Towards Effective Harnessing Of Crowd Sensing And Drive-by Sensing For High-Resolution Urban Heat Analysis: Challenges And Approaches

by

Seyed Navid Hashemi Tonekaboni

(Under the Direction of Lakshmish Ramaswamy)

Abstract

Large fractions of urban populations around the globe are under an increased threat of extreme heat events. Anthropogenic climate change, coupled with rapid urbanization, have exacerbated these threats. Most of the current urban heat studies have relied upon conventional data sources such as satellites and weather stations to map and analyze the urban heat islands (UHIs). However, these data sources lack the spatial and temporal resolutions required to accurately capture the temperature variations in space and time.

Towards overcoming these limitations, this thesis explores the challenges of harnessing modern data collection paradigms, namely crowdsourcing (through human-borne sensors) and drive-by sensing (through vehicle-borne sensors) for UHI analysis. This thesis proposes a three-tier framework called Smart Community-Centric Urban Thermal Sensing (SCOUTS) for efficiently gathering temperature data through

crowdsourcing and drive-by sensing, integrating them with data from satellite and weather stations and performing innovative analysis to map and study UHIs.

While crowdsourcing and drive-by sensing are inexpensive data collection strategies, harnessing them in an efficient manner for UHI analysis poses several research challenges. This thesis addresses two major challenges in crowdsourcing and drive-by sensing for UHI analysis, respectively. The first is to detect human-borne temperature sensors that are placed anomalously and hence fail to accurately represent the actual outdoor environment. The proposed scheme for detection of anomalously placed sensors is based on our novel feature selection and classification design.

The second major challenge that we address is to select public transportation vehicles (city buses) for sensor deployment so as to maximize the spatio-temporal coverage value of the data collected through the drive-by sensing paradigm for UHI analysis. In this regard, we make two unique research contributions: formulating the bus selection problem as an optimization problem and introducing our cost-aware approaches to enhance the spatiotemporal coverage. This thesis reports a series of experiments demonstrating the benefits and limitations of our approaches for detecting anomalously placed sensors in thermal crowdsensing and for sensor deployment in drive-by sensing.

INDEX WORDS: Urban Heat Analysis, Crowd Sensing, Drive-by Sensing, Urban Heat Island, Spatiotemporal Coverage

Towards Effective Harnessing Of Crowd Sensing And Drive-by Sensing For High-Resolution Urban Heat Analysis: Challenges And Approaches

by

Seyed Navid Hashemi Tonekaboni

B.S., Azad University - South Tehran Branch, 2012

A Dissertation Submitted to the Graduate Faculty
of The University of Georgia in Partial Fulfillment
of the
Requirements for the Degree

Doctor of Philosophy

Athens, Georgia

2019

Towards Effective Harnessing Of Crowd Sensing And Drive-by Sensing For High-Resolution Urban Heat Analysis: Challenges And Approaches

by

Seyed Navid Hashemi Tonekaboni

Approved:

Major Professor:	Lakshmish Ramaswamy
Committee:	Hamid R. Arabnia
	Deepak Mishra

Electronic Version Approved:

Suzanne Barbour
Dean of the Graduate School
The University of Georgia
August 2019

This work is dedicated to my parents, Samira and Amir. Without their love and support over the years none of this would have been possible.

Acknowledgments

I would like to deeply thank my advisor, Dr. Lakshmish Ramaswamy, for his support, knowledge, and encouragement. His willingness to give me the opportunity to work on research topics which I have been really passionate about and providing me with the chance to work with an interdisciplinary research group made my Ph.D. experience much more rewarding. I would also like to truly appreciate Dr. Hamid Arabnia for his ever-present support and mentorship during my Ph.D. journey by providing me with his insightful advice and feedback. I am also sincerely thankful to Dr. Mishra for his sincere support, guidance, and willingness to offer help whenever I needed it.

Thanks are due to my colleagues at the Data Intensive and Pervasive Systems (DIPS) Lab at the Computer Science Department: Sujeet Kulkarni for his commitment, and hard work. To Himanshu Pendyala and Omid Setayeshfar, who contributed to this research by developing our mobile applications. I am also very thankful to Dr. Andrew Grundstein as an excellent academic advisor in our interdisciplinary research group and Yanzhe Yin as a great researcher from the Department of Geography. I am very grateful to all the faculty and staff members

of the Computer Science Department. I will always be grateful for the opportunity that I was given in this department to learn and to grow.

Above all, I wholeheartedly thank my parents, Samira and Amir, and my brother, Majid, for their love, patience, encouragement, and support during my Ph.D. program. I would also like to express my sincere gratitude to friends who believed in me, supported me, and encouraged me throughout my academic career. Special thanks to Sahar Voghoei, Hamed Yaghoubian, Saber Soleymani, Soroush Omidvar, Delaram Yazdansepas, and Nasrin Rouhani for their kindness and strong support during the past few years.

This research has been partially funded by the National Science Foundation's (NSF) S&CC: Smart Connected Communities program under Grant Number 1637277. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the NSF.

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

1.1 URBAN HEAT ANALYSIS

Our planet is becoming increasingly warmer, more populated, and more urbanized. Due to global warming, deadly heat waves are becoming more common, and heat stresses (e.g., on morbidity and mortality) are exacerbated in cities because cities are warmer than surrounding rural areas. Unfortunately, urban heat phenomena hit disproportionately vulnerable communities such as the people who are living in poorly-planned neighborhoods, the communities who cannot afford air conditioning, infants and older people, construction workers, and city outdoor workers. Therefore, it is of crucial importance to meticulously track and analyze urban heat hazards to come up with effective strategies to make urban areas more livable.

Urban heat hazard has been studied well in low spatio-temporal resolution;

however, there is a great need to study this phenomenon in high spatio-temporal resolution. High-resolution urban heat analysis considerably helps city officials to devise effective strategies to minimize heat-related health problems of various at-risk communities. Specifically, it can help the vulnerable community of outdoor workers such as those who are involved in construction, sanitation, public works, and mail delivery [Gubernot et al., 2014]. Due to global warming, ambient heat exposure is predicted to be a prominent safety concern for many employees in the near future [Gubernot et al., 2014]. Therefore, it is vital to measure and understand the urban heat hazard accurately.

1.2 MOTIVATIONS AND RESEARCH OBJECTIVES

To date, urban heat vulnerability research has mostly focused on generating coarse-grained heat maps of cities using satellite images with low spatio-temporal resolutions to quantify the heat hazard [Duan et al., 2018, Jacob et al., 2004, Jiménez-Muñoz et al., 2014]. While some recent works propose incorporating data from nearby static weather stations, they fail to reflect the spatial variations of air temperature in urban areas due to the limited availability of weather stations.

The primary motivation behind this study is the importance of this avenue of research and the impact of growing heat hazards on our lives. The increased heat events have direct and indirect influences on our lives. As an example of their

direct impact, the Center for Disease Control and Prevention (CDC) reported that around 618 people in the United States are killed by extreme heat every year [of Disease Control and Prevention, 2017]. This phenomenon leads to many heat-related illnesses such as heat syncope, heat stroke, and heat exhaustion, which are preventable [Luber and McGeehin, 2008]. Talking about the indirect impacts of increasing heat events, studies show that when exposed to warmer temperatures, plants grow more vigorously and produce more pollen than they otherwise would [Schmidt, 2016]. This higher level of pollen triggers and exacerbates the allergic effects on 400 million people in the world who suffer from allergic rhinitis and 300 million people who are dealing with asthma [Lake et al., 2016]. Therefore, this phenomenon has many direct and indirect impacts on our lives, and due to their ever-growing trends, the global research community has focused on understanding and mitigating their effects on our cities and our planet as a whole.

The primary objective of this thesis is to test the viability and effectiveness of harnessing smart devices and low-cost sensors to incorporate different sensing technologies and generate heatmaps with high spatiotemporal resolution. In this regard, we have proposed and implemented software and hardware solutions to address the limitations. Furthermore, we proposed approaches to add smart features into our design, which could be leveraged by various sensing frameworks.

In this study, we present a hybrid framework to incorporate the observations from remote sensing, in-situ-based sensing, crowd sensing, and drive-by sensing technologies. For this purpose, a scalable and robust smart sensor-based architecture is designed to leverage a variety of human and vehicle-borne sensors.

Although each sensing paradigm has its intrinsic limitations, our proposed architecture leverages the synergy provided by them. As a result, it can generate hyper-local heatmaps of cities. One of the major contributions of this study is demonstrating the inability of conventional UHI mapping approaches to capture the spatio-temporal variations within the heatmaps, compared to high-resolution heatmaps generated by the hybrid sensing approach. The results from this study help researchers to find a correlation between biophysical parameters of different geographical areas and the heat events associated with those locations.

1.3 PROBLEM STATEMENTS AND CONTRIBUTIONS

To be specific, there are three main problems which have been addressed in this dissertation. In the following paragraphs, we first explain each problem, then the contributions of this study to resolve those problems are discussed.

The first problem is the lack of a comprehensive and scalable framework to accurately capture the temperature variations and to create hyperlocal heatmaps with high spatiotemporal resolution. Therefore, we propose and implement a three-layer framework called Smart Community Centric Urban Thermal Sensing (SCOUTS). SCOUTS incorporates the results of four different sensing paradigms: remote sensing, in-situ-based sensing, crowd sensing, and drive-by sensing to provide heatmaps with high spatiotemporal resolution. In particular, different software and hardware solutions are designed and implemented to integrate crowd

sensing and drive-by sensing paradigms into our framework.

The second problem is the absence of a mechanism to detect anomalous placement of sensors in temperature crowd sensing, which significantly limits crowd sensing-based approaches by enforcing the participant to follow specific instructions. To address this problem, a lightweight filtering approach is proposed and implemented to detect anomalous placement of temperature sensors. By using low-cost temperature sensors, the proposed technique can be leveraged into various crowd sensing frameworks which focus on collecting different environmental features.

The third problem is the budget limitation in drive-by sensing approaches, where there is a limited number of sensors, and we need to select a subset of vehicles to mount the sensors to maximize the sensing coverage. To address this issue, we formulated the problem as an optimization problem. Then, our cost-aware approach is proposed and implemented to efficiently select a subset of buses for sensor mounting to maximize the spatiotemporal sensing coverage, considering that there are some dynamic hotspots in the cities where their continuous sensing is of greater importance compared to other areas.

1.4 DISSERTATION ORGANIZATION

Chapter 2 provides background material and related works about Urban Heat Islands and different sensing paradigms. Chapter 3 focuses on our framework, the implementation details, and its preliminary results. In chapter 4, the anomalous

sensor placement, which is a prevalent problem in environmental crowd sensing applications, is discussed. It is followed by explaining our proposed approach to effectively detect and filter these anomalies. Chapter 5 focuses on enhancing spatiotemporal coverage in drive-by sensing, where we propose our near-optimal cost-aware approach. The dissertation concludes in chapter 6 with a summary of this study, followed by future directions for this research.

Chapter 2

MOTIVATION, BACKGROUND, AND RELATED WORK

2.1 URBAN HEAT ISLANDS

Urban Heat Island (UHI) is an urban area that is noticeably warmer compared with the adjacent rural areas due to urbanization. This phenomenon was first described by an amateur meteorologist named Luke Howard about two hundred years ago, although he was not the one to name the phenomenon. He published his findings in a book titled The Climate of London in the 1810s [Howard, 1820]. Different factors are contributing to a city's UHI such as the increased surface area of buildings, decreased moisture and vegetation, greater heat capacities of

Figure 2.1: Urban Heat Island

building material, and anthropogenic heat waste from vehicles and different types of machinery [Taha, 1997]. The heat absorbed during the day is radiated out at night, which ultimately leads to an increase in the relative nighttime temperatures [Kuras et al., 2015]. Therefore, this phenomenon has altered the urban climate not only during the daytime but also in the nighttime.

Although the UHI has been widely studied in large cities around the world, findings show that this phenomenon also exists in smaller scales [Buckley et al., 2008, Grathwohl et al., 2006, Pinho and Orgaz, 2000]. To make it more clear, Figure 2.1 depicts the UHI phenomenon. We can see that the temperature experienced in urban areas are higher than the suburban and rural areas. The focus of most studies has been on the orange semicircle of this figure, while the red line which depicts the variations in smaller scales has been underestimated in the research community. One of the main reasons is the fact that low-resolution satel-

lite imageries have been the primary source to generate heatmaps and to analyze the UHI phenomenon. Therefore, there is a need to leverage different technologies which enable us to capture the temperature variations in high resolution and to perform hyperlocal analysis on the urban heat phenomena.

2.2 REMOTE SENSING METHODOLOGIES

Remote sensing is the primary sensing paradigm to analyze UHIs. Different models have been introduced to derive the Land Surface Temperature (LST) using satellite data [Jin and Dickinson, 2010]. In these methods, researchers use the thermal bands of the satellite imagery that captures the reflectance data from the earth. Then, they perform atmospheric correction procedure [Hadjimitsis et al., 2010], which is the process of removing the effects of unsteady atmosphere on the reflectance values of the images taken by satellite sensors. Finally, they use existing land cover/ land use (LCLU) maps along with the emissivity (a measure that shows the effectiveness of emitting energy as thermal radiation) values associated with different land surfaces to derive the heat maps of different geographical areas.

There are different satellites with Thermal Infrared Sensor (TIRS) that provide images with thermal bands to be used as the input for the heatmap generation models [Kuenzer and Dech, 2013]. For example, Landsat 8 satellite which has been widely used by the research community has TIRS with two spectral bands in the longwave infrared region [NASA, 2019]. It provides images with 100-meter spatial resolution every sixteen days for any given area on the earth. The Advanced

Spaceborne Thermal Emission and Reflection Radiometer (ASTER) [JPL, 2016] and the Moderate Resolution Imaging Spectroradiometer (MODIS) [EOS, 2013] are two other satellite sensors with 90-meter and 1000-meter spatial resolution, and 16-days and daily temporal resolution, respectively. All these images have been the primary data source for remote sensing researchers to apply their models and to generate heatmaps.

Yuan et al. [Yuan and Bauer, 2007] used Landsat Thematic Mapper (TM) and Enhanced Thematic Mapper Plus (ETM+) data to estimate the LST of different seasons for Twin Cities in Minnesota. They compared the normalized difference vegetation index (NDVI) and percent impervious surface as indicators of surface urban heat island effects in Landsat imagery by investigating the relationships between the land surface temperature (LST), percent Impervious Surface Area (%ISA), and the NDVI. Their findings show a robust linear relationship between the LST and %ISA, which suggests that impervious surface area accounts for most of the variation in land surface temperature dynamics. Although their study showed impressive results, their conclusion is based on only one area and four different dates.

In another study, based on the Landsat 8 images, authors [Tan et al., 2017], used the mono-window algorithm proposed by Qin et al. [Qin et al., 2001] back in 2001 as the base algorithm to make some improvements. The improved algorithm is based on the radiative transfer equation, which states that the sensor-observed radiance is always impacted by atmospheric transmittance and ground emission. Other than coming up with new parameters for the original equation, they recal-

culate the land surface emissivity values of the original study. In the same study, Tan et al., modified the algorithm proposed by Mao et al. [Mao et al., 2005] to make a comparison between the two modified algorithms.

We implemented both models and compared the results with our in situ-based data collection happened at the same time when satellite images were taken. Although the Qin's model worked better for the few cloud-free dates compared to the Mao's algorithm, we could not find any consistency between the heatmaps generated by the models and the on-the-ground readings using either Industrial Infrared (IR) guns or our temperature data loggers. For example, when we ran the Qin's model on the satellite image of 27 July 2017, although it was a cloud-free day, the heatmap showed temperatures ranging from 0 °C to 22 °C, while the weather station's sensor showed around 31 °C. Figure. 2.2 shows a part of the heatmaps generated by the two discussed models, and we can observe how the two models produced heatmaps differently.

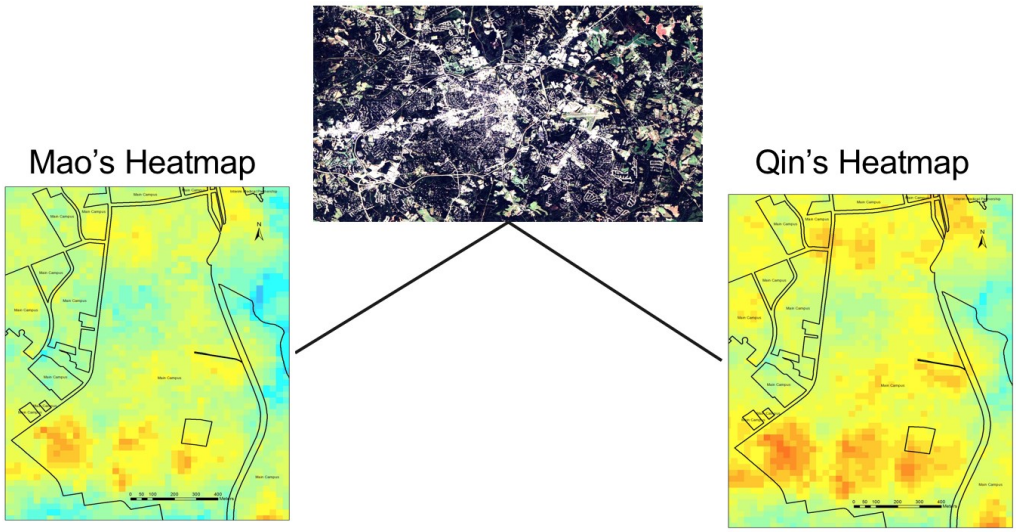


Figure 2.2: Heat Map Retrieved by Different Models

2.3 IN-SITU-BASED METHODOLOGIES

In in-situ-based techniques, researchers mainly rely on readings from weather stations to measure the temperature of different areas, and the spatial resolution of the network of weather stations is their main limitation. The ambient air temperature can be affected by microclimate differences due to the density of vegetation, buildings material, and street orientation. Thus, a limited number of weather stations cannot capture the heterogeneity of ambient air temperature. A study [Kuras et al., 2015] suggests that heterogeneity in heat exposure exists even within an urban neighborhood. In another study, authors [Bernhard et al., 2015] raised the same issue of the difference between readings from the nearest weather station and the experienced individual heat exposure.

In order to better understand the spatial variation of the ambient air temperature, we collected data on different dates using standard temperature sensors during the year of 2017. For instance, we conducted an experiment on 10 July 2017 when 25 temperature sensors (Kestrel DROP environmental data loggers with $0.5\,^{\circ}\mathrm{C}$ accuracy) were installed in an area of about 0.16 km^2 at the University of Georgia (UGA) campus for 90 minutes in the morning between 9:45 AM and 11:15 AM. For accurate measurement of the ambient air temperature, sensors were installed about one meter above the ground over different surfaces such as asphalt, grass, sand, and concrete. Figure 2.3 illustrates the exact location of each sensor on the map.

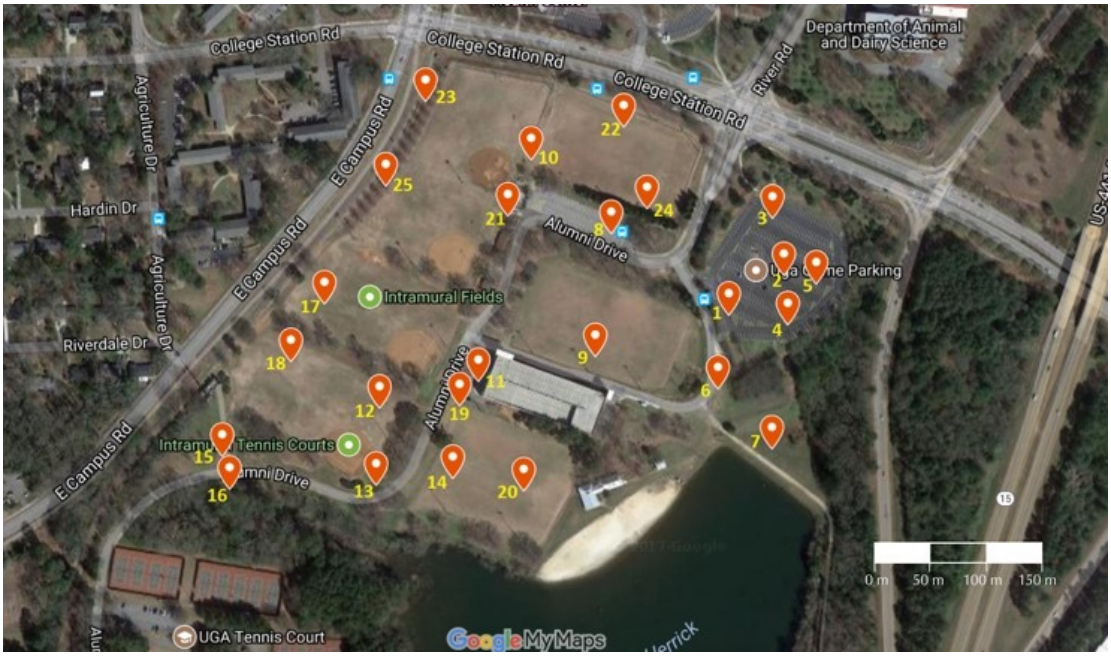


Figure 2.3: Temperature Sensors Locations

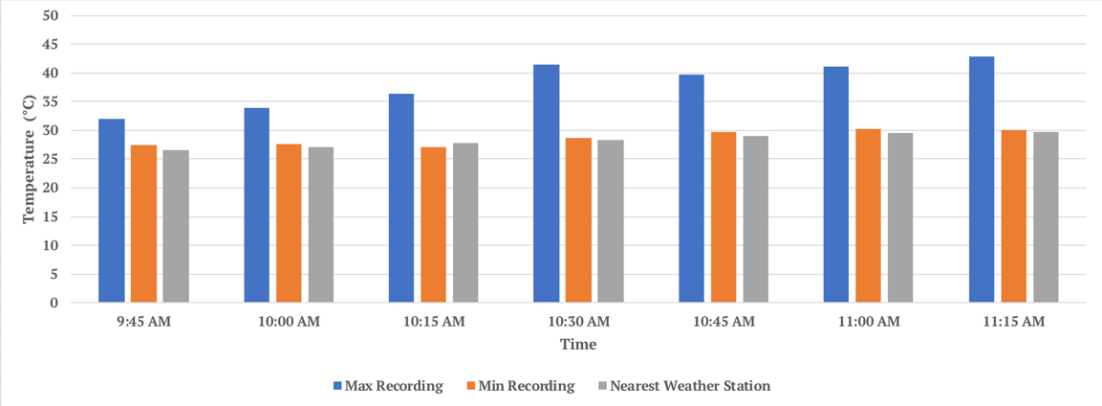


Figure 2.4: Highest and Lowest Sensor Recoding vs Nearest Weather Station Reading

Collected data shows the high variation of air temperature within a small area. On average, there was a 9.5 °C difference between the highest and the lowest recorded temperature at the same given points of time. As depicted in Figure 2.4, we also compared the sensor readings with that of the nearest weather station (1.75 km away from the study area), and we observed an average of 5 °C difference between the recorded mean temperatures. These measurements represent the high variation of ambient air temperature in response to different factors such as the land cover or shading, which is not obtainable by a limited number of weather stations.

However, it is impractical, at least in the near future, to have tens of thousands of sensors installed on different parts of any given area because of the exorbitant cost that it would impose. Therefore, crowd sensing technologies emerged to fill this gap. For this purpose, Michael Goodchild [Goodchild, 2007] introduced the term Volunteered Geographic Information (VGI), as a particular case of mobile

crowd sensing. VGI, which utilizes the user-generated geospatial contents, is even more prominent when monitoring of highly dynamic features such as the ambient air temperature is required.

2.4 CROWD SENSING

The proliferation of mobile devices in recent years has enabled smart crowd sensing approaches which could be used as a practical approach for the UHI studies. Mobile Crowd Sensing (MCS) term was first introduced by Ganti et al. [Ganti et al., 2011] as an application of the Internet of Things, where the individuals who carry sensing devices share some data to measure and map a phenomenon. Based on the type of phenomena being monitored, MCS can be classified into two categories of individual and community-sensing. In urban heat analysis domain, individual heat exposure map is the result of the first category, while integrated urban heat exposure map pertains to the second category. Community-sensing, which is also referred to as participatory sensing [Burke et al., 2006], has been used in different environmental studies. For example, Hasenfratz et al. [Hasenfratz et al., 2012] implemented GasMobile as an air quality measurement system. They attached low-cost sensors to mobile phones to measure the ozone concentration of the air and consequently create air pollution maps.

Crowd sensing has many advantages over traditional sensor networks. First, due to the mobility of users, broad areas can be covered [Zhang et al., 2015]. Secondly, more often than not, there is no need to deploy and maintain sensors as

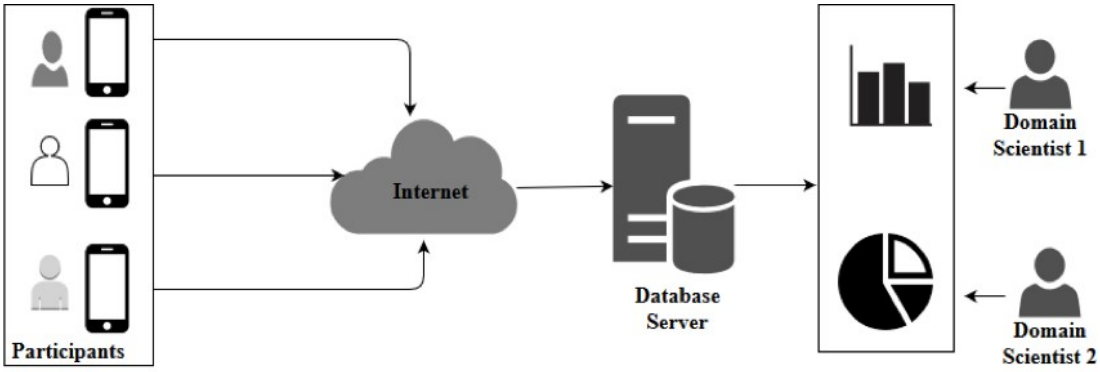


Figure 2.5: Conventional Model for Crowd Sensing Application

they are integrated into mobile phones [Aoki et al., 2008]. Lastly, the availability of software development tools for mobile phones makes application development and deployment relatively easy [Heggen et al., 2013, Kanhere, 2011]. Advantages of crowd sensing have led to an increase in mobile sensing applications. On the other hand, there are some challenges associated with the design and implementation of such applications. Data quality, filtering, anomaly detection, effective and adaptive sensor installation, and resource consumption management are different research challenges in various crowd sensing frameworks.

A conventional model for crowd sensing applications is depicted in Figure 2.5 where the mobile devices carried by participants send the data to the server, which later on will be used by domain scientist to study the target phenomena. In these applications, domain scientists usually specify their areas of interest on the map so that the crowd sensing agents would know which areas are targeted for sensing purposes [OGrady et al., 2016].

One of the best examples of crowd sensing in the context of heat monitoring is the project done by researchers at Wageningen University in Netherlands [Overeem et al., 2013]. They proposed a crowdsourcing method to measure air temperature using smartphone battery temperatures. They failed to take the fact into account that temperature of smartphone batteries can be impacted by many different factors, including the phones model, screen size, number of running applications, and the OS version. They assumed that phones are usually carried inside the pockets close to the users body and introduced a heat transfer model. Their model is based on some certain assumptions which make it limited and non-scalable.

In another studies, [Chapman et al., 2017] and [Meier et al., 2017] implemented two different case studies by using low-cost weather stations to quantify the urban heat island of London and Berlin, respectively. They used the network of amateur weather stations called Netatmo, which is configured to monitor the temperature of the local environment. Although they aimed to motivate atmospheric scientists towards the use of crowdsourcing in urban meteorology, they only used static sensors. Their study fails to measure and analyze the community-centric and personalized heat exposure. More importantly, their approach is limited to a network of specific stationary sensors, which makes it non-scalable. Also in the first study [Chapman et al., 2017], they did not have access to metadata to analyze the quality of their crowd-sensed data (e.g., quality of instrumentation and station location).

2.5 DRIVE-BY SENSING

Drive-by sensing was first introduced as a new network paradigm and predicted to be a viral sensing approach. This sensing technology, which adopts different vehicles as its sensing agents, is very practical in cases where a vast number of sensors are required to be deployed in an area, or the sensors are very costly [Hull et al., 2006, Lee and Gerla, 2010]. The prevalence of portable sensors and ubiquitous devices had made this platform even more attractive due to its cost-effectiveness and the unique sensing opportunities. This approach can either employ cars [Hull et al., 2006], buses [Gao et al., 2016], taxis [Eriksson et al., 2008], or vans [Li et al., 2016]. Considering that many vehicles are commuting all around the cities, this platform has many potentials in various urban studies.

Drive-by sensing agents could be categorized into two groups of dedicated and non-dedicated vehicles [Genc et al., 2013, Habibzadeh et al., 2017]. Dedicated vehicles are driven solely for data collection purposes, such as the ones used in Google Street View. On the other hand, non-dedicated vehicles are the ones that have their own schedule, such as city buses. For instance, in the City Scanner project [Anjomshoaa et al., 2018], portable sensors are deployed on top of the existing garbage trucks to collect data in Cambridge, Massachusetts. Considering that the schedule of the hosting vehicles in the latter approach remains unaltered, it would be a much cheaper choice compared to having dedicated vehicles.

We incorporated drive-by sensing paradigm into our framework and mounted our sensors on 30 different buses of Athens Clarke County Transit and UGA Campus Transit, during the Summer of 2018 and 2019. We also mounted our sensors

on the 5 different shuttles of Arizona State University (ASU) for two weeks during August 2018. The ASU shuttles travel around the Tempe and Phoenix area in Arizona. The two images in Figure.2.6 show Athens buses and ASU shuttles, respectively.

Figure 2.6: Athens Clarke County and Arizona State University Buses

2.6 CHAPTER SUMMARY

Considering that remote sensing approaches fail to capture the highly dynamic nature of urban heat islands, some researchers tried to incorporate in situ-based methodologies to overcome the limitations. In these studies, researchers mainly analyze the readings from weather stations to measure the temperature of different areas. Most of these stations provide data every a few minutes, but the spatial resolution of the network of weather stations is their main limitation.

There are a few studies trying to measure and analyze the urban heat tem-

perature through crowd sensing. [Overeem et al., 2013] proposed a crowd sensing method to measure air temperature using smartphone battery temperatures. Their offered method suffers from inherent problems. Their model is mainly based on certain assumptions, which makes it non-scalable. Two other studies, [Chapman et al., 2017] and [Meier et al., 2017] failed to measure and analyze the community-centric and personalized heat exposure. More importantly, their approach is limited to a network of specific stationary sensors, which makes it non-scalable.

In summary, considering that each sensing approach has its own shortcomings, there is a need to come up with a comprehensive and hybrid approach to leverage the synergy provided by different sensing technologies.

Chapter 3

SCOUTS FRAMEWORK

Chapter Overview[1]

In this chapter, we present a multi-layer approach to tracking the actual heat experienced by individuals and communities with very high spatiotemporal resolution. The proposed framework, Smart Community-centric Urban Thermal Sensing (SCOUTS), seamlessly support a variety of human, and vehicle-borne sensors in conjunction with satellite and weather station data to accurately map the heat hazards of urban regions and communities.

[1]This chapter partially appears as:
N. H. Tonekaboni et al., "SCOUTS: A Smart Community Centric Urban Heat Monitoring Framework." In Proceedings of the 1st ACM SIGSPATIAL Workshop on Advances on Resilient and Intelligent Cities. ACM, 2018.

3.1 FRAMEWORK OVERVIEW

The overarching goal of this framework is to efficiently integrate and analyze data from multiple diverse sources such as human, and vehicle-borne sensors, satellites and weather stations for effective identification and tracking of heat stress risks of individuals and urban communities. Mobile Crowdsensing is an essential component of the SCOUTS as it helps us gather data at much finer spatiotemporal granularities compared to traditional sources. Since the framework has to support highly heterogeneous data sources seamlessly, it adopts the concept of data virtualization. The key strength of data virtualization is that it allows the users of the framework to retrieve and analyze data from diverse sources without needing to know the technical details about the data or its source. The framework is composed of three layers, namely physical, virtualized data, and modeling/ notification layers. Figure .3.1 shows the high-level architecture of this framework based on a multi-layer bottom-up model.

The physical layer is composed of the actual data sources such as temperature sensors, mobile devices, satellites, and weather stations. SCOUTS is designed to support a variety of small temperature and humidity data loggers (such as Kestrel DROP) which can be attached to clothing or backpacks. We have developed iOS and Android applications to communicate with the sensors and upload the data to the cloud. Weather stations are another data source providing us with readings of environmental features such as temperature and humidity. GPS information, temperature, and humidity are continuously uploaded to the virtualized data layer.

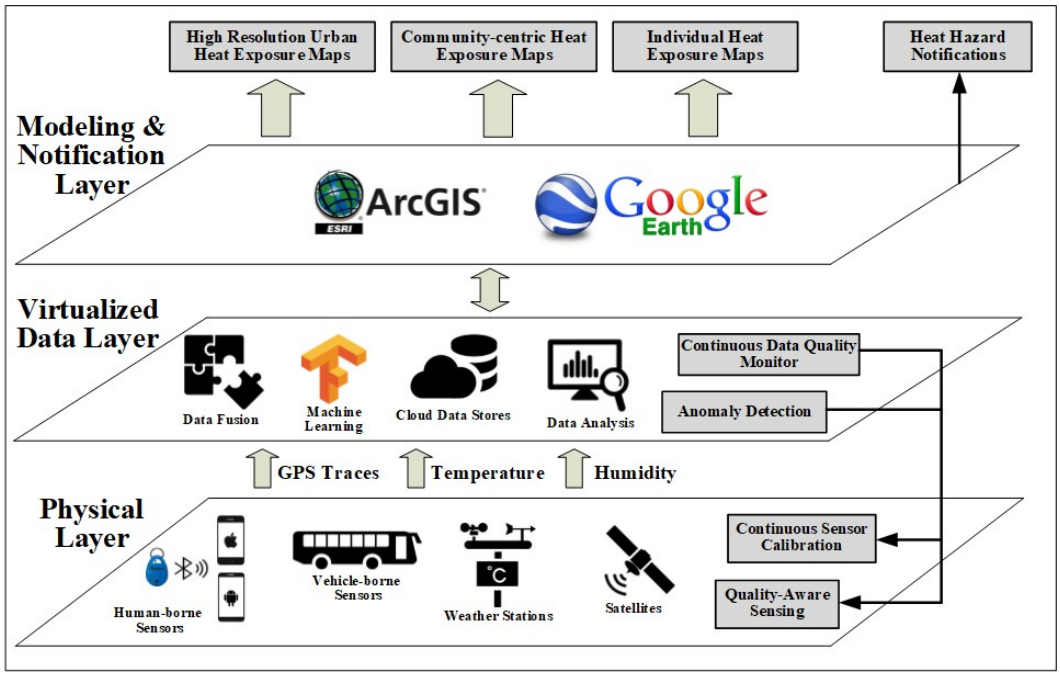


Figure 3.1: High level architecture of the SCOUTS framework

Virtualized data layer consists of cloud-based data stores along with data analytics and data fusion tools. In conjunction with cloud storage which ensures data accessibility, data analysis tools enable spatiotemporal data mining. Access to frequent arrays of geo-located and time-stamped data enables this layer to send quality-aware sensing feedbacks and calibration information to the physical layer. For instance, based on the heat hazard probability of a given area, configuration information will be sent to the physical layer to update the data collection frequency of sensors accordingly. Moreover, this layer integrates data streams coming from heterogeneous sources and perform anomaly detection techniques to assure

the integrity of data.

The third layer is the modeling and notification layer, to which the analyzed and filtered data is fed. Data plotting and visualization are the two main tasks of this layer which yields urban heat maps, individual, and community-centric heat exposure maps with high spatio-temporal granularities. It also generates heat hazard notifications, sending alerts to individuals or communities who are at the risk of extreme heat events.

There are some research challenges associated with the full implementation of this framework. Other than the data quality measures and calibration methods, mechanisms to detect misplacement of sensors and filter out the sensor readings, which are not the true representative of the outdoor air temperature is of crucial importance. In other words, the sensors which are not exposed to the outdoor environment in the crowdsensing procedure needs to be detected and filtered out. Community-centric clustering of heat stresses is another challenge where organizations can devise insightful strategies to mitigate heat hazard effects on their respective communities. Considering that the data from this framework correlates with health-related issues, a new level of privacy consideration, beyond anonymity, is required so that businesses such as insurance companies will not be able to take advantage of the heat exposure profile of communities to impose higher health insurance rates. Incentivization mechanism is the other challenge to motivate volunteers to participate, especially in the areas with more potential extreme heat events.

3.2 IMPLEMENTATION DETAILS

3.2.1 Crowd Sensing: Mobile Applications and Kestrel DROP Sensors

For the crowd sensing section of our framework, we used Kestrel DROP sensors, which are small and accurate environmental data loggers. As shown in Figure 3.2, these sensors can be easily attached to the backpack or the belt loop of volunteers to collect data. These waterproof sensors use Bluetooth Low Energy (BLE) to communicate and have 0.5 °C accuracy with 0.1 °C resolution. Although these sensors can efficiently work with Kestrel Connect mobile applications to report the data, there are some limitations associated with them. First, they do not provide location information, which is an essential element of crowd sensing to map each temperature reading to its location. Secondly, data reporting of their mobile application is manual, meaning that each user needs to send the data via email or text manually. Then, they have to clean the memory; otherwise, the memory of the sensor gets full, and they stop working. Lastly, participants are allowed to change the configurations such as the logging frequency, or even disconnect from a sensor and connect to another Kestrel sensor within their Bluetooth range. Having access to these configurations make the data collection process much harder because extra efforts are required to train the users and ask them to follow the instructions strictly.

In order to solve the limitations mentioned above, we developed our own iOS and Android mobile applications (Figure 3.3). These mobile applications connect

Figure 3.2: Kestrel DROP human-borne sensors

to the Kestrel sensors through BLE, read the data from them, get the GPS data from the mobile devices, integrate the location information to each temperature data point, and finally send the data to our server. For our server, we have used Google Cloud Platform and implemented the services to work with the mobile applications using Java Servlet. Our applications are designed to be very simple, and no configuration option is given to the users so as to minimize participants' intervention in the data collection process. In our implementation, admin sets the required configurations such as the data logging frequency through our server. Each participant is provided with a sensor, and only for the first time, the app asks for their sensor ID. After that, participants simply need to open the app, and it automatically connects to the sensors to start the data collection process. The applications do not require the mobile devices to have an active Internet con-

nection; they store the data on the devices until there is network connectivity to send the stored data to the server.

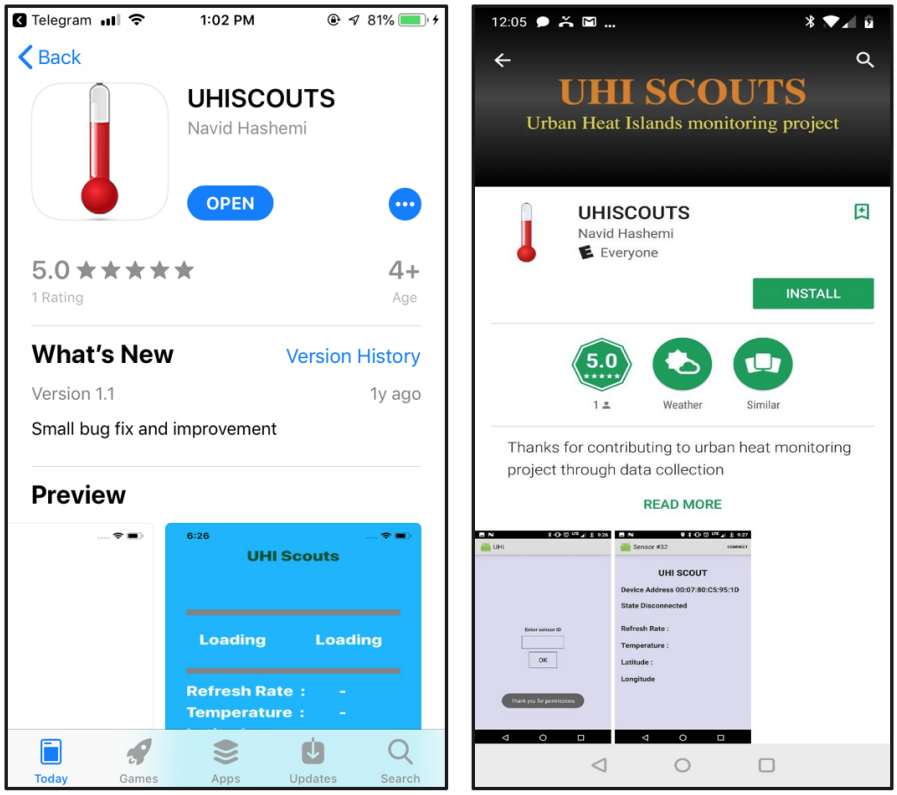


Figure 3.3: SCOUTS iOS and Android Mobile Applications

3.2.2 Drive-By Sensing: Vehicle-borne sensors and GPS Data

For drive-by sensing, we designed and assembled our own standalone temperature sensors with enhanced battery and memory. Our DIY sensors are constructed using Arduino microcontroller boards, DS18B20 1-wire digital temperature sensors with 0.5C accuracy, low-power GPS FeatherWing boards, and lithium-ion batteries. Figure 3.4 shows the process of soldering, assembling, and mounting our sensors on the buses. Considering that the engines are in the back of the buses, we found the bike racks in front of the buses to be the best place to mount our sensors.

Our vehicle-borne sensors worked pretty well; however, we learned that having the GPS sensors active all the time to get the location data every 5 seconds is pretty battery intensive. To solve this problem, we implemented a web service architecture, as shown in Figure 3.5, to minimize the number of calls to the GPS sensor. For this purpose, we called the APIs provided by the two companies, Avail Technologies and GMV Synchromatics, which provides integrated transportation solutions for Athens Transit and UGA Campus Transit, respectively. SCOUTS call their services through HTTP requests, then their reply in JSON format is fed into SCOUTS endpoint. In the next step, the JSON data is parsed and inserted into a MySQL database every five seconds. We used Apache web server to host our Java code.

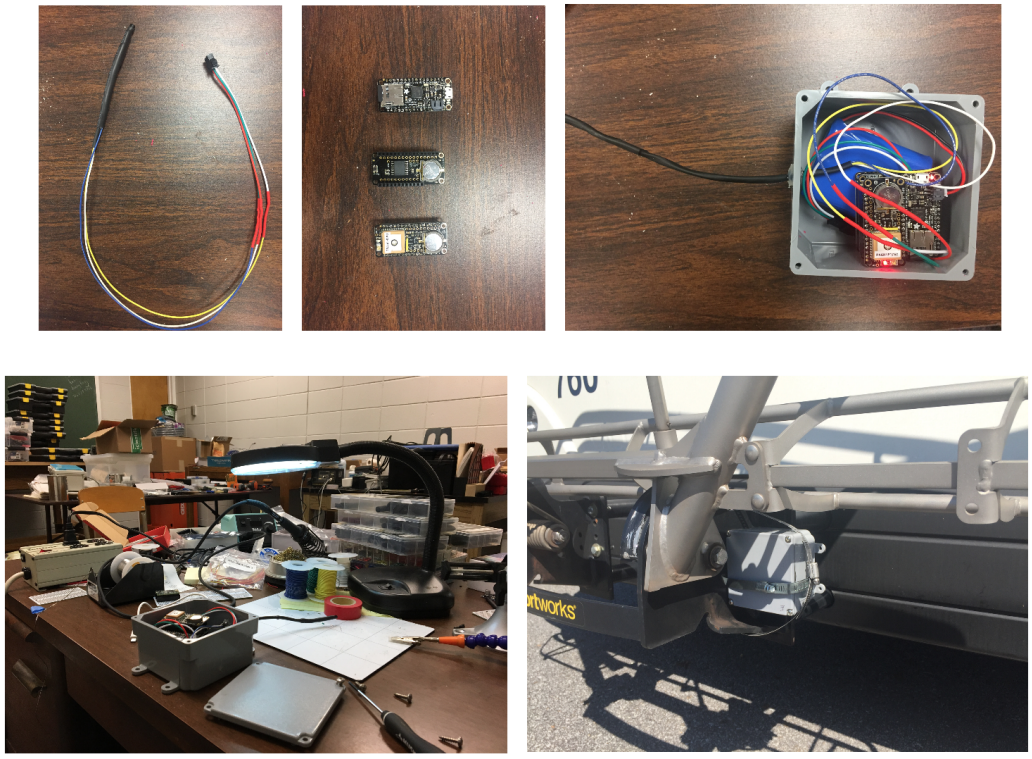


Figure 3.4: DIY Vehicle-borne Sensors

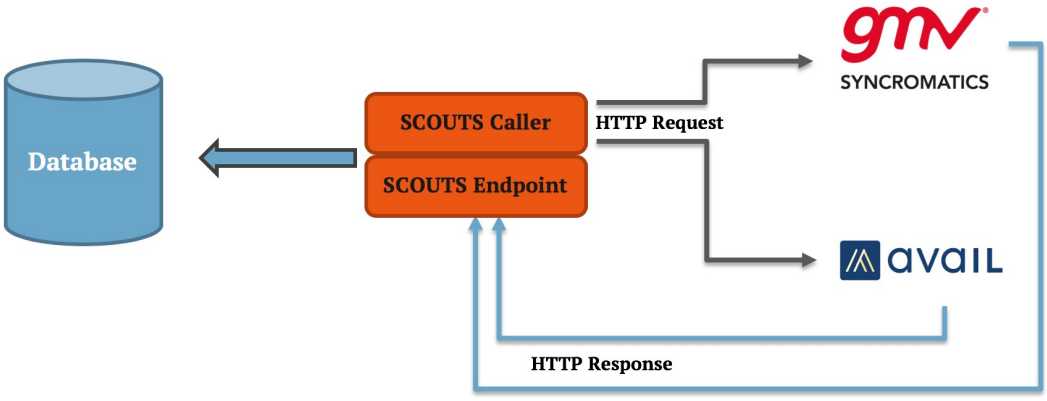


Figure 3.5: Web Service Architecture for GPS Data Collection

In order to support extensibility as an essential principle of software design, we used REST service so that different components such as mobile applications in crowd sensing can be easily replaceable. Another reason to choose REST over SOAP is its simplicity. In our implementation, we only used GET method of the REST service to retrieve a simple textual data in short intervals. Therefore, the current implementation can be easily extended for future developments.

Using this solution, we have extended the battery life of our drive-by sensing sensors from five days to more than twenty days. We still use the GPS boards in our sensors; however, we call them once every six hours to make sure the clock is synchronized. In other words, the GPS sensors are used to make sure all the geo-tagged data from buses precisely match the temperature data using their timestamps.

3.3 PRELIMINARY RESULTS

To have a comprehensive understanding of the heat hazards and the effectiveness of such a crowd sensing framework, we had a couple of rounds of data collection. The most cloud-free day we could find was 28 Sept 2017, when four individuals were walking around the UGA campus with sensors attached to their backpacks around noon time. The heat exposure map in Figure.3.6 shows different walking routes and the experienced air temperature by individuals during the 45-minute course of data collection. The range of temperature experienced by each person shows the high spatio-temporal heterogeneity in the air temperature.

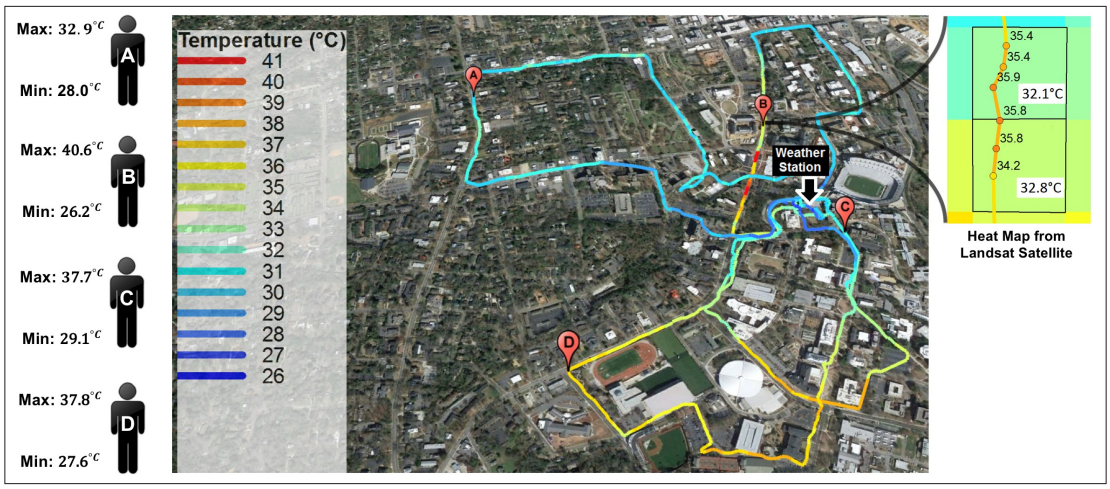


Figure 3.6: Temperature Map of 28 Sept 2017

The only weather station in the area is located on the map, as well as the exact location of each person at 12:05 PM (the time when Landsat 8 satellite image was taken). Although the individuals experienced up to about 14 °C variation in the highlighted routes, the stationary weather station recorded only around 1 °C variation during the same period. The experiment underlines the unreliability of weather stations for measuring the precise individually experienced temperature, which is a crucial factor in heat-related health issues.

Now, we compare the crowd-sensed data with the satellite-based heatmap data. Figure.3.6 shows the zoomed location of person B in the heat map generated using Landsat 8 image. While each 30-meter by 30-meter re-sampled pixel represents a single temperature value, the actual recorded temperature values are denoted on the yellow line. Although this cloud-free experiment shows around 3 °C difference between the two approaches, there is no consistency in this dif-

ference among different locations and dates. Therefore, there is an uncertainty in satellite-based heatmaps, and they may not represent the precise temperature values. For instance, we recorded up to 21 °C difference between the temperature values in satellite-based heatmaps as opposed to the crowd-sensed readings.

In the left diagram of Figure. 3.7, we make a comparison between the temperature values derived from the crowd sensing approach (the four volunteers), the remote sensing-based approach (the heat-maps generated from Landsat 8 satellite, at 12:05 PM), and the in-situ-based approach (the nearest weather station) in a 30-minute window of 15 minutes before and after the time when the satellite image was taken. The Landsat readings correspond to the cell values of the heatmaps where each volunteer was at 12:05 PM, exactly when the satellite image was captured. For instance, the temperature recorded by person D was about 37 °C, while the satellite-based heat-map shows 33 °C for that location. Although the latter represents LST, the temperature difference is considerable. Another interesting observation is the temperature range experienced by each individual where there is no consistency in what different individuals experienced. While person A experienced around 2 °C variation in the 30-minute period, person B experienced a temperature range of around 9 °C.

The right diagram in Figure. 3.7 represents the average temperature experienced by the four volunteers walking around the campus compared to the average temperature reading of the weather station in the same 45-minute period. This graph clearly shows that a static weather station cannot accurately capture the variability of the experienced temperature by individuals in an area. While one of

the volunteers experienced a slightly lower temperature compared to that of the weather station, three other volunteers were exposed to a higher temperature.

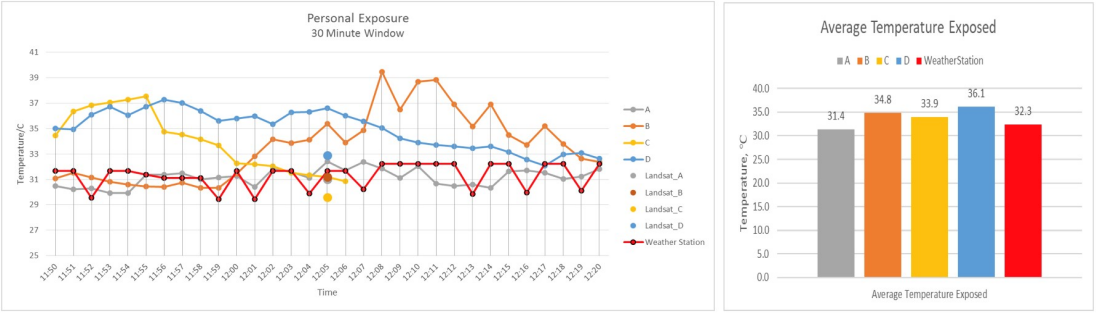
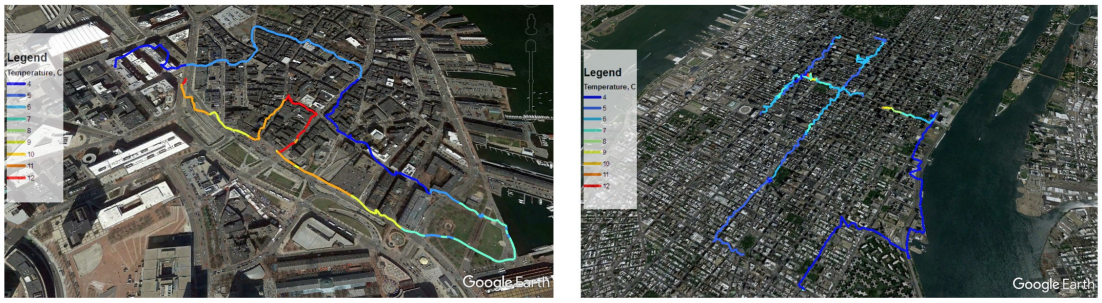


Figure 3.7: Comparison between Weather Station Readings, Landsat8 Heatmap, and the Actual Heat Exposure

We have also had the same observation of having distinct temperature readings from the nearest weather stations and the experienced individual heat exposure. The two maps in Figure 3.8 depict the variation in experienced temperature in New York City and Boston, respectively. Similarly, we observed that the nearest weather stations were not able to capture the actual temperature variation experienced by people in their daily life.



Boston Downtown, 21 Dec 2017, 1:40-2:34 PM

New York City, 24 Dec 2017, 3:20-9:38 PM

Figure 3.8: Temperature Crowd Sensing in Boston and New York City

Furthermore, we have integrated and analyzed our drive-by sensing data for different locations and dates so as to demonstrate the scalability of our framework. The following results generated by comparing drive-by sensing data and remote sensing-based data restate our claims that a comprehensive and hybrid framework like SCOUTS is needed to be able to accurately analyze different urban heat phenomena with high spatiotemporal resolution.

Figure 3.9 enables us to make a comparison between the heatmap generated from Landsat 8 satellite imagery and 20 minutes of drive-by sensing data (10 minutes before and after the time when the satellite passed the Athens area) on 14 August 2018. It is clearly shown that on this date, the satellite-based heatmap represents a higher temperature compared to the vehicle-borne sensors, which were placed very close to the land surface. Considering that the buses were consistently on the roads, all their data belong to the same land cover type of asphalt. Therefore, we tried to find a correlation between the drive-by sensing data and that of the satellite-based data; however, we could not find any consistent relationship between them. In Figure 3.10, we can see two zoomed-in areas of the same map of Figure 3.9. For instance, while the satellite-based heat-map cells of the left figure shows 39 °C, the drive-by sensing readings from the same cells are 5 °C to 9 °C lower.

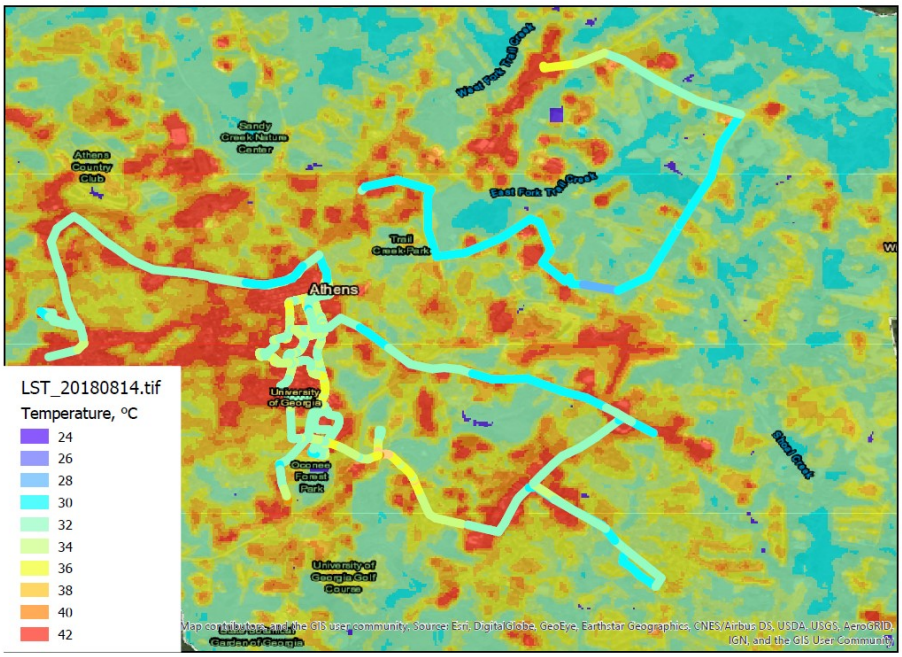


Figure 3.9: Drive-by Sensing Heatmap vs Remote Sensing-based Heatmap of Athens on 14 August 2018

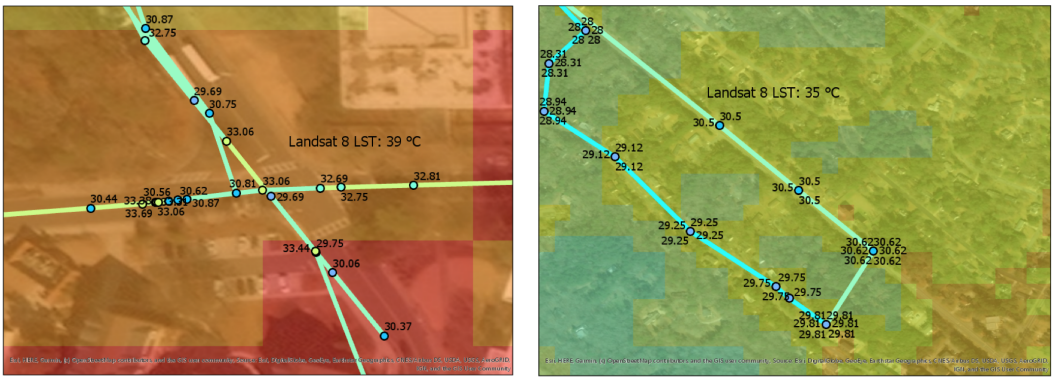


Figure 3.10: Drive-by Sensing Data Points vs Remote Sensing-based Heatmap Cells in Athens

In order to show that our observations do not belong to a given geographical area, we also mounted our drive-by sensing sensors on the shuttles of Arizona State University. Figure 3.11 generated by comparing the satellite-based heatmaps and the drive-by sensing data collected from Arizona. Figure 3.11 shows an area in Tempe, close to the Arizona State University campus. Similarly, the drive-by sensing data in the figures belong to 10 minutes before and after the time when the satellite passed the Tempe area. In this area, there was only one bus with sensors, which is highlighted on the map. Although it is not always the case, in this date we again saw that the satellite-based data represent higher temperature readings compare to drive-by sensing data.

Figure 3.12 shows two zoomed-in locations within the same area. These heatmaps show that our claim about inconsistency in different sensing approach applies to other geographical areas while they may have completely different land cover types. For instance, the satellite-based heatmap cells on the right image show the temperature of 50 °C, while the drive-by sensing readings within the same cells reported around 10 °C lower temperature values. It should be mentioned that the color legend used for Figure 3.12 is the same the one used for Figure 3.11.

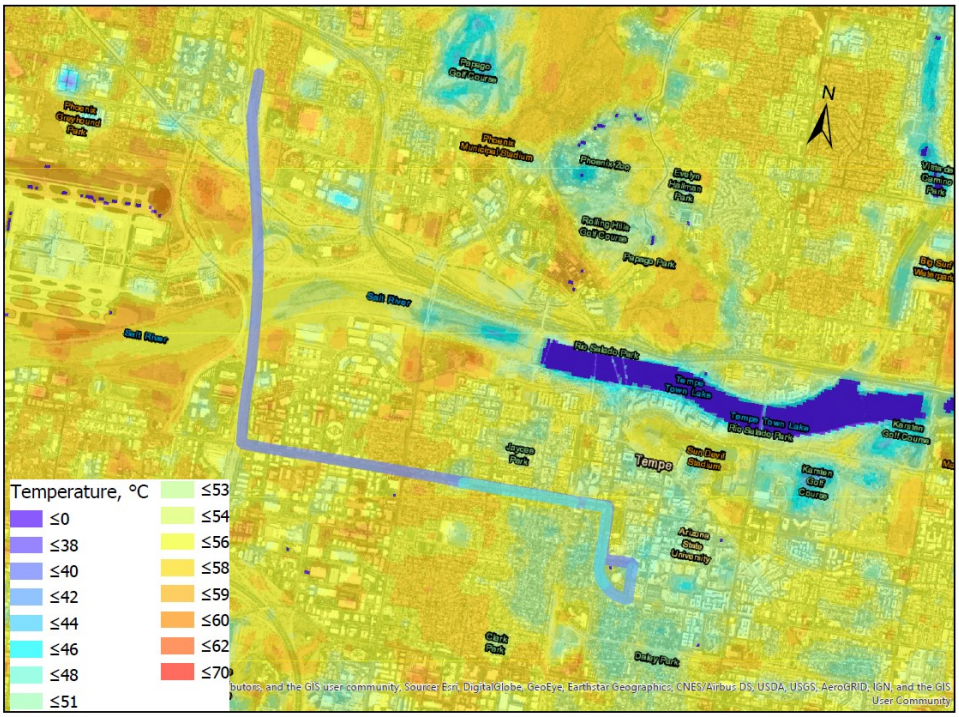


Figure 3.11: Drive-by Sensing Heatmap vs Remote Sensing-based Heatmap of Tempe on 3 August 2018

Figure 3.12: Drive-by Sensing Data Points vs Remote Sensing-based Heatmap Cells in Tempe

3.4 CHAPTER SUMMARY

Extreme heat events are increasingly becoming a threat to human health and wellness in many countries around the world. It is of crucial importance to analyze and mitigate them in a practical way using the available technologies. In this chapter, we first demonstrate the limitations of current approaches to quantify heat hazards. Then, we present the SCOUTS framework, to address the limitations, and to create heat exposure maps with high spatio-temporal resolution. SCOUTS helps city officials to understand the heat variation in urban areas better and devise mechanisms to make the cities more resilient. Finally, we present preliminary results to show the effectiveness of such a framework to analyze and mitigate urban heat hazards.

Chapter 4

ANOMALY DETECTION IN CROWD SENSING

Chapter Overview[1]

Although crowd sensing is a powerful sensing paradigm, we identified an essential limitation in sensing environmental features. There is no mechanism to detect the wrong placement of sensors in crowdsensing, and this problem significantly limits the applications of this sensing paradigm by enforcing participants to follow strict instructions. In other terms, if users misplace the sensors in a way that they are not exposed to the natural outdoor environment, they report wrong information to the system. Therefore, there should be a mechanism to detect and filter out the erroneous data. To resolve this problem, we developed a new ap-

[1]This chapter partially appears as:
N. H. Tonekaboni et al., "Edge-Based Anomalous Sensor Placement Detection for Participatory Sensing of Urban Heat Islands." 2018 IEEE International Smart Cities Conference (ISC2).

proach to detect wrongly-placed temperature sensors in a semi-real-time manner. We introduce a sliding window technique in conjunction with supervised learning classifiers to detect anomalously-placed sensors. This approach is based on the empirical observation that temperature readings show more frequent fluctuations while exposed to the outdoor environment. We conduct a series of comparative performance analysis on different classifiers, including SVM, Logistic Regression, and Random Forest. Our approach can be integrated into different participatory sensing applications by adding low-cost temperature sensors.

4.1 INTRODUCTION

The recent proliferation of mobile devices has enabled crowd sensing (also referred to as participatory sensing) as a paradigm for sensing environmental phenomena [Ganti et al., 2011]. Crowd sensing can facilitate environmental monitoring studies such as ambient air temperature, humidity, noise, and air pollution analysis. Although crowd sensing research studies have considerably increased due to the ever-increasing prevalence of low-cost sensors, data integrity has become one of the major challenges. Sensor failure, transmission error, and infrequent system behavior can affect the integrity of data. Therefore, there is a need for effective mechanisms to ensure the quality of data coming from multiple resources.

Urban Heat Islands (UHI), which is the result of changes in urban climate compared with adjacent rural areas due to urbanization, has been an issue of increasing significance in recent years. Consequently, environments have been trans-

forming from native vegetation to human-made infrastructure with much higher thermal-storage capacity [Luber and McGeehin, 2008]. Even at a smaller scale, some city blocks or buildings can create their own UHIs. Factors contributing to these heated areas are the increasing surface area of buildings, anthropogenic heat waste from vehicles, the higher heat capacity of building materials, and decreasing vegetation.

Most of the current approaches use satellite-based remote sensing and weather stations for monitoring and analyzing UHIs. These data sources fail to capture the highly dynamic nature of this phenomena due to their limitations, such as cloud coverage and low spatiotemporal resolution. Recently, researchers have started to explore human and vehicle-borne sensors as a newer approach to augment traditional data sources.

A major challenge of crowd-sensed temperature data is that sensors might not be situated according to the given instructions [Kuras et al., 2017]. For example, a temperature sensor might be inside an air-conditioned car or inside a bag, which affects the integrity of the collected data. It is crucial to design effective mechanisms for detecting anomalous sensor placements to ensure the integrity of the crowd-sensed data. On the other hand, deployment of these techniques close enough to the source of data (i.e., at the edge) helps avoid unnecessary data transmission.

Anomalies are defined as subsets of observations inconsistent with the data set which are categorized into three groups: point anomalies, contextual anomalies, and collective anomalies. Point anomalies are individual data instances that

are anomalous compared with the remainder of the dataset. If a data instance is anomalous in a specific context, but not otherwise, it is called a contextual anomaly. Collective anomalies are defined as the collection of data instances which are anomalous with respect to the entire data set. In these kinds of outliers, individual data instances may not be anomalies by themselves, but their joint occurrence as a collection creates the anomalies [Chandola et al., 2009]. In this chapter, we are mainly focusing on detecting collective anomalies caused by anomalous sensor placements.

Due to the highly dynamic nature of temperature and ever-changing ambient conditions, there is no gold standard reference to verify the data integrity of sensor readings. The most efficient way is to find patterns in the sensor data streams so as to filter the input data. In this chapter, we focus on the ambient air temperature measurements to show the feasibility of using lightweight models at the edge to filter out the outliers in a semi-real-time manner. In other words, the focus is on collective outlier detection using our proposed approach.

Most of the current crowd sensing systems primarily rely on the users to follow the provided instructions for data collection. In this chapter, we introduce a novel anomalous sensor placement detection technique that analyzes temporal patterns in the sensor data streams. Our approach is based on temperature readings, which show more frequent fluctuations while exposed to the outdoor environment. Furthermore, our technique is very lightweight, making it appropriate for the edge deployments. We also report a series of experiments to demonstrate the effectiveness of the method.

4.2 BACKGROUND AND RELATED WORK

4.2.1 CROWD SENSING FOR UHI ANALYSIS

UHI studies have traditionally relied upon data from satellites and weather stations. Researchers have used satellite images with thermal infrared bands to derive heat maps for different areas. They utilize different models based on the emissivity of land surfaces, as a measure of effectiveness in emitting energy as thermal radiation, to calculate the land surface temperature. As mentioned in the introduction, these traditional data sources have some limitations. For instance, the Landsat 8 satellite, which has been widely used in the research community takes an image from a given geographical area every sixteen days. These images also suffer from the coarse spatial resolution of 100 meters. Also, the limited number of weather stations fails to capture the heterogeneity of air temperature in urban areas.

In order to overcome these limitations, researchers have recently explored augmenting traditional data sources with participatory sensing. In participatory sensing approaches, small and low-cost sensors (samples shown in Figure. 4.1) are used by participants to collect data with high spatiotemporal resolution.

In the ambient air temperature studies, we are only interested in data coming from correctly-placed sensors. Therefore, there is a need to have mechanisms to identify the anomalously-placed sensors such as the ones inadvertently put inside a users pocket or bag. In addition, any data collected within the buildings, cars, and any other temperature-controlled environment needs to be detected.

Figure 4.1: Kestrel DROP and iButton temperature sensors.

4.2.2 RELATED WORKS

Some researchers [Klepeis et al., 2001, Kuras et al., 2017, Middel et al., 2016] have done a comprehensive study on the challenges of personal heat exposure research. They claim that the sensor placement is a primary factor to consider in designing data collection protocols for measuring heat exposure. For this purpose, thermal sensors need to be mounted as an external attachment to a backpack, exposed to the outdoor environment. In another study [Bourgeois et al., 2003] equipped research participants with temperature-logging sensors called iButton to measure the air temperature surrounding individuals as they went about their daily lives. Participants were asked to record the time periods when they were not carrying the sensor. This manual approach is not a scalable solution for large crowd sensing frameworks. The imposed limitations, mainly due to unscalability of their approach, did not allow them to do a comprehensive study.

In a separate participatory sensing research study, [Hasenfratz et al., 2012] introduced GasMobile, a portable air quality measurement system based on off-the-shelf components to be used by a large number of people. To ensure the accuracy of the collected data, they have exploited the sensor readings near static reference stations to recalibrate their sensors frequently. Although it is a reasonable strategy, they failed to consider the problem of sensor misplacement as a significant factor resulting in erroneous data collection. Interestingly, there is an on-board temperature sensor on their designed hardware, which provides the opportunity of analyzing temperature data stream simultaneously to detect the outliers caused by sensor misplacements.

It should be mentioned that the detection model we are proposing is designed based on experimental data, where the same behavior is observed in different low-cost temperature sensors. In general, temperature sensors which are exposed to ambient atmosphere show more frequent fluctuations in their temperature readings, compared to those from the sensors placed in climate-controlled settings. Wind effects, solar radiation, and longwave radiation, which is the radiation emitted from the Earth's surface [Erell et al., 2005] are different factors which influence a sensors behavior. As a result, each U.S. Climate Reference Network (USCRN) station contain three thermometers in a shielded setting, and the observed temperature values from all these three sensors are used to come up with a single official USCRN temperature value every hour. This single official value is either the median or the average of the three recorded values [Shlain, Accessed: 2018].

Exploiting statistical and data mining techniques is the best way to detect

anomalously-placed sensors. Finding patterns in subsequences of temperature time series and consequently identifying the structural similarities between those patterns is an effective way to separate different classes of data [Singh and Upadhyaya, 2012]. Considering that there are only two classes (i.e., correctly-placed and anomalously-placed sensor readings), supervised machine learning models based on the historical data enable us to perform binary classification of subsequences of temporal temperature data. Therefore, we are primarily dealing with a binary classification problem.

4.3 OVERVIEW OF OUR APPROACH

Figure.4.2 shows the high-level architecture of the participatory air temperature sensing and the anomaly detection model. This system utilizes low-cost ambient air temperature sensors to collect data with high spatiotemporal granularity. We used Kestrel temperature sensors (environmental data loggers) in our research. The system architecture is scalable and can be extended to support different temperature sensors via Bluetooth. Sensors are carried by users or mounted on the vehicles for data collection, and the mobile application acts as an intermediary component to communicate with the sensor and the cloud server. The communication between sensors and mobile devices is handled using the Bluetooth Low Energy protocol; on the other hand, the communication between mobile devices and the server is done using web services over the Internet. Temperature readings are synchronized with GPS sensors on mobile devices based on their timestamps,

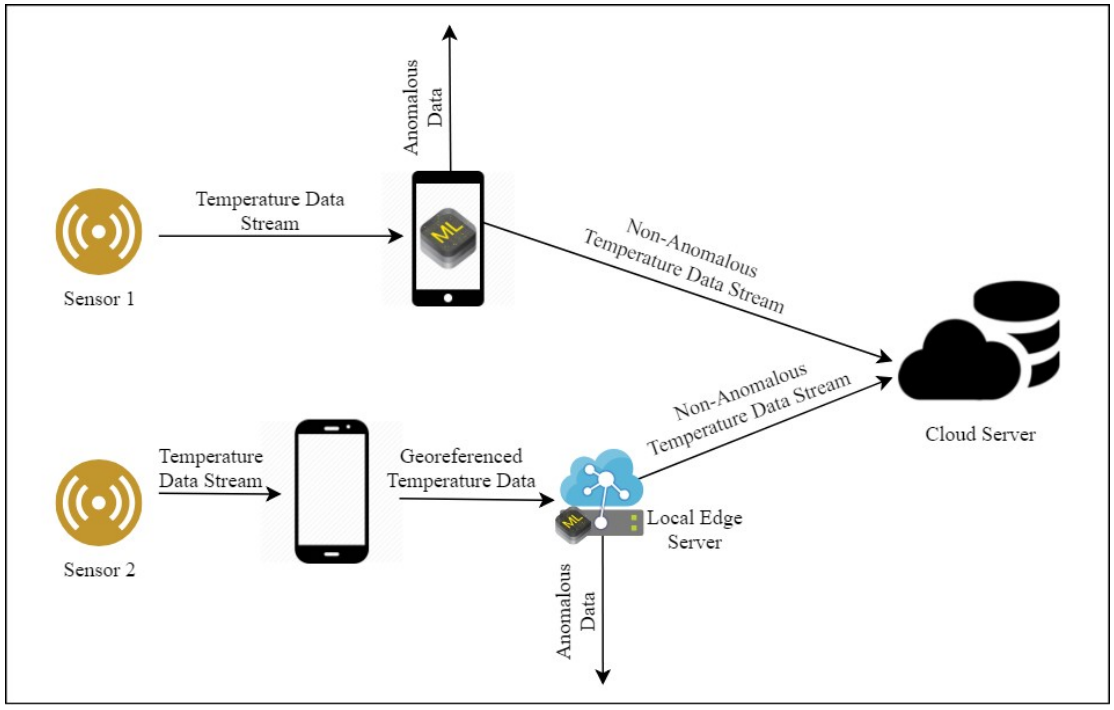


Figure 4.2: High level system architecture.

and then the spatiotemporal temperature readings will be sent to the server.

As illustrated in Figure.4.2, the lightweight binary classification model uses the incoming sequence of temperature data to do the semi-real-time filtering at the edge nodes. The anomaly detector model can be deployed either on the mobile devices or on the local edge servers to filter out the unwanted data.

The non-anomalous data stream will be sent to the cloud server once a mobile device connects to the Internet, either via WiFi or cellular network. As depicted in Figure.4.2, multiple users can simultaneously collect temperature data and upload them to the server. Anomaly detection model is used to mark anomalous

subsequences of temperature time-series data, and the filtered data will be stored for further analysis.

We primarily used Kestrel DROP D2 sensors for temperature data collection. These sensors record temperature readings with $(+/-)0.5\,^{\circ}\mathrm{C}$ accuracy, have $0.1\,^{\circ}\mathrm{C}$ resolution, and cover the temperature range of $-10\,^{\circ}\mathrm{C}$ to $55\,^{\circ}\mathrm{C}$ [Steffensen, n.d]. These small sensors communicate with our mobile applications through Bluetooth Low Energy.

4.3.1 DATA COLLECTION

In order to analyze the impact of temperature sensor placements, we collected the data from August 2017 to October 2017 by four volunteers using 12 different sensors. Data samples were collected from the morning to the evening on different dates over a span of three months. In total, we have collected 131 time-series data samples, equal to 77 hours of temperature data; and 5-second data logging rate was used during data collections. These collections involved temporal temperature reading samples from human-borne, vehicle-mounted, and stationary sensors. We labeled the data collected in climate-controlled settings as Class A (Anomalous), and the data collected by the sensors which were exposed to the outdoor environment as Class NA (Non-Anomalous). Table 4.1 shows the labels associated with each data sample.

As can be seen in Table 4.1, we further sub-categorize the data based on the traveling mode and the sensor placement to cover different possible erroneous situations where temperature data were collected. The sub-categorization helps us

Table 4.1: Data Collection Subcategories

Lable/Class ↓	User/Sensor Movements → (Travelling Mode) / Sensor Position ↓	Stationary	Walking	Bus	Car	Total	
A (Anomalous)	*stationary_in_ac_home*	6				6	91
	walking_in_bag		14			14	
	walking_in_pocket		16			16	
	in_ac_car				13	13	
	in_ac_car_in_pocket				7	7	
	in_ac_car_in_bag				7	7	
	in_ac_bus_on_bag			5		5	
	in_ac_bus_on_belt_loop			5		5	
	in_ac_bus_in_pocket			9		9	
	in_ac_bus_in_bag			9		9	
NA (Non-Anomalous)	*stationary_outside_in_sun*	8				8	40
	stationary_outside_in_shadow	6				6	
	walking_on_belt_loop		8			8	
	walking_outside_bag		9			9	
	outside_car				9	9	
	Total	20	47	28	36	**131**	

understand the possible common patterns and variations in temporal temperature readings used for anomaly detection.

To better understand the behavior of time series data in different scenarios, we have plotted data from different classes to do a comparative analysis. For example, in the car experiment shown in Figure. 4.3, a user carried four different sensors while traveling in a car. One sensor was placed correctly outside the car and exposed to outdoor ambient air temperature. The other three sensors were anomalously placed inside the car, one placed beside the gear, the other one in the drivers pocket, and the third inside a backpack.

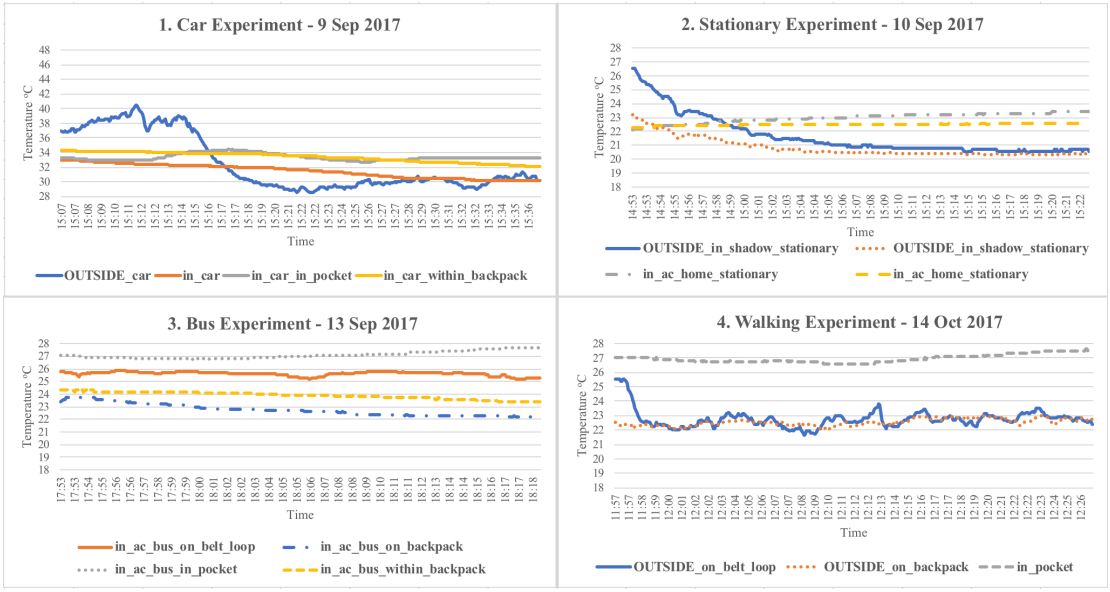


Figure 4.3: Experimenting Different Scenarios

4.3.2 KEY OBSERVATION

Based on the data plotted in Figure.4.3, we observed that when temperature sensors are placed correctly, the data show frequent fluctuations, while the otherwise placed sensors produce much smoother plots. For example, the data from the car experiment on 9 Sept 2017 shows that the correctly-placed sensor yielded much more fluctuations compared to the other three misplaced ones.

The same behavior has been observed during different experiment scenarios: sensors inside a bus, sensors moving at walking speed, and stationary sensors. We hypothesize that these fluctuations occurred because of wind effects, air flows, solar radiation, and longwave radiation from the Earth.

4.3.3 SLIDING WINDOW FOR TEMPERATURE TIME SERIES DATA

Domain plays an essential role in defining anomalies in time series data. In specific anomaly detection problems, properties of a subsection of time series data can be more critical than properties of the entire data stream. The sliding window algorithm helps to extract all such subsequences in a given stream of data [Yu et al., 2014]. Moving the window with a small offset, lets us extract a higher number of subsequences from time-series data. The sliding window is used to convert a window of sequential data to a single output, then all the subsequences in the time series are mapped to their respective values where supervised learning algorithms can be applied. Thus, with the help of the sliding window method, we can utilize conventional machine learning algorithms to solve sequential pattern recognition problems [Dietterich, 2002].

The window size plays a vital role in our anomaly detection analysis. On the one hand, smaller windows would lead to faster algorithms to provide real-time filtering. On the other hand, larger window sizes provide us with more data to better understand the temperature variations. We found that 5-minute and 10-minute window sizes are the best choices which satisfy the needs from both ends. In other terms, these window sizes contain sufficient temperature readings and are small enough to perform semi-real-time anomaly detection.

4.3.4 FEATURE SELECTION

The discussed feature extraction approach is used to classify patterns in the temperature data stream. For instance, let's consider a temperature time series with starting index as 1: $D = \{24, 25, 24, 24, 23, 25, 25, 26\}$. Therefore, series of consecutive temperature differences can be defined as: $C = \{1, -1, 0, -1, 2, 0, 1\}$, where each value represents the difference between the two adjacent temperature readings. The following features that are discussed in this section use the series C and D stated above:

1. STANDARD DEVIATION OF CONSECUTIVE TEMPERATURE DIFFERENCES (StdDevTempDiff): The high value of the standard deviation of a consecutive temperature difference (e.g., series C) in a given window represents the fluctuations in the temperature time-series data. This feature plays an essential role in our algorithm by separating the anomalous from non-anomalous sensor data.

2. ZERO-CROSSING RATE (ZeroCrossRate): Zero-crossing is defined as a change in sign from positive to negative or vice versa in sequential data [Bachu et al., 2008]. Therefore, zero-crossing points in a consecutive temperature difference (e.g., series C) are the points where a change in sign occurs. This measure signifies the change in surrounding conditions caused by solar radiation, wind speed, or longwave radiation.

   Zero-crossing rate [Gouyon et al., 2000] is the total number of zero-crossings in a sequential data divided by the total number of instances in that se-

quence. This rate for a given subsequence shows variations in the surrounding environment, and the high value of zero-crossing rate represents the fluctuations in temperature time series data. For instance, the zero-crossing indexes of series C are $Z = \{1, 4\}$, and the zero-crossing rate will be the total instances in series Z divided by the total instances in series C, therefore, the zero-crossing $rate = 2/7 = 0.28$

3. STANDARD DEVIATION OF ZERO-CROSSING POINT WEIGHTS (ZeroCrossWtStdDev): The anomaly detection problem is not that simple. We observed that in many cases, although the sensors are not exposed to the outdoor natural environment, there are some fluctuations for a small amount of time. For example, sensors might record different values due to the air circulation in indoor environments. As depicted in the second experiment of Figure.4.3, the time series indicated by *in_ac_home_stationary* show small fluctuations at around 15:15 which result in a high zero-crossing rate that might wrongly be classified as a non-anomalous sensor data.

   We define zero-crossing point weight as the difference between temperature reading at the zero-crossing point and the mean of the temperature readings in the given subsequence. Temporal temperature subsequence is then reconstructed by replacing temperature readings at zero-crossing points with corresponding zero-crossing point weights, and other temperature readings are replaced by zero. The score representing zero-crossing point weights in the subsequence is then calculated by taking the standard deviation of the reconstructed subsequence.

Using the example series of C, D, and Z, series containing zero-crossing point weights are defined by: Mean of series $D = 24.5$; zero-crossing point indexes of series D are $T = \{2, 5\}$; series of zero-crossing point weights are $W = \{0, 0.5, 0, 0, -1.5, 0, 0, 0\}$; and the feature value is standard deviation of values in series W.

A high value of standard deviation for zero-crossing point weights indicates fluctuations in temperature time series and appear as high peaks and low valleys. The feature helps in reducing the total number of false negatives.

4. NON-ZERO TEMPERATURE DIFFERENCE RATE

   (NonZeroTempDiffRate): Non-zero temperature difference rate indicates the total number of non-zero temperature differences in consecutive temperature difference series, which is divided by the total number of instances in the series. From series C, the total number of non-zero temperature differences equals 5, and the total number of instances in series C is equal to 7. Hence, the non-zero temperature difference rate is 5/7 (0.71).

   In some cases, when the temperature is either increasing or decreasing in a particular direction, correctly-placed temperature readings show fewer fluctuations. As a result, the corresponding subsequence may get wrongly classified as anomalously-placed. The high value of non-zero temperature difference rate indicates the frequent changes in temperature values. This feature helps in reducing the total number of false positives [Mahadevan et al., 2010]. Finally, to train the classifiers, feature extraction module extracts the features using the sliding window technique from crowd-sensed

temporal temperature data and converts the subsequences into data points.

4.3.5 ANOMALY DETECTION WORKFLOW

Our proposed model is designed to recognize the anomalous subsequences in the temperature time series $\{T_1, T_2, T_3, \cdots, T_n\}$. For this purpose, we used the sliding window approach to convert the time series outlier detection problem into a binary classification problem. Then, supervised learning models are used to detect anomalous subsequences. Figure. 4.4 represents the steps of our anomaly detection workflow.

Based on the data logging rate of a sensor, and also the size of the sliding window, a subsequence of the temperature data stream will be chosen. Next, the features mentioned earlier will be extracted. Then, our supervised learning models classify the subsequences using the extracted features. If the subsequence is classified as an anomalous, it will be filtered out. Otherwise, it will be sent to the server for further analysis. In the next step, based on the defined offset, the sliding window will move forward, and the next subsequence will be chosen for classification. This process continues to the end of the temperature data stream.

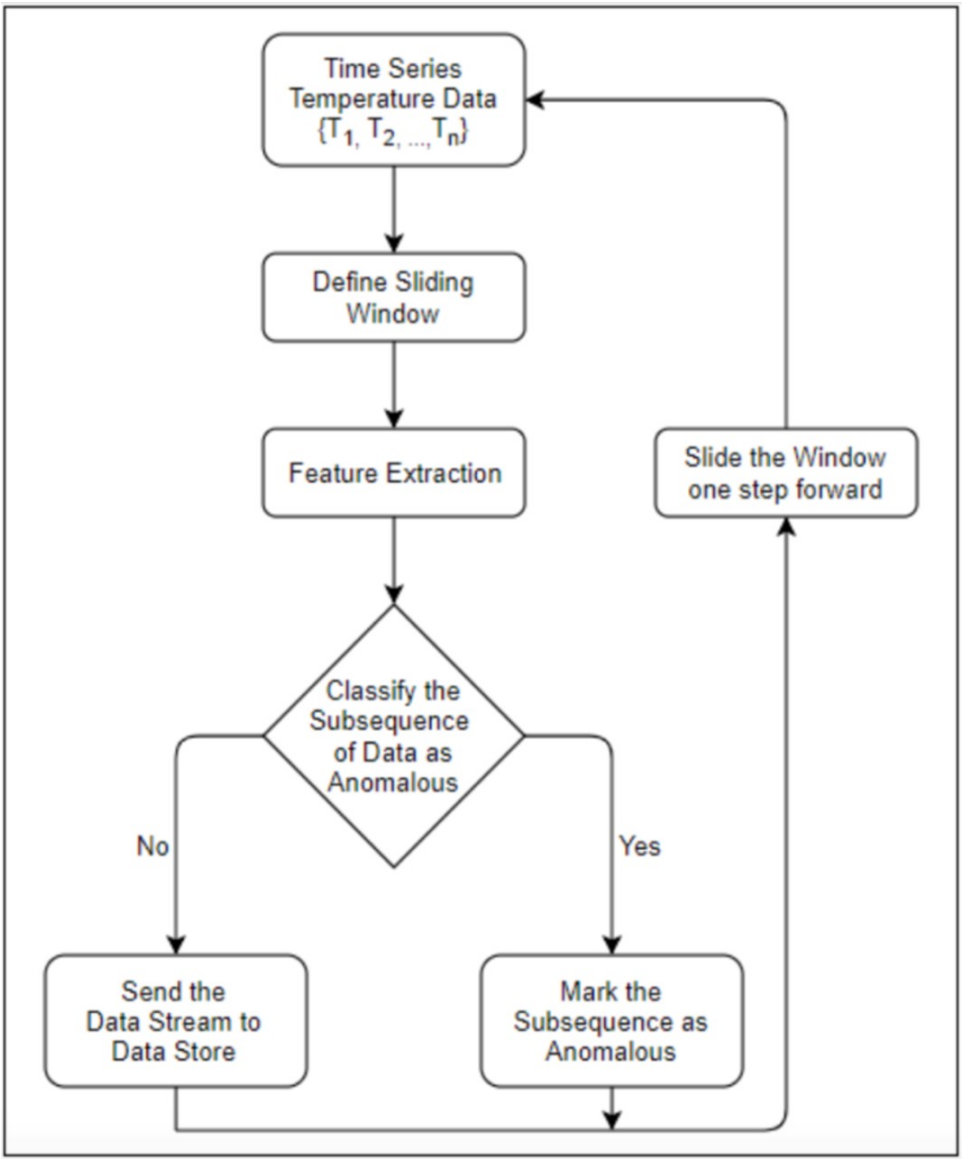


Figure 4.4: Anomaly Detection Based on Sliding Window.

4.4 EXPERIMENTAL RESULTS

4.4.1 EXPERIMENTAL SETUP

In order to generate the training and testing data sets, five different data logging rates of 5, 10, 20, 30, and 60 seconds are used. We also came up with 5-minute and 10-minute sliding windows with an offset of one for the feature extraction. We developed a feature extraction module in Python programming language, which chooses a subsequence of time series data based on the window size and offset, then extracts statistical features to generate the data sets for classification.

In this chapter, we study the performance of three common statistical binary classification methods, i.e., Support Vector Machine (SVM), Linear Regression, and Random Forest, to detect anomalous subsequences in temperature time series data. To train the models, classifiers in scikit-learn Python package [Steffensen, 2019] are used. The details of the performance comparison analysis will be discussed later in this section.

A linear kernel with standardized input data is used to train the SVM classifier. Standardization is the process through which variables are re-scaled to have a mean of zero and a standard deviation of one. It brings data into a standard format, which enhances the comparison process. Based on the analysis of underlying data distribution, we used the linear kernel to train SVM. Nonlinear kernel functions map data points to higher dimensional feature spaces to achieve linear separability [Widodo and Yang, 2007], which increase their computational complexity and make them unsuitable for resource-constrained edge devices.

For training Logistic Regression classifier, training data are standardized, and 12 penalty parameters are used. For the Random Forest classifier, 100 trees are used as estimators and split quality is measured based on gini criterion; as a statistical measure of the degree of variation represented in a set of values.

In our data set, 71% of the data were collected from anomalously-placed sensors as opposed to 29%, which were placed correctly. To understand the feature-wise class distribution, we plot the four selected features to know how these features have been distributed. Considering that feature values depends on the size of the sliding window, based on some experiments, we came up with two window sizes of 5-minute and 10-minute. In the following sections, we mainly focus on the 5-minute window size, as it is promised to be both lightweight and accurate.

Figure.4.5 shows the class-wise distribution of features. As can be observed in the box plots, data points between the lower quartile and the upper quartile of the two classes do not overlap with each, which makes the data linearly separable.

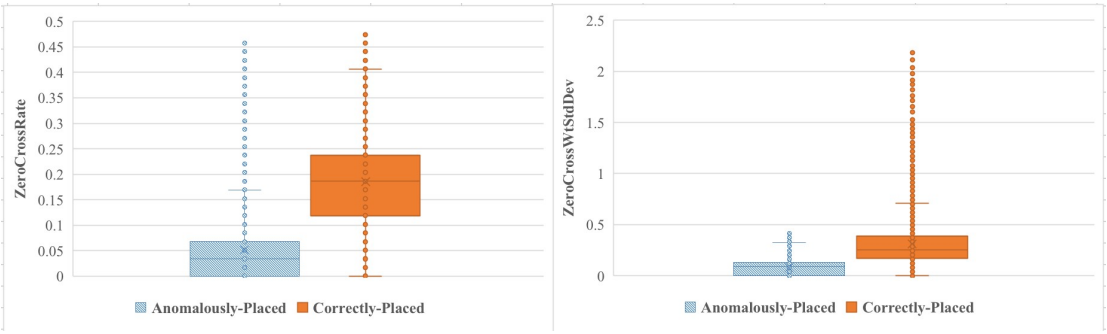


Figure 4.5: Class Distribution of Features

In addition, Figure.4.6 represents the distributions in a 2D feature space. Although class boundaries overlap with each other, anomalous data have lower val-

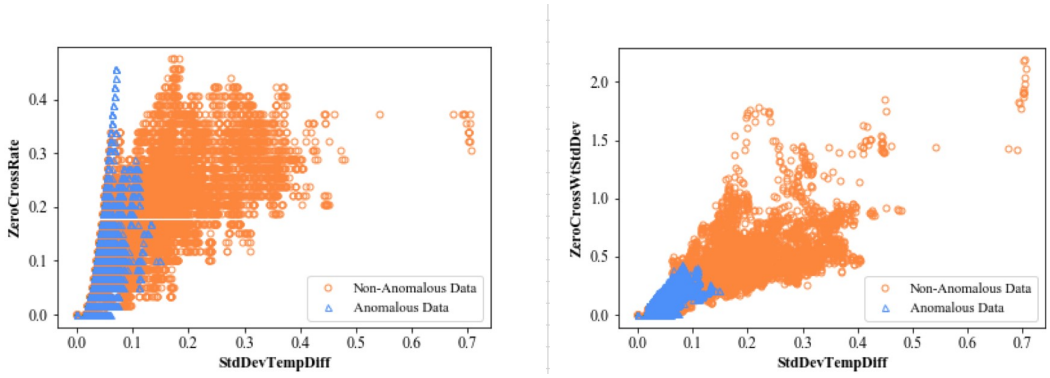


Figure 4.6: Distributions in Feature Space

ues and are prominently clustered around the origin. On the other hand, non-anomalous data points have higher values and are spread away from the origin.

4.4.2 PERFORMANCE METRICS

In our approach, we use binary classification to identify anomalous sub-sequences. Sensitivity, specificity, and macro F1-score are three performance measures that we have used to evaluate the performance of classifiers [Parambath et al., 2014, Van Asch, 2013]. Anomalous temporal temperature subsequences are defined as positive class and non-anomalous ones as the negative class. Sensitivity helps us understand the probability with which the proposed method identifies the unexposed subsequences. Mathematically, sensitivity is defined as [Yu et al., 2014]:

$$Sensitivity = TruePositives/(TruePositives + FalseNegatives)$$

Specificity gives us an idea of how effectively a classifier identifies negative classes, and it is defined as [Sokolova and Lapalme, 2009]:

$$Specificity = TrueNegatives/(TrueNegatives + FalsePositives)$$

Macro F1-score is defined as the arithmetic mean of F1-score of both positive and negative classes [Sokolova and Lapalme, 2009, Yang et al., 1999] which represents a more balanced and less biased view compared to the other metrics. The F1-score formula is:

$$F1 - score = 2 * (Precision * Recall)/(Precision + Recall)$$

The three aforementioned testing metrics provide a comprehensive understanding of the models' performances.

4.4.3 CLASSIFICATION USING ALL FEATURES

We trained the SVM, logistic regression, and random forest classifiers with all the four extracted features and evaluated their performance using sensitivity, specificity, and macro F1-score. As we can see in the plots of Figure.4.7, SVM with 20 seconds logging rate shows an overall better performance using different performance metrics. Based on the comparative analysis, we noted that the best model which makes a balance between performance, data logging frequency, and the sliding window size to satisfy the semi-real-time characteristic of the system is SVM with 20 seconds frequency and 5-minute window size.

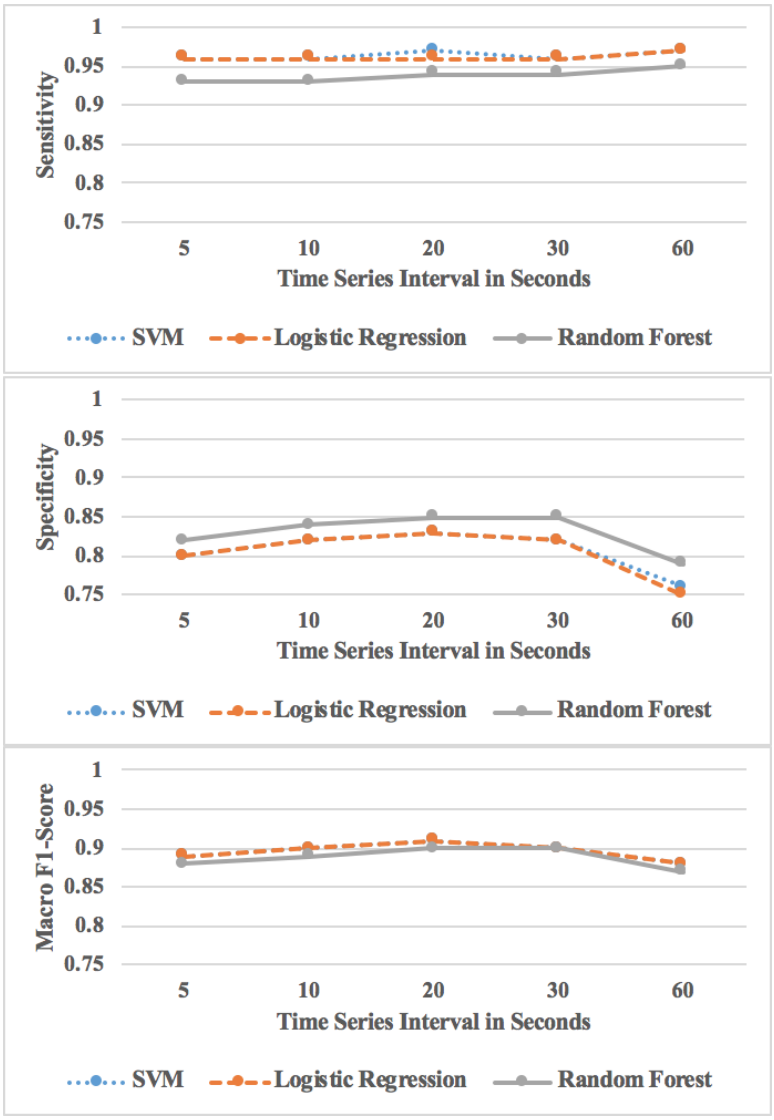


Figure 4.7: Performance of the models trained on all features

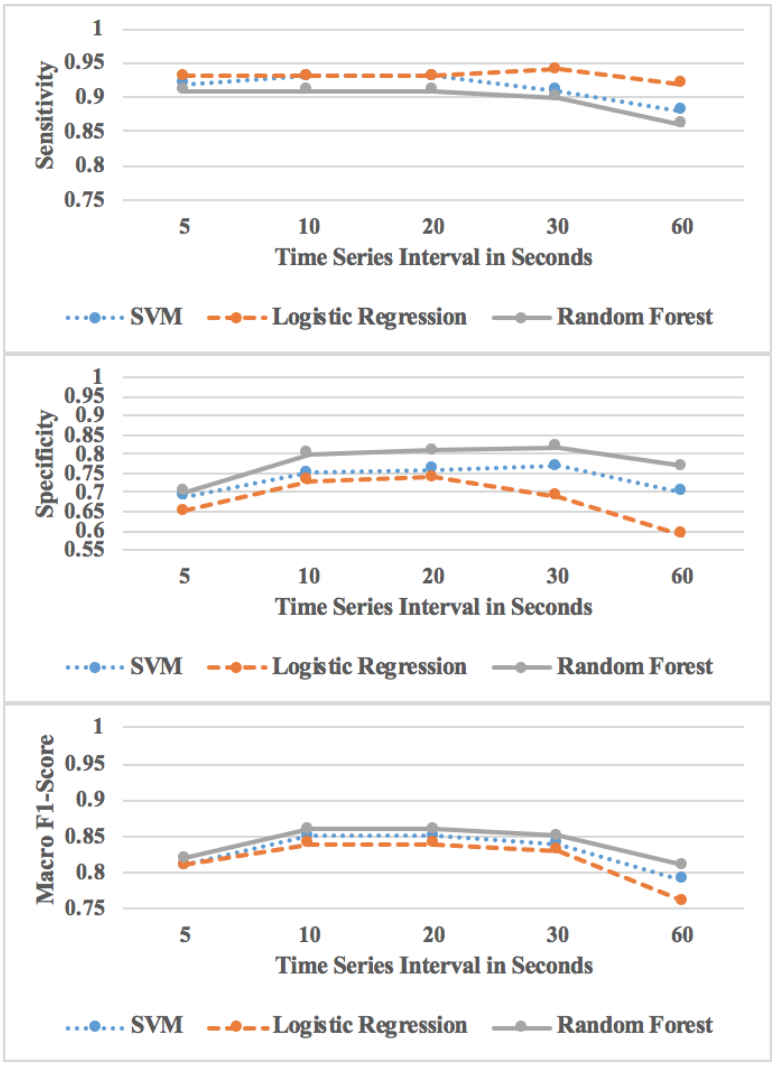


Figure 4.8: Performance of the models using only zero-crossing rate

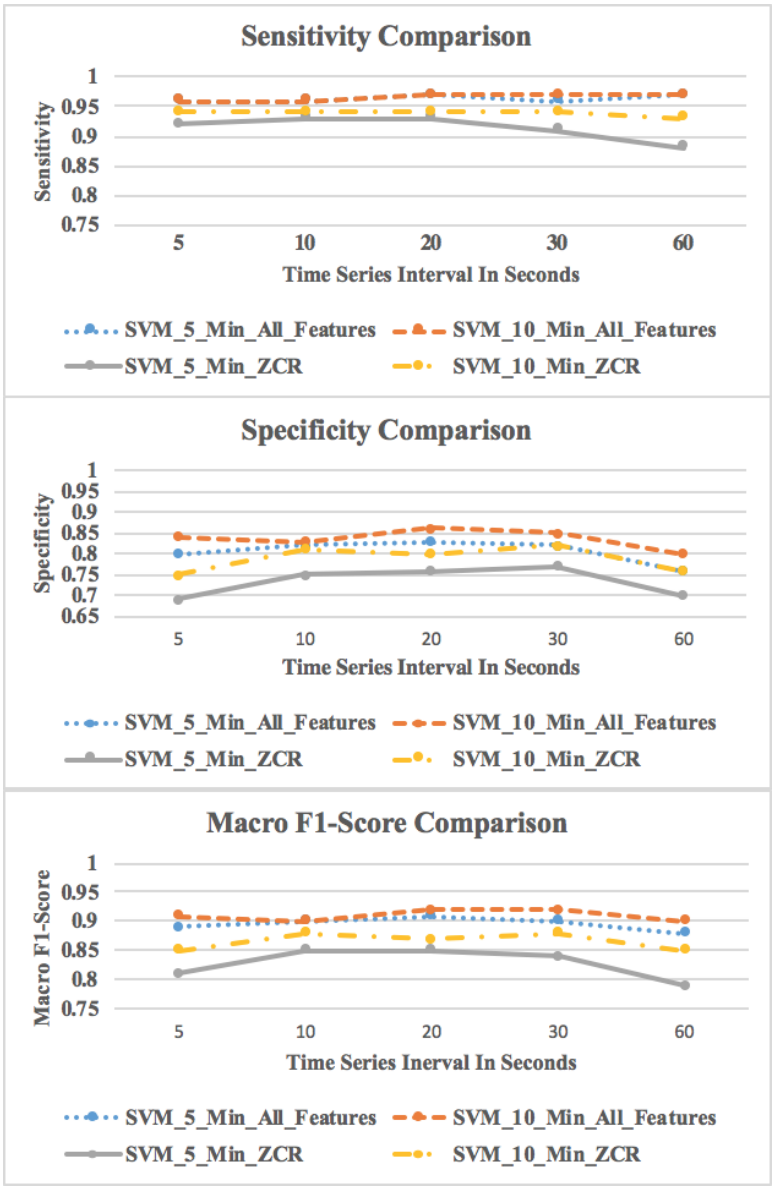


Figure 4.9: Performance of SVM classifier

4.4.4 CLASSIFICATION USING ONLY ZERO-CROSSING FEATURE

In another experiment, we trained the model using only the zero-crossing rate feature. This feature effectively captures fluctuations observed in the readings of temperature sensors exposed to the outdoor environment. Figure.4.8 depicts the accuracy of the model using different metrics.

4.4.5 PERFORMANCE OF SVM

Overall, we observed that SVM is the best classifier for our anomaly detection purpose. In this section, we are comparing the performance of this classifier using all four features as opposed to using only the zero-crossing rate feature. We are also extending our analysis to compare the 5-minute window size with 10-minute ones. In Figure.4.9, we can observe that SVM classifier trained with 5-minute window size with all the four features on the data collected with 20 seconds frequency shows a reasonable performance regarding the sensitivity and the macro F1-score, while its specificity is slightly lower than the 10-minute window. It suggests that a comparable performance can be achieved by using all features with smaller window size, which help our model respond faster.

4.5 CHAPTER SUMMARY

Due to many limitations of traditional data sources such as satellites and weather stations, crowd-sensed temperature data have been increasingly leveraged for an-

alyzing the Urban Heat Islands (UHI) phenomena. However, one of the main challenges in using human-borne sensors for UHI studies is that the sensors might be erroneously placed, which will heavily compromise the integrity of the collected data.

In this chapter, a semi-real-time approach towards detecting anomalously-placed sensors is developed, which is based on the key observations in which the temperature readings from correctly-placed sensors show frequent fluctuations. Using the sliding window approach, we introduce zero-crossing point weight and non-zero temperature difference rate to quantify the fluctuations in time series data. Our technique utilizes standard binary classification methods such as SVM, logistic regression, and random forest for identifying the anomalous subsequences of data. We performed a number of participatory sensing experiments to analyze the performance of our approach. The results prove the effectiveness of our method in identifying anomalous sensor placements.

Chapter 5

COVERAGE MAXIMIZATION IN DRIVE-BY SENSING

Chapter Overview

Many natural phenomena and physical properties such as sound, temperature, and magnetic fields on the earth are continuous signals, both spatially and temporally. To study these environmental features accurately, we need to have a consistent monitoring infrastructure to capture their spatiotemporal variations. On the other hand, public transportation vehicles such as city buses provide a cost-effective platform for environmental sensing. Based on different applications, various types of sensors and cameras could be mounted on these vehicles to provide constant monitoring of the target features such as noise pollution, temperature, and air pollution in different urban areas. In this chapter, we propose and develop

our efficient approach to mount a limited number of sensors on the buses, given their trajectory data to maximize their spatiotemporal sensing coverage. We consider that there are some pre-defined hotspots in the cities where their continuous sensing is of greater importance compared to other areas. First, we formulate this vehicle selection problem as an optimization problem, then explain our proposed method while adhering to cost constraints. Finally, we evaluate our approach using the real-world trajectory data collected from more than twenty buses in the city of Athens, Georgia.

5.1 INTRODUCTION

Drive-by sensing is a category of mobile sensing [Lee and Gerla, 2010] in which the participants are vehicles. This sensing paradigm provides an excellent opportunity, especially where there are cost constraints, and the target properties are dynamic. Public transportation vehicles move around the cities frequently, so they provide an excellent infrastructure to have systematic sensing from different areas at different times of the day. Furthermore, the predefined paths of the public transportation vehicles are close to the daily commute routes of city dwellers. Therefore, drive-by sensing paradigm provides a more accurate reading from environmental features compared to other sensing technologies such as remote sensing.

Apart from the cost associated with sensors and the sensing infrastructure, the process of mounting sensors on the buses and maintaining them impose a

separate cost. For instance, in the City Scanner project [Anjomshoaa et al., 2018], researchers proposed a modular sensing architecture to be mounted on top of the garbage trucks to collect a multitude of city features. Most of these sensors are expensive, so there should be some selection mechanisms to choose a subset of running vehicles for mounting the limited number of sensors. On the other hand, there are some Areas of Interest (AOIs) in the cities where their continuous monitoring has a higher priority. For example, in the context of urban heat analysis, the locations with a high variation in temperature create AOIs, and their continuous sensing is more important compared to other areas.

Although leveraging this sensing paradigm for monitoring purposes is beneficial for different applications, there is a need for cost-aware sensing mechanisms. In this study, we focus on this addressing problem, which is common in various sensing applications. Our specific contributions in this domain can be summarized as follows:

- Formally defining and formulating the cost-aware bus selection problem as an optimization problem.
- Providing an efficient approach to select a near-optimal subset of buses given the cost constraints.
- Addressing the problem of dynamicity of the hotspots and taking the importance of their continuous monitoring into consideration within the decision process.

The rest of the chapter is organized as follows. In Section 2, we present some

backgrounds and related works in this domain. In section 3, the sensor placement problem is defined and formulated. Then, two simple selection approaches are discussed. Section 4 focuses on cost-aware sensing approaches where our methods are introduced. In section 5, we perform experimental evaluations to analyze the performance of the proposed algorithms. In the last section, we conclude this chapter and discuss future directions for this study.

5.2 BACKGROUND AND RELATED WORKS

Multiple studies focus on maximizing the sensing coverage in mobile crowdsensing given different scenarios. These studies assume that all the participants are already equipped with the required sensing devices, and they investigate various approaches to distribute sensing tasks while minimizing recruitment costs.

Guo et al. [Guo et al., 2016] propose a worker selection approach under two situations: either based on the intentional movement of sensing agents for time-sensitive tasks or based on their unintentional movement for tasks which are not time-sensitive. They evaluate their algorithms using D4D dataset [Blondel et al., 2012] which contains individual call detail records for customers of Orange Group during two weeks in Ivory Coast. Each data point has its id, latitude, and longitude, so the authors extracted the users mobility traces corresponding to the cell towers. Then, they show how their algorithm outperforms the previous approaches like that of discussed in [Engelbrecht, 2014] as a particle swarm optimization (PSO) solution.

Campioni et al. [Campioni et al., 2018] study recruitment algorithms aimed at selecting participants within a crowdsensing network in a way that the most sensing data is obtained for the lowest possible cost. However, like many other studies in this domain, they assumed all the participants are equipped with the sensing devices. They developed different approximation solutions to solve the vehicle recruitment problem for both the temporal and spatiotemporal variants.

He et al. [He et al., 2015] present a new participant recruitment strategy for vehicle-based crowdsourcing by predicting the future trajectory of participants. They evaluated the two proposed algorithms by comparing to other existing methods with unpredictable mobility patterns, and they were able to achieve an average of 15% improvement in terms of crowdsourcing quality.

In another study, Yi et al. [Yi et al., 2017] propose a fast algorithm for vehicle participant recruitment problem, which achieves a linear-time complexity at the sacrifice of a slightly lower sensing quality. They claim that their method is 50 times faster than the state-of-art algorithms while it only sacrifices 5% of the sensing quality by testing on more than 1000 vehicles.

Wang et al. [Wang et al., 2018] proposed a system model based on the predictable trajectory of public transports through a cloud management platform which interacts with static based stations for distributing the sensing tasks to buses with embedded sensors. In their design, they assume that each public transport vehicle needs to be paid a sensing reward to perform the crowdsensing. Accordingly, their approximate algorithm called efficient combination query algorithm adopts a greedy approach to efficiently distribute the sensing reward until

it reaches the limited budget. This research, like the other studies discussed in this section, assumes that all the drive-by sensing vehicles are equipped with the required sensors and receive a reward per each sensing task.

5.3 SENSOR PLACEMENT PROBLEM

The objective of this chapter is to find an optimal bus selection approach to mount a limited number of sensors on the buses to maximize their spatiotemporal coverage. For this purpose, there are two main assumptions:

1. Trajectory data of the buses are available. In other terms, the routes that each bus traverse is known. Using the GPS data and the timestamps associated with them, we can estimate the location of each bus at a particular point of time.

2. There are some hotspot locations where their continuous sensing is of higher importance compared to other areas.

To formulate the sensor placement problem, we model the study area as a grid of square cells, as shown in Figure.5.1. The dimension of each cell is a configurable parameter and represents the spatial granularity of the sensing. We define matrix A, where an arbitrary cell of the grid is represented as a_{ij}:

$$A = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mn} \end{bmatrix}$$

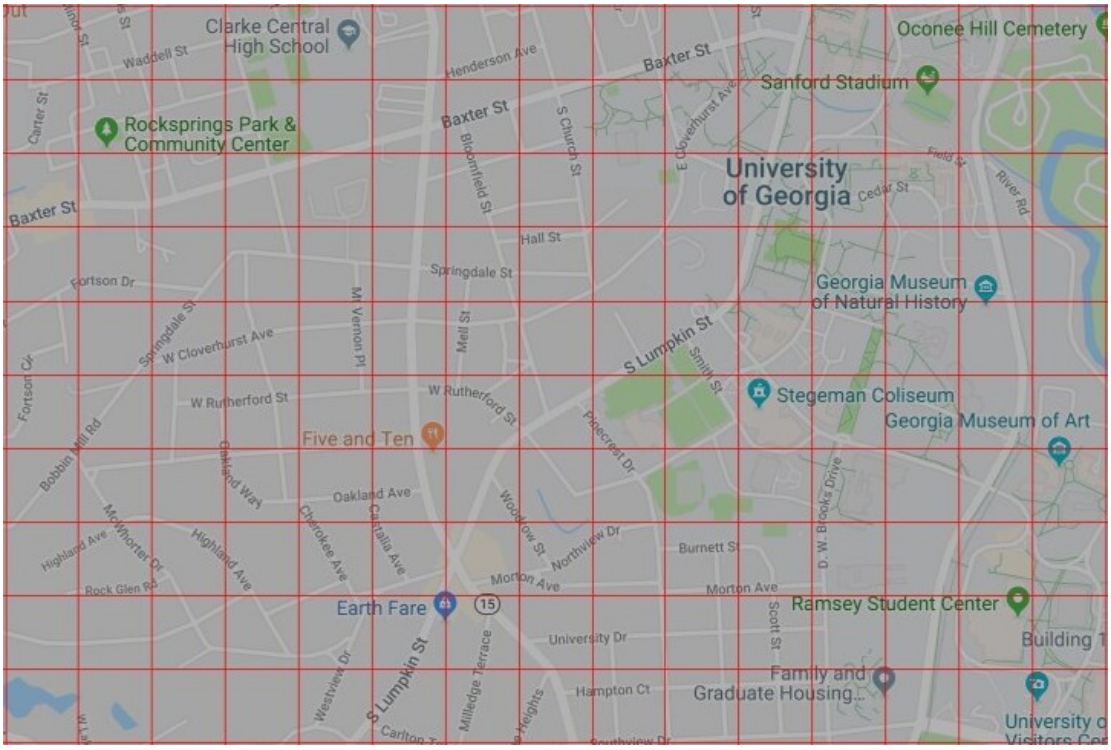


Figure 5.1: A sample grid representation

To consider the hotspot locations, we should be able to assign different weights to the cells. Therefore, matrix W is defined where each grid cell is associated with a weight:

$$W = \begin{bmatrix} w_{11} & \dots & w_{1n} \\ : & & : \\ w_{m1} & \dots & w_{mn} \end{bmatrix}$$

Time is modeled as a vector of $T = \{t_1, t_2, ..., t_l\}$ where each t_k is a time slot with configurable duration. The sum of these time slots is 24 hours (1 day), and the duration of each slot represents the granularity along the temporal dimension. For example, if we need to have a reading of an environmental feature every 30 minutes, each t_k denotes a 30-minute time slot.

The set of $B = \{b_1, b_2, ..., b_p\}$ represents all the buses available in the city where each b_λ represents an individual bus. If a bus b_λ carries a sensor (i.e., it is selected for sensor deployment), it can obtain a reading from the cell a_{ij} in the time slot t_k if and only if b_λ is present within a_{ij}'s boundaries for at least some duration of time slot t_k (i.e., b_λ has traversed through a_{ij} in time slot t_k). Please note that a bus can traverse through multiple cells during a time slot. Also, multiple buses can traverse through a given cell during a given time slot (in which case, we obtain duplicate values).

Considering the limited Number of Sensors (NS), we define BS:

$$BS = \{BS_1, BS_2, ..., BS_q\}$$

as the set of all possible bus combinations, where: $BS \subseteq B \quad \& \quad |BS| \leq NS$. For instance BS_1 can be represented as:

$$BS_1 = \{b_5, b_{18}, b_{24}\}$$

OBJECTIVE FUNCTION

In this section, we define our objective function. For this purpose, let's suppose that the Selected Bus Set of $SBS^* = \{b_l, b_k, b_p\}$ represents the set of 3 buses selected for sensor deployment. ($SBS \subseteq B \quad \xi \quad |SBS| \leq NS$)

Having laid out the model, we now define the Coverage Value (CV) of the selected bus set (SBS^*) with respect to a cell a_{ij} at a time slot t_k as follows:

$$CV(BS_x, a_{ij}, t_k) = \begin{cases} w_{ij}^{t_k}, & \text{if } \{\exists b_i \in BS_x | b_i \; is \; in \; a_{ij} \; at \; t_k\} \\ 0, & \text{otherwise} \end{cases} \tag{5.1}$$

And the Cumulative Coverage Value of SBS^* is defined as:

$$CCV(BS_x) = \sum_{t_k \in T} \sum_{\forall a_{ij} \in A} CV(BS_x, a_{ij}, t_k) \tag{5.2}$$

Furthermore, we define Minimum Coverage Value as:

$$MinCV(BS_x) = \min_{\forall t_k \in T} (\sum_{\forall a_{ij} \in A} CV(BS_x, a_{ij}, t_k)) \tag{5.3}$$

MinCV denotes the lowest coverage value gained in all the time slots, which plays an essential role in our objective function to choose the SBS^* with highest spatial coverage during the whole sensing period. Finally, our objective function

is defined as follows:

$$SBS^* = \begin{cases} SB_x, & \text{if } (CCV(BS_x) > \bigwedge\limits_{BS-\{BS_x\}} CCV(BS)) \\ SB_x, & \text{if } (CCV(BS_x) = \sum\limits_{\forall_{BS_i \in BS}} CCV(BS_i) \wedge \\ & \quad MinCV(BS_x) = \sum\limits_{\forall_{BS_i \in SB}} MinCV(BS_i)) \end{cases} \tag{5.4}$$

In other words, the objective function chooses a bus set if its CCV is higher than all the CCV of other bus sets. If more than one bus set had the same CCV, the SBS^* with the highest MinCV would be selected.

In order to better understand the definitions mentioned above, we provide some examples in the following paragraphs. In the example shown in Figure 5.2, we have a grid with 16 cells without any hotspot. The routes that each bus passed during a time slot is depicted using the dotted lines. Let's suppose that there are two bus selections named BS_1 and BS_2, where:

$$BS_1 = \{bus_1, bus_2, bus_3\}$$

$$BS_2 = \{bus_3, bus_4, bus_5\}$$

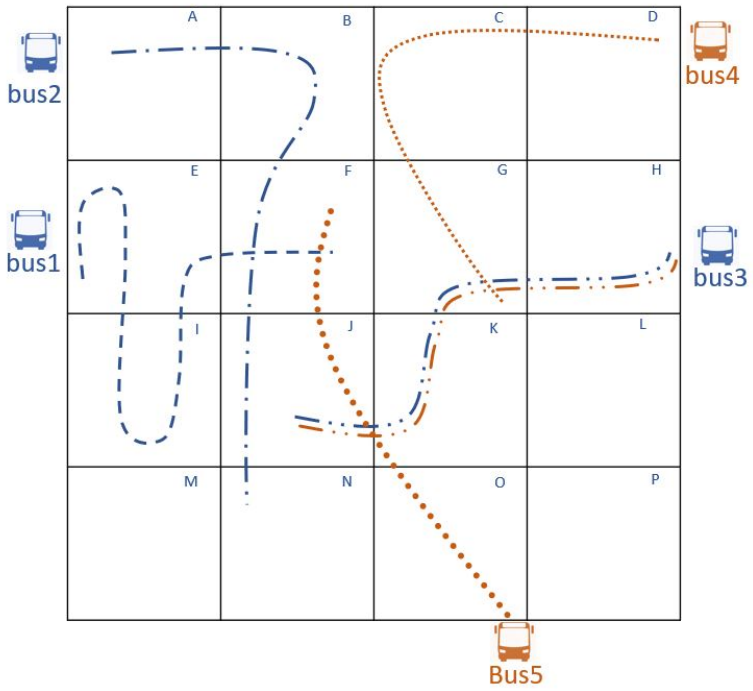


Figure 5.2: An Example of a Bus Selection Coverage in One Time Slot

We can see that bus_3 is selected in both sets, while the other two buses are different. Table 5.1 represents the number of cells passed by each bus during a given time slot. For instance, bus_1 passed three different cells (E,F, and I); therefore, it gets the value of 3 in Table 5.1.

Table 5.1: Calculating Bus Coverage Value at t_l

$\sum w_{ij}$	b_1	b_2	b_3	b_4	b_5
BS_1	3	5	4		
BS_2			4	3	4
BS_3					

Table 5.2: Total Sensed Cells Per Each Sensing Period

$\cup w_{ij}$	t_1	t_2	t_3
BS_1	**10**	12	9
BS_2	**8**	10	10
BS_3	10	11	10

Considering that some of the covered cells by different buses in a bus selection might be the same, we generate Table 5.2 from the previous table, where the union operator is used to exclude the overlaps. In other words, we only need to know whether a bus selection, as a whole, has a reading from a cell in each time slot or not. Therefore, the union operator allows us to exclude the additional readings from the same cell. As it can be seen in Table 5.2, although the sum of the cells in the first row of Table 5.1 is 12, the union of them equals to 10; because cells F and J are covered by two separate buses of the same bus selection of BS_1. The first two bolded cells of Table 5.2 are generated based on Table 5.1, and the other cells are assumed to have the other values to be used for explaining the next steps.

The example we saw in Figure.5.2 was for one time slot. In the next step, we want to continue with the same example but for three consecutive time slots. In Figure.5.3 the grid on the back corresponds to the same bus selection of BS_1 which we had in Figure.5.2. Considering that during the first time slot, BS_1 met 10 different cells, this selection gains the coverage values of 10 for t_1. During the second time slot, the three buses (bus_1, bus_2, and bus_3) continued their routes and sensed 12 different cells. Although some of the cells were already sensed during t_1,

these cells are counted again in t_2, because we only consider the overlaps within a same time slot. Therefore, BS_1 gets the coverage value of 12 in t_2. Using the same logic, BS_1 collects 9 coverage values during t_3. Looking back at Table 5.2, we can see these coverage values corresponds to the first row of the table. Figure 5.4 provides a comprehensive overview of the example discussed above. We can see all the cells that are sensed during the whole time period.

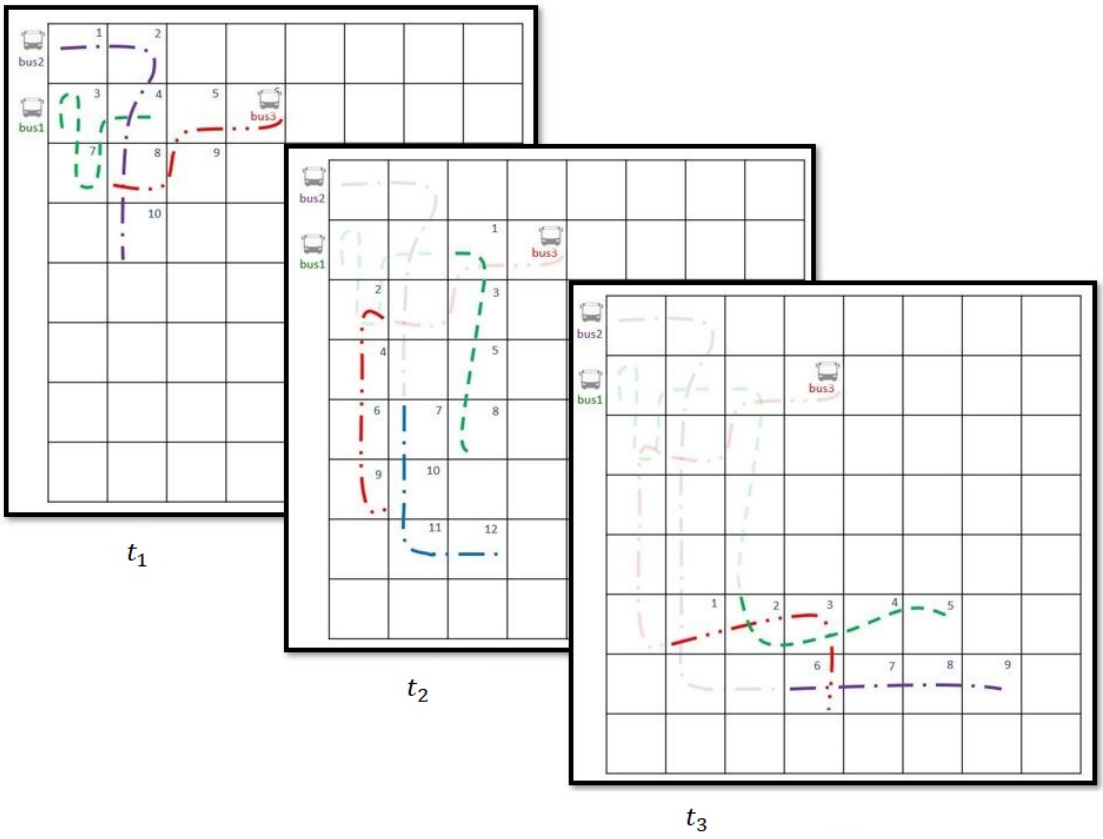


Figure 5.3: An Example of Bus Selection (BS_1) Coverage in Three Consecutive Time Slots

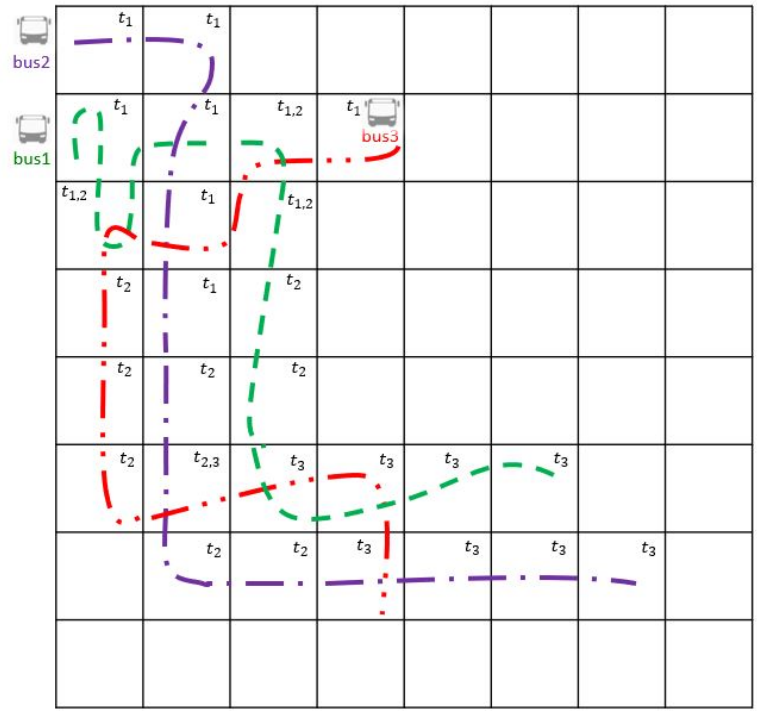


Figure 5.4: Bus Selection BS_1 in Whole Time Period of t_1 to t_3

Figure 5.5 shows the coverage of our third bus selection BS_3 in the time period of t_1 to t_3, which correspond to the third row of Table 5.2. This bus selection earned coverage value of 10 in t_1, coverage value of 11 in t_2, and coverage value of 10 in t_3. Similarly, we consider the overlapped cells within each time slot; however, BS_3 collected some coverage values from the same cells in different time slots. The cells which are counted more than once are shown in Figure.5.6 where more than one t_i is written in a same cell.

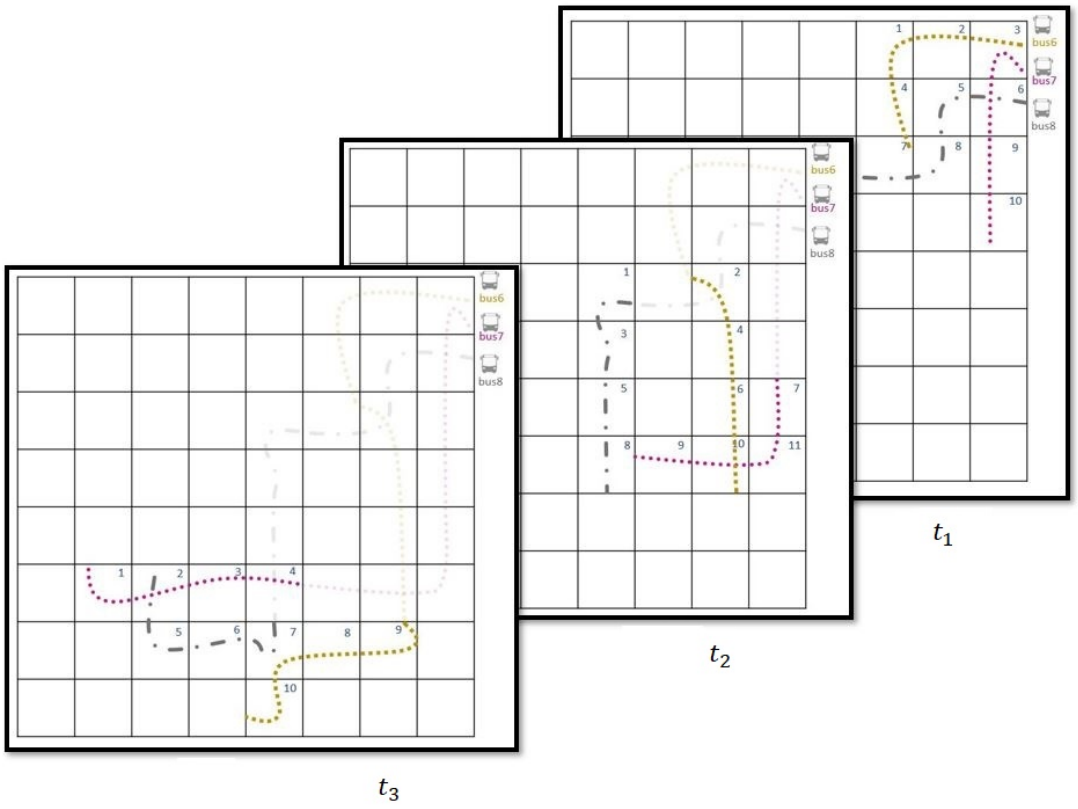


Figure 5.5: An Example of Bus Selection (BS_3) Coverage in Three Consecutive Time Slots

In the next step, we generate the Total Coverage Value for each bus selection during the whole time period. The first column in Table. 5.3 represents this total coverage value for each BS_i. Because of the fact that we used union operator in the previous steps to exclude the overlaps, the values of this column are basically the sum of the values in each row of Table 5.2. The second column of Table. 5.3 shows the minimum value of each row of Table 5.2. In other terms, this column shows the minimum coverage values that each bus selection was able to

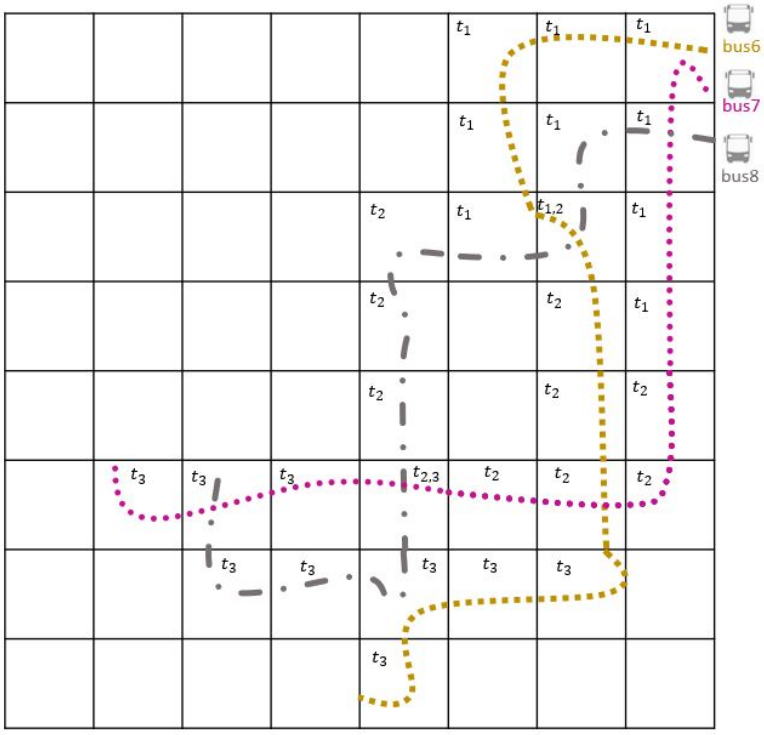


Figure 5.6: Bus Selection S_3 in Whole Time Period of t_1 to t_3

earn during each time slot. The minimum coverage values will be used later in our objective function, where we want to choose the most optimal bus selection with the best spatiotemporal coverage.

Table 5.3: Total Sensing Coverage Value for Each Bus Selection During the Whole Time Period

	CCV(BS_x)	MinCV(BS_x)
BS_1	31	9
BS_2	28	**8**
BS_3	31	10

To better understand how the different weights of hotspots can affect the total sensing coverage value, Figure 5.7 depicts the previous example with BS_1 and BS_2 while the grid cells have different weights. Table 5.4 shows the updated bus selection coverage values at t_1. Furthermore, Table 5.5 represents the updated total coverage value during the first time slot.

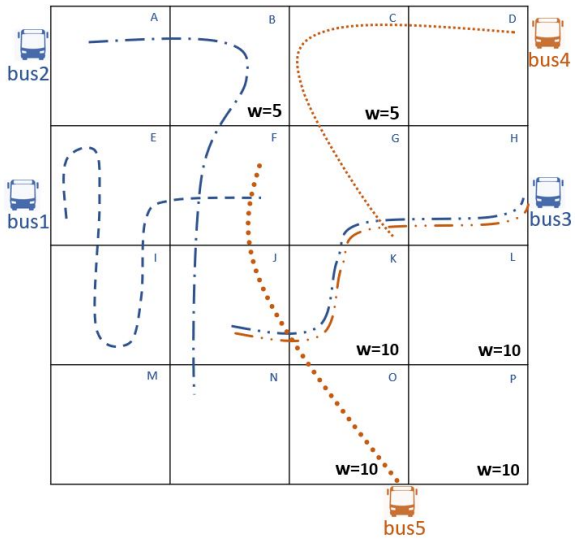


Figure 5.7: An example of a grid with AOIs of different weights

Table 5.4: Calculating Bus Selection Coverage Value at t_l with AOIs

$\sum w_{ij}$	b_1	b_2	b_3	b_4	b_5
BS_1	9	3	13		
BS_2			13	7	22

Table 5.5: Total Sensed Cells Per Each Sensing Period with AOIs

$\cup w_{ij}$	t_1	t_2	t_3
S_1	23	...	...
S_2	30	...	...

5.3.1 SIMPLE APPROACHES

NAIVE APPROACH

The simplest approach to solve the problem is to mount sensors on a randomly selected set of buses. Considering that the random selection fails to consider the requirements defined by our objective function, we cannot have any conclusion on their effectiveness.

EXHAUSTIVE APPROACH

The other approach which considers our discussed objective function is the exhaustive method. In this approach, all the possible combinations of n buses taken r at a time, where r is equal to the number of sensors (NS), is computed. Then, the bus combination with highest CCV will be chosen.

As shown in pseudocode of our exhaustive algorithm in Figure.5.8, we first need to create the grid structure based on the given size for each cell by using the latitude and longitude of the area. Next, the algorithm generates the matrix W, where the weights associated with each grid are provided by domain scientists based on the target phenomena to be monitored. Besides, this algorithm creates all the possible bus selections with X different buses where X is equal to or less

than the number of sensors. Furthermore, it calculates the set of time slots within the total sensing period. All the data mentioned above provide the inputs of our primary exhaustive function. This main function is responsible for choosing the most optimal bus selection, which has the highest spatial and temporal coverage.

This algorithm calls two other functions. The first function, which is called $CCV\_Calculation$, determines the cumulative coverage value earned by each given bus selection by looping through the set of buses, the cells within the grid structure, and the weights associated with each grid cell. Furthermore, it calculates the minimum coverage value during different time slots for each grid cell. The second function, called SBS, chooses the best selection by applying the objective function. In other terms, it finds the bus selection with the highest cumulative coverage value, and if this value happens to be the same for more than one selection, it chooses the selection which its minimum coverage value is maximum.

Although this method is computationally expensive (its runtime grows factorially in terms of the number of bus combinations), it is guaranteed to choose the best possible bus combination where the cumulative coverage value is greater than all other bus selections.

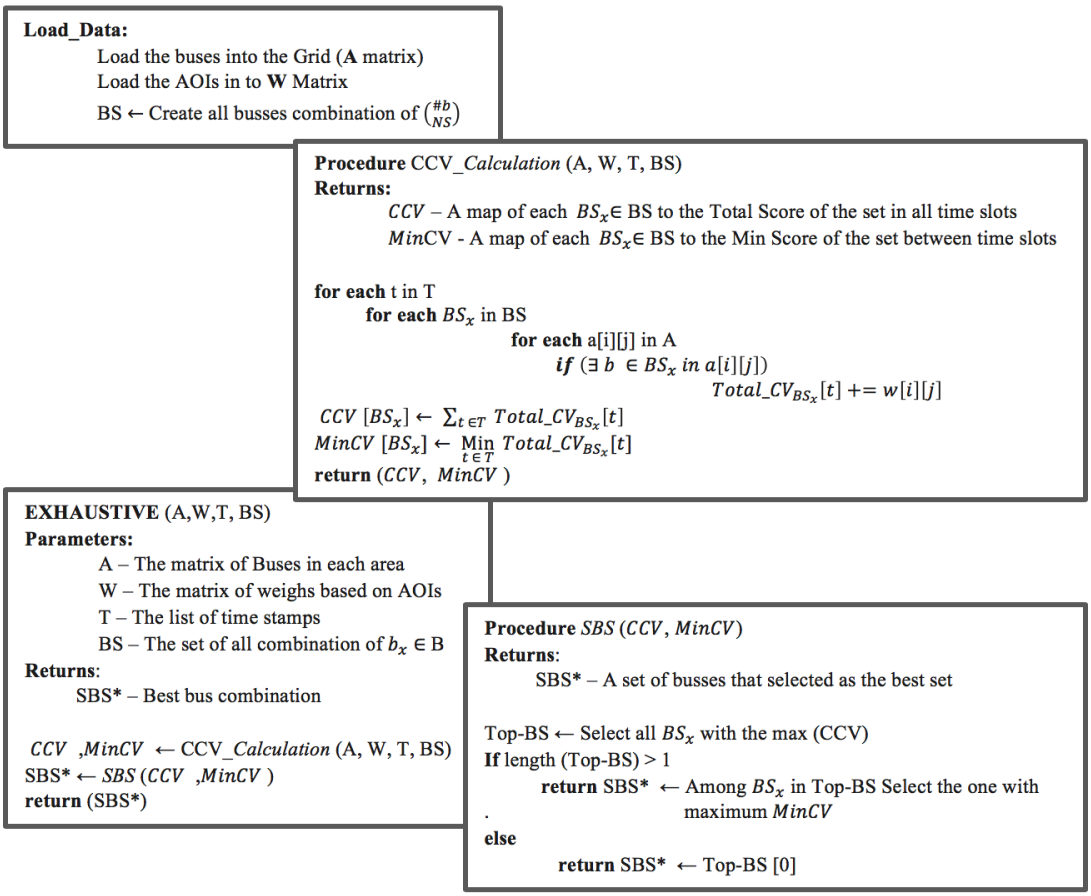


Figure 5.8: Pseudocode of Our Exhausitve Approach

5.4 COST-AWARE APPROACHES

Considering that the exhaustive approach calculates all the r–combinations of the set of buses where r is the limited number of sensing devices, running the algorithm for large data sets leads to extremely long processing time. There are many applications where the sensing parameters, such as the coverage values associated with each hotspot, changes quickly. Thus, we have to unmount and mount our

sensors on a new subset of buses to monitor the target environmental features in a dynamic setting. For instance, a football game may necessitate extra surveillance coverage. Therefore, there should be mechanisms to select an optimal subset of buses to mount surveillance cameras and monitor the areas around the stadium for that particular day. As a result, there is a need for cost-aware approaches with a fast decision process to choose the optimal subset of public transportation vehicles to cover the target areas.

To provide a better understanding of the scale of real-world applications, Table 5.6 provides the number of buses in some selected cities around the world. It also represents the number of different bus combinations if 5%, 10%, or 20% of the buses were supposed to be selected. For instance, there are 639 buses in Atlanta. If we want to select 32 buses out of 639 which traverse around this city, we need to calculate the cumulative coverage value of around 1.03E+54 different bus selections.

Furthermore, Figure. 5.9 provides a graph in blue showing how the number of combinations grows compared to the graph in orange which represents the linear growth. The huge growth of bus combinations makes the exhaustive approaches impractical for real-world applications.

Table 5.6: Combinations of Different Bus Selection in Selected Cities

City	Number of Buses	Number of Sensors (~5%)	Number of Combinations	Number of Sensors (~10%)	Number of Combinations	Number of Sensors (~20%)	Number of Combinations
Atlanta	639	32	1.03E+54	64	1.1E+89	128	3.8E+137
New York City, Tokyo	1400	70	2.4E+119	140	1.6E+196	280	4.7E+302
Washington	1500	75	9.9E+127	150	2.09E+210	300	2.4E+324
Los Angeles	2400	120	3.1E+205	240	1.9E+337	480	7.6E+519
Karachi	7400	370	2.0E+636	740	8.5E+1042	1480	1.8E+1606
Beijing	24347	1217	4.7E+2096	2435	3.7E+3435	4869	5.1E+5288

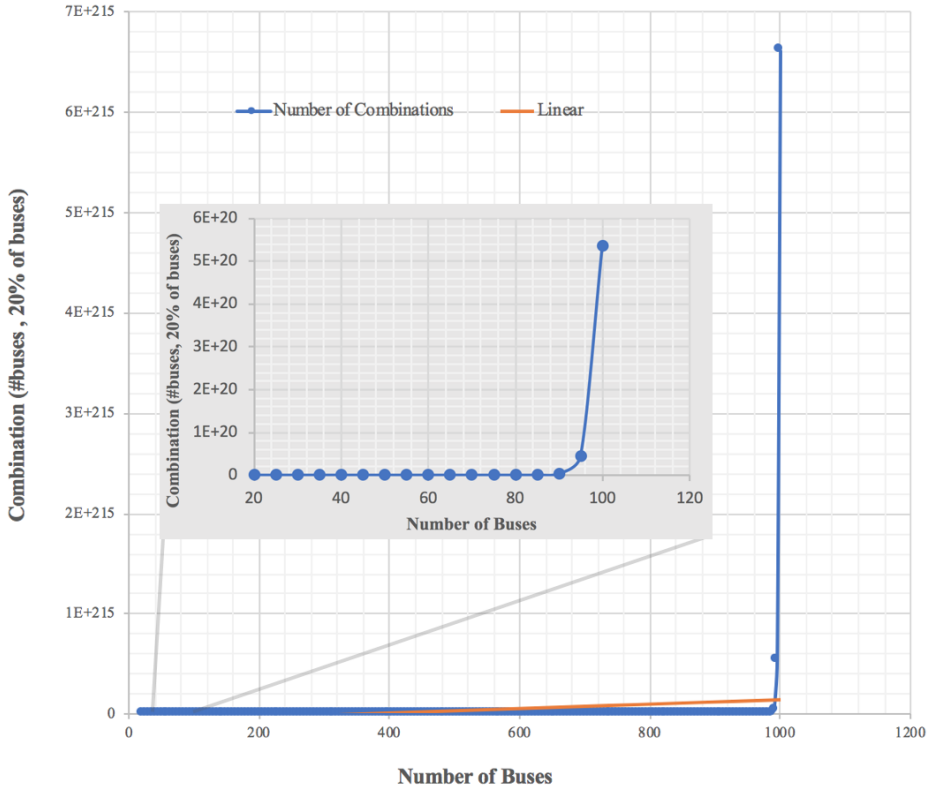


Figure 5.9: Growth of Bus Selection Combinations

To resolve the limitations associated with the exhaustive approach, in this section, we propose our cost-aware sensing approach, which can be used by various sensing frameworks where selecting a subset of vehicles is required.

5.4.1 HOTSPOT-BASED APPROACH

The hotspot-based approach is designed based on the relative importance of various areas in an urban region. The importance of a particular area is indicated by the weight (w_{ij}) assigned to the corresponding grid cell. The default weight of each grid cell is assumed to be one.

In this approach, instead of running the aforementioned exhaustive algorithm, we only consider cells that are hotspots, i.e., cells that correspond to areas with higher importance levels as indicated by their respective weight values. The threshold of the weight values for a cell to be considered a hotspot is a configuration parameter, and it is specified at the time of running the algorithm.

In other words, this approach excludes buses that do not pass through any hotspot cells. We then execute the exhaustive algorithm on the reduced set of buses. Excluding buses that do not pass through hotspots significantly reduces the number of bus combinations that need to be considered, thus making the algorithm more efficient.

As shown in pseudocode of our hotspot-based algorithm in Figure.5.10, the grid structure, the matrix of weights, the list of timestamps, and the list of all available buses are fed into our main function. Next, it filters the list of available buses to exclude the ones which do not pass any of the hotspots. Then, it generates

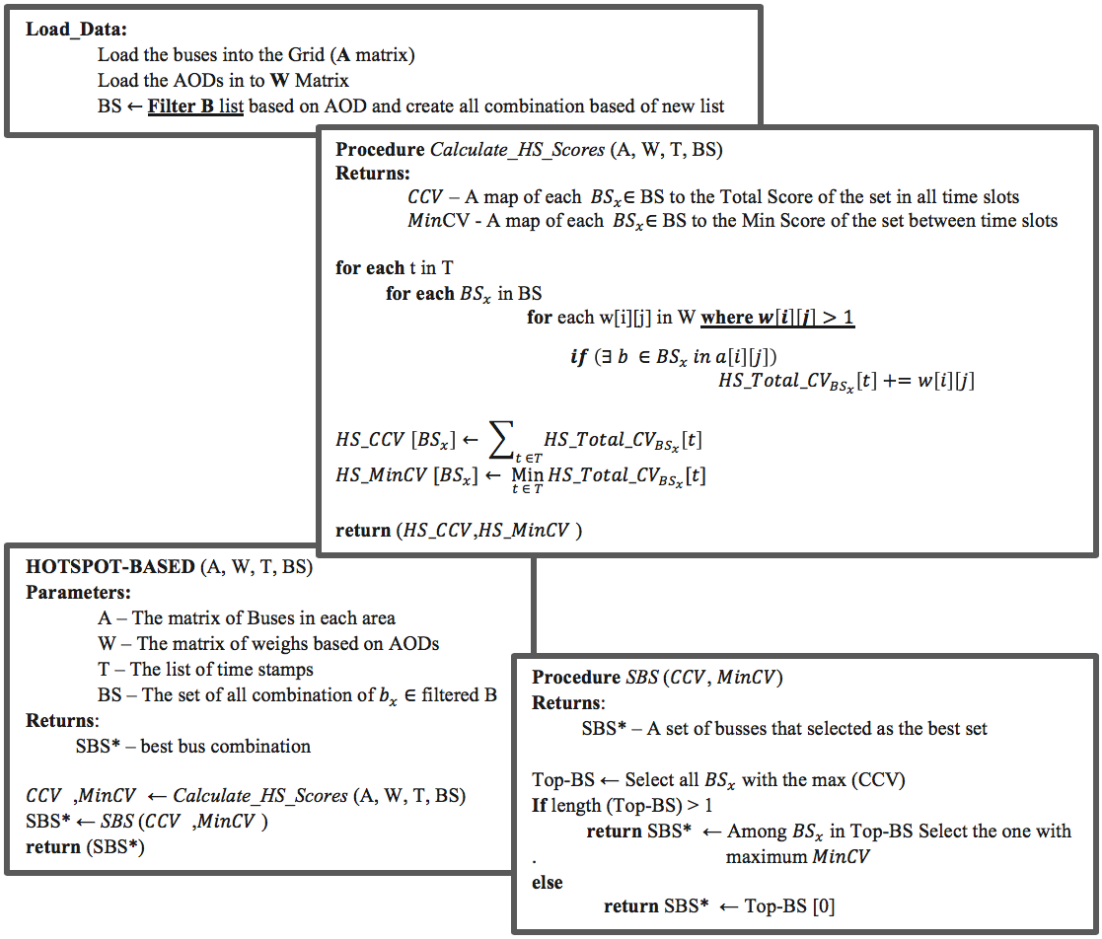


Figure 5.10: Pseudocode of Our Hotspot-based Approach

all the possible combinations from the updated list of buses. The main procedure of this algorithm invokes two other functions to calculate the new coverage values and choose the best selection.

The method called *Calculate_HS_Scores* calculates the new cumulative coverage value for each bus selection. This function loops through the set of buses, the

hotspot cells within the grid, and the matrix of weights. Besides, it calculates the minimum coverage value during different time slots for each hotspot cell. On the other hand, SBS select the most optimal subset of buses by comparing the new cumulative coverage value associated with each bus selection. It also considers the second condition of our objective function for cases where more than one bus selection gained the highest cumulative coverage value. In that condition, this function chooses the selection which its minimum coverage value is maximum.

Our hotspot-based approach is considered as the first stage of our solution. In other terms, this algorithm can be either used standalone or act as the initial step of our near-optimal solution, which runs slightly slower, but is more rewarding. Our genetic algorithm, which is founded on top of this hotspot-based algorithm, is discussed in the following section.

5.4.2 COST-AWARE GENETIC ALGORITHM

Considering that our hotspot-based approach only focuses on the hotspot locations to make the decision, we propose a genetic algorithm approach geared toward our coverage maximization problem, which allows the algorithm to consider locations other than the hotspots. This algorithm uses the output of our hotspot-based algorithm as its input and provides the algorithm with the chance to explore the bus selections which have not passed any hotspot, but gained higher total coverage values.

In our genetic algorithm, chromosome representation is used to encode the candidate buses to be chosen for a bus selection. For instance, if we wanted to

select three buses out of twenty, we would have a chromosome representation like that of Figure 5.11, where there are exactly three 1s. This condition guides the algorithm to always select the number of buses proportional to the number of sensors.

b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9	b_{10}	b_{11}	b_{12}	b_{13}	b_{14}	b_{15}	b_{16}	b_{17}	b_{18}	b_{19}	b_{20}
1	**1**	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 5.11: A sample chromosome representation

The crossover (recombination) operator combines the genetic representation of two parents to create a new generation. In our design, the algorithm randomly selects a single crossover point in the chromosome representation of the two parents and recombine them like the example shown in Figure 5.12. Thus, the bits to the right of the selected crossover point are swapped between the two parent chromosomes to generate two new chromosomes.

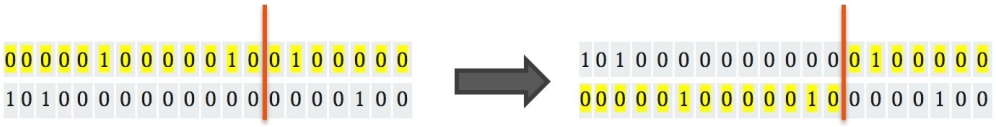


Figure 5.12: A sample crossover operation

After each crossover operation, our algorithm checks whether the number of 1s in each new child chromosome still corresponds to the number of sensors or not. If the condition is not met, the mutation operation will be used to randomly flip bits in each child until the condition is satisfied. Figure 5.13 depicts a crossover which invalidates the condition as mentioned earlier; therefore, the mutation operation comes into the picture to solve the inconsistency.

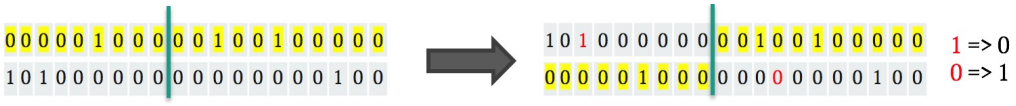


Figure 5.13: A sample mutation operation

The fitness function is the same as the objective function of our exhaustive algorithm. So, the fitness function is to find the bus selection with the highest consecutive coverage value. Moreover, if the consecutive coverage value happens to be the same for different bus selections, the one which its minimum coverage values in different time slots are higher than the others will be chosen as the best selection.

In the selection and replacement phase, our initial population is chosen from the results of our hotspot-based approach. In other terms, we first calculate the actual consecutive coverage value for the bus selections generated by the hotspot-based approach and sort them. Next, based on the experimental setup, we select our initial chromosome population from the sorted list. Then, in each iteration, based on the coverage values, we discard the worst 20% of the population and replace them with new children generated from the parents coming from the top 20% of the population.

As shown in the pseudocode of our cost-aware genetic algorithm in Figure.5.14, the grid structure, the matrix of weights, the list of timestamps, the list of all available buses, and the number of sensors are the input parameters for this algorithm. The algorithm initializes with the population of bus selections which are the top selections of our hotspot-based algorithm. Next, it performs the crossover operation based on the replacement criteria (and mutation, if required). The crossover

and mutation steps provide the algorithm with the chance to consider new bus selections that were disregarded by the hotspot-based algorithm. The coverage values for the new generation is recalculated, and the algorithm continues the iteration until the termination condition is satisfied. We designed the termination condition using the OR operator. We stop the iteration if the algorithm starts converging; otherwise, it continues until the assigned iteration threshold is met.

It should be mentioned that our genetic algorithm design can easily integrate an incremental setting of sensors. In other words, if there would be a scenario where there is a need to mount new sensors, while there are already some sensors mounted on the buses, we can fix the bits (1s) associated with those buses in the chromosome representation throughout the selection and replacement phase.

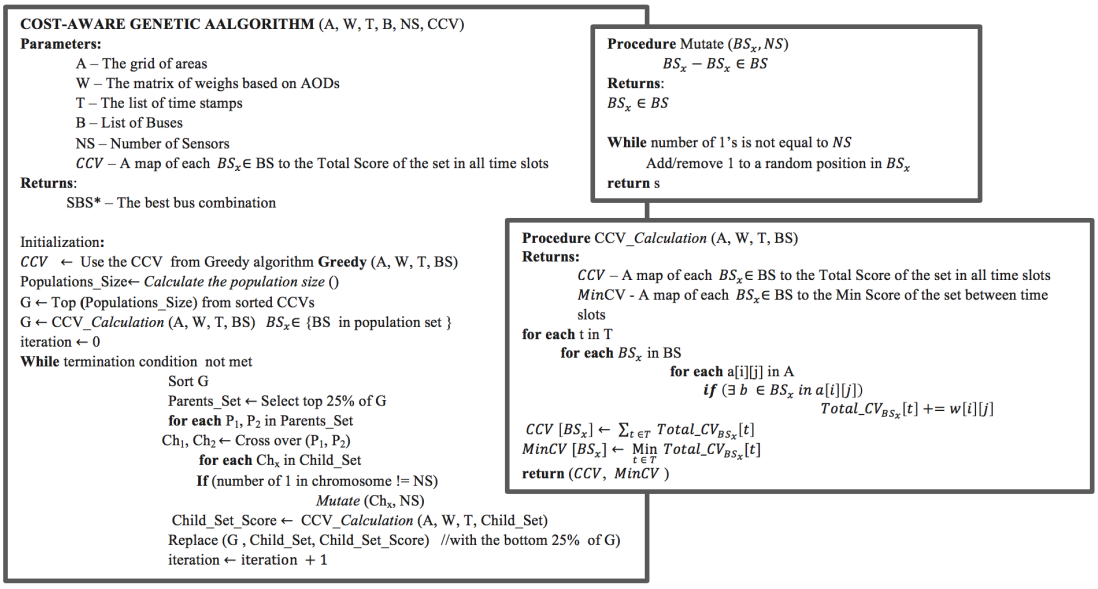


Figure 5.14: Pseudocode of Our Hotspot-Based Genetic Algorithm Approach

5.5 EVALUATION EXPERIMENTS

5.5.1 EXPERIMENTAL SETTINGS

To solve the coverage maximization problem, we create a grid covering the whole area while each cell corresponds to a 90-meter by 90-meter area on earth. In this case study for urban heat analysis, the hotspots and their corresponding weights are determined by the heatmaps generated from satellite imagery. In other terms, the importance of each hotspot to be targeted by drive-by sensing vehicles is defined by analyzing the history of heatmaps generated by Landsat 8 satellite imagery.

Considering that all the bus routes were within the Athens Clarke County, we chose the four corners of the triangle to represent the boundaries of our grid as depicted in Figure 5.15. Then, we created a grid like that of Figure 5.16 to cover the whole area.

In this experimental setup, we assume that there are only three sensors and tested our proposed algorithms on 5-hour trajectory data collected from twenty buses of the Athens Transit public bus system. Our real-world dataset has more than 61,000 data points and provides the GPS data of each bus every 5 seconds. In the setup, there were seven different hotspots, and their weights varied between 2 to 8. It should be mentioned that these hotspots covered less than 0.075% of the whole grid. Some selected hotspots in Athens Clarke County area and the weights assigned to each of them is depicted in Figure 5.17.

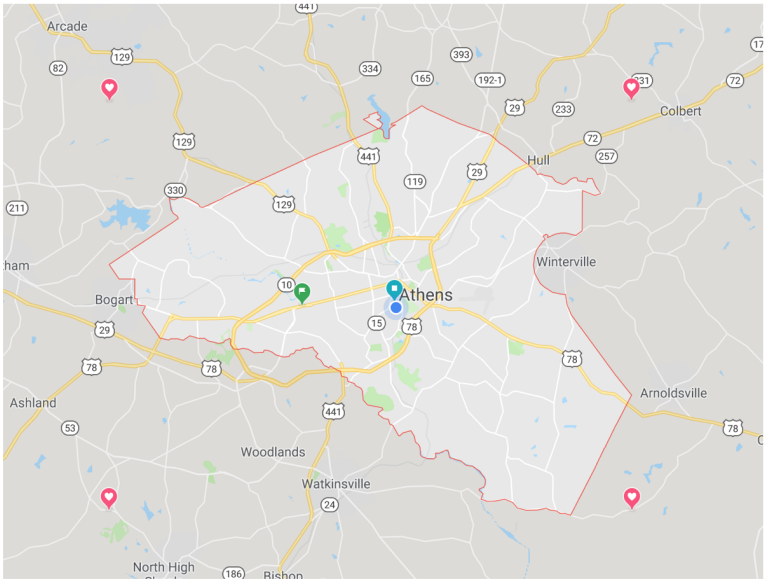
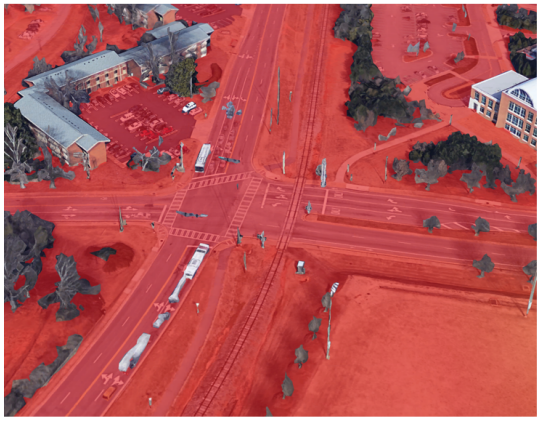


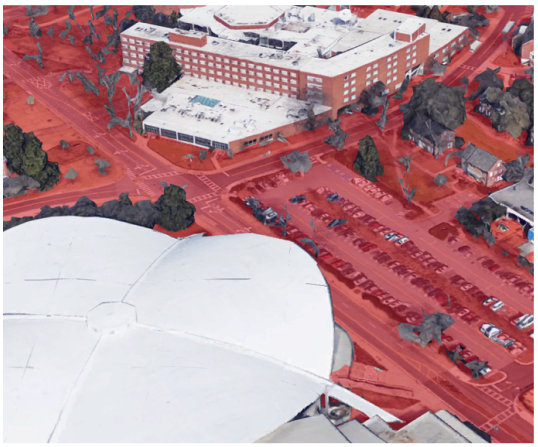
Figure 5.15: The Boundaries of Our Grid for Athens Clarke County

Figure 5.16: The Grid Structure for Athens Clarke County



UGA Health Center Intersection, W=8

Five Points Intersection, W=8



Georgia Center Intersection, W=6

Athens Regional Medical Center, W=6

Bolton Dining Commons, W=4

E Broad – S Lumpkin Intersection, W=2

Figure 5.17: Selected Hotspots (AOIs) in Athens Clarke County

5.5.2 RESULTS

We tested the three approaches called: 1) the exhaustive approach, 2) the hotspot-based approach, and 3) the cost-aware genetic algorithm approach on the real-world data collected from the city buses of Athens Clarke County. In this section, we compare the results of different approaches.

Figure 5.18 shows the result of our exhaustive algorithm on all the possible bus combinations, i.e., the combination of 20 buses taken 3 at a time which is equal to 1140. The x-axis represents the CCV range, and the y-axis represents the number of bus selections which belong to each range.

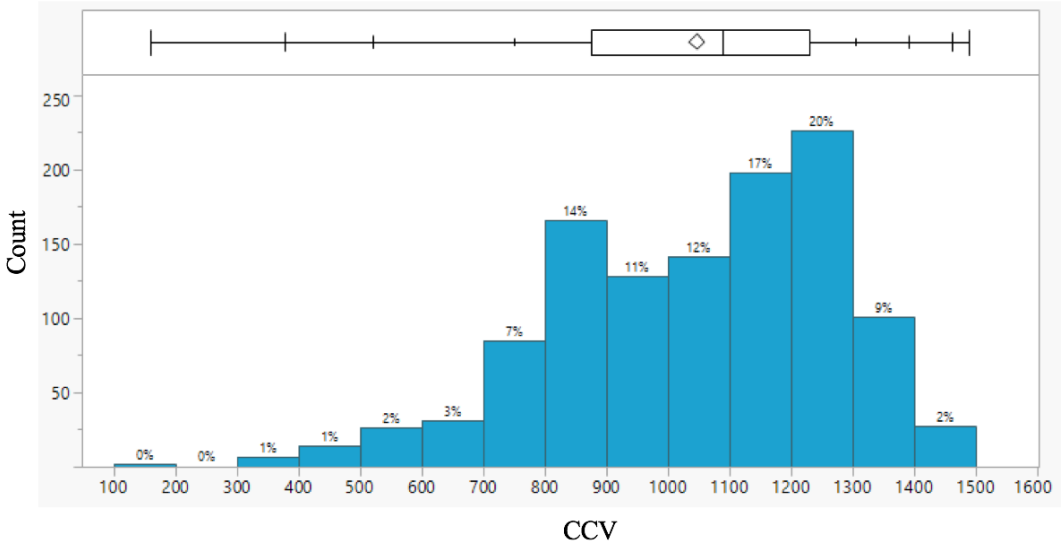


Figure 5.18: Results from the Exhaustive Approach

The exhaustive algorithm chooses the best bus selection $(B741, B764, B766)$ that gained the highest CCV of 1489. Figure 5.19 shows the trajectory data of this bus selection on the map, which illustrates how well our proposed objective

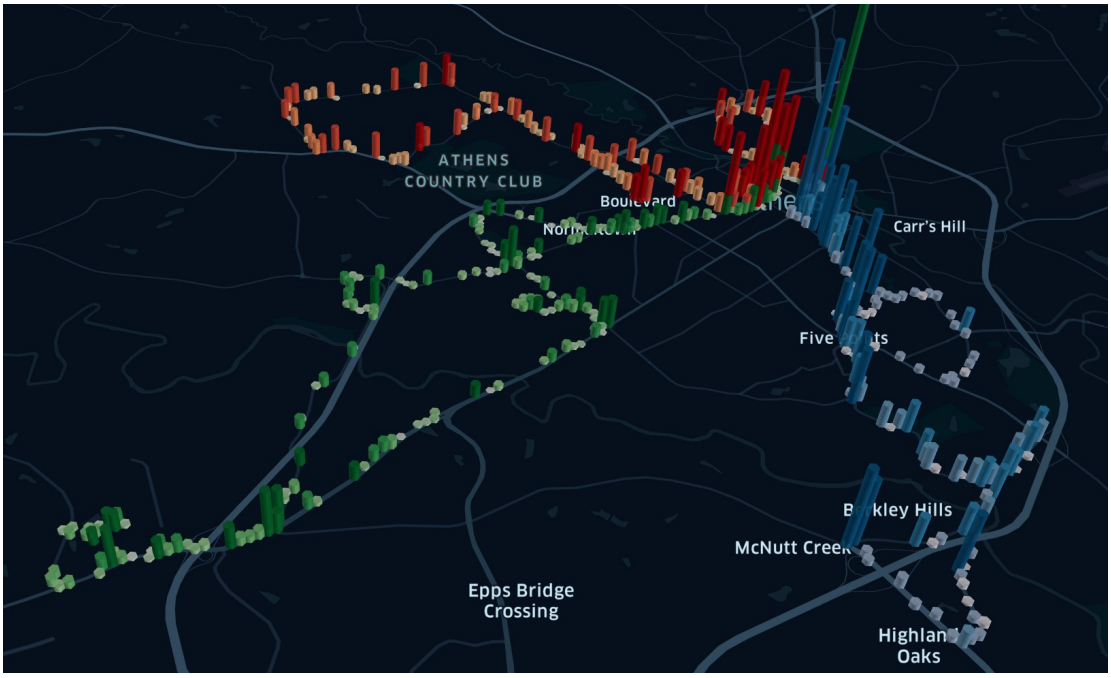


Figure 5.19: Trajectory Map of the Bus Selection: {B741, B764, B766}

function was able to select buses with the highest spatiotemporal coverage and the lowest amount of overlaps. The height of the bars in this map depicts the frequency of GPS readings in those cells.

Figure 5.20 depicts the result of our hotspot-based approach. In this algorithm, only the grid cells which correspond to hotspots are counted, and all other cells are assigned to have a weight of zero. Therefore, the graph shows lower CCVs. This approach selects a bus selection $(B753, B754, B766)$ with the CCV of 1411 (the hotspot-based CCV is 460). Although this CCV is not as good as the one chosen by the exhaustive algorithm, it runs more than 79 times faster. Considering that

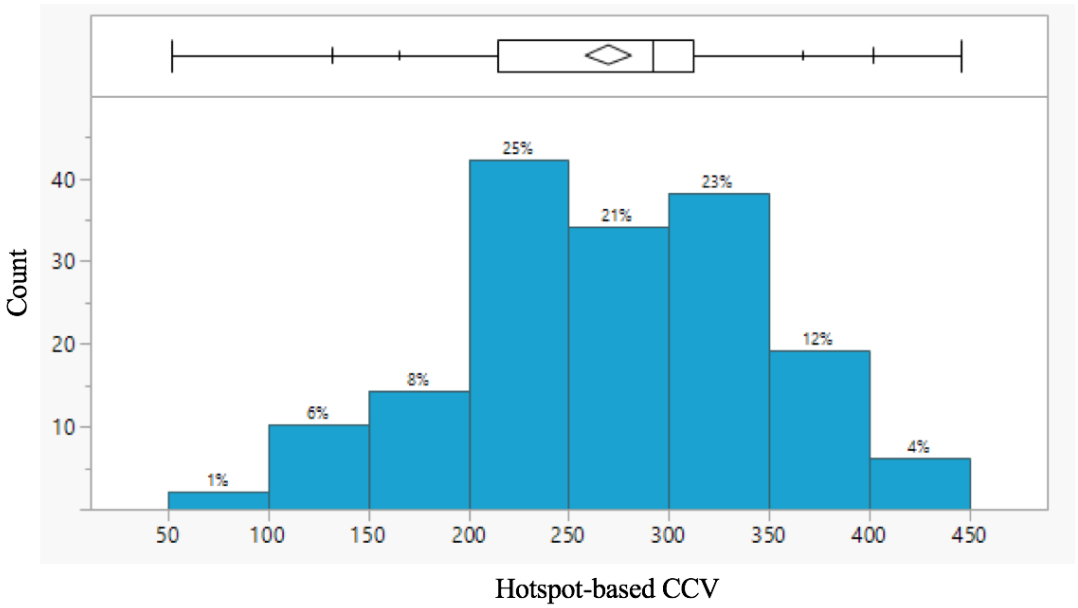


Figure 5.20: Results from the Hotspot-based Approach

the exhaustive approach has a runtime with factorial growth (in terms of number of bus combinations), the efficiency of our hotspot-based algorithm is even more pronounced as the size of our dataset grows.

Figure 5.21 depicts the final population of an example run of our genetic algorithm. We set the population size to be 40. Although we specified the maximum iteration of 20, on average the algorithm converged after 7 iterations. Our algorithm was able to find five bus combinations (black dots) with higher CCVs compared to the best selection in the initial chromosome population (red dots). Therefore, it was able to increase the CCV from 1411 to 1446. In this experimental setup, our algorithm works 6 times faster than the exhaustive algorithm.

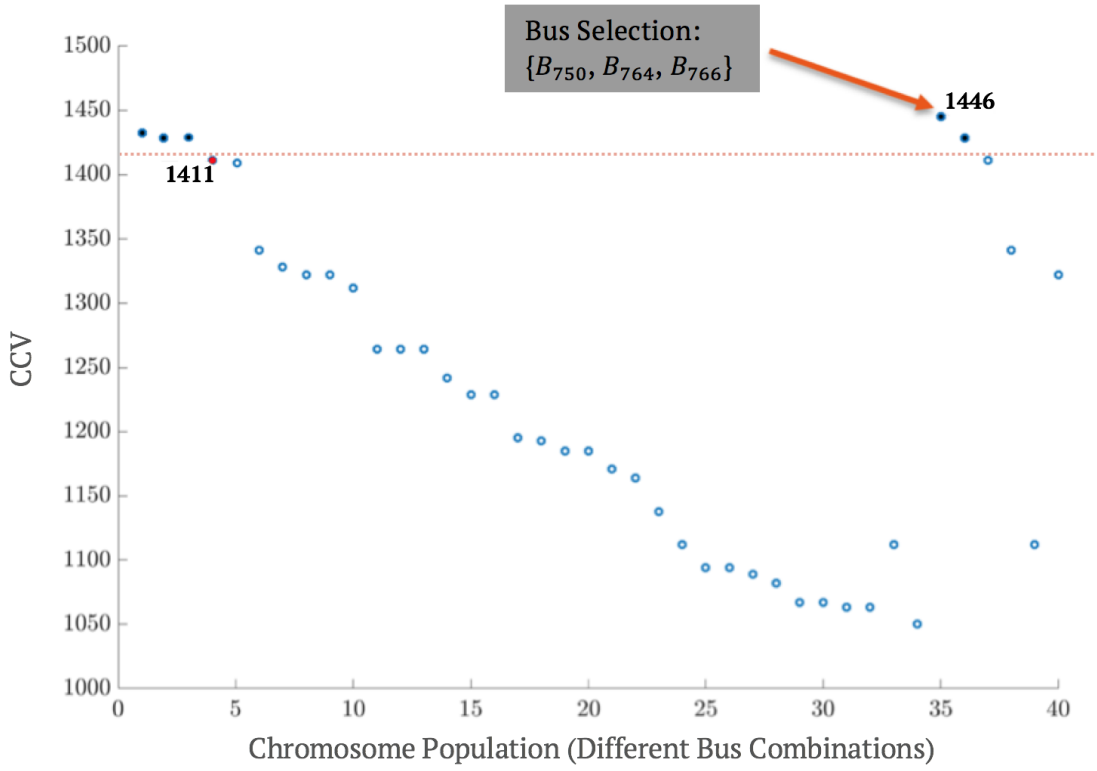


Figure 5.21: Visualization of an Example Run of the Cost-Aware Genetic Algorithm

In order to compare the performance of our three algorithms, we tested them for a varying number of sensors (3,4, and 5). In other terms, we ran the algorithms for different bus combination sizes. Figure 5.22 shows the cumulative coverage value earned by each algorithm. Although the exhaustive approach provides the highest CCV, the coverage values gained by the other two algorithms are comparable. On the other hand, the runtime of the two latter approaches is considerably lower. Figure 5.23 depicts the runtime comparison for these four

bus combination sizes. It shows how fast the runtime of our exhaustive approach grows compared to the other algorithms. Furthermore, we tested the algorithms for a varying number of buses (20, 16, 12, and 8) and a fixed number of sensors (4). Figure 5.24 shows the cumulative coverage value earned by each algorithm, and Figure 5.25 shows the runtime comparison of the three algorithms.

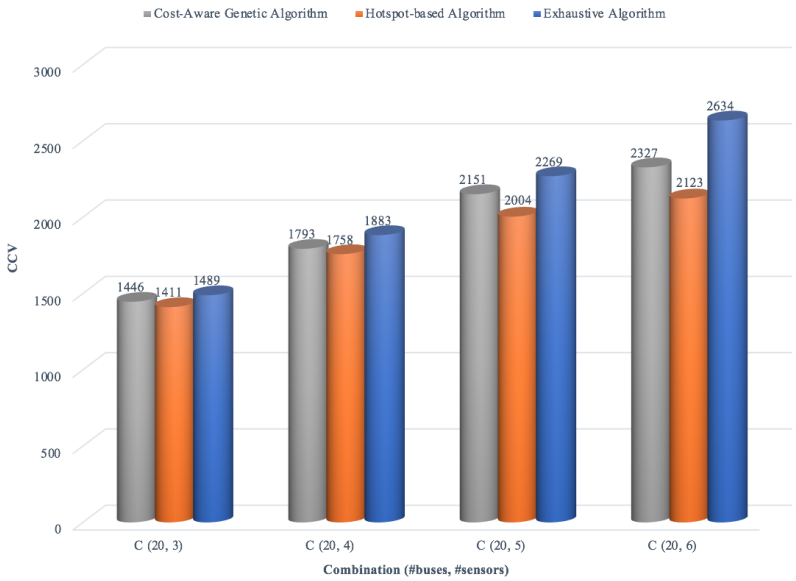


Figure 5.22: CCV Comparison for Different Number of Sensors

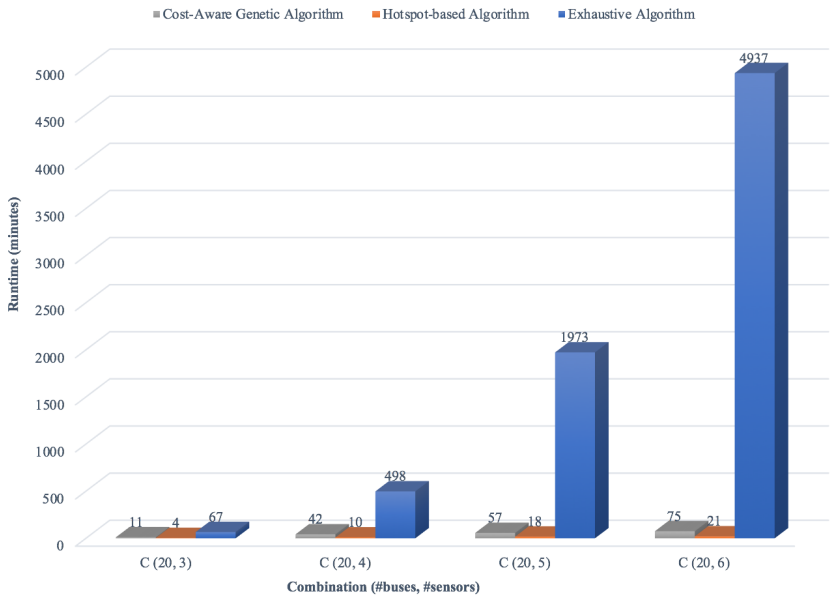


Figure 5.23: Runtime Comparison for Different Number of Sensors

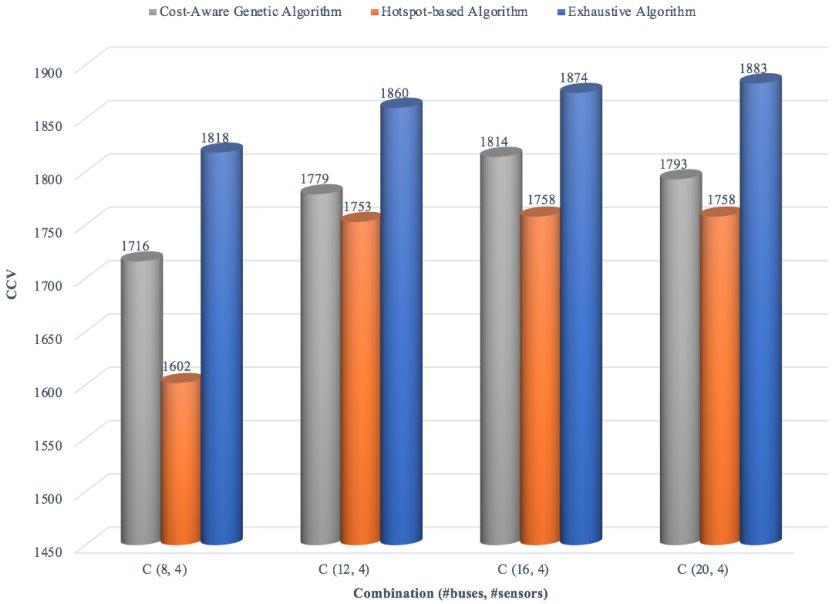


Figure 5.24: CCV Comparison for Different Number of Buses

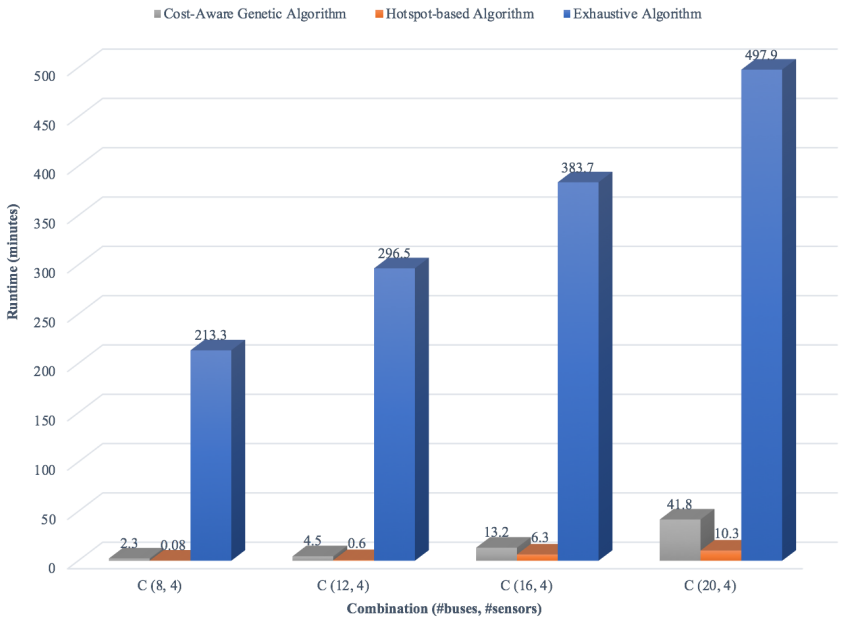


Figure 5.25: Runtime Comparison for Different Number of Buses

To elaborate on the efficiency of our proposed genetic algorithm with a near-optimal solution, Table 5.7 provides the runtime data under two conditions: selecting 3 buses out of 20 versus selecting 5 buses out of 40 (we have 40 buses just in Athens Clarke County). It also provides the runtime results while the algorithm ran in parallel on 6 cores. For instance, we can see that in the parallel mode, our cost-aware genetic algorithm runs 144 times faster than that of the exhaustive algorithm.

Table 5.7: Runtime Comparison of the Algorithms

		One Core Runtime (Minutes)			Six Core Runtime (Minutes)		
	$\binom{\#Buses}{\#Sensors}$	Exhaustive	Greedy	Genetic Algorithm	Exhaustive	Greedy	Genetic Algorithm
Athens Clarke County	$\binom{20}{3}$	282.5	4 (~70 times faster)	65.6 (~4 times faster)	66.6	3.8 (~18 times faster)	7.6 (~9 times faster)
	$\binom{40}{5}$	189849	124.6 (~1163 times faster)	2033.3 (~71 times faster)	44763	117.8 (~290 times faster)	236.7 (~144 times faster)

5.6 DISCUSSION

Greedy approaches are based on the intuitive decision-making heuristic of choosing the best solution at each step. These approaches are promised to find locally optimal choices while they might fail to find the globally optimal solution. Although these algorithms can be quite successful in solving some problems, their solutions may not lead to the best global answer. One way of addressing the bus selection problem would be a greedy strategy. The algorithm can be designed as follows: in the first step, the bus which goes through the highest number of hotspots is selected. In the next step, the algorithm chooses the second bus from the set of remaining buses. This selection is made in a way to maximize the number of sensed hotspots, excluding the ones which were already covered by the first bus. This approach follows the same logic to choose one bus at a time until it reaches the limit imposed by the number of sensors.

5.7 CHAPTER SUMMARY

There are various monitoring applications in which continuous sensing, both spatially and temporally is essential. Rather than implementing the sensing infrastructure, we can leverage available platforms such as public transportation systems which can carry different sensors and provide us with a systematic data collection paradigm. Drive-by sensing paradigm provides many sensing opportunities, especially in urban areas where the routes of public transportation vehicles are close to the daily commute paths of city residents. So, drive-by sensing can be a reliable substitute for crowd sensing. Unlike crowd sensing frameworks in which the mobility patterns of the agents can be unpredictable, drive-by sensing through public transportation allows us to have much more consistent monitoring of the target phenomena in cities.

In this chapter, we first defined and formulated the problem of choosing a subset of buses as an optimization problem. Our objective function is implemented using three different algorithms: an exhaustive approach, a hotspot-based approach, and a cost-aware genetic algorithm. Then, we compared their performance and provided experimental results. We showed how the genetic algorithm outperforms in terms of runtime as the size of our dataset grows. For instance, if we want to select 15 buses out of 60, the number of calculations required for the exhaustive approach will be more than 13 million times higher than that of the genetic algorithm. Therefore, the latter approach is very instrumental for real-world applications where the target hotspots are dynamic.

Chapter 6

CONCLUSIONS

As a result of the rapid growth of buildings, depletion of green cover, and climate change, extreme heat events are posing an increasing threat to many urban communities around the world. Subsequently, deadly heat hazards are becoming more common, and heat-related morbidity and mortality are increasing in different urban areas. So far, most urban heat vulnerability studies have focused on generating low-resolution heat maps of cities using satellite images to analyze the heat hazards. While some recent works tried incorporating data from the nearest static weather stations, they could not reflect the precise spatial variation of the air temperature in urban areas due to the limited availability of these stations.

We proposed SCOUTS framework to address the limitations associated with conventional urban heat analysis approaches. Besides, we discussed its implementation challenges along with our solutions, followed by some preliminary results from the States of Georgia and Arizona. These results confirm our claims about

the importance of hyperlocal and high-resolution heatmaps due to the high spatiotemporal variability of the ambient temperature in urban areas.

Although crowd sensing is a scalable sensing paradigm, we identified an essential limitation in the temperature data collection, which also applies to many other environmental crowd sensing applications. The limitation is that if participants misplace the temperature sensors in a way that are not exposed to the natural outdoor environment, they report wrong information to the servers. To address this problem, we developed a lightweight model to detect anomalously-placed sensors. Considering that our design is based on analyzing the temperature data, we can leverage the same approach to identify the misplacement of other types of environmental sensors by adding a low-cost temperature sensor to them. To the best of our knowledge, this is the first study on detecting the anomalously-placed sensors in the temperature crowd sensing applications.

Finally, we recognized a problem which is common in various drive-by sensing applications. The problem is how to choose an optimal combination of buses to mount the sensors so as to enhance the spatiotemporal coverage in drive-by sensing. Given the trajectory data of buses and also the boundaries of the hotspot locations in an area, we first parametrize all the variables and formulate the problem as an optimization problem. Next, we define our objective function in a way that considers the hotspots areas where their continuous monitoring is of higher importance. Then, we propose a cost-aware genetic algorithm to choose a near-optimal bus selection with high spatiotemporal coverage.

6.1 FUTURE WORK

Regarding the SCOUTS framework, we were able to design and implement crowd sensing and drive-by sensing approaches and produce high-resolution heatmaps. However, these data are first stored in our servers, and then the heatmaps are produced. There is the potential to implement the streamline of processes in order to come up with a semi-real-time framework to instantly map and analyze the temperature readings from various sensors. As a result, personalized and community-based urban heat hazard notification can be implemented which significantly benefits different at-risk communities.

Regarding our anomalous sensor placement detection approach, although we designed a lightweight algorithm which is suitable for edge devices, we have not implemented the proposed classification models on mobile devices. Therefore, there is an excellent opportunity to develop a mobile application for real-time detection and filtering of the data coming from the wrongly-placed sensors using our proposed models.

Finally, regarding our cost-aware approach to enhance the spatiotemporal coverage in drive-by sensing, our objective function is designed in a way to minimize the location-based overlaps of different sensor readings. However, we have not considered the land-cover type similarities into consideration. In other words, there is a research opportunity to address the same problem by minimizing the readings from similar land surface types. Thus, the objective function can be updated to gear toward covering different land cover types. Considering that the land cover type has a significant influence on the remote sensing-based heatmaps,

this new research opportunity can be very beneficial in finding a correlation between remote sensing-based heatmaps and the heatmaps which are produced from the drive-by sensing data.

Bibliography

D. Adams. *The Hitchhiker's Guide to the Galaxy*. San Val, 1995. ISBN 9781417642595. URL http://books.google.com/books?id=W-xMPgAACAAJ.

A. Anjomshoaa, F. Duarte, D. Rennings, T. J. Matarazzo, P. deSouza, and C. Ratti. City scanner: Building and scheduling a mobile sensing platform for smart city services. *IEEE Internet of Things Journal*, 5(6):4567–4579, 2018.

P. M. Aoki, R. Honicky, A. Mainwaring, C. Myers, E. Paulos, S. Subramanian, and A. Woodruff. *Common Sense: Mobile Environmental Sensing Platforms to Support Community Action and Citizen Science*, 1 2008. URL https://kilthub.cmu.edu/articles/Common_Sense_Mobile_Environmental_Sensing_Platforms_to_Support_Community_Action_and_Citizen_Science/6469922.

R. Bachu, S. Kopparthi, B. Adapa, and B. Barkana. Separation of voiced and unvoiced using zero crossing rate and energy of the speech signal. In *American Society for Engineering Education (ASEE) Zone Conference Proceedings*, pages 1–7, 2008.

M. C. Bernhard, S. T. Kent, M. E. Sloan, M. B. Evans, L. A. McClure, and J. M.

Gohlke. Measuring personal heat exposure in an urban and rural environment. *Environmental research*, 137:410–418, 2015.

V. D. Blondel, M. Esch, C. Chan, F. Clérot, P. Deville, E. Huens, F. Morlot, Z. Smoreda, and C. Ziemlicki. Data for development: the d4d challenge on mobile phone data. *arXiv preprint arXiv:1210.0137*, 2012.

W. Bourgeois, A.-C. Romain, J. Nicolas, and R. M. Stuetz. The use of sensor arrays for environmental monitoring: interests and limitations. *Journal of Environmental Monitoring*, 5(6):852–860, 2003.

P. I. Buckley, P. S. Market, A. R. Lupo, and N. Fox. Further studies of the heat island associated with a small midwestern city. *Atmospheric Science Letters*, 9 (4):226–230, 2008.

G. M. Budd. Wet-bulb globe temperature (wbgt)its history and its limitations. *Journal of Science and Medicine in Sport*, 11(1):20–32, 2008.

M. H. Burke, Jeffrey A.and Deborah Estrin, N. R. Andrew Parker, and M. B. S. Sasank Reddy. *Participatory sensing*, 2006.

F. Campioni, S. Choudhury, K. Salomaa, and S. G. Akl. Improved recruitment algorithms for vehicular crowdsensing networks. *IEEE Transactions on Vehicular Technology*, 68(2):1198–1207, 2018.

V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15, 2009.

L. Chapman, C. Bell, and S. Bell. Can the crowdsourcing data paradigm take atmospheric science to a new level? a case study of the urban heat island of london quantified using netatmo weather stations. *International Journal of Climatology*, 37(9):3597–3605, 2017.

C. R. Dawson. Heavy rail transit ridership report, first quarter 2009. *apta. com. American Public Transportation Association. http://www. apta. com/resources/statistics/Documents/Ridership/2009_q2_ridership_APTA. pdf. Retrieved*, pages 08–16, 2009.

T. G. Dietterich. Machine learning for sequential data: A review. In *Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)*, pages 15–30. Springer, 2002.

S.-B. Duan, Z.-L. Li, H. Wu, P. Leng, M. Gao, and C. Wang. Radiance-based validation of land surface temperature products derived from collection 6 modis thermal infrared data. *International journal of applied earth observation and geoinformation*, 70:84–92, 2018.

A. Engelbrecht. Particle swarm optimization. In *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 381–406. ACM, 2014.

EOS. Modis mcd43a4 imagery, 2013. URL https://eos.com/modis-mcd43a4/. Accessed: July 2019.

E. Erell, V. Leal, and E. Maldonado. Measurement of air temperature in the presence of a large radiant flux: an assessment of passively ventilated thermometer screens. *Boundary-layer meteorology*, 114(1):205–231, 2005.

J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan. The pothole patrol: using a mobile sensor network for road surface monitoring. In *Proceedings of the 6th international conference on Mobile systems, applications, and services*, pages 29–39. ACM, 2008.

C. for Disease Control and Prevention. Natural disasters and severe weather, 2017. URL www.cdc.gov/disasters/extremeheat/heat_guide.html. Accessed: 19 June 2017.

R. K. Ganti, F. Ye, and H. Lei. Mobile crowdsensing: current state and future challenges. *IEEE Communications Magazine*, 49(11):32–39, 2011.

Y. Gao, W. Dong, K. Guo, X. Liu, Y. Chen, X. Liu, J. Bu, and C. Chen. Mosaic: A low-cost mobile sensing system for urban air quality monitoring. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE, 2016.

Y. Genc, Y. Zhu, M. Dejori, and F. Dierkes. Crowd sourcing parking management using vehicles as mobile sensors, Mar. 7 2013. US Patent App. 13/527,114.

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

F. Gouyon, F. Pachet, O. Delerue, et al. On the use of zero-crossing rate for an application of classification of percussive sounds. In *Proceedings of the COST G-6 conference on Digital Audio Effects (DAFX-00), Verona, Italy*, 2000.

K. Grathwohl, S. Scheiner, L. R. Brandt, and A. R. Lupo. Analysis of weather data collected from two locations in a small urban community. *Transactions of the Missouri Academy of Science*, 2006.

D. M. Gubernot, G. B. Anderson, and K. L. Hunting. The epidemiology of occupational heat-related morbidity and mortality in the united states: a review of the literature and assessment of research needs in a changing climate. *International journal of biometeorology*, 58(8):1779, 2014.

B. Guo, Y. Liu, W. Wu, Z. Yu, and Q. Han. Activecrowd: A framework for optimized multitask allocation in mobile crowdsensing systems. *IEEE Transactions on Human-Machine Systems*, 47(3):392–403, 2016.

H. Habibzadeh, Z. Qin, T. Soyata, and B. Kantarci. Large-scale distributed dedicated-and non-dedicated smart city sensing systems. *IEEE Sensors Journal*, 17(23):7649–7658, 2017.

D. G. Hadjimitsis, G. Papadavid, A. Agapiou, K. Themistocleous, M. Hadjimitsis, A. Retalis, S. Michaelides, N. Chrysoulakis, L. Toulios, and C. Clayton. Atmospheric correction for satellite remotely sensed data intended for agricultural applications: impact on vegetation indices. *Natural Hazards and Earth System Sciences*, 10(1):89–95, 2010.

S. L. Harlan, A. J. Brazel, L. Prashad, W. L. Stefanov, and L. Larsen. Neighborhood microclimates and vulnerability to heat stress. *Social science & medicine*, 63(11):2847–2863, 2006.

D. Hasenfratz, O. Saukh, S. Sturzenegger, and L. Thiele. Participatory air pollution monitoring using smartphones. *Mobile Sensing*, 1:1–5, 2012.

Z. He, J. Cao, and X. Liu. High quality participant recruitment in vehicle-based crowdsourcing using predictable mobility. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 2542–2550. IEEE, 2015.

S. Heggen, A. Adagale, and J. Payton. Lowering the barrier for crowdsensing application development. In *International Conference on Mobile Computing, Applications, and Services*, pages 1–18. Springer, 2013.

Z. A. Holden, A. E. Klene, R. F. Keefe, and G. G. Moisen. Design and evaluation of an inexpensive radiation shield for monitoring surface air temperatures. *Agricultural and forest meteorology*, 180:281–286, 2013.

L. Howard. *The Climate of London: Deduced from Meteorological Observations Made at Different Places in the Neighbourhood of the Metropolis. In Two Volumes*, volume 2. W. Phillips, 1820.

B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden. Cartel: a distributed mobile sensor computing system. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 125–138. ACM, 2006.

M. Infosystems. The new york city bus system, 2019. URL https://www.ny.com/transportation/buses/. Accessed: 4 July 2019.

F. Jacob, F. Petitcolin, T. Schmugge, E. Vermote, A. French, and K. Ogawa. Comparison of land surface emissivity and radiometric temperature derived from modis and aster sensors. *Remote Sensing of Environment*, 90(2):137–152, 2004.

J. C. Jiménez-Muñoz, J. A. Sobrino, D. Skoković, C. Mattar, and J. Cristóbal. Land surface temperature retrieval methods from landsat-8 thermal infrared sensor data. *IEEE Geoscience and remote sensing letters*, 11(10):1840–1843, 2014.

M. Jin and R. E. Dickinson. Land surface skin temperature climatology: Benefitting from the strengths of satellite observations. *Environmental Research Letters*, 5(4):044004, 2010.

JPL. Advanced spaceborne thermal emission and reflection radiometer, 2016. URL https://asterweb.jpl.nasa.gov/. Accessed: July 2019.

S. S. Kanhere. Participatory sensing: Crowdsourcing data from mobile smartphones in urban spaces. In *2011 IEEE 12th International Conference on Mobile Data Management*, volume 2, pages 3–6. IEEE, 2011.

N. E. Klepeis, W. C. Nelson, W. R. Ott, J. P. Robinson, A. M. Tsang, P. Switzer, J. V. Behar, S. C. Hern, and W. H. Engelmann. The national human activity pattern survey (nhaps): a resource for assessing exposure to environmental

pollutants. *Journal of Exposure Science and Environmental Epidemiology*, 11 (3):231, 2001.

C. Kuenzer and S. Dech. Thermal infrared remote sensing. *Remote Sensing and Digital Image Processing. doi*, 10(1007):978–94, 2013.

E. Kuras, D. Hondula, and J. Brown-Saracino. Heterogeneity in individually experienced temperatures (iets) within an urban neighborhood: insights from a new approach to measuring heat exposure. *International journal of biometeorology*, 59(10):1363–1372, 2015.

E. R. Kuras, M. B. Richardson, M. M. Calkins, K. L. Ebi, J. J. Hess, K. W. Kintziger, M. A. Jagger, A. Middel, A. A. Scott, J. T. Spector, et al. Opportunities and challenges for personal heat exposure research. *Environmental health perspectives*, 125(8):085001, 2017.

I. R. Lake, N. R. Jones, M. Agnew, C. M. Goodess, F. Giorgi, L. Hamaoui-Laguel, M. A. Semenov, F. Solomon, J. Storkey, R. Vautard, et al. Climate change and future pollen allergy in europe. *Environmental health perspectives*, 125(3):385–391, 2016.

U. Lee and M. Gerla. A survey of urban vehicular sensing platforms. *Computer Networks*, 54(4):527–544, 2010.

C.-M. Li, B. Liu, R.-F. Qin, and N. Yang. An urban mobile monitoring system integrating remote sensing and environmental sensors. In *Design, manufacturing*

and mechatronics: Proceedings of the 2015 International Conference on Design, Manufacturing and Mechatronics (ICDMM2015), pages 510–519. World Scientific, 2016.

G. Luber and M. McGeehin. Climate change and extreme heat events. *American journal of preventive medicine*, 35(5):429–435, 2008.

V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos. Anomaly detection in crowded scenes. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1975–1981. IEEE, 2010.

M.-W. Mak and S.-Y. Kung. Low-power svm classifiers for sound event classification on mobile devices. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1985–1988. IEEE, 2012.

H. Mansoor. Karachi needs 8,000 new buses to ease transport woes, cm told, 2017. URL https://www.dawn.com/news/1323617. Accessed: 4 July 2019.

K. Mao, Z. Qin, J. Shi, and P. Gong. A practical split-window algorithm for retrieving land-surface temperature from modis data. *International Journal of Remote Sensing*, 26(15):3181–3204, 2005.

M. A. McGeehin and M. Mirabelli. The potential impacts of climate variability and change on temperature-related morbidity and mortality in the united states. *Environmental health perspectives*, 109(suppl 2):185–189, 2001.

F. Meier, D. Fenner, T. Grassmann, M. Otto, and D. Scherer. Crowdsourcing air

temperature from citizen weather stations for urban climate research. *Urban Climate*, 19:170–191, 2017.

K. Meters. kestrelmeters.com, 2015. URL https://kestrelmeters.com/products/kestrel-drop. Accessed: 26 June 2019.

A. Middel, N. Selover, B. Hagen, and N. Chhetri. Impact of shade on outdoor thermal comforta seasonal field study in tempe, arizona. *International journal of biometeorology*, 60(12):1849–1861, 2016.

C. Muller, L. Chapman, S. Johnston, C. Kidd, S. Illingworth, G. Foody, A. Overeem, and R. Leigh. Crowdsourcing for climate and atmospheric sciences: current status and future potential. *International Journal of Climatology*, 35 (11):3185–3203, 2015.

A. G. Murray. *World trolleybus encyclopaedia*. Trolleybooks, 2000.

NASA. Landsat 8 oli and tirs, 2019. URL https://landsat.gsfc.nasa.gov/landsat-data-continuity-mission/. Accessed: July 2019.

E. Nosrati, A. S. Kashi, Y. Darabian, and S. N. H. Tonekaboni. Register flooding attacks detection in ip multimedia subsystems by using adaptive z-score cusum algorithm. In *ICIMU 2011: Proceedings of the 5th international Conference on Information Technology & Multimedia*, pages 1–4. IEEE, 2011.

C. of Disease Control and Prevention. About extreme heat, 2017. URL https://www.cdc.gov/disasters/extremeheat/heat_guide.html. Accessed: 26 June 2019.

A. Overeem, J. R. Robinson, H. Leijnse, G.-J. Steeneveld, B. P. Horn, and R. Uijlenhoet. Crowdsourcing urban air temperatures from smartphone battery temperatures. *Geophysical Research Letters*, 40(15):4081–4085, 2013.

M. J. OGrady, C. Muldoon, D. Carr, J. Wan, B. Kroon, and G. M. OHare. Intelligent sensing for citizen science. *Mobile Networks and Applications*, 21(2): 375–385, 2016.

S. P. Parambath, N. Usunier, and Y. Grandvalet. Optimizing f-measures by cost-sensitive classification. In *Advances in Neural Information Processing Systems*, pages 2123–2131, 2014.

O. Pinho and M. M. Orgaz. The urban heat island in a small city in coastal portugal. *International Journal of biometeorology*, 44(4):198–203, 2000.

Z. Qin, A. Karnieli, and P. Berliner. A mono-window algorithm for retrieving land surface temperature from landsat tm data and its application to the israel-egypt border region. *International journal of remote sensing*, 22(18):3719–3746, 2001.

A. Roman. Top 100 transit bus fleets: Despite stimulus, funding issues remain for top 100. *Metro*, 105(8), 2009.

C. W. Schmidt. Pollen overload: seasonal allergies in a changing climate, 2016.

U. P. Service. Us postal service cited by us labor departments osha after heat-related death of medford, mass., mail carrier in july heat wave cited for inadequate heat stress program and communication of heat hazards to carriers, 2003.

URL https://www.osha.gov/news/newsreleases/region1/12162013-0. Accessed: 26 June 2019.

W. M. Service. Washington metropolitan area transit authority, 2019. URL https://www.wmata.com/service/bus/. Accessed: 4 July 2019.

A. Shlain. U.s. climate reference network (uscrn) measurements, Accessed: 2018. URL https://www.ncdc.noaa.gov/crn/measurements.html.

K. Singh and S. Upadhyaya. Outlier detection: applications and techniques. *International Journal of Computer Science Issues (IJCSI)*, 9(1):307, 2012.

M. Sokolova and G. Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, 2009.

N. Steffensen. scikit-learn: machine learning in python scikit-learn 0.16.1 documentation, 2019. URL https://scikit-learn.org.

N. Steffensen. Kestrel drop certificate of conformity, n.d. URL http://bit.ly/2KestrelDROP-Certificate.

H. Taha. Urban climates and heat islands: albedo, evapotranspiration, and anthropogenic heat. *Energy and buildings*, 25(2):99–103, 1997.

K. Tan, Z. Liao, P. Du, and L. Wu. Land surface temperature retrieval from landsat 8 data and validation with geosensor network. *Frontiers of Earth Science*, 11(1):20–34, 2017.

J. Tourist. Tokyo toei buses, 2017. URL https://www.japanvisitor.com/japan-transport/tokyo-buses. Accessed: 4 July 2019.

V. Van Asch. Macro-and micro-averaged evaluation measures [[basic draft]]. *Belgium: CLiPS*, pages 1–27, 2013.

C. Wang, C. Li, C. Qin, W. Wang, and X. Li. Maximizing spatial–temporal coverage in mobile crowd-sensing based on public transports with predictable trajectory. *International Journal of Distributed Sensor Networks*, 14(8): 1550147718795351, 2018.

A. Widodo and B.-S. Yang. Support vector machine in machine condition monitoring and fault diagnosis. *Mechanical systems and signal processing*, 21(6): 2560–2574, 2007.

J. Xiang, P. Bi, D. Pisaniello, and A. Hansen. Health impacts of workplace heat exposure: an epidemiological review. *Industrial health*, pages 2012–0145, 2013.

J. Yang, Q. Liu, W. Dai, and R. Ding. Fluid dynamic analysis and experimental study of a low radiation error temperature sensor. *Physics Letters A*, 381(4): 177–183, 2017.

Y. Yang, X. Liu, et al. A re-examination of text categorization methods. *ACM*, 99(8):99, 1999.

K. Yi, R. Du, L. Liu, Q. Chen, and K. Gao. Fast participant recruitment algorithm for large-scale vehicle-based mobile crowd sensing. *Pervasive and Mobile Computing*, 38:188–199, 2017.

Y. Yu, Y. Zhu, S. Li, and D. Wan. Time series outlier detection based on sliding window prediction. *Mathematical problems in Engineering*, 2014, 2014.

F. Yuan and M. E. Bauer. Comparison of impervious surface area and normalized difference vegetation index as indicators of surface urban heat island effects in landsat imagery. *Remote Sensing of environment*, 106(3):375–386, 2007.

X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao. Incentives for mobile crowd sensing: A survey. *IEEE Communications Surveys & Tutorials*, 18(1):54–67, 2015.

X. Zhang, Z. Yang, and Y. Liu. Vehicle-based bi-objective crowdsourcing. *IEEE Transactions on Intelligent Transportation Systems*, 1(99):1–9, 2018.