APPROVAL SHEET

Title of Dissertation: S.T.O.M.P.: Sentry Tele-Operation and Monitoring Protocol

Name of Candidate: Tejas Sathe
Master of Science, 2017

Dissertation and Abstract Approved:
Cynthia Matuszek
Assistant Professor
Computer Science and Electrical Engineering

Date Approved: ________________

Name: Tejas Ajay Sathe

Degree and date to be conferred: M.S., 2017

Secondary education: Abhinava Vidyalaya English Medium High School, Pune, Maharashtra, India, May, 2009.

Collegiate institutions attended:

1. University of Pune, Bachelor of Engineering, May, 2015.
2. University of Maryland, Baltimore County, Master of Science, May, 2017.

Major: Computer Science

Professional positions held:

1. Software Engineering Intern, EquityZen Inc., New York, NY.
2. Research Assistant, University of Maryland, Baltimore County, Baltimore, MD.

ABSTRACT

Title of Document:	S.T.O.M.P.: SENTRY TELE-OPERATION AND MONITORING PROTOCOL
	Tejas Ajay Sathe, M.S., 2017
Directed By:	Type Your Advisor’s Title, Name, and Department

Sentry duty has been employed since ancient times to ensure peace and mitigate disturbances on the grounds of any structure; be it college campuses, corporate offices, or even military establishments; this is the first area of focus. Good reconnaissance is also extremely important, especially in situations like Search and Rescue; this is the second area of focus. This thesis addresses both these areas from a robotics standpoint. The research involves equipping a robot with a sensing suite to generate a real-time map of the environment, run predictions on the network activity and estimate the number of people present in any structure. The results of this research are stated as the activity profile generated over a week, and a map of a narrow hallway to substantiate the claims the thesis makes.

S.T.O.M.P.: SENTRY TELE-OPERATION AND MONITORING PROTOCOL

By

Tejas Ajay Sathe

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, Baltimore County, in partial fulfillment
of the requirements for the degree of
Master of Science
2017

© Copyright by
[Tejas Ajay Sathe]
[2017]

Preface

This thesis on “S.T.O.M.P.: Sentry Tele-Operation and Monitoring Protocol” is research that tries to answer two main questions: whether we generate a real-time map accurate enough to serve as an alternative to a camera in low-visibility situations, and whether network activity can serve as an accurate indicator of the number of people present in any structure. This dissertation has been written to fulfill the graduation requirements of the M.S. Computer Science program at the University of Maryland, Baltimore County. It was written in the period between January and April 2017.

This research was carried out under the purview of the Interactive Robotics and Language (IRAL) Lab at UMBC. I was Research Assistant associated with IRAL from January, 2017 to May, 2017. This topic was chosen to accommodate my areas of interest, which are Robotics, Network Security and Machine Learning.

Acknowledgements

I would like to thank all my advisors for their excellent guidance during this entire process. I would also like to thank my colleagues in the IRAL lab for their outstanding inputs and ideas whenever I was in a logjam. I enjoyed laying out my ideas and receiving critique from all of them.

Table of Contents

List of Figures

Chapter 1: Introduction

Anomaly Detection

In this age, almost everything runs on computers. Even legacy industrial processes have now been automated and computerized. One big advantage of doing this is it reduces human effort and involvement. Even though some human intervention is required, automation has made it possible for the humans on the factory floor to monitor all processes from a computer screen in the comfort of their offices. This also provides a safer work environment, since if the processes or the environment are hazardous to humans on direct contact, like nuclear reactors, the operators only have to get physically involved only if there is an emergency; however, we are at a point where even damage control can be done from behind a computer screen.

In essence, the human component of monitoring any particular event has been reduced. There are several situations that require constant monitoring. Another example is the Intensive Care Unit, where a patient needs to be under watch all time to avoid health problems. Again, in this case, this can be done using highly sophisticated equipment connected to the patient that transmits readings to a computer. The patient needs to be disturbed only in case there is some problem with their vitals.

Yet another example is sentry duty. For a very long time, soldier camps or important facilities were guarded by human sentries, who would work in shifts patrolling the perimeter and report on any anomalous behavior. Like the aforementioned examples, this has also become easier and more effective with the advent of technology. Nowadays, video cameras and perimeter breach detectors are ubiquitous. Again, they allow an operator to view multiple places on the facility at the same time and direct forces a particular place in case there is a break-in.

The idea of robotic sentries was introduced by science fiction media, like video games, e.g. the ED-209 from RoboCop. A sentry robot that can detect anomalies is one of the objectives of this research.

Objective #1*: *To build a system capable of:

- performing sentry duty while being tele-operated remotely
- mapping the environment using various sensors
- detecting anomalous readings/behavior

Reconnaissance

Emergency response is a very important field. Rescue specialists cannot be someone who just decided to try it out, they have to be trained for a variety of scenarios. Disasters are extremely unpredictable, and the responders have to be able to improvise in order to function effectively and reduce the number of casualties. An example of a branch of emergency response is the fire department. Firemen are tasked not only with putting out fires, but also free trapped people and save as many of them as they can.

It is unwise to just rush into a burning building, since that can be dangerous to the firemen themselves. Some reconnaissance has to be done, which can help them plan the course of action. This can be done after reaching the disaster zone. Several factors like the topography, the extent of the damage, etc. has to be studied before they can go in and start mitigating the situation. A robot can be a big help in such cases. In places where humans cannot be sent to do reconnaissance, a robot can be sent to gather essential information, map the environment and relay that data back to the first responders. This is the second focus of this research.

Objective #2: To build a system capable of:

- performing real-time reconnaissance with very little knowledge of the situation
- interpreting the data and converting it to information valuable to the operator
- transmitting this information back to the operator reliably and intuitively

Problem Statement

To design a tele-operated sentry robot system that can be used for anomaly detection and reconnaissance.

Organization of the thesis

In Chapter 2, we will discuss literature that already exists on the two main topics of this thesis:

i) Sensor automation

ii) Network discovery

iii) Wireless activity tracking

In Chapter 3, the motivation behind this research will be discussed in a little more detail. This includes the questions it is trying to answer, and how the implementation will be structured.

In Chapter 4, the actual implementation will be discussed in detail, which includes the hardware and the software architectures, and how they combine to ultimately give life to the system.

In Chapter 5, we will discuss how the data was collected and analyzed. We will also discuss how the data is being visualized using an intuitive web interface that allows the operator to see the state of the environment at a glance.

Chapter 6 will include the conclusion, and what work can be done in the future.

Chapter 2: Related Work

Everett, Hobart R. designed a controlled prototype robot [1] that served two purposes: it acted as an autonomous sentry and served as a platform for evaluation of sensors and their corresponding interfaces.

Harrigton et al. [2] described a prototype robot system that functioned as a part of the existing security system at a high security facility. This system possessed the ability to navigate without the need for external inputs, like beacons, and the date link facilities to communicate information either directly to an electronic system to a central command center. It also had modes to operate on remote manual control or autonomously.

Åkerberg et al. [3] presented an article which talked about the major prerequisites for industrial process automation using Industrial Wireless Sensor Networks, or IWSNs. They also addressed issues like safety, security and availability before IWSNs can be adopted as an everyday part of the industrial process automation. They further talk about adopting the IEEE 802.15.4 as a standard for communication in the network, and the challenges involved with the same. IEEE 802.15.4 is a technical standard which defines the operation of a low-rate wireless personal area network (LR-WPAN). It specifies the physical layer and media access control for LR-WPANs, and is maintained by the IEEE 802.15 working group, which defined the standard in 2003. ZigBee and WirelessHART are two protocols that use the 802.15.4 standard, and the authors discuss how the latter is more effective since the former is not specifically designed for reliable real-time cyclic communication.

Kevin Chin Yiu Shum, et al. [5] applied estimation location algorithms integrated into a wireless network, to monitor the behavior of mobile users. In their approach, they used an open source programmable Linksys router as a wireless sensor to obtain information about wireless traffic like the SSID (Service Set Identifier) and the ESSID (Extended Service Set Identifier), RSSI (Received Signal Strength Indicator), Traffic Rate, Traffic Frequency, etc. By using multiple routers programmed in the same way, they were able to provide a framework for system administrators to monitor and analyze wireless traffic.

Theoretically, the relationship between RSSI values and the inter-distance between a mobile device and an access point obeys the inverse square law. In reality, this is not the case since a lot of noise can be introduced by environmental factors such as interference and reflection. The Location Fingerprinting that they have stated in this paper utilizes signal strength to determine location. They divided it into two phases: on-line and off-line.

In the off-line phase, snapshots of the Received Signal Strength (RSS) vectors are collected in every location. Each snapshot contains two values: Si and APi, which signifies the RSS received from the *i*th access point, identified by the Access Point Identifier, API. In the on-line phase, the RSS vector of the mobile device is measured and the generalized weighted distance Lp was computed. They adopted the Least Square method in matching the signal strength snapshot with the trained fingerprint database.

Lorincz et al. [5] set out to design a common protocol and software framework that could integrate devices such as wearable sensors that track vitals, handheld computers, location tracking tags etc. into disaster response scenarios. The result of this was CodeBlue, a new architecture that allows wireless monitoring and tracking of patients and first responders. They identified specific issues related to inter-operation of the various devices, like establishing communication, long range, prioritisation of bandwidth usage, security, and location tracking. They also explore RF based location tracking and argue that in an incident of mass casualty vital sign sensors can be placed on multiple patients; this would allow emergency responders to pinpoint the location of a patient who suddenly requires medical attention.

They argue that by installing battery operated RF sensors in the building beforehand, they would be able to use the location tracking in low visibility situations, like if the building were on fire. They built a system on this concept called MoteTrack, specifically designed for emergency response. They populated the building with beacons nodes and used a combination of those to triangulate their location.

80% of the location estimates are within 3 meters of the true location. These are 74 location estimates collected over one floor of the Computer Science building at Harvard University. These beacons broadcast periodic messages in a tuple format, {sourceID, powerLevel}. sourceID identifies the beacon and powerLevel indicates the amount of power used to broadcast the message. A combination of the two can be used to track location.

Yang et al. [6] presented a miniaturized telemetered ambulatory monitoring device in a ring configuration. The ring sensors will be worn by the patient at all times, which allows their health to be monitored around the clock. It is equipped with LEDs and photo detectors, while pulse oximetry technology is implemented for monitoring pulse waves and the saturation of oxygen in blood. Abnormal health conditions and accidents are detected by this apparatus. This system, they argue, is especially useful for monitoring of patients that cannot be transported to a hospital; either their condition is benign enough that remote monitoring is sufficient, or they're unwilling, or moving them can worsen their condition. It is also useful to monitor the health of difficult and/or non-compliant patients, like people with dementia. They developed a comfortable wearable to make monitoring easy.

Toschi et al. [7] present a survey on home area networks designed using the M2M (Machine-to-Machine) methodology. They also review the standards, architectures and initiatives created to enable M2M communication and integration in several different environments, especially at the smart home domain. They also pointed out the differences between them and identified trends for the future.

They also explain the concepts through a variety of visuals; for example, they illustrated a hype cycle of emerging technologies, where they mapped expectations with various stages of development of the system. The Hype Cycle for Emerging Technologies report [10] is the longest-running annual Hype Cycle, providing a cross-industry perspective on the technologies and trends that business strategists, chief innovation officers, R&D leaders, entrepreneurs, global market developers and emerging-technology teams should consider in developing emerging-technology portfolios.

Another interesting graphic they included in their paper was a cost-rate diagram, where they showed the relationship between the data rate, the applications it can be used in and the cost per bit of data as an X-Y coordinate system.

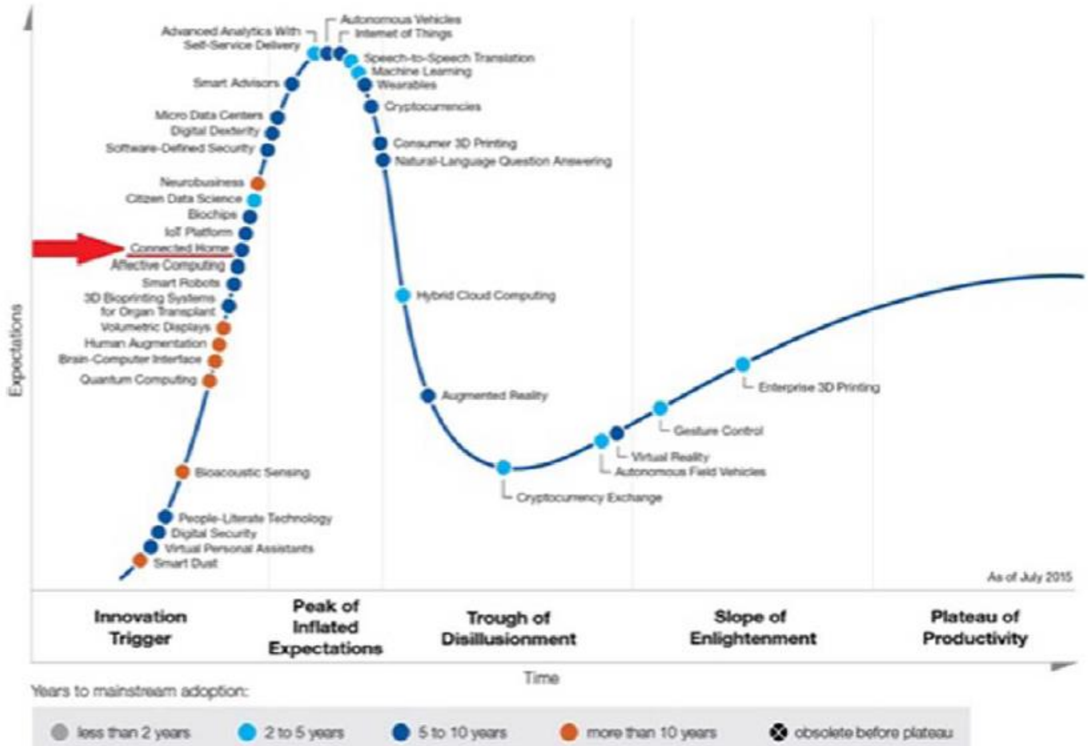

Fig. 1: Hype Cycle of Emerging Technologies

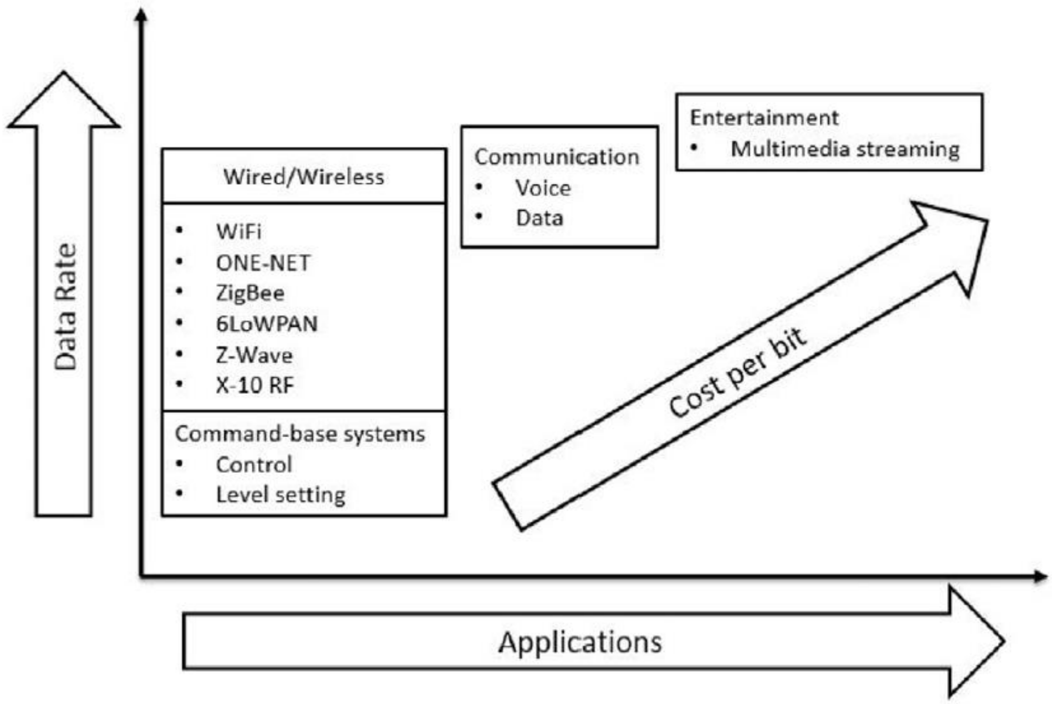

Fig. 2: Cost-Rate Diagram

Sunehra et al. [8] presented the implementation details of two schemes for home automation and control; the first one presents a prototype of a HAS (Home Automation System) for remotely controlling the appliances at home through the subject of email. The second scheme uses Bluetooth technology for controlling the devices when we are at home. In the first method , the home appliances are controlled by email. A Wi-Fi connected Raspberry Pi acts as the listener to the email messages. The user can send an ON or OFF message as an email to the Raspberry Pi, which then switches the relays either on or off. The system also sends an email response to the user with the status of the operation. In the second method, a Bluetooth controller on an Android phone is used to control all appliances.

Chapter 3: Motivation

Research Questions

The main idea behind this research was to design a sentry robot that accomplishes the following objectives:

Generate a real-time map of the environment using ultrasonic distance sensors (also maps the environment, and flags deviations from the predicted normal state of parameters)

Monitor network traffic and predict the number of people in the vicinity.

The mapping portion of this research can be used in conditions that are hostile to humans; for example, a burning building. In such cases, a lot of factors that can provide accurate information to rescue specialists can be compromised, visibility being the foremost. Mapping an environment using sensors can provide useful in this case. The point under investigation here is whether the operator can maneuver a rescue robot well enough and efficiently enough given just the output of the sensing equipment on board.

The aforementioned deviation from normal state is based mainly on two parameters: volume of wireless traffic and number of people present in the building at that time. The main question this research aims to answer is whether there is a predictable pattern to people's' presence and Internet activity, via mainly through wireless services like Bluetooth and Wi-Fi. If there is, are sudden changes to that an accurate indicator of anomalous activity. As a side note, anomalous activity in this context does not necessarily mean malicious intent. As an example, the building being empty during a weekday in Spring break is also flagged as anomalous, in comparison to a normal weekday when there are expected to be people on campus.

Research Design

The robot used in this research is the Beam Telepresence Robot, manufactured by Suitable Tech. This system also provides a GUI to the user through which they can configure the Beam, connect it to a wireless network and control it remotely. It can also be controlled from an external network, from a location in a different state, even a different country. That is the whole point of a telepresence robot. The reason an already established telepresence system was chosen for this research is that maneuverability of the robot itself was not the main focus.

The Beam was then equipped with a full sensor suite complete with an Arduino microprocessor. The Arduino was powered and controlled by a laptop. A laptop was used only because it was convenient to rectify any bugs and problems on the spot. But the proof of concept is sound, and a microcomputer like the Raspberry Pi can just as easily be mounted and used to control the microprocessor and sensors.

The sensor suite comprised of three ultrasonic distance sensors which measured the distance between the Beam and the nearest obstacle in three directions: North, West and East.

The wireless traffic analysis component consists of a program written on the laptop (or a microcomputer as a proof of concept). This program was written in Python on Ubuntu. It runs two processes in parallel; the first one interfaces with the Arduino, gathers distance readings and generates the map of the environment, and the second one sniffs TCP packets on the current network in 10 second bursts. This gives us a measure of how many packets are flowing across the network in unit time.

The Beam is maneuvered using the standard GUI that Suitable Tech provides, the program gathers and processes data in the background, and the web based GUI reports this to the operator.

Chapter 4: Methodology

Hardware Implementation

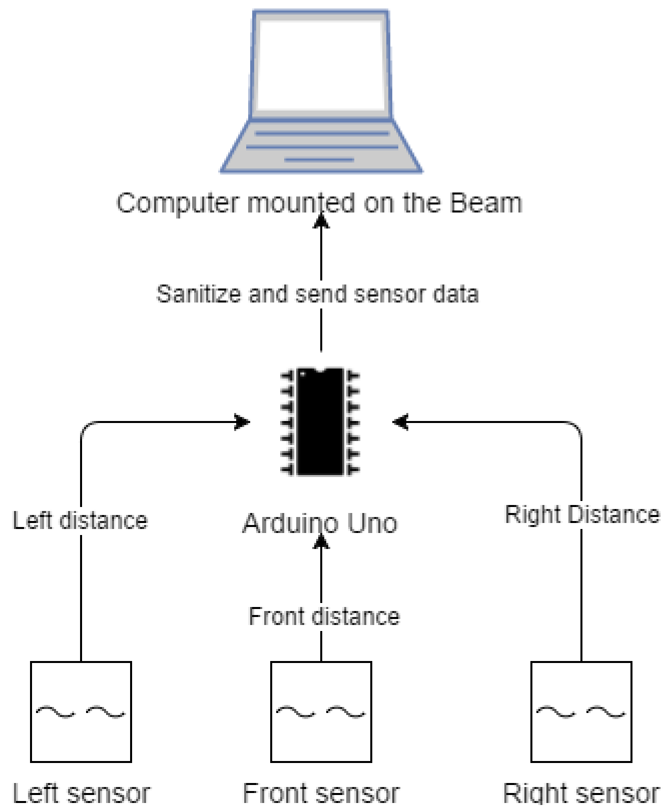

Fig. 3: Hardware Implementation

Since we are using a robot that can already be controlled and manoeuvred, the only implementable hardware was the sensing suite. As mentioned before, we equipped the Beam with a computer, an Arduino and three distance sensors.

As can be seen, there is a clear hierarchy of the devices. The sensors communicate only with the Arduino, and the implementation beyond that is abstracted for them. This is how they work:

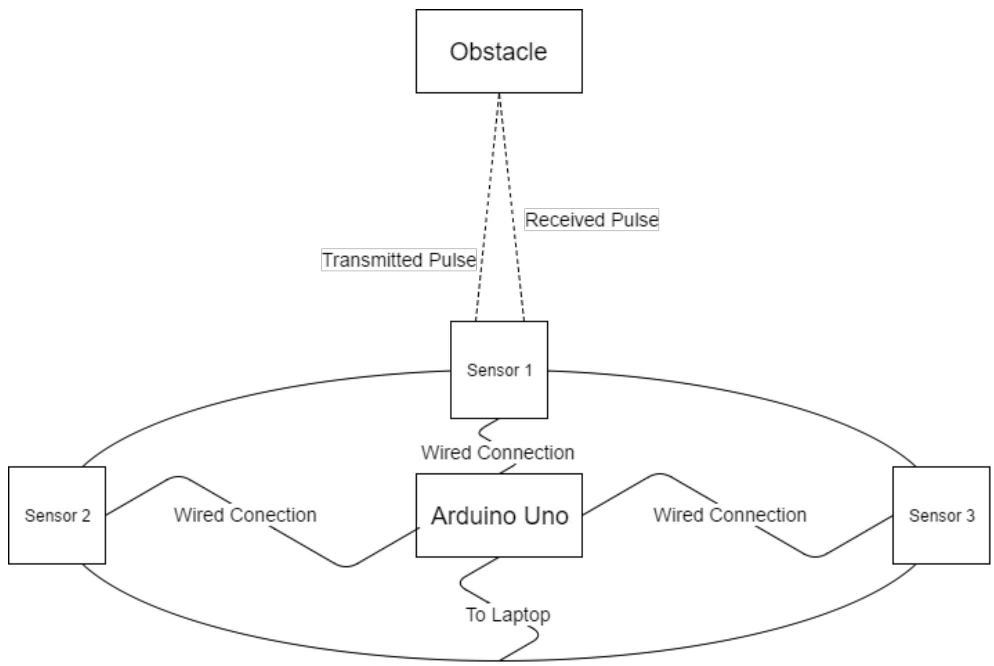

Fig. 4: Sensor Suite Architecture

The above diagram is a top view of the Beam and sensor suite. The Arduino and the sensors are attached to the Beam's base. The sensors emit an ultrasonic pulse that bounces off the nearest obstacle in the path. The distance is calculated using the time difference between emission and receipt of the pulse. This is the mapping component of the system.

Since this data is analog, it has to be converted to digital first. The environment mapping is done assuming the distance of the sensors is in centimeters, so the digital data is further converted to centimeters. The three sensors then independently and simultaneously report their readings to the Arduino, which transforms the data to the required format and sends it to a program that monitors the ports of the Arduino to which the sensors are connected. These readings are then used to generate a map, as discussed later.

Software Implementation

The software implementation has many distributed components. Since the Beam can be controlled from a remote location, it made sense to design a system that gives the user the same freedom. The whole point of this system is to give search and rescue specialists a platform for effective reconnaissance.

As shown in the diagram, the entire system comprises of at most four different computers; the database can be created and maintained on a separate machine, but it is more convenient to have it on the same machine that performs the analytics. You could also split the central server into two units: one that performs the analytics and the other that serves up the GUI. But, unless there is analysis that is so substantial that it requires its own machine, that would be overkill.

For this project, however, the server and the database are hosted on the computer mounted on the Beam itself, since it is convenient to perform quick debugging and troubleshooting in case the system malfunctions while on the move. But it has been tested as an independent server-and-computer, and it does work.

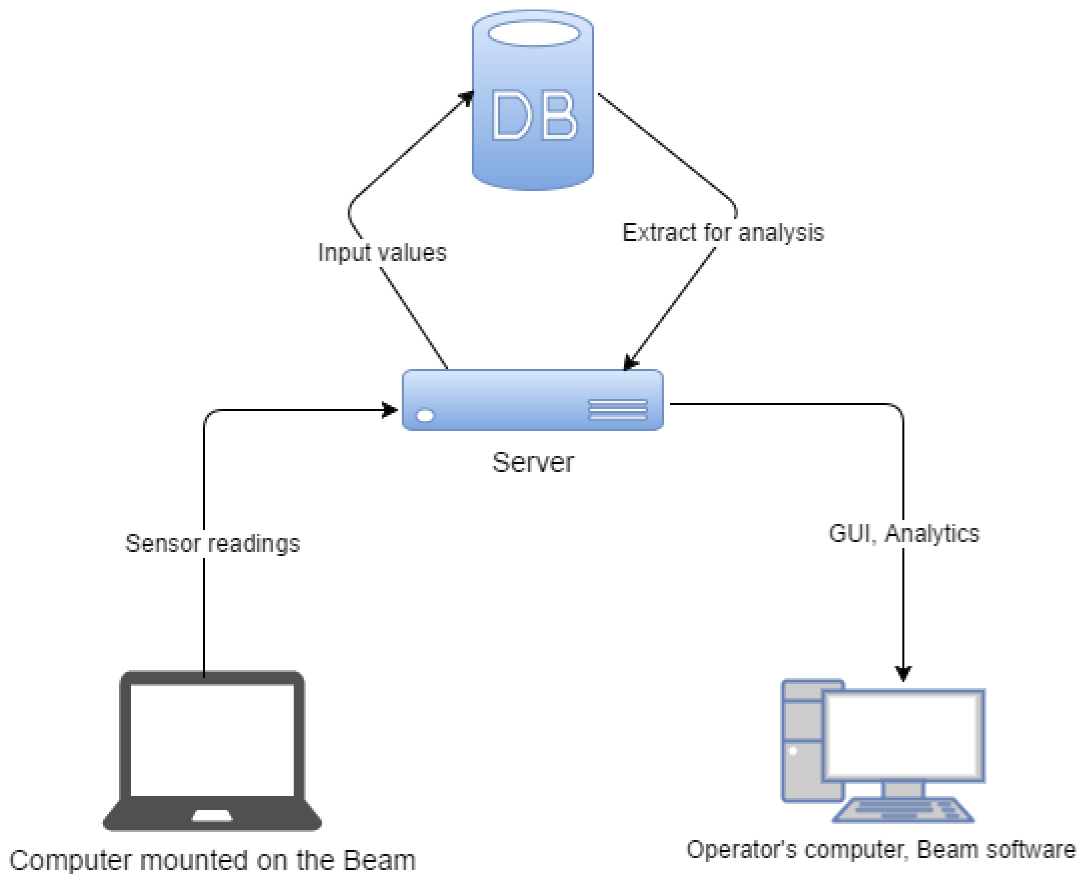

Fig. 7: Software Implementation

In the above diagram, even though it says “Computer mounted on the Beam”, it is actually an abstraction of Fig. (the one with the IC). That subsystem sends the sensor readings (which include the distance sensors as well as the packet sniffer) to the server. The server uses the distance readings to generate the map, which it integrates into the UI it serves. In the data collection phase, it also stored the packet sniffer readings and ran a learning algorithm on them. This is another feature embedded in the GUI, which is the output of the analytics on the sniffer data.

The third computer is the one used by the operator. In this case, the word "computer" is a fairly generic term for any device that has a web browser and can access websites. That includes mobile devices like cell phones and tablets. The idea is that the user will open the GUI the server provides and also the one Suitable Tech provides for controlling the Beam (shown in the Fig. 7). While the Beam is maneuvered, our web application also gives the operator a map and other information like the predicted number of people in the area at that moment in time.

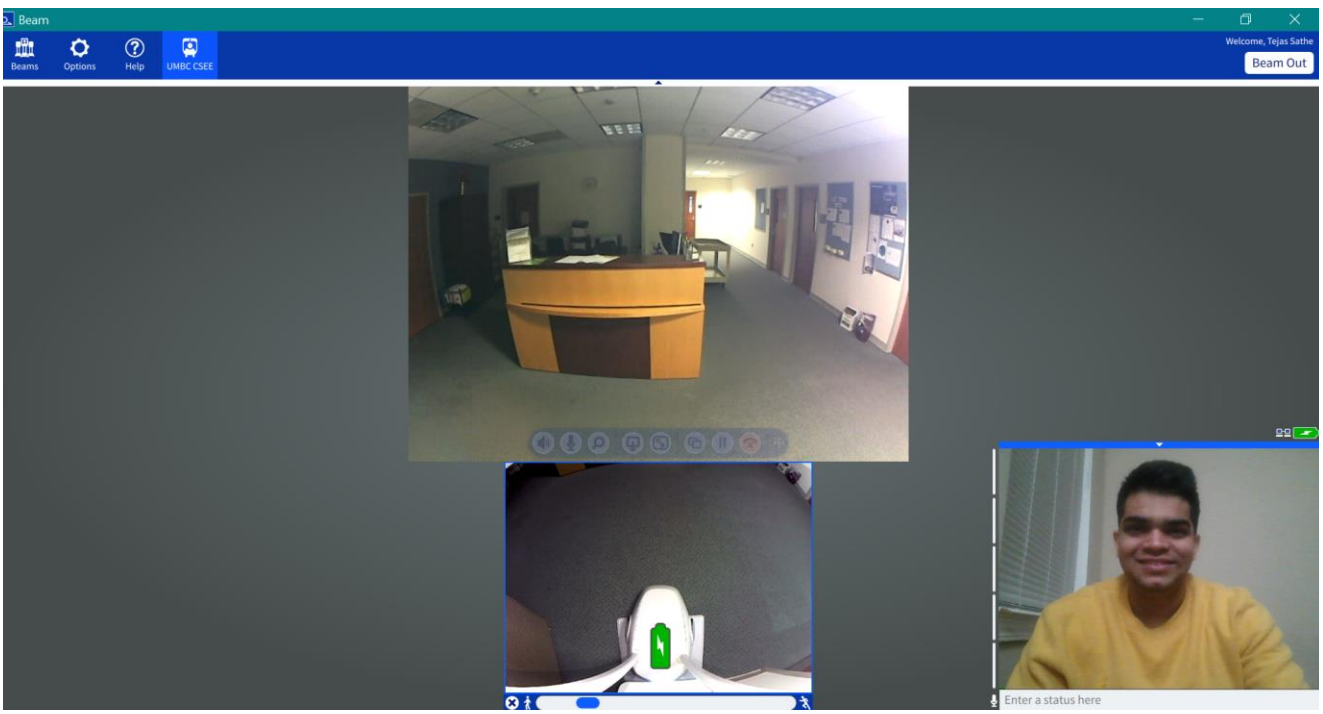

Fig. 6: Beam Software

Chapter 5: Data Collection, Analysis and Visualization

Data Collection

Data was collected thrice a day for two weeks. The dataset consists of the day of the week, the time of day, and the number of TCP (Transmission Control Protocol) packets detected traveling across the network at that point in time. In reality, many types of packets are flowing across the network, e.g. UDP (User Datagram Protocol), ARP (Address Resolution Protocol) etc. But we are sniffing only TCP packets since a person's Internet activity is an exchange of TCP packets.

When someone enters a URL in the browser and hits Enter, a TCP connection is established between the server of that website and the person accessing it. This allows the server and the user to exchange data using HTTP (Hyper Text Transfer Protocol). The connection is established in a process called a handshake.

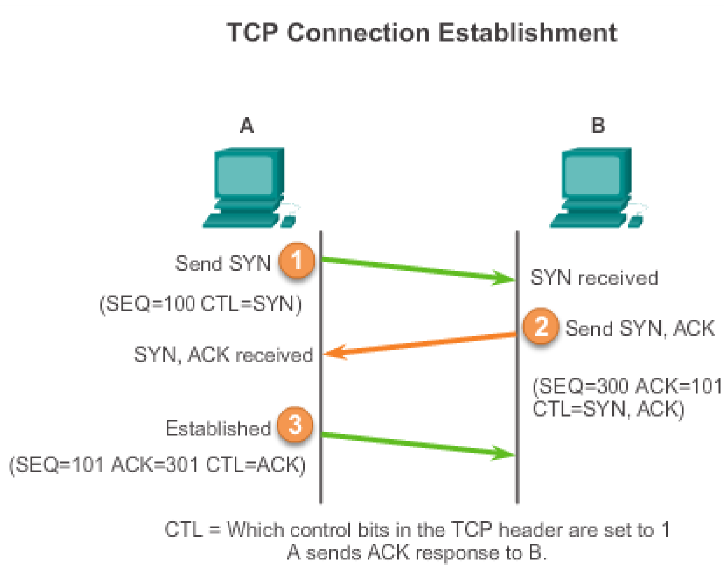

Fig. 8: TCP Handshake

First, the initiator sends a SYN request to the listener. The listener then sends the same SYN back with an additional ACK (acknowledgement). This shows the listener has received the request sent by the initiator. Finally, the initiator sends the ACK back, and the connection is established. This is a dedicated connection and is terminated when the initiator closes the browser, or ceases the activity.

As mentioned before, the data consists of the day of the week, the time of day, and the number of packets detected in ten seconds; it was transformed into integer values to make the learning easier. The day was labeled according to its position in the week (1 for Monday, 2 for Tuesday, etc.), and the time of day was taken as the number of seconds elapsed since midnight. Here's a snapshot of the data:

Day	Time	# of packets
4	36404	86
4	36414	100
4	36425	157
4	36435	334
4	36445	72
4	36455	50
4	36466	53

Fig. 9: Dataset Sample

This is only a small sample; in reality, there are over 1200 rows of data collected over the period of a week, at various points in the day. For example, it was observed that the traffic was low early in the morning, since there are very few people on the campus actively looking something up. While at lunchtime, there are many more students, doing something on their phones and/or laptops. Thus, the traffic reading at that time shoots up.

Data was also gathered using one more method; the number of unique IP addresses on the network was noted at an interval of 30 minutes. That gives a fair estimate of the number of people present in the vicinity at that moment in time. Since the computer that ran that program detected IP addresses of the devices connected to the same access point, it gives a fairly accurate reading of the number of devices connected to the same access point, and therefore, in the area. Assuming an average of 2 devices per person (since everyone has at least a computer and a phone connected to the network), half the count can be considered as the count of people present, at least as a proof of concept.

Data Analysis

This was stored in a table in a SQLite database. When it was ready to be analyzed, it was extracted in Jupyter Notebook, and a Random Forest Classifier was trained on with the day of the week and time of day as inputs and the number of packets as output. This model predicts the output with an accuracy of 90.3%.

The Random Forest Classifier was chosen since it yielded the best accuracy without over-fitting. Other methods that were tried were K Nearest Neighbors, Binary Tree Classifier, etc. The Random Forest Classifier is an ensemble learning method for classification and regression. It constructs multiple decision trees while training, runs the prediction on all of them and returns the class that is the most frequently reported by the set of decision trees.

The unique IP address data was gathered for two days, and the prediction is done using Newton's Backward Interpolation technique. The value passed to the function is the number of hours elapsed since midnight on the Monday of that week. For example, if you wanted to calculate the time of day for 11:30 a.m. on Thursday, that would be 3*24 + 11.5 = 83.5 hours.

A combination of the unique IP address and packet volume data is used to calculate an activity index, which is calculated using the following formula:

$$\frac{\#\ of\ people\ predicted\ at\ time\ t}{\#\ of\ packets\ at\ time\ t} * 10$$

The result of this is an index between 0 and 10, 10 being highly active and 0 being almost deserted.

Visualization

Fig. 10: GUI Screenshot

The data collected in real time by the system is visualized using a web based GUI designed in Django.

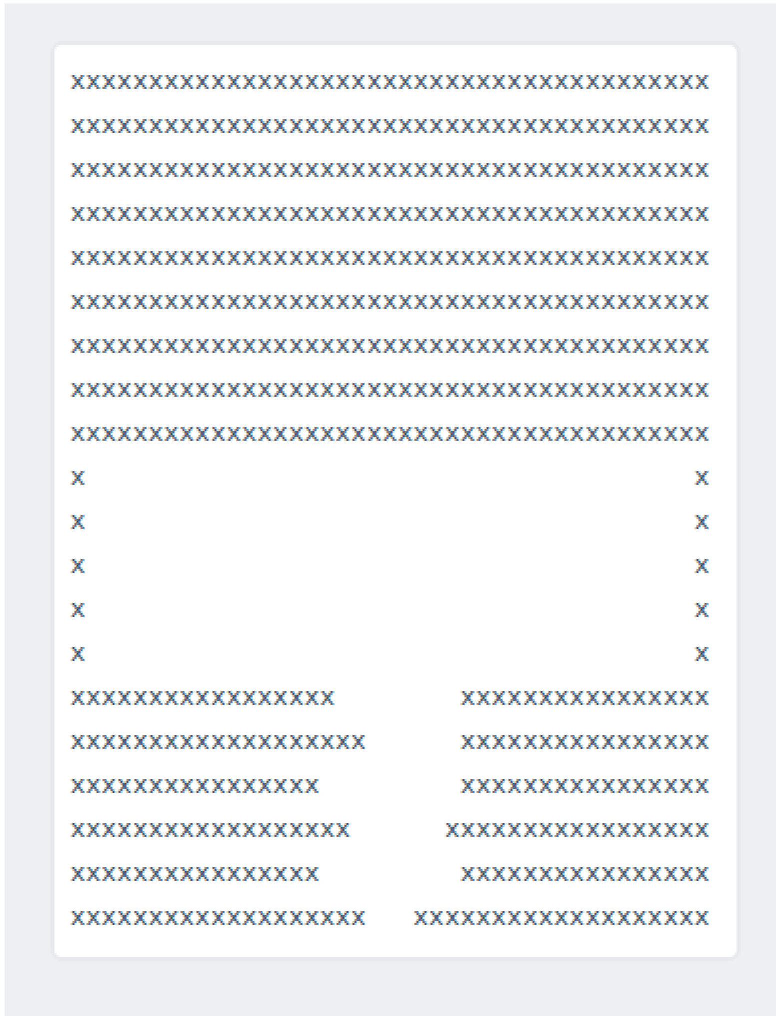

Fig. 11: Real-time Map Pane

The first pane shows a live map of the environment. The x's represent obstacles. The rows on top represent the distance of the nearest obstacle from the front sensor, and the ones on the left and right represent nearest obstacle from left and right sensors. Thus, the map shown can be used to maneuver through areas with low visibility.

There is also an option to download the map, which can be saved to file and used later.

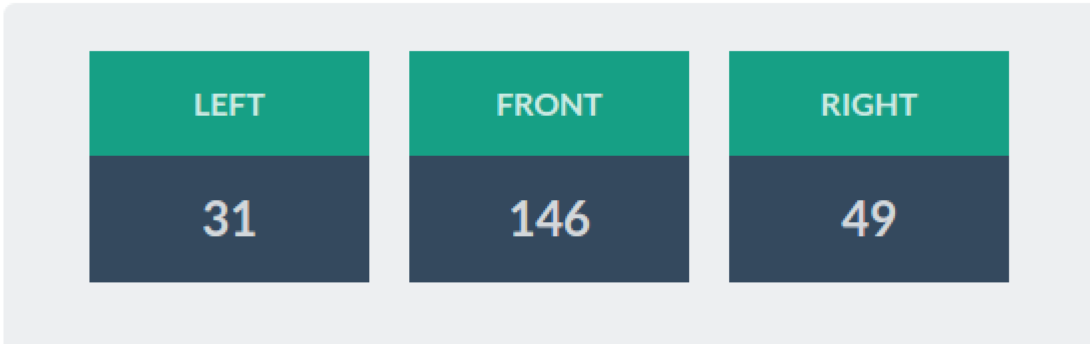

Fig. 12: Non-erroneous Sensor Readings Pane

The second pane shows real time distance readings from the sensors, left, front and right. If the distance falls below 20 cm, it turns red, since the system is too close to that particular border. The idea is these readings combined with the map can provide a reliable means of controlling the robot in low visibility.

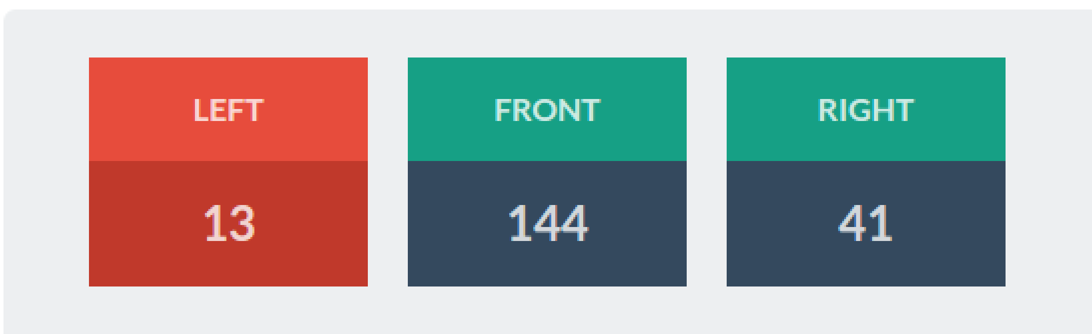

Fig. 13: Erroneous Sensor Readings Pane

In the case shown, the system is too close to the left border, and the operator should ideally move it towards the right before continuing on.

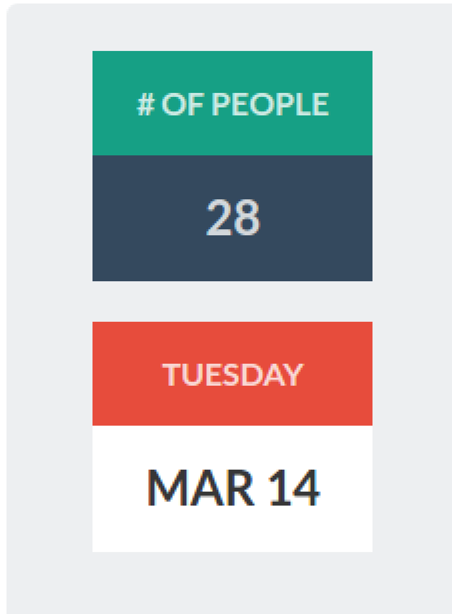

Fig. 14: Predicted Number of People Pane

The third pane shows the predicted number of people and the activity index, which is calculated from the unique IP addresses, as mentioned in the Data Analysis section.

Chapter 6: Results

Activity Index

As mentioned before, the activity index is a parameter that depicts how active the network is at any particular time on a scale of 0 to 10. It is calculated by taking the ratio of the estimated number of people present and the number of packets flowing across the network, both at that particular time. It is then scaled down or up (depending on the value) to fall between the 0 and 10 range, which then serves as an indicator.

The activity indices observed over the period of a week are presented in the graphs shown below:

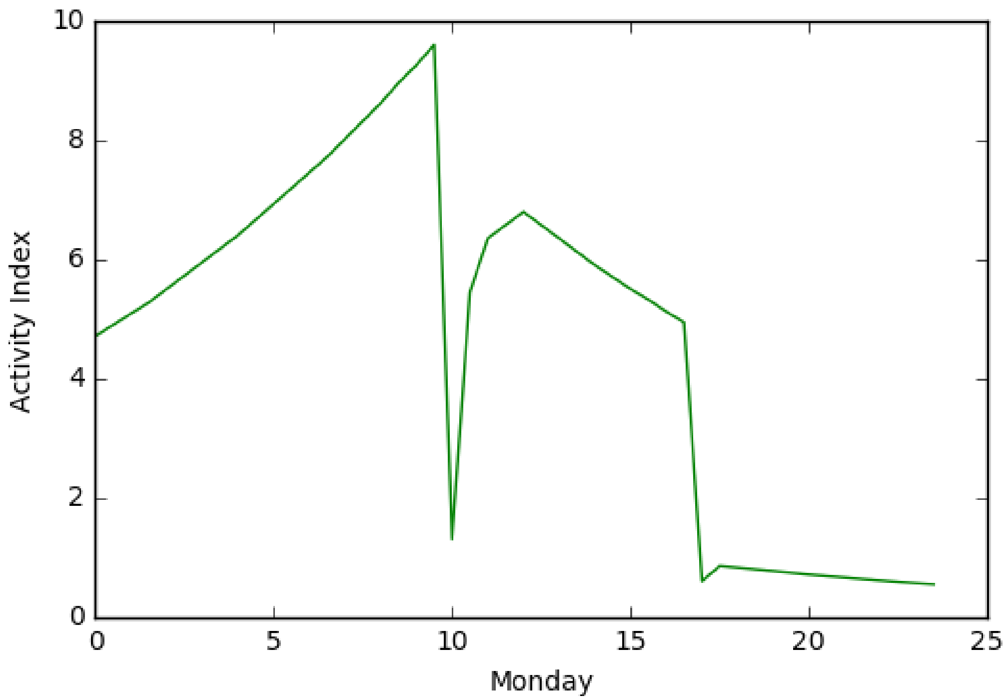

Fig. 17: Activity Index Graph – Monday

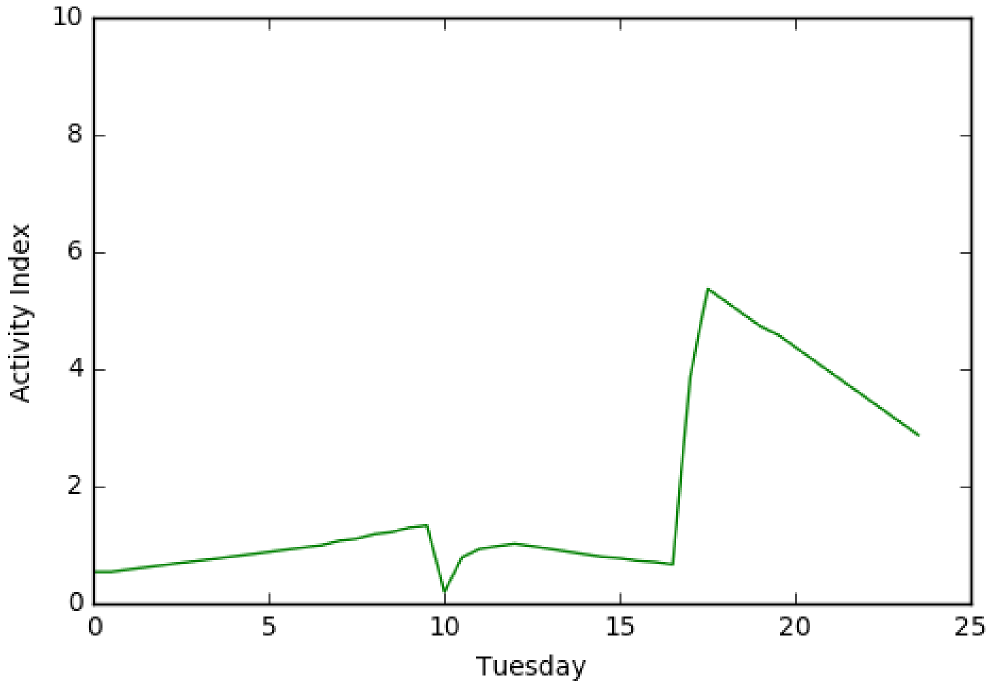

Fig. 16: Activity Index Graph – Tuesday

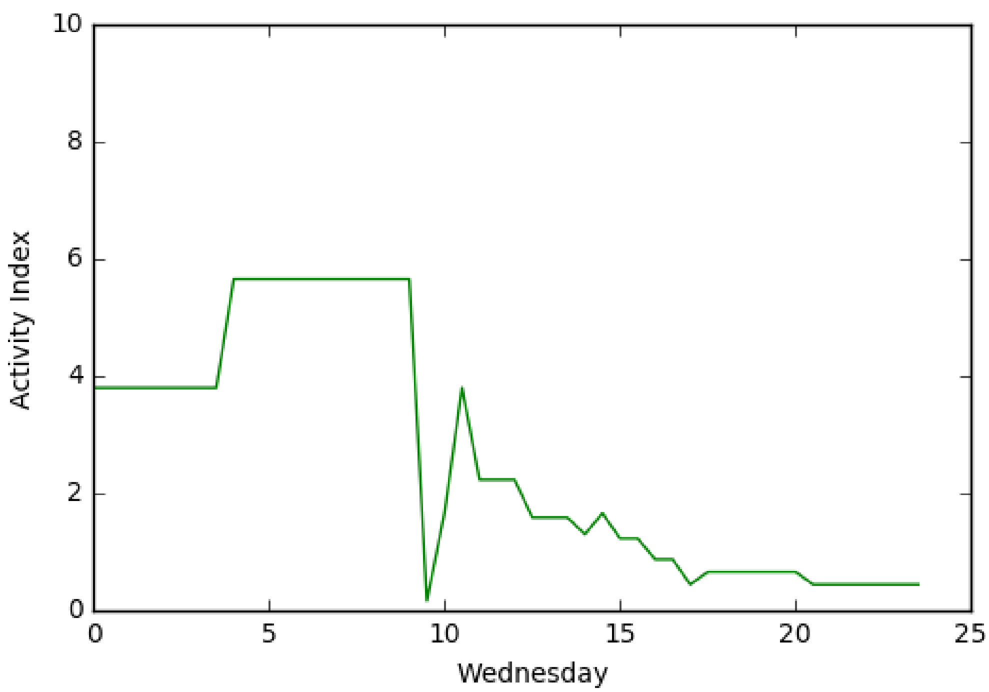

Fig. 17: Activity Index Graph – Wednesday

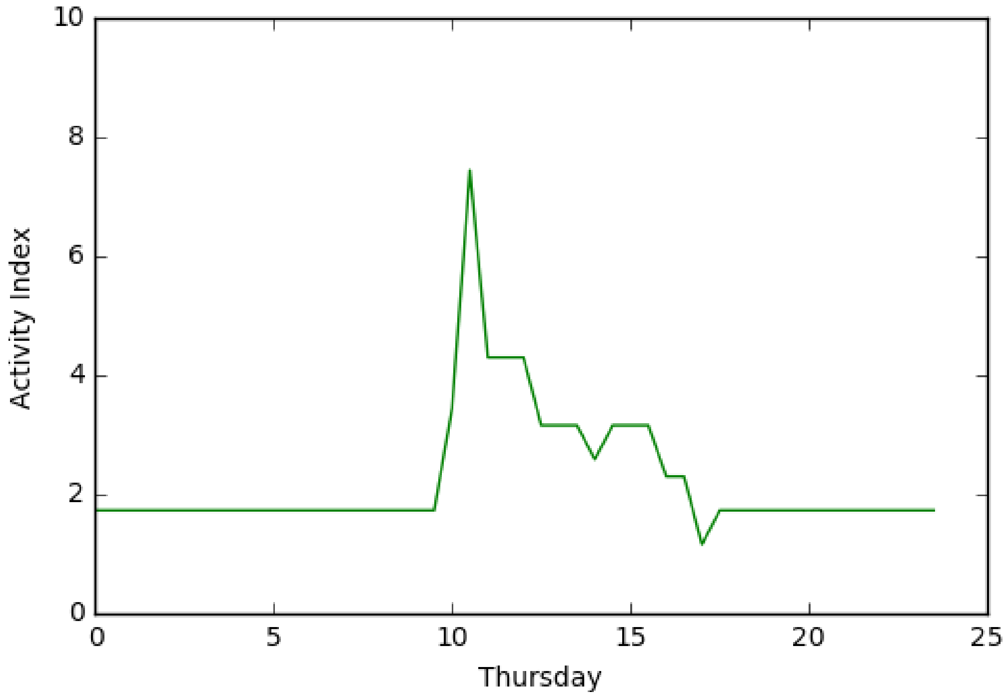

Fig. 18: Activity Index Graph - Thursday

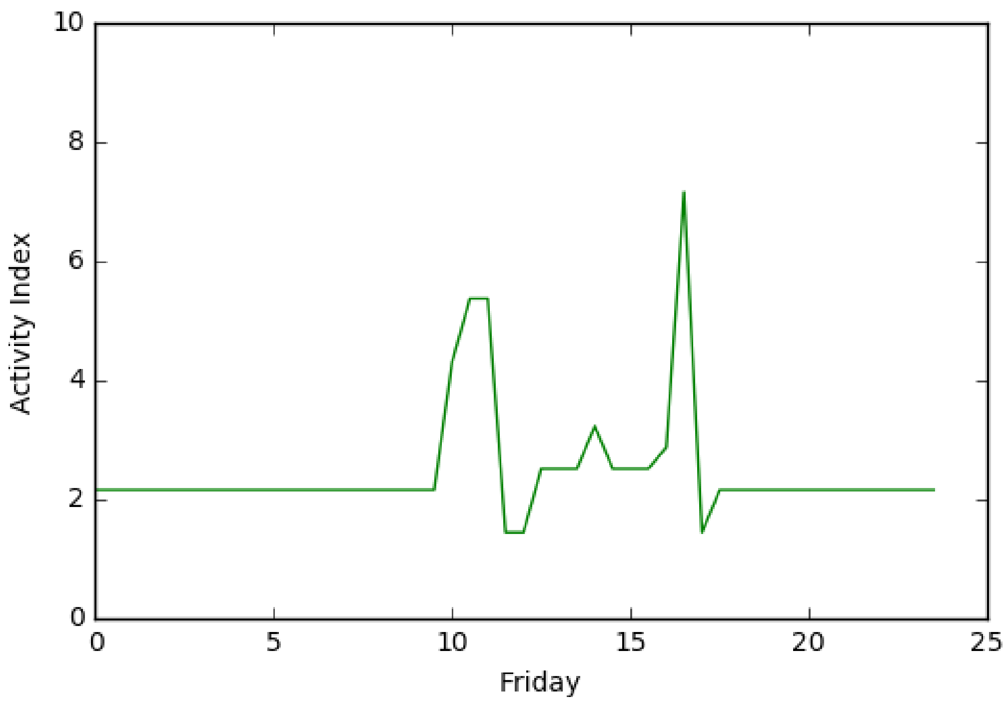

Fig. 19: Activity Index Graph - Friday

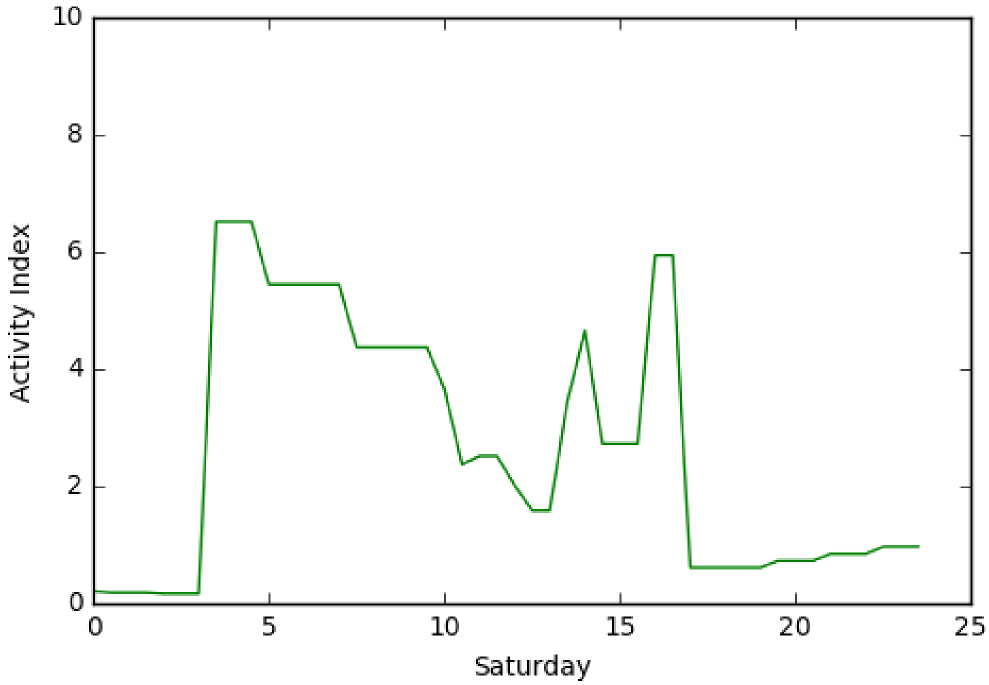

Fig. 20: Activity Index Graph - Saturday

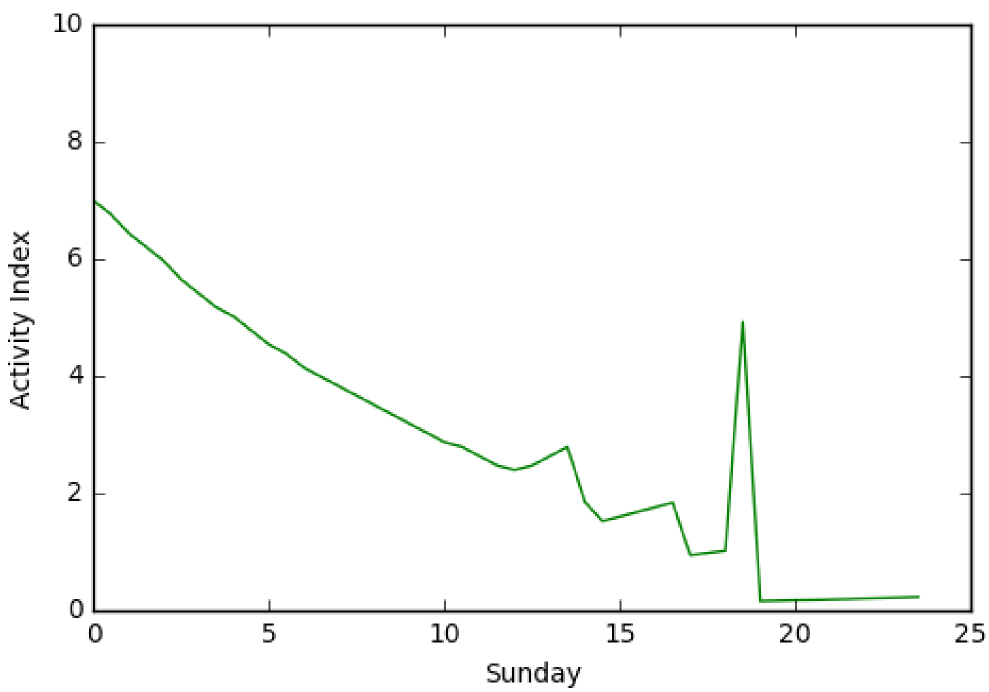

Fig. 21: Activity Index Graph – Sunday

We predicted the time period between 10 a.m. and 4 p.m. to be the most active. As can be seen, most graphs follow this rule more or less; there might be a slight shift in the activity zones, but it is true in general. For example, we predicted the activity would reduce as the day progresses after 3 p.m., and people start leaving the campus. It reaches close to 0 at night, when there is almost no one in the department building, since the day is over.

The readings fluctuate and there are outliers for mainly two reasons:

1. The graphs have been generated from data predicted over one week, but the activity profile for one day will vary from the same day of some other week.
2. The subnetwork structure is more complicated than we had imagined, and there is no concrete way of saying from which part of campus the activity is being detected. A longer-term study and in-depth understanding of the configuration of the university network will provide greater specificity of traffic patterns during data collection and analysis, to include identifying and isolating only traffic exchanged within a single building. However, as a proof of concept, the approach is sound and paves the way for follow-on research.

Real-time Map

The Beam, equipped with the distance sensors was maneuvered through a narrow hallway. Since the maps used in the demonstration was constructed using random distance readings, it looks winding instead of parallel. But since the hallway is a perfect rectangle, we expected the readings to be equidistant, and the map to be a rectangle too, which was achieved. This is what that map looks like (since the distance in front was more than 200 cm, there are no full rows of Xs):

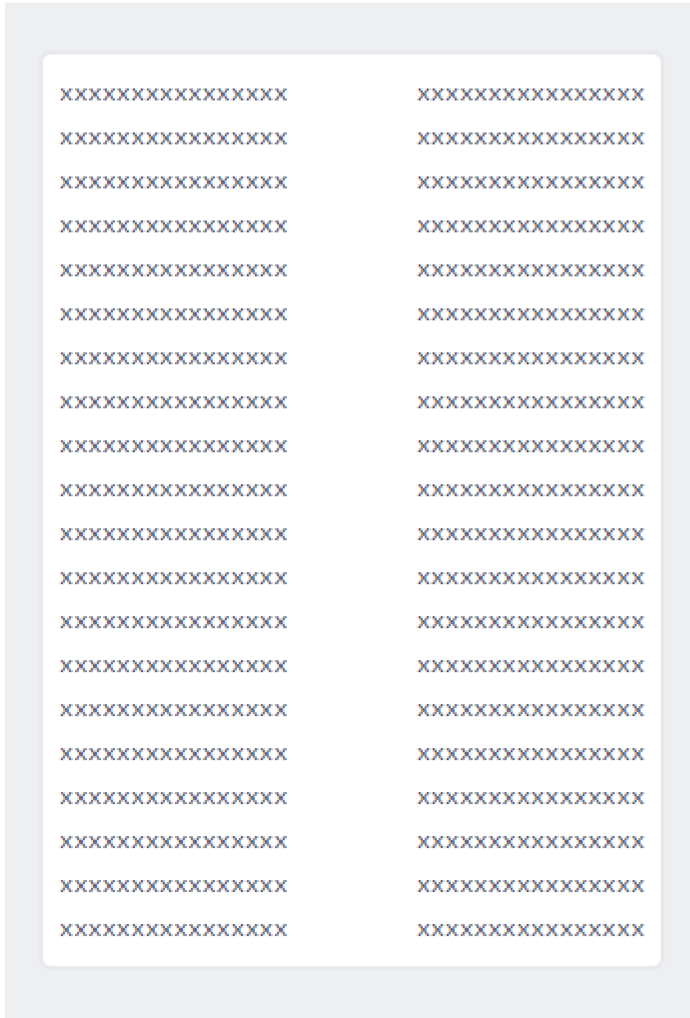

Fig. 23: Real-time Map – Narrow Hallway

Chapter 7: Conclusion and Future Work

Conclusion

This thesis provides a detailed description of the system that was built to server two main purposes: anomaly detection and reconnaissance. The main objective behind this research was to provide an intuitive interface to the operator to be able to perform the aforementioned objectives effectively. This thesis also discussed the various methods the data was collected, analyzed and presented to the user.

The first objective was to build a system that can be tele-operated to perform sentry duty, map the environment and report anomalous behavior; this was achieved by mounting ultrasonic distance sensors to the Beam robot, and by using predictive learning on the data that was gathered.

The second objective was to build a system that can perform real-time reconnaissance, interpret the information from its sensors in a format readable by the operator, and transmit information back reliably and intuitively. This was achieved by creating a web GUI to assist the operator to collate and receive all the data in one place.

Future Work

The prediction of the number of people can be improved by adding proprietary wireless drivers that allow promiscuous mode; this will allow the computer to get the actual location of the devices connected to the wireless network. This will improve the accuracy of the prediction, thus making the information displayed more relevant to the operator.

The robot can be made even more powerful by equipping it with a variety of sensors; right now it has only ultrasonic distance sensors, but by adding other sensors like temperature, light etc. the overall utility of the system can be enhanced. For example, adding light or heat sensors will also allow the operator to see a heat map of a burning building, enabling them to choose a path for maneuvering the robot through areas of comparatively lower heat. Heat sensors will also allow the operator to view heat signatures of survivors in hazardous zones, like nuclear disaster sites.

Thus, a lot of work can be done on automating sentries. Sentry robots can also be tailor-made for different situations, as this thesis shows.

Bibliography

[1] Everett, Hobart R. A microprocessor controlled autonomous sentry robot, *Monterey, California. Naval Postgraduate School.* Aug. 1982

[2] Harrington, J.J., & Klarer, P.R. (1987). SIR-1: An autonomous mobile sentry robot. United States.

[3] Johan Åkerberg, Mikael Gidlund, Mats Björkman. Future research challenges in wireless sensor and actuator networks targeting industrial automation, *9th IEEE International Conference on Industrial Informatics*. 2011

[4] Kevin Chin Yiu Shum, Quan Jia Cheng, Joseph Kee Yin Ng, Donna Ng. A Signal Strength based Location Estimation Algorithm within a Wireless Network, *IEEE International Conference on Advanced Information Networking and Applications.* 2011

[5] K. Lorincz, D.J. Malan, T.R.F. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, S. Moulton. Sensor networks for emergency response: challenges and opportunities, *IEEE Pervasive Computing.* 2004

[6] Boo-Ho Yang, Sokwoo Rhee. Development of the ring sensor for healthcare automation, *Robotics and Autonomous Systems.* Feb. 2000

[7] Guilherme Mussi Toschi, Leonardo Barreto Campos, Carlos Eduardo Cugnasca Home automation networks: A survey, *Computer Standards & Interfaces.* Feb. 2017

[8] Dhiraj Sunehra, Veena M. Implementation of Interactive Home Automation Systems Based on Email and Bluetooth Technologies, *International Conference on Information Processing (ICIP).* 2015

[9] Neaz Md. Morshed, G.M. Muid-Ur-Rahman, Md. Rezaul Karim, Hasan U. Zaman. Microcontroller Based Home Automation System Using Bluetooth, GSM, Wi-Fi and DTMF, *International Conference on Advances in Electrical Engineering (ICAEE).* 2015

[10] Gartner's 2015 Hype Cycle for Emerging Technologies Identifies the Computing Innovations That Organizations Should Monitor
http://www.gartner.com/newsroom/id/3114217