W}}} \frac{\text{Performance of } M}{\text{Performance of } OPT}$$

where $G(T, W, U)$ is an arbitrary input of tasks, workers and their utilities, V is the set of all possible input orders of tasks and the workers, and v is one order in V.

3 Dynamic Task Assignment Algorithms

Solutions to the dynamic task assignment problem roughly fall into two modes: batch mode and real-time mode. Batch mode periodically processes a set of workers and tasks that arrive within a specific time interval. Real-time mode makes an assignment immediately when a worker or a task appears on the platform. Both modes are able to handle dynamically arrived workers and tasks. However, only the real-time mode is suited for stringent real-time requirement, *i.e.*, tasks should be assigned immediately upon arrival.

3.1 Batch Mode

The basic idea of the batch mode is to periodically make an assignment in the static scenario, *i.e.*, both workers and tasks have already appeared on the platform. All the existing solutions in the batch mode [13, 14, 21, 20, 6] aim to maximize the total utility. According to the methods to conduct the static assignment, the batch mode can be further categorized as *maxmimum flow based* methods [13, 14, 21] and *greedy based* methods [20, 6].

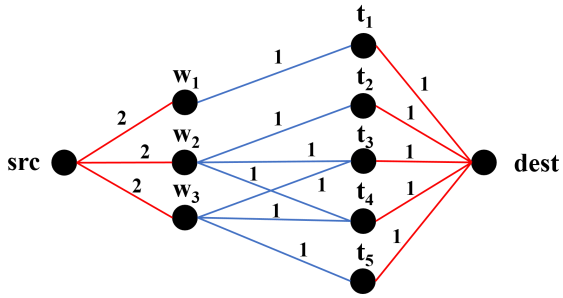


Figure 3: The procedure of reducing DTA to the maximum flow problem.

Maximum Flow Based Methods. Kazemi *et al.* [13] are the first to use a batch mode solution to the DTA problem in spatial crowdsourcing. Since they aim to maximize the number of performed tasks (*i.e.*, the utility between each worker and task is 1), the basic idea is to reduce the instance of DTA into an instance of maximum flow problem. Fig. 3 illustrates the procedure of the reduction. First, the capacity of the edge (src, w_i) is c_{w_i} because each worker can only perform c_{w_i} tasks at most. Second, since workers can only perform tasks that are in their regions (*e.g.*, w_1 can only perform t_1), the vertex mapped from w_i can transfer flow to only some of the vertices mapped from those tasks (*e.g.*, the edge between w_1 and t_1). The capacity of the edge between workers and tasks is 1 because a worker would not repeatedly perform the same task. Last, the capacity of the edge $(t_j, dest)$ is c_{t_j} because a task can only have at most c_{t_j} assigned workers. By reducing to the maximum flow problem, any algorithm that computes the maximum flow in the network can be used to solve the instance, *e.g.*, Ford-Fulkerson algorithm [15]. Finally, the assignment between workers and tasks can be induced through the flow and capacity between w_i and t_j in the instance of maximum flow problem. Consequently, to solve the DTA problem we repeat this step for every batch.

Multiple heuristics techniques have been proposed to optimize solutions in the batch mode. Hient *et al.* [21] introduce a Least Location Entropy Priority (LLEP) strategy to seek a global optimal by considering future coming workers. They use entropy of a location to measure the total number of workers in that location as well as the relative proportion of their future visits to that location. A higher priority is given to tasks located in areas with smaller location entropy, because those tasks have a lower chance of being completed by other workers. A Nearest Neighbor Priority (NNP) Strategy is also proposed to minimize the total travel distance of workers.

Greedy Based Methods. Greedy is a straightforward batch mode solution to the DTA problem. Hien *et al.* [20] propose to always select the worker who has the maximum number of unperformed tasks. Cheng *et al.* [6] design a greedy strategy to always select the pair of worker and task with the maximum utility. The benefit of

the greedy methods is that they are usually efficient and a few techniques can help improve the effectiveness of the methods. Some successful optimization techniques include prediction the arrivals of tasks and workers [6], divide-and-conquer [6], etc.

Summary. A comprehensive experimental comparison among batch-based solutions can be found in [5]. LLEP [21] is more effective but less efficient due to the time complexity of maximum flow problem and NNP [21] is more efficient. Note that assignment algorithms in the batch mode are online algorithms and it is possible to analyze their theoretical guarantees under partial information, *i.e.*, competitive ratio. However, all the methods in the batch mode [13, 14, 21, 20, 6] can only guarantee their effectiveness within a batch but there is no guarantee on the global performance. It remains open whether the batch mode is competitive under the adversarial model, random order model and i.i.d model.

3.2 Real-time Mode

Solutions in the real-time mode to the DTA problem decide the assignment once a worker or a task appears on the platform and are thus more challenging than those in the batch mode. Existing solutions in the real-time mode vary in objective goals. Popular optimization objectives include minimizing the total travel distance between workers and tasks [12, 1, 17, 25] such that the average waiting time of tasks (*e.g.*, passengers in taxi dispatching services [31]) is minimized, and maximizing the total utility between workers and tasks [26, 19, 28, 7, 8].

Minimizing the Total Distance. Greedy [12] can be a naive method to immunize the total distance. It matches each new arrival request to its currently nearest unmatched worker. The competitive ratio of Greedy is $O(2^n - 1)$ under adversarial model [12]. In [12], the authors also propose another method, called Permutation, to further improve the competitive ratio to $O(2n-1)$. The basic idea of Permutation is to make an assignment for each task according to the result of the offline minimum weighted matching (*e.g.*, Hungarian algorithm). Since a deterministic method may easily obtain a worse competitive ratio under the adversarial model, other researchers [1, 17] utilize the Hierarchically Separated Tree (HST) [9] to design a randomized algorithm such that a log-scale competitive ratio can be obtained. Specifically, they first embed the metric space into an HST. Then, they use HST-Greedy [17] and HST-Reassignment [1] to achieve the ratio of $O(\log^3 n)$ and $O(\log^2 n)$. However, these studies [12, 1] mainly focus on analyzing the worst-case competitive ratios of the proposed online algorithms, while [25] studies the performance of these algorithms in practice (*i.e.*, Random Order Model). Particularly, they observe a surprising result that the simple and efficient greedy algorithm, which has been considered as the worst due to its exponential worst-case competitive ratio, is significantly more effective than other algorithms. They further show that the competitive ratio of the worst case of the Greedy algorithm is actually a constant of 3.195 in the average-case analysis.

Maximizing the Total Utility. In order to maximize the total utility between workers and tasks, Tong *et al.* [26] propose a Hungarian-based method called TGOA with competitive ratio $1/4$ under random order model. They also propose a greedy-based method called TGOA-Greedy with competitive ratio $1/8$ under the same model. Both [26] and [19] propose a threshold-based method to maximize the utility in bipartite matching [26] and trichromatic matching [19]. A threshold of utility is sampled beforehand and the method arbitrarily choose an assignment only if the utility between the worker and the task is above the threshold. In practice, prediction is also used to improve the effectiveness and efficiency of the real-time methods [28, 24, 7, 8].

Summary. A comprehensive experimental comparison among solutions in the real-time mode which minimize the total distance can be found in [25]. From large-scale evaluations, a surprising result is observed that the simple greedy algorithm may be still competitive in the random order model, which is more practical than the adversarial model. Nevertheless, comprehensive evaluations among solutions in the real-time mode in the goal of maximizing the total utility are still missing.

We summarize existing solutions in the real-time mode in Table 1. Constant competitive ratio is usually achievable under the random order model and the i.i.d model. The competitive ratio under i.i.d model is usually

Table 1: Comparisons of existing real-time solutions to the dynamic task assignment problem.

Objective	Method	Analysis Model	Competitive Ratio
Minimize Distance	Greedy [12]	Adversarial	$O(2^n - 1)$
	Permutation [12]	Adversarial	$O(2n - 1)$
	HST-Greedy [17]	Adversarial	$O(\log^3 n)$
	HST-Reassignment [1]	Adversarial	$O(\log^2 n)$
	Greedy [25]	Random order	3.195 in worst case
Maximize Utility	TGOA [26]	Random Order	0.25
	TGOA-Greedy [26]	Random Order	0.125
	Basic-Threshold [19]	Random Order	$1/(3e \ln(U_{max} + 1))$
	POLAR-OP [28]	i.i.d	0.47
	ADAP [7]	i.i.d	$0.5 - \epsilon$
	NADAP [8]	i.i.d	0.295

higher, because prediction is usually helpful to improve the effectiveness of method in practice [28, 7, 8].

4 Conclusion

In this article, we formulate the generic Dynamic Task Assignment (DTA) problem for spatial crowdsourcing and briefly review the state-of-the-art solutions to the DTA problem. As an emerging research topic, DTA is far from mature. We list some of the open questions below. One interesting open problem is whether Greedy can achieve constant competitive ratio under the random order model for the DTA problem when minimizing total distance [25]. Another open issue is whether existing spatial indexes, which support moving object queries, can be extended to support the online data processing in spatial crowdsourcing. Finally, well-defined benchmarks to test and compare different spatial crowdsourcing data processing techniques are still missing. We envision this paper to not only raise awareness of DTA in the database community but also invite the database researchers to advance this promising area.

References

[1] N. Bansal, N. Buchbinder, A. Gupta, and J. Naor. A randomized $o(log^2 k)$-competitive algorithm for metric bipartite matching. *Algorithmica*, 68(2):390–403, 2014.

[2] A. Borodin and R. El-Yaniv. *Online computation and competitive analysis*. Cambridge University Press, 2005.

[3] L. Chen, D. Lee, and T. Milo. Data-driven crowdsourcing: Management, mining, and applications. In *31st IEEE International Conference on Data Engineering*, ICDE '15, pages 1527–1529, 2015.

[4] L. Chen and C. Shahabi. Spatial crowdsourcing: Challenges and opportunities. *IEEE Data Engineering Bulletin*, 39(4):14–25, 2016.

[5] P. Cheng, X. Jian, and L. Chen. An experimental evaluation of task assignment in spatial crowdsourcing. *Proceedings of the VLDB Endowment*, 11(11):1428–1440, 2018.

[6] P. Cheng, X. Lian, L. Chen, and C. Shahabi. Prediction-based task assignment in spatial crowdsourcing. In *33rd IEEE International Conference on Data Engineering*, ICDE '17, pages 997–1008, 2017.

[7] J. P. Dickerson, K. A. Sankararaman, A. Srinivasan, and P. Xu. Allocation problems in ride-sharing platforms: Online matching with offline reusable resources. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, AAAI '18, pages 1007–1014, 2018.

[8] J. P. Dickerson, K. A. Sankararaman, A. Srinivasan, and P. Xu. Assigning tasks to workers based on historical data: Online task assignment with two-sided arrivals. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '18, pages 318–326, 2018.

[9] J. Fakcharoenphol, S. Rao, and K. Talwar. A tight bound on approximating arbitrary metrics by tree metrics. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, STOC '13, pages 448–455, 2003.

[10] J. Feldman, A. Mehta, V. Mirrokni, and S. Muthukrishnan. Online stochastic matching: Beating 1-1/e. In *FOCS 2009*.

[11] H. Garcia-Molina, M. Joglekar, A. Marcus, A. G. Parameswaran, and V. Verroios. Challenges in data crowdsourcing. *IEEE Transactions on Knowledge and Data Engineering*, 28(4):901–911, 2016.

[12] B. Kalyanasundaram and K. Pruhs. Online weighted matching. *Journal of Algorithms*, 14(3):478–488, 1993.

[13] L. Kazemi and C. Shahabi. Geocrowd: enabling query answering with spatial crowdsourcing. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '12, pages 189–198, 2012.

[14] L. Kazemi, C. Shahabi, and L. Chen. Geotrucrowd: trustworthy query answering with spatial crowdsourcing. In *Proceedings of the 21st International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '13, pages 304–313, 2013.

[15] J. Kleinberg and E. Tardos. *Algorithm design*. Pearson Education India, 2006.

[16] G. Li, J. Wang, Y. Zheng, and M. Franklin. Crowdsourced data management: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 28(9):2296–2319, 2016.

[17] A. Meyerson, A. Nanavati, and L. Poplawski. Randomized online algorithms for minimum metric bipartite matching. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '06, 2006.

[18] M. Musthag and D. Ganesan. Labor dynamics in a mobile micro-task market. In *2013 ACM SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, 2013.

[19] T. Song, Y. Tong, L. Wang, J. She, B. Yao, L. Chen, and K. Xu. Trichromatic online matching in real-time spatial crowdsourcing. In *33rd IEEE International Conference on Data Engineering*, ICDE '17, pages 1009–1020, 2017.

[20] H. To, L. Fan, L. Tran, and C. Shahabi. Real-time task assignment in hyperlocal spatial crowdsourcing under budget constraints. In *2016 IEEE International Conference on Pervasive Computing and Communications*, PerCom '16, pages 1–8, 2016.

[21] H. To, C. Shahabi, and L. Kazemi. A server-assigned spatial crowdsourcing framework. *ACM Trans. Spatial Algorithms and Systems*, 1(1):2:1–2:28, 2015.

[22] Y. Tong, L. Chen, and C. Shahabi. Spatial crowdsourcing: Challenges, techniques, and applications. *Proceedings of the VLDB Endowment*, 10(12):1988–1991, 2017.

[23] Y. Tong, L. Chen, Z. Zhou, H. V. Jagadish, L. Shou, and W. Lv. SLADE: A smart large-scale task decomposer in crowdsourcing. *IEEE Transactions on Knowledge and Data Engineering*, 30(8):1588–1601, 2018.

[24] Y. Tong, Y. Chen, Z. Zhou, L. Chen, J. Wang, Q. Yang, J. Ye, and W. Lv. The simpler the better: A unified approach to predicting original taxi demands based on large-scale online platforms. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, pages 1653–1662, 2017.

[25] Y. Tong, J. She, B. Ding, L. Chen, T. Wo, and K. Xu. Online minimum matching in real-time spatial data: experiments and analysis. *Proceedings of the VLDB Endowment*, 9(12):1053–1064, 2016.

[26] Y. Tong, J. She, B. Ding, L. Wang, and L. Chen. Online mobile micro-task allocation in spatial crowdsourcing. In *32nd IEEE International Conference on Data Engineering*, ICDE '16, pages 49–60, 2016.

[27] Y. Tong, L. Wang, Z. Zhou, L. Chen, B. Du, and J. Ye. Dynamic pricing in spatial crowdsourcing: A matching-based approach. In *Proceedings of the 2018 International Conference on Management of Data*, SIGMOD '18, pages 773–788, 2018.

[28] Y. Tong, L. Wang, Z. Zhou, B. Ding, L. Chen, J. Ye, and K. Xu. Flexible online task assignment in real-time spatial data. *Proceedings of the VLDB Endowment*, 10(11):1334–1345, 2017.

[29] Y. Tong, Y. Zeng, Z. Zhou, L. Chen, J. Ye, and K. Xu. A unified approach to route planning for shared mobility. *Proceedings of the VLDB Endowment*, 11(11):1633–1646, 2018.

[30] Y. Zeng, Y. Tong, L. Chen, and Z. Zhou. Latency-oriented task completion via spatial crowdsourcing. In *34rd IEEE International Conference on Data Engineering*, ICDE '18, pages 317–328, 2018.

[31] L. Zhang, T. Hu, Y. Min, G. Wu, J. Zhang, P. Feng, P. Gong, and J. Ye. A taxi order dispatch model based on combinatorial optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '18, pages 2151–2159, 2017.

Spatiotemporal Clustering in Urban Transportation - a Bus Route Case Study in Washington D.C.

Xiqi Fei, Olga Gkountouna
George Mason University, USA
{xfei, ogkounto}@gmu.edu

Abstract

Public buses are an important part of the urban transportation mix. However, a considerable disadvantage of buses is their slow speed, which is in part due to frequent stops, but also due to the lack of segregation from other vehicles in traffic. As such, assessing bus routes and the respective sections that are prone to congestion is an important aspect of route planning, scheduling, and the creation of dedicated bus lanes. In this work we use bus tracking data from the Washington Metropolitan Area Transit Authority, to discover speed patterns of specific bus routes in relation to the road network throughout the day. Specifically, we focus on using these patterns to identify free flow segments, bus stop locations, traffic light locations and road segments prone to congestion.

1 Introduction

Buses, like other forms of public transportation, provide an essential service to users that depend on this service to commute to and from work, and to other places. Such services are especially important in large cities, where increasing vehicular traffic flows continues to a be a major challenge for urban planners, who must content with associated road congestion in cities. However, buses face several challenges, one of which is having notoriously slow speeds (as low as 17mph for some bus routes in DC), thus resulting in longer commute times for passengers. Besides frequent stops, which are prescribed for this means of transportation, the speed is also impacted by the lack of segregation from other vehicular traffic. As such, the assessment of traffic conditions along bus routes forms an integral part of route planning, scheduling, and the creation of dedicated bus lanes in cities.

In our study, we discretize a bus route and calculate the average speed using odometer and time stamp values to discover patterns of slowdowns throughout a 24-hour period. This slowdown pattern may be persistent throughout the day, random, or may appear at specific times of the day. Each of these cases are due to different causes. Random slowdowns throughout the day are indicative of traffic lights. More persistent slowdowns are indicative of bus stops, while time-dependent slowdowns are more likely related to traffic congestion. From a public transportation planning perspective, route segments prone to traffic congestion would be prime candidates for dedicated bus lanes.

We performed our analysis on real (Metrobus) data from WMATA [20], the public transport authority of the Washington DC area. We cluster all road segments along a bus route, using features derived from the bus speeds at each segment, and sampled at hourly intervals. Our results reveal different categories of road segments which can be associated with free-flow of traffic, and different types of slow-downs.

The remainder of this paper is structured as follows. Section 2 presents a brief survey of the related work on bus data. Section 3 includes an overview of the main challenges we faced in working with this type of data. In Section 4, we present our method, experimental setup and an evaluation of our Washington D.C. Metrobus case study. Finally, Section 5 concludes the paper.

2 Related Work

Bus data has received considerable attention for the estimation of traffic conditions. Floating car data (FCD) or probe vehicle data (PVD) refers to the use of data generated by one vehicle as a sample to assess to overall traffic conditions (cork swimming in the river). PVD from automobiles have previously been studied to estimate travel times and traffic conditions [15, 10, 2, 18, 22, 21] and traffic speed [9]. Specifically, as it relates to bus data, the focus of our work, a number of works [12, 16, 17, 3, 1] have used this data to study travel times and related traffic flows in urban areas. It was shown in [3] that the difference between travel times of a bus and that of an car was relatively stable, and that buses with automated vehicle locators (AVL) can be used as a probes to collect travel time data at regular intervals with minimum cost. AVL bus data is used for characterizing the performance of arterial roads in Oregon [1]. [12] examine real-time sensitivity between buses and cars to study the feasibility of a real bus probe application in an urban traffic environment. [17] predicted travel times under heterogeneous traffic conditions by applying a Kalman filtering technique to GPS data collected from buses.Further, [16] use bus probe data to evaluate the travel time variability and the level of service of roads. Kumar et al. [8] developed a bus arrival time prediction system, considering both spatial and temporal variations of travel times. In [7] a simulation technique was used to study the influence of these stops on traffic flow under heterogeneous traffic conditions.

The location optimization of bus stops has also been the focus of several works. Saka [14] developed a model for determining optimum bus-stop spacing in urban areas, with the aim of decreasing travel time, headway, and the fleet size. Chien et al. [5] focus on optimizing bus routes in areas with a commuter (many-to-one) travel pattern. [4] address the problem of optimizing the placement of bus stop locations, with the goal to improve the accessibility of a bus service. [13] used a GIS-based methodology to identify hazardous bus stop locations prone to auto-pedestrian collisions. [6] developed a spatial interaction coverage model for identifying bus stop redundancy in order to optimize transit planning. A work more relevant to ours [11] proposes a methodology to de-noise GPS AVL data, identify bus stops, and detect time schedule information.

While [11] clusters all the bus recordings along a route to form groups, with each group corresponding to one stop, our approach aims to discover the different categories of segments within the bus route. Ideally, all stops should appear in one cluster, the traffic lights in another, etc.

3 Challenges

While most existing approaches are based on GPS data, in our study we use odometer readings recorded using a bus AVL system. The data comprises of bus trips for different routes collected during a 24-hour period. Each trip consists of a time series of odometer readings from a specific bus, with sampling interval varying from 1 to 10 seconds.

3.1 Rate of recordings

The rate at which the location and time stamp information were recorded is not constant. It varies for different buses, as well as for the same bus over time. The time delay between any two consecutive measurements varies from one to several seconds. To overcome this inconsistency, we discretized time into constant time intervals, and calculated the average bus speed over those specific periods.

3.2 Odometer alignment

Even though they are more reliable than GPS, still the odometer readings between any two different bus trips are not perfectly aligned. Specifically, two buses that follow the same path may be at different locations after 1,000 odometer-measured feet. One reason for this misalignment is the choice of lane, especially when turning

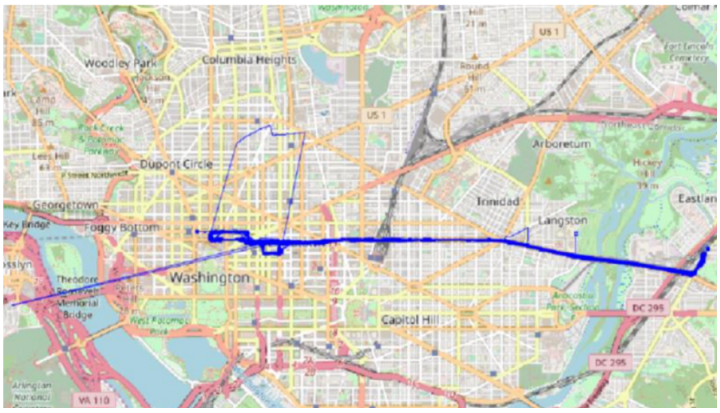


Figure 1: Bus trips of X2 route.

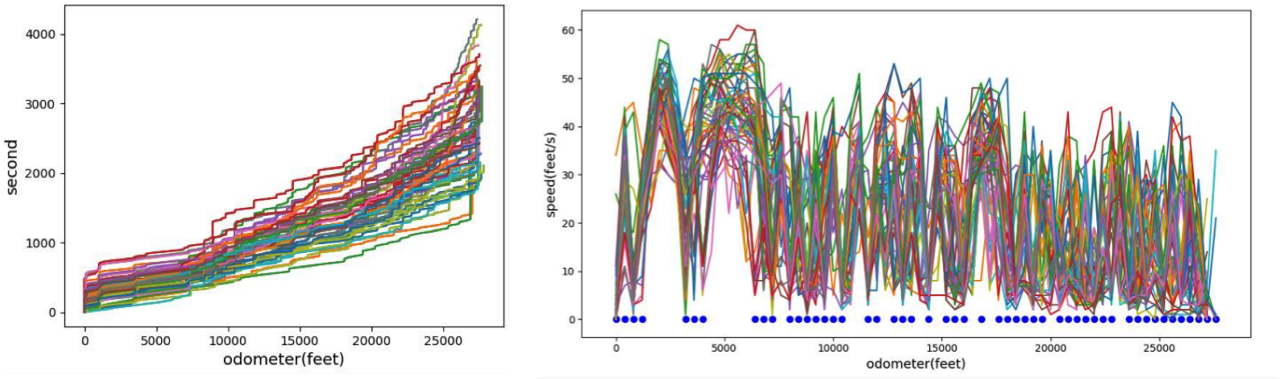
the bus. A bus that goes around the outer lane to turn will record a longer odometer distance compared to a bus that has used the inner lane in the same direction of traffic flow. Another reason is that the pressure of the tires may contribute to recording more feet over the same distance travelled. This misalignment creates a problem wherein when we divide the odometer space into segments of 200ft, these segments are not the same for every bus trip. Consequently, this makes the locations of bus stops appear differently for every bus. To address these issues, we plan to align the bus routes as part of our future work, using a special indication of bus stop locations. Whenever a bus enters or exits a geo-fence around a bus stop, it is recorded in the data. These recordings should include all bus stops, regardless of whether the bus actually opened its doors or not.

3.3 Bus stop location identification

We want to know the bus stop locations in the odometer space, i.e., how far each stop is from the beginning of the route. This forms an important part of this research; we aim to identify what segments of the route have traffic patterns indicating the existence of a bus stop, as opposed to slow traffic due to congestion, or traffic lights. The dataset contains indications of an ”Open door” whenever passengers board or disembark the vehicle, and which happens almost exclusively at bus stop locations. Combining these indications from all the bus trips is not a trivial task as the odometers of different bus routes are not aligned. Using the geo-fence based indication mentioned in challenge 3.2 can solve this misalignment. However, this method includes all bus stop locations, even if no buses ever stop there (for example a very unpopular bus stop, or an old bus stop that is no longer in use). To identify valid bus stops as our ground truth, we consider a road segment as containing a bus stop only if more than a minimum number of buses per day open their doors at that location.

4 Case Study: Washington D.C. Metrobus

The main bus service in the Washington D.C. metropolitan area, Metrobus, provides more than 400,000 trips each weekday, and serves 11,500 bus stops in the District of Columbia, Maryland, and Virginia respectively. Metrobus has more than 1,500 buses operating on 325 routes, and is the sixth busiest bus agency in the United States [20]. Over the last two decades, there has been gradual decrease in the ridership of Metrobus, particular due to increased congestion, and which has resulted in significant lose of revenue [19]. This makes the specific study area and bus service of particular interest in the study of traffic flow dynamics. The sections that follow provide an overview of our findings from an analysis of real data acquired from the Metrobus service.



(a) Time travelled vs. Distance covered.

(b) Average Speed vs. Distance covered.

Figure 2: Relationship between travel time, average speed and odometer readings of all the buses.

4.1 Data

We used bus trips from the X2 route of the Washington D.C. metropolitan area shown in Figure 1. Our dataset contained 489 trips from 10/04/2016. During data cleaning, we removed any bus trips that significantly deviated from the examined route. This resulted in a dataset containing 58 trips travelling on one direction. We understand the limitation of performing analysis on such a small dataset, and for only a 24 hour period. Future work will undertake a more in-depth analysis, using a larger collection of trips that are acquired for a much longer time period.

4.2 Preprocessing

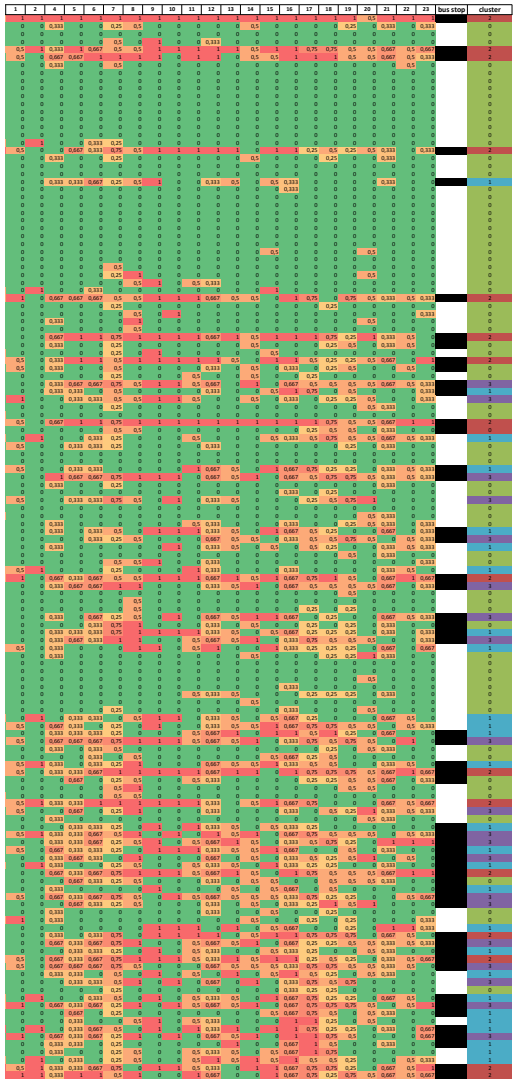
We discretized the bus route into 200ft segments. This segment length was considered neither too small to be noisy, nor large enough so that several bus stops and/or traffic lights would be contained within any single segment. The odometer readings (i.e., distance covered) versus the time from the beginning of the trip are shown in Figure 2(a), for all the buses in our data collection.

Using the above routes, we estimated the average speed of every bus, for each 200ft road segment. These results are visualized in Figure 2(b). The blue dots at the bottom of this figure correspond to indications of a bus door opening. These indications are used in our evaluation as ground truth for where bus stops are located. To avoid false indications (e.g. when the bus driver opens a door for an emergency passenger request), we consider a road segment having a bus stop only if at least five buses have opened their doors at that segment.

4.3 Clustering Segments

We use the aforementioned bus speeds to derive a set of features for clustering road segments along the bus route. Our first approach uses the percentage of slow buses calculated over hourly time buckets. Given a speed threshold τ, we consider a bus as "slow" if it moves at a speed $v < \tau$. For our experiments, we set the value of τ to be 10km/h. Our second approach uses the average speed of all buses calculated for each road segment for each hourly time bucket. Using these approaches, we generate a set of 24 features per road segment. Principal component analysis was then used to reduce our feature space to 4 principal components, following which these latent features were fed to a k-means clustering algorithm. A k value of 4 was used since we expect to discover 4 types of road segments.

Intuitively, we expect *bus stop* locations to be those where almost all buses would stop. On the other hand, *traffic lights* should cause some buses to stop at a red light, while others would pass with a green light, regardless



(a) Fraction of slow buses

(b) Average bus speed

Figure 3: Clustering results.

of time of the day. *Congestion* should affect specific road segments at specific times of the day (e.g. rush hours). This would be a periodic phenomenon, and it would be very interesting to examine this aspect as part of future research. Finally, in *free flow* segments, buses should have relatively high values of speed throughout the day. These segments are identified as those having between 0 and 20 slow buses for at least 20 out of the 24 hour buckets of the day. The remaining segments are labelled as other congestion or traffic lights. The results of our two approaches are presented in the following section.

4.4 Experimental Results

A visualization of our results is presented in Figure 3. Each row corresponds to a 200ft road segment along the X2 bus route. Columns labels 1 through 24 are the hourly buckets, and correspond to the 24 derived features. For Figure 3(a), these features are the fraction of slow buses, colored in a scale from green to red, with red depicting a high number of slow buses, while green depicts few or no slow buses. In the case of Figure 3(b), the

Table 1: Summary of Results for the Fraction of Slow Buses Approach.

Cluster Label	0	1	2	3	Total
Bus stop	5	16	15	16	52
Other congestion or traffic light	14	11	1	4	30
Free flow	55	0	0	0	55
Total	74	27	16	20	137

Table 2: Summary of Results for the Average Bus Speed Approach.

Cluster Label	0	1	2	3	Total
Bus stop	0	24	27	1	52
Other congestion or traffic light	0	5	14	11	30
Free flow	20	0	4	31	55
Total	20	29	45	43	137

features are the average hourly bus speeds for each segment. In this Figure, green shows high speeds, yellow and orange show moderate speeds, while red depicts low bus speeds. The buses start from an origin outside the city, where there is typically less traffic, and travel towards a destination in the center, which is more prone to congestion. This explains why there are many more green cells towards the top (origin) and more red cells at the bottom (destination). The 25th column contains the locations of bus stops depicted as black cells. White cells correspond to road segments that do not include bus stops. The final column shows the derived cluster label of each segment. For clarity, in both Figures 3(a) and 3(b) cluster '0' is colored green, '1' is blue, '2' is red, and '3' is purple. Note however that the numbering of the cluster labels is random. Clusters of the same number in the two figures do not necessarily correspond to each other. In the following we evaluate the quality of our findings.

4.4.1 Using the Fraction of Slow Buses.

The results of our cluster analysis are shown in Table 1. Cluster label 0 includes all the road segments of free-flow. Cluster 1 appears to be more related to congestion as in several of its segments there contain slow buses at certain hours, where there are no bus stops. We note that most road segments of cluster 1 are located towards the city center, compared to most free-flow segments (label 0) that are located outside the city center, and thus less prone to congestion. Most members of cluster 2 correspond to bus stops. However, all the bus stops are almost evenly distributed among labels 1, 2 and 3. In particular, 15 out of 16 road segments of cluster 2 contain bus stops (Precision=0.94), while 15 out of the 52 bus stops in total were labeled as cluster 2 (Recall=0.29). What remains to be tested is which road segments correspond to traffic lights, for which no data was available in this study. This data could be aligned with collected odometer readings to verify if one or more clusters correspond to these locations.

4.4.2 Using the Average Bus Speed.

Table 2 shows the results of using the average bus speed in our analysis. The members of cluster 0 correspond only to road segments of free-flow, where the average bus speed is high throughout the day. These segments are mainly located outside the city center, with no bus stops. Compared to the use of the percentage of slow buses, several free-flow segments are now placed in cluster 3. Those segments are closer to the city center, but are not bus stop or traffic light locations. Thus they are able to maintain high average speeds throughout the day.

Most bus stops are distributed between only two clusters, 1 and 2. While the majority of them (28 of 53) are in cluster 2, the purity of bus stop locations in cluster 1 is larger, with a precision of 0.83, compared to 0.60 for bust stops in cluster 2. The members of cluster 1 have low average speeds throughout the day, implying frequent

bus stops. On the other hand, the members of cluster 2 have lower speeds at certain hours of the day, which indicates that they are also prone to traffic, or that the corresponding bus stops are more popular at certain hours of the day. This explains why cluster 2 contains 31% of "other congestion" segments. Cluster 3 almost never coincides with a bus stop and contains road segments with moderate, but not low, average speeds.

5 Conclusions

This analysis is a small step in the direction of spatiotemporal analysis of latent traffic patterns. We explored a data-driven approach to assess the traffic characteristics of road segments using public transportation data. In particular, we use data mining techniques to identify latent speed patterns in bus traffic related flows: traffic-light related delays, bus stop related delays, or free-flow. The lack of sufficient data is an obvious limitation of our analysis. More data are needed in order to assess the validity of our approach, but we believe that our preliminary results show a very promising direction of research.

Acknowledgement

This research has been supported by National Science Foundation "AitF: Collaborative Research: Modeling movement on transportation networks using uncertain data" grant NSF-CCF 1637541.

We would like to thank the Washington Metropolitan Area Transit Authority (WMATA) for providing us with their data.

References

[1] R. Bertini and S. Tantiyanugulchai. Transit buses as traffic probes: Use of geolocation data for empirical evaluation. *Transportation Research Record: Journal of the Transportation Research Board*, 1870(1870):35–45, 2004.

[2] A. Bhaskar, E. Chung, and A. Dumont. Fusing loop detector and probe vehicle data to estimate travel time statistics on signalized urban networks. *Comp.-Aided Civil and Infrastruct. Engineering*, 26(6):433–450, 2011.

[3] P. Chakroborty and S. Kikuchi. Using bus travel time data to estimate travel times on urban corridors. *Transportation Research Record: Journal of the Transportation Research Board*, 1870(1870):18–25, 2004.

[4] S. I. Chien and Z. Qin. Optimization of bus stop locations for improving transit accessibility. *Transportation planning and Technology*, 27(3):211–227, 2004.

[5] S. I.-J. Chien, B. V. Dimitrijevic, and L. N. Spasovic. Optimization of bus route planning in urban commuter networks. *Journal of Public Transportation*, 6(1):4, 2003.

[6] E. M. Delmelle, S. Li, and A. T. Murray. Identifying bus stop redundancy: A gis-based spatial optimization approach. *Computers, Environment and Urban Systems*, 36(5):445–455, 2012.

[7] R. Z. Koshy and V. T. Arasan. Influence of bus stops on flow characteristics of mixed traffic. *Journal of transportation engineering*, 131(8):640–643, 2005.

[8] B. A. Kumar, L. Vanajakshi, and S. C. Subramanian. Bus travel time prediction using a time-space discretization approach. *Transportation Research Part C: Emerging Technologies*, 79:308–332, 2017.

[9] Q. Ou, R. L. Bertini, H. van Lint, and S. P. Hoogendoorn. A theoretical framework for traffic speed estimation by fusing low-resolution probe vehicle data. *IEEE Trans. Intelligent Transportation Systems*, 12(3):747–756, 2011.

[10] D. Pfoser, S. Brakatsoulas, P. Brosch, M. Umlauft, N. Tryfona, and G. Tsironis. Dynamic travel time provision for road networks. In *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '08, pages 68:1–68:4, 2008.

[11] F. Pinelli, F. Calabrese, and E. P. Bouillet. Robust bus-stop identification and denoising methodology. In *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*, pages 2298–2303. IEEE, 2013.

[12] W. Pu, J. Lin, and L. Long. Real-time estimation of urban street segment travel time using buses as speed probes. *Transportation Research Record: Journal of the Transportation Research Board*, 2129(2129):81–89, 2009.

[13] S. S. Pulugurtha and V. K. Vanapalli. Hazardous bus stops identification: An illustration using gis. *Journal of Public Transportation*, 11(2):4, 2008.

[14] A. A. Saka. Model for determining optimum bus-stop spacingin urban areas. *Journal of Transportation Engineering*, 127(3):195–199, 2001.

[15] R.-P. Schäfer, K.-U. Thiessenhusen, and P. Wagner. A traffic information system by means of real-time floating-car data. In *ITS world congress*, volume 2, 2002.

[16] N. Uno, F. Kurauchi, H. Tamura, and Y. Iida. Using bus probe data for analysis of travel time variability. *Journal of Intelligent Transportation Systems*, 13(1):2–15, 2009.

[17] L. Vanajakshi, S. C. Subramanian, and R. Sivanandan. Travel time prediction under heterogeneous traffic conditions using global positioning system data from buses. *IET intelligent transport systems*, 3(1):1–9, 2009.

[18] Y. Wang, Y. Zheng, and Y. Xue. Travel time estimation of a path using sparse trajectories. In *The 20th ACM SIGKDD*, pages 25–34, 2014.

[19] Washington Metropolitan Area Transit Authority. FY2017 Budget: Ridership and Revenue, October, 2015.

[20] WMATA. Washington Metropolitan Area Transit Authority. `https://www.wmata.com`.

[21] X. Zhan, S. V. Ukkusuri, and C. Yang. A bayesian mixture model for short-term average link travel time estimation using large-scale limited information trip-based data. *Automation in Construction*, 72:237–246, 2016.

[22] F. Zheng and H. Van Zuylen. Urban link travel time estimation based on sparse probe vehicle data. *Trans. Res. Part C: Emerging Technologies*, 31:145–157, 2013.

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

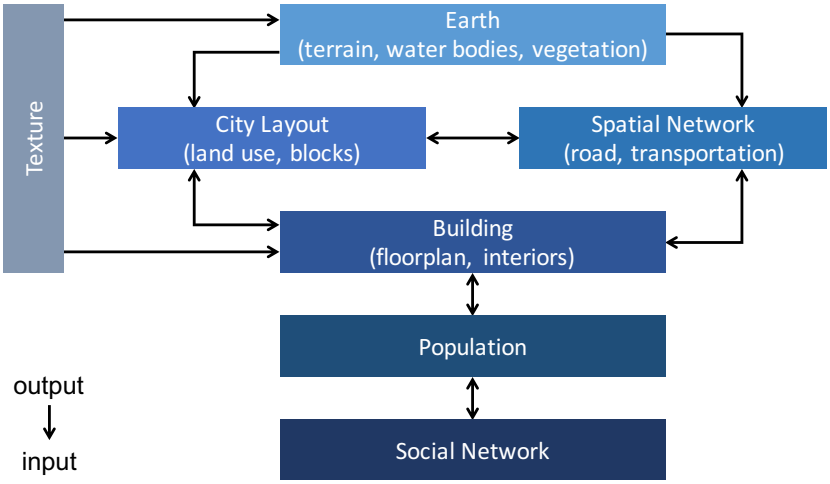


Figure 1: Procedural city generation and input-output relationships

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.

join today!

SIGSPATIAL & ACM

www.sigspatial.org **www.acm.org**

The **ACM Special Interest Group on Spatial Information** (SIGSPATIAL) addresses issues related to the acquisition, management, and processing of spatially-related information with a focus on algorithmic, geometric, and visual considerations. The scope includes, but is not limited to, geographic information systems (GIS).

The **Association for Computing Machinery** (ACM) is an educational and scientific computing society which works to advance computing as a science and a profession. Benefits include subscriptions to *Communications of the ACM*, *MemberNet*, *TechNews* and *CareerNews*, full and unlimited access to online courses and books, discounts on conferences and the option to subscribe to the ACM Digital Library.

- ❑ SIGSPATIAL (ACM Member) ..... $ 15
- ❑ SIGSPATIAL (ACM Student Member & Non-ACM Student Member) ..... $ 6
- ❑ SIGSPATIAL (Non-ACM Member) ..... $ 15
- ❑ ACM Professional Membership ($99) & SIGSPATIAL ($15) $114
- ❑ ACM Professional Membership ($99) & SIGSPATIAL ($15) & ACM Digital Library ($99) $213
- ❑ ACM Student Membership ($19) & SIGSPATIAL ($6) ..... $ 25

payment information

Name _____

ACM Member # _____

Mailing Address _____

_____

City/State/Province _____

ZIP/Postal Code/Country _____

Email _____

Mobile Phone _____

Fax _____

Mailing List Restriction
ACM occasionally makes its mailing list available to computer-related organizations, educational institutions and sister societies. All email addresses remain strictly confidential. Check one of the following if you wish to restrict the use of your name:

- ❑ ACM announcements only
- ❑ ACM and other sister society announcements
- ❑ ACM subscription and renewal notices only

Credit Card Type: ❑ AMEX ❑ VISA ❑ MC

Credit Card # _____

Exp. Date _____

Signature _____

Make check or money order payable to ACM, Inc

ACM accepts U.S. dollars or equivalent in foreign currency. Prices include surface delivery charge. Expedited Air Service, which is a partial air freight delivery service, is available outside North America. Contact ACM for more information.

Questions? Contact:
ACM Headquarters
2 Penn Plaza, Suite 701
New York, NY 10121-0701
voice: 212-626-0500
fax: 212-944-1318
email: acmhelp@acm.org

Remit to:
ACM
General Post Office
P.O. Box 30777
New York, NY 10087-0777

SIGAPP

www.acm.org/joinsigs

Association for Computing Machinery

Advancing Computing as a Science & Profession

The SIGSPATIAL Special

ACM SIGSPATIAL
http://www.sigspatial.org