THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Edge-Based Internet-of-Things Sensors over LoRaWAN

BOYANG ZHOU
SPRING 2022

A thesis
submitted in partial fulfillment
of the requirements
for baccalaureate degrees
in Computer Science and Mathematics
with honors in Computer Science

Reviewed and approved* by the following:

Sven G. Bilén
Professor of Engineering Design, Electrical Engineering,
and Aerospace Engineering
Thesis Supervisor

Jesse L. Barlow
Professor of Computer Science and Engineering
Honors Adviser

* Electronic approvals are on file.

ABSTRACT

Enabled by advances in sensor and networking technology, the “smart city” concept is being implemented in many important applications for optimizing living environments. Smart city implementations leverage Internet of Things networks to create ecosystems that seek to optimize the ecosystems in which they are deployed. In this thesis, we describe a proof-of-concept Internet of Things network implemented for a university campus, including what are the steps, challenges, and why it is necessary to utilize the idea of smart city to actively monitor the current environment. This thesis provides an overview of our own smart city implementation. Specifically, we describe three types of IoT sensors, namely sanitizer dispenser sensors, air quality monitoring sensors, and room occupancy sensors. We also evaluate the possibility for data fusion and discuss edge computing.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

My utmost thanks and appreciation to

Dr. Sven Bilén

for his generous help, support, valuable advice,

and his assistance at every stage of this thesis

Dr. Anna Mazzucato

for her gracious help since my sophomore year

and guiding my undergraduate career

Dr. Jesse Barlow

for being my honors advisor and his support

My Parents

for their unwavering support of and belief in me

Chapter 1

Introduction

1.1 Motivation

In recent years, the rapid development of electronics, particularly wireless sensors, and cloud-based services has enabled a concept commonly referred to as the “smart city.” Although this term is seeing broad usage, there is no consensus on its definition; hence, we first explore what this term means.

To understand the term “smart city,” we can review other terms used to describe similar concepts. The most common terms used interchangeably with “smart city” are “intelligent city,” “digital city,” and “technicity.” These terms emphasize the technological development of a city, with smart city emphasizing the use of technology to greatly improve and optimize the efficiency of public services. Another term that is frequently used is “sustainable city,” which refers creating a sustainable environment and decreasing pollution, and could refer to the entire city or an area within the city. Lastly, “well-being city” describes maximizing quality of life when developing improvements to the status quo [Dameri, 2013].

To understand further what all these terms mean as they relate to daily life, we can look at the domains into which smart city initiatives are classified. Typically, smart city initiatives can be considered through six dimensions: economy, environment, governance, living, mobility, and people. *Smart economy* refers to the business opportunities that are created or optimized through the use of information technology (IT). This may include anything ranging from manufacturing processes to constructing more effective business models for existing industries. *Smart* *environment* employs technology to monitor resources and adjust to more effectively allocate the resources. *Smart governance* refers to using IT to improve public services and make well-informed decisions. As for living, mobility, and people, all three domains deal with the micro-level. *Smart living* encompasses using IT to analyze and improve an individual's lifestyle; *smart mobility* focuses on enabling better modes of transportation; and *smart people* refers to having better access to resources like education and having more options to boost quality of life, such as work from home instead of physically traveling to the office [Camero & Alba, 2019; Ismagilova et al., 2019].

Though smart city concepts have been implemented across various dimensions and with numerous applications, it is IT that provides the core technical underpinnings for smart city initiatives. One of the key challenges for implementation is the collection and processing of large quantities of data. After deploying sensors, a huge amount of sensor data needs to be handled as large amounts are required to discover and generate implementation solutions. Hence, it is important to have intelligent methods to handle these data. One of the important phases of processing the gathered information is called data fusion.

Data fusion may be categorized into three types (which may also be hybridized and combined): complementary, redundant, and cooperative [Nakamura et al., 2007]. *Complementary* refers to combining small chunks of data to form a larger picture. For example, a single smoke detector may not be able to determine definitively whether or not a room is on fire, but adding a temperature sensor will help form a clearer picture. *Redundant* refers to sensing the same thing more than once to assure the accuracy of data. *Cooperative* implies gathering data from multiple types of sensors and producing new knowledge. For example, a store may recognize that merchandise has been removed from a shelf, but sensors at the checkout counter did not subsequently sense that same merchandise being paid for after two hours. This then may be categorized as a possible shoplifting event. Neither knowing individually that merchandise was removed from the shelf or sensing if merchandise is at the counter can determine whether theft has occurred, but with the two pieces of information together, probable theft can be inferred. Hence, data fusion is a valuable data processing technique. Through data fusion, situations can be discerned that are otherwise impossible or very difficult to do so, meaning a better grasp of the situation can be obtained [Wang et al., 2017].

In addition to the passive collection of data, engaging with the local community is also important for cities to become smarter. In Amsterdam's Climate Street project, in addition to collecting environmental data, its developers also facilitated communication with citizens, which turned out to be quite effective. Within two years, this busy retail street successfully reduced its annual carbon dioxide emissions from 3400 to 1276 tons [GSMA, 2013]. This was accomplished by implementing the following. Electricity meters were interconnected such that supply and demand between different areas can be balanced. Trash bins are connected so that they only are emptied when full. In addition to introducing many smart devices, the government also uses mobile services to interact with the community. The interaction with the local community enabled the developer to better understand what people living in the city need and also helped the citizens understand clearly the project's goals. It turned out that these two-way interactions helped in reducing energy consumption [Buck & While, 2017].

Recently, there has been increased emphasis on the need for smart cities to make more connections with their citizens as they are the ones who drive and benefit from these developments. Many researchers suggest that the planners of smart cities should approach solving their residents' issues through closer interactions and better communication with them, which brings it in line with how the smart city initiative was initially conceived, i.e., to generate more effective solutions to social issues while also satisficing the needs of government and the public [Trencher, 2019].

In addition to analyzing data intelligently, another challenge when designing a smart city is to determine where data should be analyzed so that the least amount of resource can be used. A few years ago, designs for smart cities generally used a data center in the cloud, to which data from sensors were sent and the data were also processed in the cloud. Though relatively straightforward to implement, the often-times high data bandwidths required to get all the data into the cloud was a limiting factor and could also be a bottleneck with respect to real-time service. This is for two main reasons. In addition to the bandwidth required to transport all of the data from the edge to the cloud, processing all that data in the cloud heavily load the systems. One workaround for these two issues is to use edge computing. Rather than sending all data to the cloud to process, we instead could do a portion of the processing work at the edge. Considering the large amount of edge sensors in one system, this could significantly reduce the bandwidth usage and workload in the cloud [Cicirelli et al., 2017]. Edge-computing is further introduced in Section 3.3.3.

It is quite clear that smart cities can and will provide numerous benefits and are needed to address the anticipated increases in urbanization. The populations of big cities increase every year, and the trend shows that this will continue. The UN estimates that, by 2050, nearly 70% of the world's population will be living in urban areas [UN, 2018]. The demand for limited resources and facilities will only increase as the urban population increases. Hence, it is imperative to collect and manage data to better use available resources. As the traditional approach of increased resources to meet the needs of increased populations becomes unrealistic, the smart city approach has been proposed as a solution [Gavalas et al., 2017].

Smart city initiatives have been popularly discussed for over a decade, yet the concept of the smart city is still expanding, with new application areas being explored. For example, the novel concept of “smart floating cities” to address rising sea levels, particularly cities that are already below sea level, and the large number of cities built along waterfronts. The concept of floating cities has been developed to address the shortage of land and other results of climate change, [Kirimtat et al., 2015]. The idea is to develop a city that is eventually self-sufficient through the sustainable management of resources. This also benefits citizens since making a city sustainable can also optimize the living environment [Kirimtat et al., 2020].

As there is no single formula for a smart city, there are many factors to consider in their implementation. One needs to put a number of physical devices into use and a data command center, either a physical one or one on the cloud, to manage all the devices, process all the data, and determine how to react. Smart cities require many sensors collecting data, then these data can be processed with the purpose of taking certain actions.

Seeing the current development toward and the prospect of the smart city, we realized that making our own living area “smarter” could improve people’s lives. In the fall of 2020, after several months of the COVID-19 pandemic and the need to open up again, it was realized that the need and demand for hand sanitizer in public facilities like universities would increase significantly, especially when the fall semester started and the students returned to campus. Relying fully on janitors to check and refill every single hand sanitizer dispenser is an inefficient strategy, as the flow of people through different places in a building could vary significantly, and constantly checking all of them would significantly add to the janitors’ workload. On the other hand, by using the smart city approach, installing sensors in sanitizer dispensers and monitoring the usage of the dispensers could ensure efficiency and safety. Janitors would no longer need to check all dispensers every time and only need to refill ones that are detected to be almost empty. Furthermore, having the data of the usage of a large number of sensors can help us to understand the flow of people inside buildings. With that information, we can then identify potential issues that may occur in the building, such as queuing to enter and leave building staircases. This application above, which is detailed in this thesis, is an example of a smart city approach to addressing pressing issues.

1.2 Thesis Contributions

In describing how we constructed a demonstration IoT sensor network and how we analyzed the collected data, this thesis provides insights on how to implement the smart city approach in a workspace environment. Previous publications mainly focus on the applications of smart city from a commercial and living perspective, but reports of applications on a university campus are fairly limited. As mentioned above, comprehensively converting a place to be “smarter” requires multiple steps. One of the first few steps is to deploy different kinds of sensors and analyze the collected data. This thesis provides an example of how we have planned out the sensors we need to use in accordance with the goals for improvements. It also shows the details of how we have configured the entire network to successfully connect everything and how we processed data to get the most use of it.

1.3 Thesis Outline

The remainder of the thesis is organized as follows. Chapter 2 introduces background on Internet of Things. Chapter 3 describes the three sensor types employed in this project in chronological order. Chapter 4 discusses the possibility of using data fusion to provide unique and valuable insight. Chapter 5 summarizes the thesis and addresses future work that can be developed for the continuation of this project.

Chapter 2

Background

2.1 Internet of Things

To successfully implement smart city initiatives, they must be implemented using appropriate architecture. Useful and accurate data are collected, problems are diagnosed, and solutions are identified and implemented. This process requires lots of data to be collected and analyzed to verify the correct actions are occurring and to continue improving. The need for large streams of data has driven the implementation of Internet of Things [Kirimtat, 2020].

The term Internet of Things, or IoT, was first used over a decade ago by Kevin Ashton, founder of a research group that works on sensor networks. At the time, he defined IoT as “a system where the Internet is connected to the physical world via ubiquitous sensors, including radio frequency identification (RFID)” [Ashton et al., 2009]. Fast forward to today, the term IoT is no longer limited to a system of simple and single-use sensors anymore. As networked systems, especially wireless technologies, develop rapidly, many of the elements of the initial definition have changed as that definition now is too narrow. However, there is still not a definition that is generally agreed upon, despite numerous attempts [DeFranco, 2021]. In order to better understand what the term Internet of Things refers to, we can reference multiple definitions covering various aspects of IoT [Patel & Patel, 2016]. The definition of IoT should include the following elements. First, IoT is the concept of connecting any device to the Internet directly or through/between other devices. Second, data are collected and shared between sensors and/or to a central control system (which could exist in the cloud). Third, the data exchanged are also analyzed, either at the edge or at the central control system. Fourth, these data may trigger events/actions. In short, IoT connects many devices together and data are collected, shared, and analyzed, and subsequently the insight gleaned from the data may trigger event(s).

A typical IoT implementation generally has three layers: sensing, network, and application layers [Nord et al., 2019]. These three layers include the entire data flow cycle, from data collection to data evaluation and processing, and finally making certain changes or improvements to the environment [Yan et al., 2014].

The sensing layer, sometimes also referred to as the device layer, is the layer for all the sensing devices. Typically, an IoT-based sensor will be equipped with a radio frequency modem. Early IoT devices utilized RFID, which can be categorized into passive, semi-passive, and active based on how the radio interface is powered. Passive RFID devices do not have their own power supply. An RFID reader provides the energy needed. Examples of this may include passports and some electronic toll tags. Semi-passive RFID devices power their own microcontroller, but communications still require energy provided by the reader. Lastly, active RFID tags do not obtain any energy needed for communication from the reader. Active RFID devices allow for actively sending messages to a central controlling unit when necessary or requested [Lee & Lee, 2015].

In addition to RFID, some other technologies like Near Field Communication (NFC) and wireless sensor network (WSN) protocols (e.g., WiFi, Bluetooth, LoRaWAN) are also used to connect sensors. NFC enables devices to communicate over a relatively short distance [Faulkner, 2017]. WSNs enable the construction of sensor networks that collect data from the environment and transmit them to a data collector for real-time and post analysis. For industries that rely heavily on real-time information and analysis, WSNs can decrease costs by being able to take immediate action if needed. For example, they are widely used for cold-chain logistics and tracking systems, for which the need to preform preventive maintenance could be assessed from real-time analysis [Lee & Lee, 2015].

Whereas the sensing layer focuses on the sensor end of the network, the network layer acts as a “middleman” in the IoT network, enabling the seamless information sharing with other connected devices. A new challenge for IoT networks is implementation of device-to-device (D2D) communication. Whereas the typical communication path of device to central controller is fairly straightforward, D2D communication remains challenging as existing network layer protocols are not completely compatible and have many limitations when it comes to D2D. Challenges may include addressing, routing, mobility, resource optimization, security, quality of service (QoS), etc. In the following paragraphs, we explore each challenge.

As IoT networks must deal with a large number of devices, each device much be correctly identified, which requires consideration of the following. First, it needs to be flexible enough to allocate addresses at any time. Second, devices that have multiple interfaces need to be equipped with address duplication detection. In other words, it is important that a device having several interfaces knows itself that the same address is being assigned to all interfaces. In general, this means a hardware address is the unique identifier, and not the network address. This can ensure that throughout the network, this one device can be recognized as a single device. Third, since some devices may not need to connect to the network for a very long time, and the time they connect to the network is rather short, address recycling may be supported. This can prevent a large allocation of addresses that are not in continuous use. Lastly, IoT devices should support self-configuration of addresses. This is because some of the devices may be deployed to places where access by humans is not easily possible or, indeed, impossible. Equipped with self-configuration enables the devices to start up to send data and communicate with the network in the shortest time.

Since using D2D communication protocols in IoT networks no longer requires all data to be routed through a central network, traditional routing techniques must be modified. IoT network routing protocols must consider several factors, namely multi-copy (MC) routing, uni-directional (UD) routing, device and network constraints metrics, and information-/data-pulling by devices. MC routing simply means that multiple copies of a same piece of data is needed at the same time, and these copies are sent to various destinations using different routes. UD routing, as the name suggests, only requires routing in one direction. This means that, throughout the transmission process, the acknowledgement message (ACK) that are used by most routing methods is no longer needed. Having UD routing is especially useful when encountering asymmetric connectivity, where transmission power level or interference level in the vicinity of the devices constantly changes. A routing protocol needs to ensure their connectivity is reliable enough, meaning that constrained resources like memory, throughput thresholds, interference, etc. need to be considered. It also needs to allow devices to broadcast a query in the network so that the devices can request data from each other, allowing the D2D communication to occur smoothly [Bello et al., 2017].

Due to the nature of the IoT networks, it is quite common that devices move around. Hence, when designing an IoT network, we also need to ensure stable connectivity can be maintained for mobile devices. Moreover, for any network, it is important to keep transmitted data secure. This is a challenge for IoT networks as the generally limited capabilities of IoT devices are not capable of handling the computational work for strong encryption. Therefore, security protocols that require much less computational power need to be developed and adapted for IoT networks.

QoS on IoT networks should also be guaranteed. This is especially critical for applications that require high reliability like medical monitoring equipment. In order to reach a required QoS level, the following factors should be considered. Due to the nature of IoT networks, in general, one network comprises multiple devices across multiple sensing dimensions. For example, a health monitoring service may include a heart rate monitor, video camera, alarm systems, and several other elements. Each piece of the overall service may have different QoS requirements. Although operated by different applications with disparate QoS requirements, we need to make sure they can work together coherently.

Most IoT networks can generate a significant amount of traffic. This makes the traffic prioritization task even more essential. Some services only require intermittent transmission of data to function correctly, whereas others rely on continuous data feeds. If the intermittent transmissions use almost all network resources for communication, this may lead to services that rely on continuous data transmission to be unable to receive the desired data, which could lead to disastrous results. Hence, it is crucial to set priority levels for all traffic to avoid such situations.

In short, these challenges are mainly caused by the heterogeneous nature of the network environment, the large scale of the network, maintaining devices' reachability while ensuring their mobility, having consistent service with different forms of traffic, etc. [Bello et al., 2016]. While most of them have workarounds, there still is not a perfect solution for D2D communication to solve all issues. The ideal solution for the IoT network layer remains to be discovered.

At the application layer, or what some people also refer to as the management level, the task is to store and analyze the collected data [Bello et al., 2013]. Due to the nature of IoT sensors, it is very easy to generate a huge amount of data. An efficient way to process and save these data is to use cloud computing. By providing certain "pay-as-you-go" access to a shared pool of resources, cloud computing reduces the expense for data management to minimum and, hence, is considered to be an ideal back-end solution for IoT network [Lee & Lee, 2015; Barbosa & Charão, 2012].

Integrating pieces ranging from the sensor end through to the management system, IoT systems help achieve optimization that previously impossible or extremely costly to implement. With sensors and actuators becoming more powerful and portable at the same time, it is easier and easier to implement IoT systems. This helps to innovate many industries, in unprecedented and sometimes unexpected ways.

2.2 Low-Power Wide-Area Networks

The networking protocol we used to connect sensors to the Internet is called Long-Range Wide-Area Network, or LoRaWAN. As described by the LoRa Alliance, LoRaWAN is designed to optimize a Low-Power Wide-Area Network (LPWAN) [Mekki, 2018]. As the name suggests, the two main features of LPWANs are low power, which translates to low bit rates, and wide area, which means they can cover large distances. Low bit rate refers to the transmission rate between sensors and from sensors to the gateway. Depending on the requirements, the transmissions can be as long as 10 km. In addition to these two features, LPWANs are also low cost. Due to the simplicity of its network protocol, using LPWAN largely reduces the hardware design cost and the network cost [Silva et al., 2017]. Because of all these benefits, LoRaWAN is the best choice for constructing our sensor network.

There are primarily two frequency bands that are used for LoRa networks. For our system, the frequency band that we are using is “US915”. Spanning from 902 to 928 MHz, US915 is a relatively wide frequency band with 26-MHz bandwidth available. This frequency band enables efficient transmission of data over long ranges [Hardesty, 2019]. The “433 MHz” band is also another choice, spanning from 433.5 to 434.5 MHz. With the same transmission power, a signal sent from a 915-MHz LoRa antenna travels roughly three fourths of the distance of that from a 433-MHz antenna [Hardesty, 2019]. However, by FCC Regulation 47 C.F.R. §15.231, the use of this frequency band is restricted. This leaves US915 as the only frequency band below 1 GHz. Sub-GHz data communication is what affords LoRaWAN its low power consumption and long-range transmission.

Chapter 3

Types of Deployed Sensors

3.1 Sanitizer Dispenser Sensors

The project in this section was published in Zhao et al. [2021]. A summary is presented below; additional details may be found in the reference.

3.1.1 Initiative

The first phase of the project began in preparation for the 2020 Fall semester at Penn State University Park. At the time, it was several months into the COVID-19 pandemic, and a vaccine was still believed to be a year or so away. Behavior-related mitigation measures were the only approach to effectively protect individuals from infection. According to the CDC, using alcohol-based hand sanitizer to maintain hand hygiene is effective and important for stopping the spread of the virus [CDC, 2021]. At the time, Penn State had decided to install more than 4000 hand sanitizer dispensers on campus, aiming to reduce the spread of the disease. However, managing all these additional facilities looked to be a big challenge, specifically: How to determine if a sanitizer dispenser is full or empty without visiting it? How to determine if a sanitizer needs to be refilled? Due to the huge number of the sanitizer dispensers on campus, how to optimize the schedule and route for maintenance? Lastly, from the usage trend, how can we adjust the station deployment to better accommodate our community?

3.1.2 Installation

Since the dispensers (shown in Figure 3.1) were already distributed around campus, we needed to find a way to install the sensor such that is would not require significant change to the pre-existing facilities but also to be able to collect the information we needed. The sensor we used is the Radio Bridge Dry Contact Sensor (Figure 3.2). Figure 3.3 shows the structure of the box before the installation. As shown in Figure 3.4, we were able to place it into the existing dispenser station due to its small size. In order to detect when a user pushes the dispenser lever, we installed a micro-switch attached to the lever and connected to the sensor module. In such implementation, the battery in the sensor can supply power for 5 to 10 years. The dry-contact sensors are able to detect all the push operations of dispenser and also replacement of the sanitizer bottles by janitors. Inside the sensor, it also contains a magnetic switch for the purpose of status checking.

Figure 3.1 Existing Sanitizer Dispenser

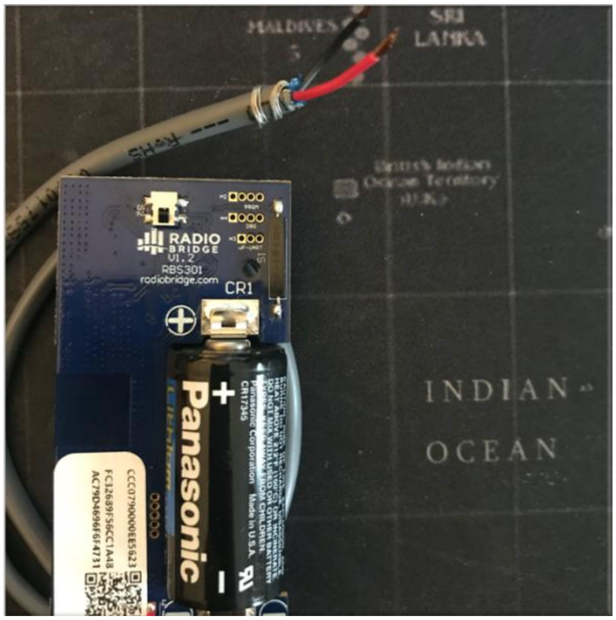

Figure 3.2 Embedded Sensor inside the Dispenser Station 1

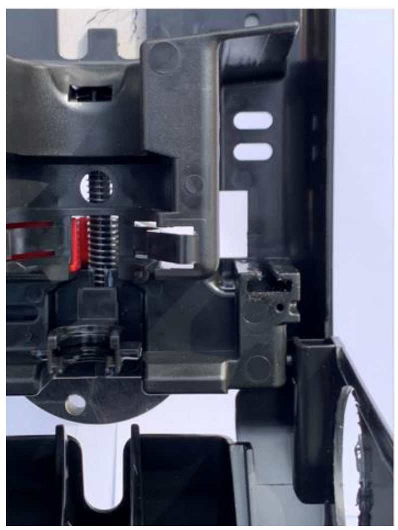

Figure 3.3 Inside the Dispenser Box before the Installation

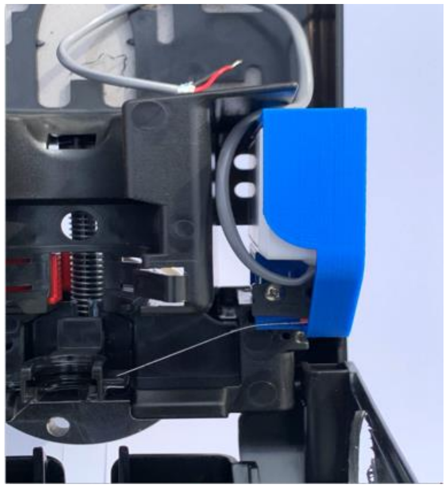

Figure 3.4 Embedded Sensor inside the Dispenser Station 2

For every push event, the sensor sends out a packet of information that includes sensor ID, location, timestamp, and signal quality. Using a LoRaWAN network over a private gateway (Multi-Tech), we are able to receive data directly from the sensors and send the sensor data to a cloud server. As introduced in the previous chapter, LoRaWAN is suitable for use here due to its low data rate and long-range capability. It can cover up to 1-to-3 miles inside buildings and up to 10 miles outside. The gateway we created can also be used for other sensors, and a single gateway can be used for hundreds of end devices. The network server, Radio Bridge Console, is provided by the sensor's manufacturer, and it can be used to store the raw data sent from sensors. Lastly, we can use the application server, TagoIO, in this project to process and analyze the data.

Sometimes misdetections happened at the sensor. The two main causes are 1) the sensors stop recording events when data are sent to the gateway, and 2) the sampling rate is limited to 4 Hz, which is relatively slow. The way to counter these issues is to temporarily save the data in the sensor module and send them out with an interval of one or two minutes. This approach, when compared to real-time, significantly reduced the number of misdetections, as shown in Figure 3.5 below.

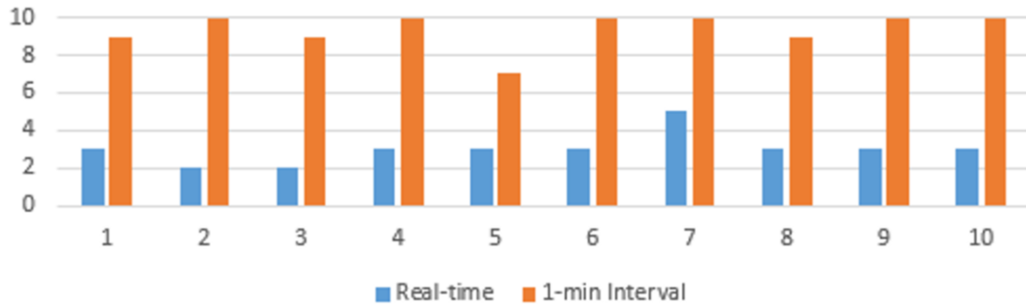

Figure 3.5 Comparison Before and After the Reconfiguration

In December 2020, we installed 24 sensors inside the Hammond Building on three different floors with the gateway installed on the second floor. The operation of sensors worked well, as shown in Figure 3.6 below, with all sensors' RSSI (Received Signal Strength Indicator) greater than −110 dBm. However, the gateway sometimes failed to capture some messages from the sensor modules. Of the 24 sensors we installed, 16 work reliably, one sensor lost connection, and the connections for seven of them are unstable. Most of the time, rebooting the sensor can solve the issue. It also turned out that weak signals may not be the only reason for poor connectivity, as sensors that are far away from the gateway worked fine, yet some sensors that are very near to the gateway have data transmission problems.

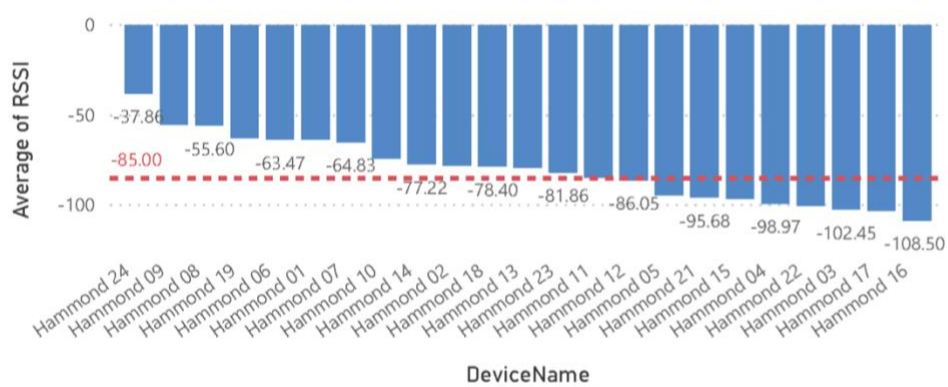

Figure 3.6 Average of RSSI by Device Name

3.2 Indoor Air Quality Monitoring

3.2.1 Initiative

In addition to determining usage of the sensorized dispensers, analysis of these data could, for example, help infer traffic patterns inside the building and overall occupancy. However, this is not very exact, with many uncertainties and possible biases remaining. For example, not all people use the hand sanitizer stations, and those using the sanitizer stations may simply prefer to use a specific dispenser station for other reasons, rather than the first one they see. Thus, in addition to the sensors installed in the sanitizer dispensers, an air quality sensor is introduced to obtain additional data within the building.

According to a report from U.S. Environmental Protection Agency (EPA), Americans spend more than 90% of their time indoors [EPA, 1989]. This makes the study of indoor air quality especially important. The EPA also points out that indoor air quality can affect people's physical and mental health [EPA, n.d.]. The COVID-19 pandemic has also exacerbated this situation. Hence, it is important to ensure adequate indoor air quality. The first step is to deploy indoor air quality monitoring sensors that detect the indexes indicating the air quality, in our case within a classroom. Such a sensor could measure the following, but not limited to, indexes: PM 2.5, ozone concentration, and carbon dioxide concentration. These collected data can signify, for example, whether the occupancy in a room is over the limit and the ventilation system is not able to keep up with the demand. In detecting behaviors like this, we can then assign the activity that was supposed to take place in that room to another room or improve the ventilation system.

3.2.2 Deployment

After researching sensor offerings from multiple companies, we found that LoRaWAN-enabled sensors capable of measuring multiple indoor air quality indexes are quite limited in the market, and those that are available are mostly from abroad and not available directly in the U.S. We eventually selected a sensor from a company called GlobalSat that takes real-time measurements of humidity, temperature, and PM 2.5 (Figure 3.7). This sensor can be connected to our existing gateway in the Hammond Building.

Figure 3.7 The GlobalSat Air Quality Monitoring Sensor

The set up of this sensor was much more complicated than for the dry contact sensor described above. It first requires the sensor connect to a terminal app called RealTerm, then sending numerous commands from this terminal app to the sensor to set up the sensor to work with a given LoRaWAN network, in our case the MultiTech conduit. After the sensor connects to the gateway, we need to connect from the gateway to the server. Namely, adding the new sensor configuration and setting up the data flow inside Radio Bridge Console. Unfortunately, the GlobalSat did not provide detailed instructions on how to set up the sensor and communication with the company yielded very little assistance. Eventually, we decided to forfeit using this sensor due to the over-complicated steps of configuration and the huge amount of wasted time in just trying to figure out how to get it connected. However, other sensors of this type should be procured and evaluated for inclusion in our network.

3.3 Edge-Computing EagleEye Sensor

3.3.1 Initiative

We first focused on determining different environmental parameters inside buildings with the goal of providing insights that could be used to optimize operations within a building. Another important piece of data is to determine directly the occupancy within a room, which is especially useful for shared working spaces. Real-time occupancy data could help people find less crowded areas without visiting every room. Such data could also provide more accurate patterns of people flow within a building, then certain optimizations could lead to better utilization of a given amount of space. In this effort, we selected the ADI EagleEye Sensor occupancy sensor for deployment (Figure 3.8).

Figure 3.8 Operating EagleEye Sensor

3.3.2 Deployment

The set up of the EagleEye sensor requires a mobile phone app that connects the sensor to the phone over Bluetooth. The app configures numerous parameters including the ceiling height, places you want it to detect, places you do not want it to detect, floor area, etc. After connecting to a Wi-Fi network with an open port 8883, the sensor then is able to connect to the Internet. The data are then shown on an online dashboard provided by the vendor, as shown in Figure 3.9 and Figure 3.10.

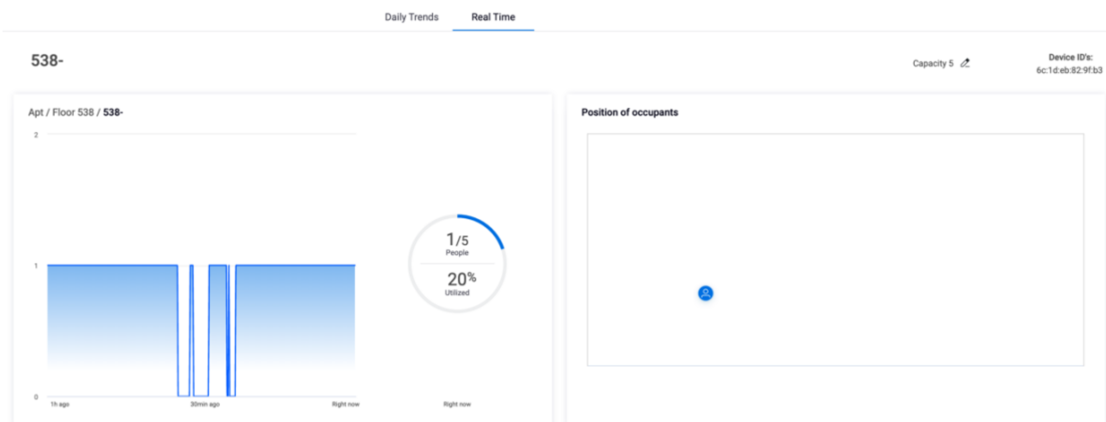

Figure 3.9 Online Dashboard Real-time Section

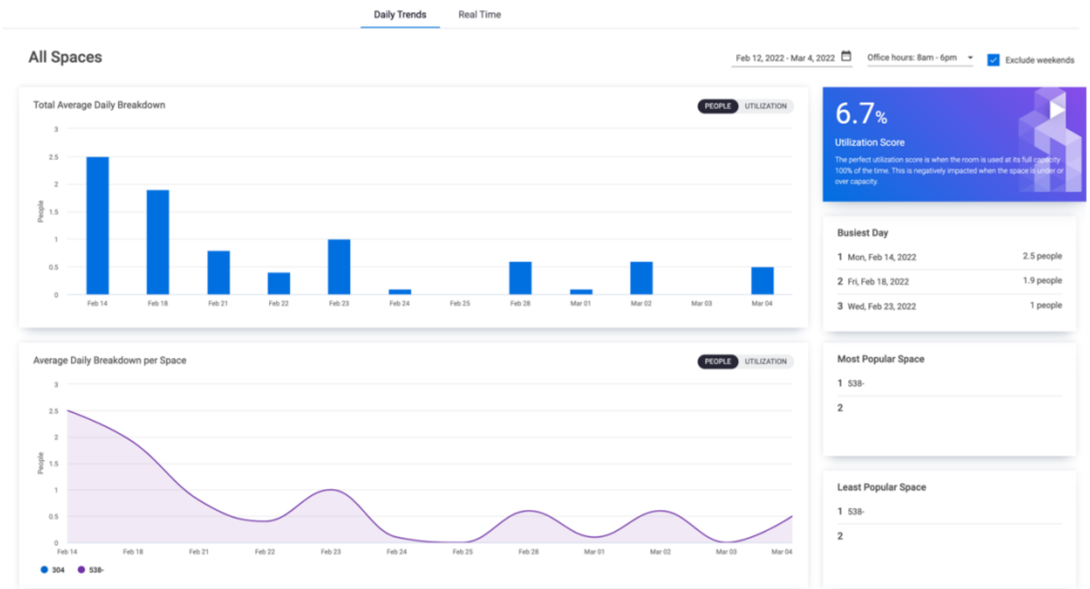

Figure 3.10 Online Dashboard Daily Trend Section

3.3.3 Edge Computing

Edge computing is beginning to be widely used in many IoT sensors due to the numerous benefits it provides. Unlike the previous two sensors, the EagleEye sensor has a different approach to sending its data in that this sensor uses edge computing. Because it is camera-based, this sensor can generate a massive amount of data, and it would be infeasible to send all the data to the cloud for processing. In addition, privacy and security concerns need to be addressed. The information needed is the current occupancy level within the room, which is a much sparse set of data. So, within the EagleEye sensor, the input data from the camera are processed into data on number and location of occupants, without detailed images. This data is presented as a real-time graph. However, the actual image shot by the camera is only accessible for a very short time when setting up the sensor on the phone. After successfully setting up the sensor, as shown in Figure 3.9, we are only able to access the position of the occupants in the room. This significantly reduces the amount of communication and saves bandwidth. In addition, this feature also provides privacy advantage as we are unable to access any graphic information from the sensor.

Chapter 4

Possibilities of Data Fusion

After successfully collecting data from sensors in an IoT network, we now focus on how we can make best use of those data. One of the common techniques for processing data is data fusion. We provide the possibilities of having data fusion for future iterations of our network.

4.1 Definition of Data Fusion

The terms “information fusion” and “sensor fusion” have been widely used and sometimes are used interchangeably. However, these two concepts are slightly different from one another. As explained by Elmenreich [2002], information fusion is more the combining of information in a broader sense. There are generally no restrictions on the sources of information. They could from sensors, databases, or even humans. Sensor fusion, on the other hand, refers to the synergy of data gathered from sensors. In both definitions, it is pointed out that the synergistic evaluation of data should help to generate a better result than what the data singularly could possibly generate. In this thesis, we focus on sensor fusion, sometimes also referred to as multisensory data fusion.

4.2 Approaches for Data Fusion

Generally, there are several prevailing statistical methods for data fusion. These two methods mainly deal with the uncertainty created from the raw measurement data.

4.2.1 Bayesian Approach

In this approach, we make use of the maximum likelihood to statistically infer a more accurate result. There are two methods that both use the Bayesian approach. The first one is called the direct method, which uses all the measured data for one particular property. These data are supposed to be consistent at any moment. Another method is called the indirect method, which is used to better identify a series of events. Below, we look at the statistical details behind these two methods.

4.2.1.1 Direct Method

We denote the sensor output to be a vector X, where $X = (x_1, x_2, \dots, x_n)$, and the object property is Θ, which may be any property of the installed sensor, such as position. Two conditional probabilities, $p(X|\Theta)$ and $p(\Theta|X)$, will be used mainly in the following statistical process: $p(X|\Theta)$ indicates the probability that the output from the sensor is X has a given property Θ, and $p(\Theta|X)$ is the probability that the sensor has property Θ given the output from the sensor is X. By Bayes' Law, we have

$$p(\Theta|X) = \frac{p(X|\Theta) \cdot p(\Theta)}{p(X)}$$

where $p(X)$ is the probability when the sensor's output is X, and $p(\Theta)$ is the probability that the property is Θ. Both probabilities are unconditional.

Assume we have the following readings from k sensors, we denote those readings as $\chi = (X^1, X^2, \dots, X^k)$. Following the process of the likelihood estimate, we then have the following:

$$p(\chi|\Theta) = \prod_{i=1}^{k} p(X^i|\Theta)$$

Now, we seek to maximize $p(\chi|\Theta)$. To make the process simpler, we take logarithms of both sides to obtain:

$$L(\Theta) = \log p(\chi|\Theta) = \sum_{i=1}^{k} \log p(X^i|\Theta)$$

If we assume the sensor readings follow a normal distribution, we then have

$$p\left(X^i\middle|\Theta\right) = \frac{1}{(2\pi)^{\frac{n}{2}}\,|C_i|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\left(X^i-\Theta\right)^{\mathrm{T}} C_i^{-1}(X_i-\Theta)\right)$$

where C_i is the variance-covariance matrix, $(\ldots)^{\mathrm{T}}$ indicates the transpose of what is inside the parentheses, and $|\ldots|$ indicates the determinant of what is inside the two vertical lines. Next, we substitute what we have for $p(X^i|\Theta)$ to $L(\Theta)$ above, yielding

$$L(\Theta) = -\frac{1}{2}\log(2\pi)^n\,|C_i|^2 - \frac{1}{2}\left(X^i-\Theta\right)^{\mathrm{T}} C_i^{-1}(X_i-\Theta)$$

Differentiating L with respect to Θ and setting that equal to 0, the value of Θ becomes:

$$\widetilde{\Theta} = \frac{\sum_{i=1}^{k} C_i^{-1}X^i}{\sum_{i=1}^{k} C_i^{-1}}$$

Following the above arithmetic process, one could find the weighted average $\widetilde{\Theta}$ of the readings from multiple sensors [Hackett & Shah, 1990].

4.2.1.2 Indirect Method

The previous section introduces the simplest case of data fusion. In most cases, however, we are more likely to be given by data measured by a set of sensors. These data are most likely not measuring the same thing but a series of related events. Hence, the direct method falls short as the measured data are not meant to be identical.

We could model all our data in term of 3-D vectors and locations. The currently unknown translation and rotation parameters are able to be computed. We denote the coordinate in our constructed coordinate system as $P_{k,}$ and the k-th object point as p_k. We then have the following:

$$P_k = Rp_k + h$$

where R and h are the rotation matrix and translation vector, respectively. The goal here is to find R and h. Shekhar et al. [1986] proposed treating translation and rotation one at a time. To compute h, the following equations are used:

$$D_i = d_i + n_i^{\mathrm{T}} h$$

$$n_i^{\mathrm{T}}$$

$$h = D_i - d_i$$

In the above two equations, D_i indicates the distances measured from the direction n_i, whereas d_i is the distance calculated from the model. With a set of data, the above equation becomes

$$\boldsymbol{C}h = \boldsymbol{d}$$

where $\boldsymbol{C} = [n_1, n_2, \dots, n_n]^{\mathrm{T}}$ and $d = [d_1, d_2, \dots, d_n]^{\mathrm{T}}$. We can also introduce the weights $w_p = \delta_{d_i}^2$, the expected errors in distance, to the equation above, i.e.,

$$w_p\boldsymbol{C}h = w_p\boldsymbol{d}$$

The above equation can then be solved using the pseudo inverse method for least squares fit [Hackett & Shah, 1990].

4.2.2 Dempster-Shafer Approach

Rather than having the traditional sense of probability, Dempster-Shafer uses another rather different concept called mass. Assigning masses to the entire subset of entities in the system, we label each one with 1 or 0, and based the label, we can then determine if any given member inside the system is in the subset or not. Having the original set S, we define the set of all subsets as the power set, 2^S.

To apply the Dempster rule, we also need to define a few more concepts. We use mass m to denote the confidence in each element of the power set, m_s the confidence in measured data from sensors, and m_0 the confidence in the previous data evidence. We then have the following equation for $A, B, C \in 2^S$:

$$m(C) = \frac{\sum_{A \cap B = C} m_s(A) m_0(B)}{1 - \sum_{A \cap B = \emptyset} m_s(A) m_0(B)}$$

This method has achieved significant success in exploring places that are not covered in the system, in which C in the above equation be interpreted [Yi et al., 2000]. Compared to the Bayesian method in which all individual points inside the system need to be measured, the Dempster-Shafer method requires less comprehensive measurement of the entire system. With incomplete measurement data of the entire system, the method allows the data to be inferred with some amount of uncertainty [Challa & Koks, 2009].

4.3 Smart City Applications

As introduced in previous sections, the goal for smart cities is not to construct generic solutions like building a weather forecast model based on cloud patterns, temperature, and humidity. Instead, the goal is to create a model that is designed for a specific system, and this system can provide unique feedback to improve, for example, quality of life [Wang et al., 2017].

Data fusion can provide significant insight. Sobolevsky et al. [2015] proposed a method using data such as social media posts and bank card transaction to infer how attractive a city is for tourists. This work is especially inspirational as it provides insights on how the smart city approach can use data fusion in a similar manner. Increasing context awareness can further help to optimize the environment with a specific goal in mind. A simple example is a bridge that is experiencing severe damaged because of severe weather. The bridge itself can then send messages to all travelers to avoid going over that bridge, and at the same time it informs local government. In this manner, drivers can be informed that the route is no longer safe (and perhaps be rerouted) and the government could take immediate action to repair the bridge. This example requires data fusion from multiple sets of IoT sensors. Data fusion helps to have a better understanding of the context so that precise decisions regarding addressing a specific issue can be made [Wang et al., 2017].

As for the sensor network developed in this thesis, with introducing additional and different types of sensors and having more data measured via those sensors, we are able to have more contextual information for the campus. Incorporating sanitizer dispenser data with indoor air quality, we could add several more sensors to have an accurate knowledge of where build is most crowded and what is the normal flow pattern for people within the building. We can then make optimizations like assigning less classroom usage in one area and more to some other areas so that people will not be concentrated in only one part of the building. Additionally, we could also deploy more ventilation devices to the area that is recognized as crowded to keep the community healthy. With the help of data fusion, many more contextual information is provided, and targeted solutions can be formulated.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In conclusion, we have successfully created an IoT sensor network that is suitable for a university campus as intended. The network is expected to include sanitizer dispenser sensors, air quality monitoring sensors, and room occupancy sensors. Because of some difficulties in commissioning, it ended up having some minor discrepancies between expectations and current implementation. Implementing the smart city approach, the sensor network could be beneficial for the community in a way to provide guidance for certain building optimization.

5.2 Future Work

5.2.1 Completing the Implementation of Smart City Initiative

The implementation described in this thesis is but the beginnings of a full-scale smart city (perhaps "smart campus") application. This thesis describes a way to set up sensors in the network and what kinds of sensors should be considered. What is next is to add more sensors to cover many other aspects, with a comprehensive view of what needs to be collected. After that, we may then be able to propose changes from the data collected. It will also take some time to complete all those changes and to see whether those changes do improve the status quo or not. From there we can do further data analysis and adjust our plan accordingly.

5.2.2 Prospect of Using Edge Computing

For a traditional networking system, data are collected on edge sensors and then sent to the cloud or a central control system. All of the data analysis and processing take place at the central control system or in the cloud, which is type of centralized data center. Currently, the majority of our system also using such an architecture. However, as the number of IoT devices increases, sending all the data to the central data center will require a considerable amount of data traffic. Moreover, if an urgent action needs to be made at the sensor end, the flow of data would still be from the sensor to the data center and then back to the sensor. This problem is especially detrimental in certain industries as many problems need to be solved in the shortest time after being detected. For example, if a sensor inside an oil-transmitting pipe detects abnormally high pressure, the pressure must be relieved as soon as possible. If the latency to this action is too high, the likelihood of the entire pipe rupturing increases significantly [Hamilton, 2019].

A way to shorten the latency is to process all or a part of data at the edge. This would enable actions to be taken in the shortest time. As for the data sent from the edge to the central management system, they can be processed before sending, making the entire data transfer process more efficient. However, this does not mean that all the data processing can be done at the edge. The main analysis and processing should be still done by the central data center. The existence of edge computing helps the data transmission efficiency and enable prompt actions to be taken.

At the current stage of our project, the sensor network includes hand sanitizer dispenser sensors and some air quality sensors. In other words, our current network is relatively small, and data transmission issues are limited. However, as the network expands, constant data communication may cause the entire system to slow down substantially or even occasionally malfunction. Edge computing may help alleviate these concerns.

BIBLIOGRAPHY

Bakıcı, T., Almirall, E., & Wareham, J. (2013). A smart city initiative: The case of Barcelona. *Journal of the Knowledge Economy, 4*(2), pp. 135-148.

Barbosa, F. P., & Charão, A. S. (2012, June). Impact of pay-as-you-go cloud platforms on software pricing and development: A review and case study. In *International Conference on Computational Science and Its Applications* (pp. 404-417). Springer, Berlin, Heidelberg.

Bello, O., & Zeadally, S. (2013). Communication issues in the Internet of Things (IoT). In *Next-Generation Wireless Technologies* (pp. 189-219). Springer, London.

Bello, O., Zeadally, S., & Badra, M. (2017). Network layer inter-operation of Device-to-Device communication technologies in Internet of Things (IoT). *Ad Hoc Networks*, *57*, pp. 52-62.

Buck, T. N., & While, A. (2017). Competitive urbanism and the limits to smart city innovation: The UK Future Cities initiative. *Urban Studies*, *54*(2), pp. 501-519.

Caragliu, A., & Del Bo, C. F. (2019). Smart innovative cities: The impact of Smart City policies on urban innovation. *Technological Forecasting and Social Change, 142*, pp. 373-383.

Camero, A., & Alba, E. (2019). Smart City and information technology: A review. C*ities, 93*, pp. 84-94.

Centers for Disease Control and Prevention. (2021) *Cleaning, Disinfection, and Hand Hygiene in Schools – A Toolkit for School Administrators.* Retrieved April 3, 2022, from https://www.cdc.gov/coronavirus/2019-ncov/community/schools-childcare/clean-disinfect-hygiene.html

Challa, S., & Koks, D. (2009). An Introduction to Bayesian and Dempster-Shafer Data Fusion. Electronic Warfare and Radar Division Systems Sciences Laboratory, DSTO, Computer

Systems Engineering, Faculty of Engineering, The University of Technology, Sydney, DSTO-TR-1436.

Chen, B., Wan, J., Celesti, A., Li, D., Abbas, H., & Zhang, Q. (2018). Edge computing in IoT-based manufacturing. *IEEE Communications Magazine*, *56*(9), pp. 103-109.

Cicirelli, F., Guerrieri, A., Spezzano, G., & Vinci, A. (2017). An edge-based platform for dynamic smart city applications. *Future Generation Computer Systems*, *76*, pp. 106-118.

Dameri, R. P. (2013). Searching for smart city definition: A comprehensive proposal. *International Journal of Computers & Technology*, *11*(5), pp. 2544-2551.

de Carvalho Silva, J., Rodrigues, J. J., Alberti, A. M., Solic, P., & Aquino, A. L. (2017, July). LoRaWAN—A low power WAN protocol for Internet of Things: A review and opportunities. In *2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)* (pp. 1-6). IEEE.

Elmenreich, W. (2002). An introduction to sensor fusion. *Vienna University of Technology, Austria*, *502*, pp. 1-28.

Faulkner, C. (2017, May 9) *What is NFC? Everything you need to know*. Techradar. https://www.techradar.com/news/what-is-nfc

Gavalas, D., Nicopolitidis, P., Kameas, A., Goumopoulos, C., Bellavista, P., Lambrinos, L., & Guo, B. (2017). Smart cities: recent trends, methodologies, and applications. *Wireless Communications and Mobile Computing*, *2017*.

GSMA. (2013, February). *Guide to Smart Cities*. GSMA. Retrieved April 2, 2022, from https://www.gsma.com/iot/wp-content/uploads/2013/02/cl_sc_guide_wp_02_131.pdf

Hackett, J. K., & Shah, M. (1990, May). Multi-sensor fusion: a perspective. In *Proceedings., IEEE International Conference on Robotics and Automation* (pp. 1324-1330).

Hamilton, E. (2019). What is edge computing: The network edge explained. *Cloudwards*. Retrieved April 3, 2022, https://www.cloudwards.net/what-is-edge-computing/#:~:text=At%20its%20simplest%2C%20that's%20what,so%20that%20data%20becomes%20actionable.

Hardesty, G. (2019, April 12) *LoRa: Long Range Wireless for Internet of Things (IoT): Frequency Bands, Antennas.* Data-alliance. https://www.data-alliance.net/blog/lora-long-range-wireless-for-internet-of-things-iot-frequency-bands/

Hardesty, G. (2019, April 12) *LoRa: dBi, dB, dBm dB(mW): Defined, Explained and Differentiated.* Data-alliance. https://www.data-alliance.net/blog/dbi-db-dbm-dbmw-defined-explained-and-differentiated/

Haxhibeqiri, J., De Poorter, E., Moerman, I., & Hoebeke, J. (2018). A survey of LoRaWAN for IoT: From technology to application. *Sensors*, *18*(11), pp. 3995.

Ismagilova, E., Hughes, L., Dwivedi, Y. K., & Raman, K. R. (2019). Smart cities: Advances in research—An information systems perspective. *International Journal of Information Management*, *47*, pp. 88-100.

Kirimtat, A., Chatzikonstantinou, I., Sariyildiz, S., & Tartar, A. (2015, May). Designing self-sufficient floating neighborhoods using computational decision support. In *2015 IEEE Congress on Evolutionary Computation (CEC)* (pp. 2261-2268). IEEE.

Karagiannis, V., Chatzimisios, P., Vazquez-Gallego, F., & Alonso-Zarate, J. (2015). A survey on application layer protocols for the Internet of Things. *Transaction on IoT and Cloud computing*, *3*(1), pp. 11-17.

Kirimtat, A., Krejcar, O., Kertesz, A., & Tasgetiren, M. F. (2020). Future trends and current state of smart city concepts: A survey. *IEEE access*, *8*, pp. 86448-86467.

Kumar, S., Tiwari, P., & Zymbler, M. (2019). Internet of Things is a revolutionary approach for future technology enhancement: a review. *Journal of Big data*, *6*(1), pp. 1-21.

Lau, B. P. L., Marakkalage, S. H., Zhou, Y., Hassan, N. U., Yuen, C., Zhang, M., & Tan, U. X. (2019). A survey of data fusion in smart city applications. *Information Fusion*, *52*, pp. 357-374.

Lee, I., & Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business horizons*, *58*(4), pp. 431-440.

Mekki, K., Bajic, E., Chaxel, F., & Meyer, F. (2018, March). Overview of cellular LPWAN technologies for IoT deployment: Sigfox, LoRaWAN, and NB-IoT. In *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM)* (pp. 197-202). IEEE.

Nakamura, E. F., Loureiro, A. A., & Frery, A. C. (2007). Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Computing Surveys (CSUR)*, *39*(3).

Nauman, A., Qadri, Y. A., Amjad, M., Zikria, Y. B., Afzal, M. K., & Kim, S. W. (2020). Multimedia Internet of Things: A comprehensive survey. *IEEE Access*, *8*, pp. 8202-8250.

Nilssen, M. (2019). To the smart city and beyond? Developing a typology of smart urban innovation. *Technological Forecasting and Social Change*, *142*, pp. 98-104.

Nord, J. H., Koohang, A., & Paliszkiewicz, J. (2019). The Internet of Things: Review and theoretical framework. *Expert Systems with Applications*, 133, pp. 97-108.

Patel, K. K., & Patel, S. M. (2016). Internet of things-IOT: definition, characteristics, architecture, enabling technologies, application & future challenges. *International Journal of Engineering Science and Computing*, *6*(5).

Ray, P. P. (2018). A survey on Internet of Things architectures. *Journal of King Saud University-Computer and Information Sciences*, 30(3), pp. 291-319.

Shafique, K., Khawaja, B. A., Sabir, F., Qazi, S., & Mustaqim, M. (2020). Internet of Things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT scenarios. *IEEE Access*, *8*, pp. 23022-23040.

Shekhar, S., Khatib, O., & Shimojo, M. (1986, April). Sensor fusion and object localization. In *Proceedings. 1986 IEEE International Conference on Robotics and Automation* (Vol. 3, pp. 1623-1628). IEEE.

Shelton, T., Zook, M., & Wiig, A. (2015). The 'actually existing smart city'. *Cambridge Journal of Regions, Economy and Society*, *8*(1), pp. 13-25.

Sobolevsky, S., Bojic, I., Belyi, A., Sitko, I., Hawelka, B., Arias, J. M., & Ratti, C. (2015, June). Scaling of city attractiveness for foreign visitors through big data of human economical and social media activity. In *2015 IEEE International Congress on Big Data* (pp. 600-607). IEEE.

Trencher, G. (2019). Towards the Smart City 2.0: Empirical evidence of using smartness as a tool for tackling social challenges. *Technological Forecasting and Social Change*, *142*, pp. 117-128.

UN, *68% of the world population projected to live in urban areas by 2050, says UN*. UN, 2018. [Online]. Available: https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html

U.S. Environmental Protection Agency. (1989). Report to Congress on indoor air quality: Volume 2. EPA/400/1-89/001C.

U.S. Environmental Protection Agency. (n.d.) *What are the trends in indoor air quality and their effects on human health?* Retrieved April 2, 2022, from https://www.epa.gov/report-environment/indoor-air-quality#note1

Wang, M., Perera, C., Jayaraman, P. P., Zhang, M., Strazdins, P., Shyamsundar, R. K., & Ranjan, R. (2017). City data fusion: Sensor data fusion in the Internet of Things. In *The Internet of Things: Breakthroughs in Research and Practice* (pp. 398-422). IGI Global.

Yan, Z., Zhang, P., & Vasilakos, A. V. (2014). A survey on trust management for Internet of Things. *Journal of Network and Computer Applications*, *42*, 120-134.

Yi, Zou, Yeong Khing Ho, Chin Seng Chua, and Xiao Wei Zhou. (2000). Multi-ultrasonic sensor fusion for autonomous mobile robots. In *Sensor Fusion: Architectures, Algorithms, and Applications IV*, vol. 4051, pp. 314-321. International Society for Optics and Photonics.

Zhao, J., Zhou, B., Butler, J., Bock, R. G., Portelli, J. P., & Bilén, S. G. (2021). IoT-based sanitizer station network: A facilities management case study on monitoring hand sanitizer dispenser usage. *Smart Cities*, *4*(3), pp. 979-994. DOI: 10.3390/smartcities4030051

ACADEMIC VITA

Boyang Zhou

Education

The Pennsylvania State University — ***University Park, PA***
Schreyer Honors College — Graduation: May 2022
Bachelor of Science in Mathematics and Computer Science
Dean's List: 6 Semesters

Honors Thesis

IoT Sensor Networks over LoRaWAN — May 2022

Focused on the analysis and presentation of data collected from Low Power Wide Area Network (LoRaWAN) sensors, aimed at optimizing real-time data communication between sensors and the network.

Publication

J. Zhao, B. Zhou, J.P. Butler, R.G. Bock, J.P. Portelli, and S.G. Bilén, "IoT-Based Sanitizer Station Network: A Facilities Management Case Study on Monitoring Hand Sanitizer Dispenser Usage," *Smart Cities*, 2021; 4(3):979-994. Available: https://doi.org/10.3390/smartcities4030051.

Course Projects

Built a Micro-CPU using Verilog in Vivado in *Computer Organization and Design* class, 2020 fall semester.

Implemented a user-space device driver for the ScatterGather drive using C. Created a peer-to-peer system in which users could place the data for their local file system on several external hosts. This project was done in *Intro to System Programming* class, 2020 fall semester.

Leadership Experience

Math Guided Study Group Leader — March 2020–Present
- Review the information students learned during the past week
- Communicate with professors and students to go through students' questions

Math Tutor — September 2019–Present
- Weekly assist 15 students one-on-one in math classes up to and including Calculus II and below

SCASD Volunteer — November 2019–March 2020
- Tutored a high school student in AP physics weekly for two hours

Technical Skills

Python, Java, C, SQL, Verilog, MATLAB, Mathematica

Honors

First Place/CodePSU 2019 Beginner Level

Member of team that beat nine other teams in the beginner level of this annual contest held by the Penn State Association for Computing Machinery.