Fall 12-2-2020

Bayes' Network and Smart Sensors – Occupancy Detection

Donald Tryon

University of Nebraska-Lincoln, dlevitryon@gmail.com

BAYES' NETWORK AND SMART SENSORS – OCCUPANCY DETECTION

by

Donald Levi Tryon

A DISSERTATION

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Doctor of Philosophy

Major: Engineering

(Construction)

Under the Supervision of Professor Avery Schwer

Lincoln, Nebraska

December, 2020

BAYES' NETWORK AND SMART SENSORS – OCCUPANCY DETECTION

Donald Levi Tryon, Ph.D.

University of Nebraska, 2020

Advisor: Avery Schwer

This dissertation presents a Bayesian analysis for determining residential occupancy using inexpensive commercially available passive infrared (PIR) motion detectors, compared against two other detectors that were used to establish ground-truth. One of the ground-truth detectors was a GPS signal from a smartphone, the second was a Bluetooth key fob. Data were gathered from four residential locations, and then analyzed to determine occupancy. The occupancy data collected from the PIR sensors were compared against ground-truth to verify the results of the PIR sensor events that were collected every minute for a week. The Bayesian training data that was used to determine the prior probability used a four-week time period collected once a minute.

Having established the correspondence between ground-truth and the PIR sensor events, the PIR data were then used to build Bayesian network conditional tables. Once the conditional tables were constructed, the Bayesian network results could be compiled and then compared against the ground-truth data.

One analysis compared the ground-truth data against the performance of individual PIR sensors and showed that there was a low correlation between the PIR motion and occupancy. Further analyses compared the ground-truth data against the performance of various groupings of PIR sensors within each residence and showed that there was a little less correlation than the individual PIR sensors method.

When Bayesian modeling was applied using historical PIR sensor data, results demonstrated an improvement in occupancy detection over the individual and grouped PIR sensor methods that were evaluated. The historical sensor data (using PIR sensor signal pulses) was successfully applied to the network, with an average of .025 ϕ correlation improvement. The historical presence data (using ground-truth data) were then applied to the same network. This step improved the ϕ correlation between the PIR sensors and ground-truth by an average of .40 over the four locations. These findings show that applying Bayesian modeling improves the accuracy of occupancy detection required for safety and efficiency, which will permit occupants to live in their homes longer.

Table of Contents

List of Figures

List of Tables

CHAPTER 1: INTRODUCTION

The need for home security is becoming more prevalent, especially for older people, due to health and safety concerns. These concerns are creating a need for security systems with reliable technology. Technological solutions have their own specific flaws, depending on the occupancy detection system they implement. These flaws can create false indications or false alerts. Most of these security systems err on the side of registering false negatives, a tendency that can be expensive since charges for regular false alarms are often levied.

Owners want reliable systems customized to their own lifestyles. A system that avoids the false alerts and other concerns can prompt fines and other penalties, eroding confidence in that system. More accurate ways to develop these system algorithms, customized for each owner, are therefore desirable. This dissertation presents a method to help prevent some of these false alerts, using a Bayesian network approach to occupancy data processing and analysis.

The goal of this dissertation is to demonstrate that a detection algorithm applied to passive infrared (PIR) sensors can enhance the detection of occupancy within a space. This study investigates: “When using a set of PIR sensors and applying the Bayesian method as an overlay, is this study able to improve the accuracy of presence detection within the system?” The hypothesis is as follows:

> H_1: When applying Bayesian modeling in conjunction with PIR sensors, the accuracy of presence detection will improve within a given location compared to independent sensor detection.

H_0: When applying Bayesian modeling in conjunction with PIR sensors, the accuracy of presence detection will have no effect on the improvement within a given location.

Chapter 2 defines what occupancy is and how an occupant is detected in this research. It also reviews literature on three main types of detection methods which can improve upon occupancy detection, such as detectors, circuit designs and analytical approach methods. Each of these three methods can improve the accuracy of occupancy detection, which then may be implemented within the project. Chapter 2 also explains the Bayesian model to help understand the principles of conditional tables, nodes and arcs which are used. Bayesian research examples of how other researchers have implemented the modeling will be presented.

Chapter 3 describes the methodology that is implemented in this research. This chapter provides the flow chart of how the research is constructed and defined. The process of how the PIR sensor data was collected, cleaned, and processed will be described in this chapter. Additionally, chapter 3 describes the independent sensor, grouped sensor and the Bayesian modeling that is applied to the sensor data to generate a prediction of occupancy.

Chapter 4 is included to provide detail on the specific architecture of the Bayesian network for this project. This chapter presents the seven main steps in building a Bayesian network as described later. After the seven steps, the construction of the Bayesian network truth tables was presented. This chapter is included to better present how the method was constructed.

Chapter 5 compares results of various methods from independent, grouped, trained Bayesian sensor hours and the trained ground-truth method. The independent sensor method is where each of the PIR motion sensors act independently of each other. The grouped method is where two groups of three PIR motion sensors are established and two of the PIR motion sensors are required to trigger an event. Trained PIR sensor hours method is where the history of the PIR motion sensors are implemented. Finally, the trained ground-truth method uses the historical data collected from the ground-truth detectors. This chapter then compares how each of these methods relate with one another. The results of the overall methods are then summarized.

Chapter 6 describes the stakeholders and the benefits of applying Bayesian network to each of their applications. This chapter also presents future research that should be done to improve the accuracy of the results. In the end of this chapter, a summary of the research project and final comments and results are established.

CHAPTER 2: LITERATURE REVIEW

This chapter explores the previous research and various analytical approaches for detecting occupancy. Each of the research discussed will provide further insight in methods that were used previously to build a deeper framework for learning how applying analytics provides improved detection predictions. Section 2.1 defines what occupancy is and how it relates to a given space. Section 2.2 demonstrates how algorithms are used to detect occupant patterns and behaviors from historical information. Section 2.3 presents four detection methods such as: 1) detectors, 2) circuit designs, 3) multiple sensors, and 4) analytical approaches. Section 2.4 describes the foundation and the basic principles for Bayesian method. Finally, Section 2.5 concludes this chapter with an overview of the research and expresses the gaps in previous research that this project attempts to fill.

2.1 Defining "Occupancy"

Occupancy is defined by Merriam-Webster as "the fact or condition of holding, possessing, or residing in or on something or the fact or condition of being occupied" (Merriam-Webster, n.d.). The definition of "occupancy" used in this dissertation is the true detection of the presence of a target within a given space. Many current systems used in commercial, industrial, and residential applications use standalone detectors, applying no further analysis to the individual detector signals, despite that this additional step could improve detection occurrences. Many methods to detect whether a space is occupied have been developed.

2.2 Occupancy Detection in the Field of Healthcare

This section discusses the applications of sensors for detecting occupancy in the field of health studies. These applications may allow users to retain residency in their homes for longer by providing necessary information indicating detection of occupancy within a given space. The use of occupancy detection is a main goal of many detection systems in healthcare, due to the desire of the elderly to stay at home for as long as it is safe to do so.

This desire to stay in one's own home is a challenge in the occupancy-detection field, since it is predicted that 23% of the world's population will be 60 or older by 2050 (Al-Shaqi, Mourshed, & Rezgui, 2016). Inside that range is the fastest growing age group in the world, namely individuals 80 years and older (Labonnote & Høyland, 2017). These large and expanding numbers are the main reason behind the need to provide care within a home or a given location. Several studies present ways to implement monitoring methods to support these age groups.

Ambient assisted living (AAL) refers to the use of different sensors to make the independent life of a person safer and more comfortable in the home environment (Demir, Köseoğlu, Sokullu, & Şeker, 2017). The AAL system can provide comfort by monitoring heating, ventilation, and air conditioning (HVAC) to provide an optimal living environment (Al-Shaqi et al., 2016). In many cases, AAL can offer continuous and real-time monitoring of not only the environment but also occupant behavior and health (Al-Shaqi et al., 2016). Some additional systems that Al-shaqi et al. identified are installed safety lights that provide warnings regarding medications, for example, when

medications are not taken. The use of AAL can provide the extra help required for the elderly to stay within their homes longer.

AAL is typically implemented using different sensors, including magnetic switches, temperature sensors, photosensors, pressure pads, water flow sensors, infrared motion sensors, power/current sensors, force sensors, smoke/heat sensor, and biosensors (Al-Shaqi et al., 2016). In addition to adding sensors that can be used to notify healthcare providers if there is a harmful accident such as a fall, Al-Shaqi et al. applied probability theories to anticipate results.

Al-Shaqi et al., considering the use of raw sensor data in their project, note that several key events must take place before the data can produce valid results. The first of these key steps is that “noise” must be eliminated from the raw data; an analysis must be defined and applied to detect the desired patterns. Such steps can include probability distribution or clustered analysis, among others. Once the analysis method is selected, it can then be modified depending on the data type. The data can then be applied to a learning method. A certain amount of training is required, depending on the activity, so that the patterns can be determined. Once the training is completed, the current data or live data can then be injected to produce a result (Al-Shaqi et al., 2016).

This method of monitoring activity using probability and prediction have been successful in test cases (Huynh, Fritz, & Schiele, 2008). Huynh et al. were able to detect certain activities using wearable sensors. Some of these activities included walking, walking while carrying things, driving a car, picking up cafeteria food, sitting at a desk, eating meals, and washing dishes. They produced their results by observing a person’s daily life activity over a 16-day period. This person was wearing two wearable sensor

devices attached to their body. During this time, the subject was required to demonstrate 75 distinct activities that were, compared to the data and grouped for matching. Machine learning was able to predict what activities the person was doing for each segment, with an overall 72.7% accuracy. This method is being used in active wear devices to predict the movements and activities of the users.

When applied to monitor elderly individuals, predictions can be quite beneficial. For example, Demir et al. demonstrated that the use of multiple sensors to gather information with machine learning would help in the diagnosis and treatment of diseases within the elderly population (Demir et al., 2017). For example, Demir et al. provided information on movements to healthcare professionals that was then used to create a plan to encourage movement for the patients. A series of sensors were deployed in the kitchen, toilet, bathroom, and bedroom. These sensors gathered an array of raw data. Using fuzzy logic, the raw data were then analyzed to find behaviors that were out of the normal ranges. The results told the end user if something was open, not open, or simply not completed using the methods of their study. The conclusion of their study was that they could successfully combine normal occupants' behavior with sensor data to predict specific events that may occur. This concept of predicting normal behavior will become useful to predict occupancy normal behavior and sensor data.

The successes of AAL systems have benefitted those elderly people who desire to continue living at home (Gokalp & Clarke, 2013). Gokalp and Clarke showed that AAL systems that implemented sensors could predict patterns or occupancy. There is a need for better-designed studies with test cases of greater duration, as many of these studies cover only a few days. They also noted that longer-term case studies might be difficult to

conduct, since the researchers would first need to get permission from both users and patients. In the end, many positive results have come from applying analytic modeling in parallel with sensors to determine occupancy and patterns.

2.3 Detection Methods

Many types of detectors and methods are used for occupancy detection. Section 2.3.1 describes the main types of detectors and presents their benefits and shortcomings when detecting objects or targets. Section 2.3.2 examines the different types of circuit designs that are used to improve sensor detection. Section 2.3.3 discusses the analytical approaches that have shown to improve the accuracy systems.

2.3.1 Detectors

Merriam-Webster defines a “detector” as a “device for detecting the presence of electromagnetic waves or radioactivity” (Merriam-Webster, n.d.). Detector construction varies, as does the way they detect objects within a given range or defined space. Guo et al. details the main types of sensors: PIR, ultrasonic, microwave, sound, light barriers, video, biometric, and pressure (Guo, Tiller, Henze, & Waters, 2010). These main sensor types are evaluated below.

2.3.1.1 Passive infrared. PIR sensors, also called “passive infrared detectors,” are electronic sensors that measure infrared light radiation from the objects that cross the field of view. PIR sensors are often used in motion detectors. They are constructed with a crystalline material at the center of the rectangle on the face of the sensor, which detects the infrared radiation. This sensor is split into two parts to allow the sensor to detect not only the infrared radiation itself, but also the change in the condition when the target enters and crosses the field of view. When the condition changes, the amount of infrared

radiation on the element changes, generating a variance in voltage output. An onboard amplifier drives an output to a relay or a microcontroller (Digikey, 2012).

PIR sensors use a Fresnel lens to focus the infrared energy emitted when a target enters the field of view. The Fresnel lens creates the detection pattern. In a detection pattern, the target must cross the PIR sensor to detect a changed event. Figure 1 is the detection pattern for ZMOTION ZEPIR0AA, a PIR sensor.

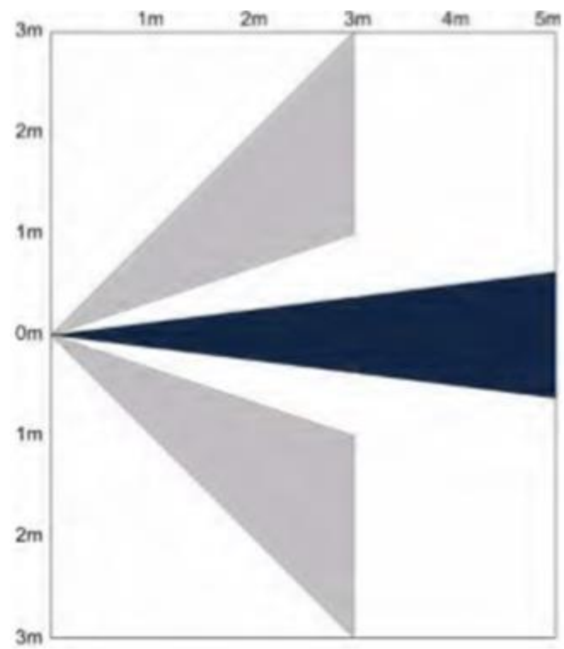

Figure 1: Detection pattern ZMOTION ZEPIR0AA sensor: The sensor provides a 60-degree cone with four beams, or detection zones, with the two inner zones providing the greater range up to 5 m (Digikey, 2012)

PIR sensors are used in both residential and commercial building management, and in security applications. PIR sensors' advantages include low cost, simplistic implementation, small size, low power usage, durability, wide lens ranges, an intuitive interface, and the ability to operate with or without an additional light source. These advantages make them practical for indoor applications, such as operating the lights and plug sockets for residential and commercial sites. They can be installed in multiple areas to increase the reliability of the detecting an object. Practical examples include placing

them at each light switch or entryway (Keller, 2000). PIR sensors are completely passive. There is no need for an auxiliary source of power or extra devices that target the objects since it uses radiation from the targets. Detection is made without contact.

However, there are cases in which PIR sensors would not be practical, due to the limitations of the sensors, namely in the environment when there are many infrared sources like the sun, the size of targets being too small, the distance from the target and objects too close or far, and the number of windows and drafts that need to be taken into account (Mathas, 2012; Digikey, 2012; Gross, 2018). Given these limitations, PIR sensors can generate false detections.

The limited number of environments in which PIR sensors can operate well are of particular concern. Sensors operate poorly outdoors with great temperature fluctuations. Temperature changes can cause false alerts. One example of such false alerts can be observed with porch lighting activating throughout the day without a legitimate trigger. The frequency of the false alerts makes PIR sensors impractical for sensitive applications. The detection is made at a distance, which can result in missed triggering events where the target is not captured due to its distance from the sensor.

This limits the acceptability of PIR sensors for security applications because they may lead to many false alarms. It is necessary for security sectors to consider the weak points of PIR sensors when designing security systems for both residential and commercial sectors. Some of the ways the security sector can improve the reliability of these weak systems include adding both additional sensors and algorithms to improve reliability and reduce false triggering events.

Energy management with the use of the PIR sensors is more feasible, since the false alerts have less influence than they would within a security system. It is less important if a light turns on when no one is present than if the security alarm triggers when no one is around, and authorities are called to attend to false alarms.

People have overcome minor issues associated with PIR sensors in several ways. Kim et al. discovered that the placement of the sensors had great effect on detection accuracy (Kim, Moon, & Yoon, 2017). Sensors placed on the walls, rather than the ceiling, detected occupants most accurately, as did sensors near doors. To further improve accuracy, they applied an algorithm in which a door sensor was added with the assumption of 30-minute occupancy (i.e., the assumption someone was still located within the room). The door sensor was used to create a secondary check for occupancy. If no motion was detected after the door sensor contact was closed, then the room was assumed to be empty. In the absence of detected motion, an occupant was assumed to be within the space until the door sensor contact changed states again. Kim et al. reported 99.8% accuracy by verifying against the camera when two to six people were in the room, with 90.1% accuracy when a single person was in the room.

The use of PIR sensors can be improved, but people primarily use these sensors for only simple applications, due to their lack of accuracy. Hence, PIR sensors are more acceptable as individual sensors within energy management systems than within security systems, where they would require more accurate sensor methods.

2.3.1.2 Ultrasonic occupancy. Ultrasonic sensors are active sensors that transmit ultrasonic waves into the air and detect waves reflected by objects (Murata, 2008). Murata Manufacturing explains ultrasonic waves as sounds that cannot be heard by

humans, normally above 20 kHz. Increasing the accuracy of ultrasonic sensors, certain materials better reflect these waves: metal, wood, concrete, glass, rubber, and paper. Other objects are harder to detect due to their poor refractory properties: cloth, cotton, and material with an undulating surface of the material (Murata, 2008).

Ultrasonic sensors do not need a direct line of sight because sonic waves reflect off surfaces and partitions to enter spaces not in direct sight of the sensor device itself. Since this sensor method uses sound waves, the color or transparency of the objects has no effect on the sensor readings. The use of sound waves also permits these devices to work in dark environments, since they do not rely on visual input for detection. Ultrasonic waves can penetrate certain materials, allowing a sensor to detect what is inside certain objects. This property allows them to detect external or deep objects, depending on the frequency of the sound wave, supporting the accurate detection of objects when dust, dirt, snow, or rain is in the environment.

The weakness of ultrasonic sensors is they use sound. They are non-functional in a vacuum, for instance, as there is no air through which sound can travel. Ultrasonic sensors are also not designed for underwater applications, where radar sensors are more effective because they use a different frequency range and construction. Furthermore, ultrasonic sensors have difficulty sensing softer materials, such as fabric, since sound waves are absorbed by the materials.

Another weakness limiting the uses of ultrasonic sensors is their sensitivity; false triggers can result from any device that can produce squeaking sounds (e.g., a squeaky fan) within the detector’s frequency. Lastly, the range of an ultrasonic sensor (about 10 m) is longer than that of a PIR sensor, with a maximum range of 5 m. The longer

range can prevent an ultrasonic sensor from being used in scenarios that require a more focused approach, such as in a specific entry way or specific segment within a space.

Ultrasonic sensors are functional for several types of detection methods, including the detection of the signal level of a continuous wave, the measurement of pulse reflection time, the utilization of the Doppler affect, the measurement of direct propagation time, and the measurement of the Karman vortex. The detection of signal level in continuous waves is typically used for counting instruments or objects that pass a certain point, such as access switches or parking at parking meters/in parking spaces. Measurements of pulse reflection time are generally used for automatic doors, level gauges, automatic changeovers of traffic signals, and the back-up sonars on automobiles. Intruder alarm systems typically use the Doppler affect in detecting occupancy in all directions from a sensor. The measurement of direct propagation time is used for densitometers and flowmeters. Flowmeters can also use the Karman vortex in various applications.

Hammoud et al. describe customizing sensitivity per room to improve the accuracy of sensors (Hammoud, Deriaz, & Konstantas, 2017). They implement a manual adjustment and an automatic adjustment method to “tune” the sensor to the specific room and the frequency for better occupant detection. These adjustments were made to help overcome some of the native limitations of ultrasonic sensors. By manually tuning a sensor, these researchers were able to produce results of 98.2% more accurate occupancy detection, compared with a non-tuned sensor within a small room with manual calibration. Hammoud et al. also showed 97.5% improved accuracy occupancy detection

with automatic calibration to tune the sensor. Calibrating the ultrasonic sensors presents an efficient method to help improve sensor accuracy when detecting moving occupants.

These strengths make ultrasonic sensors more acceptable for use within a security system because they allow the detection of an out-of-sight target. In some cases, however, increased sensitivity can create undesired false alerts. Security systems created for residential and commercial sectors must account for these weaknesses when implementing ultrasonic sensors within the design of a system attempting to detect occupancy.

2.3.1.3 Audible sound/passive acoustic. Audible sound or passive acoustic sensors involve any combination of sound recorder, detector, microphone, and/or hydrophone designed to detect and record sound. They typically use a type of a microphone and/or hydrophone for security system design. Depending on the sound produced, they can set a threshold for the amplitude, wavelength, and frequency to trigger an alarm (Tarzia, Dick, Dinda, & Memik, 2009), for example to detect glass breaking. Guo et al. state that audible sensors are known to respond to non-human environmental noises and noises from adjacent spaces and are thus prone to false alerts (Guo et al., 2010).

Tarzia et al. assert that passive acoustic sensors are functional, depending on the type of application and locations of the sensors. Their results show “that it is possible to detect the presence or absence of users with near perfect accuracy after only ten seconds of measurements” (Tarzia et al., 2009, p. 1); however, it is important to point out that they were testing the presence of people, using a computer with web camera microphones

and sensors localized within the test space. It is important to determine other triggering events that might affect passive audio sensors before their implementation in each space.

2.3.1.4 Video cameras and CO_2. In addition to PIR sensors are other types of sensors that can be used for building security and energy-management applications. Two of these technologies are CO_2 sensors and cameras. The first method is to use CO_2 sensors to detect occupancy. Jin et al. researched the method of using CO_2 sensors within a room to detect whether there are occupants in the space. Using a CO_2 detector, namely the K30 10,000 ppm CO_2 sensor (Co2meter, n.d.; Jin et al., 2015), they measured CO_2 concentrations within a room. From the CO_2 levels, Jin et al. could predict the number of occupants within the area. Their method is demonstrated in Figure 2 (Jin et al., 2015).

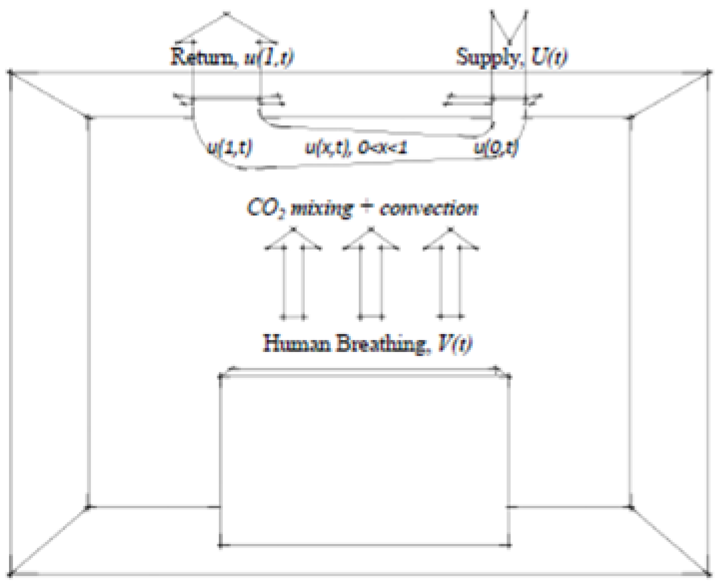

Figure 2: CO_2 model: Fresh air with CO_2 concentration U(t) enters the room from the supply vent and exits the room after convection and mixing with human breath; V(t) which rises to the ceiling, and the measured CO_2 concentration at the return vent is u(1,t) (Jin et al., 2015)

The second method is to use a camera to detect targets within view. The types of cameras to be discussed are low-resolution cameras. These low pixel cameras are selected to decrease the amount of video processing. These cameras view infrared heat

from targets rather than registering visible light. Berger and Armitage's research used a 16 by 16-pixel infrared camera to detect targets. The first step was to process the low-resolution image, which was done by rescaling the size to 64 by 64 pixels to increase the size and quality of the image. They then applied a background subtraction and Laplacian of Gaussian blob detection. Berger and Armitage could then divide the image into sections and identify whether the target had moved, as well as count the number of targets within the certain area (Berger & Armitage, 2010). The imaging processes can be seen in Figure 3.

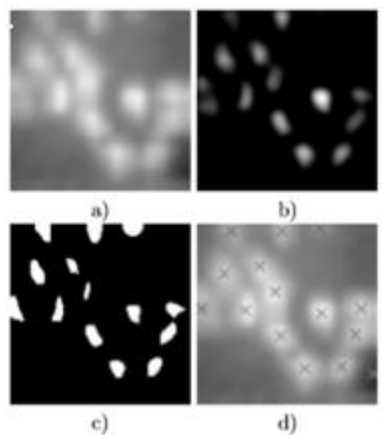

Figure 3: Infrared camera process: (a) Rescaled input image; (b) Warm objects after background subtraction and Log; (c) Binary image after area adjustment; (d) Result image (Berger & Armitage, 2010).

One key consideration with the use of cameras is that many people do not appreciate being surveyed (Eisa & Moreira, 2017). Elis and Moreira have pointed out that camera-based sensors are not widely accepted among the elderly due to their inconvenience, to computational complexity, and concerns over privacy.

CO_2 sensor and the infrared cameras can both provide another source of energy management and security applications for the detection of occupancy. Several major differences exist between the two methods that should be considered when using either of the methods. CO_2 sensors provide the accuracy needed to detect the correct number of

people within a conference room, but their measurements are delayed, which can be problematic. Jin et al. state that a “relatively long time (10–15 minutes)” is required to build up the corresponding levels of CO_2 concentration (Jin et al., 2015). This time-lag greatly weakens this method’s use for security purposes because efficient security requires a timelier response.

Infrared cameras, by contrast, can detect occupants in real time. Environmental concerns, windows, and air drafts are no longer a concern since they are filtered out with the image processing. This processing also allows more detection zones for movement detection. Berger and Armitage demonstrated this flexibility when increasing the 16-by-16-pixel resolution to 64 by 64 pixels to create more detection zones and increase accuracy.

Some of the similarities between the CO_2 sensors and infrared cameras are that both can detect how many occupants are within a certain area, increase the accuracy of detection for security and energy management applications, and adjust their algorithms to improve their accuracy for specific locations and scenarios. Some major differences also exist between these two technologies, however: time delay between the motion detections (infrared cameras require less delay); system-setup complexity (air quality adds more complexity); system costs (infrared systems can cost more depending on the quality of the cameras, but air systems can cost more to install if they are not easily accessible); and suitability for use in security requirements (for which CO_2 sensors are unsuitable, although they are more suitable for environmental applications). Elisa & Moreira presents a comparison between PIR and cameras for each technology property, which are shown in Table 1.

Table 1: Sensing technologies, a properties comparison (Eisa & Moreira, 2017).

Property	PIR	Camera
Location detection	Low	High
Presence detection	Medium	High
Tracking	Single user	Multi-users
Resolution	Single bit (on/off)	High
Cost	Low	High
Privacy concern	Low	High
Battery life	High	NA
Require data processing	Medium	High
Localization accuracy	Low (room-level)	High

2.3.2 Circuit design

The accuracy of detection can be improved by implementing a secondary circuit design into the base sensors, so they form a network. Instead of using a single standalone sensor, a linked secondary sensor can greatly increase the accuracy of the system as a whole. Two ways this accuracy improvement can be made are with the addition of circuitry in a series or in a parallel configuration.

Secondary inputs to occupancy detection can also be added to help decrease the number of false alarms. The use of several types of sensors that can help improve and build upon the network is preferable. For example, infrared sensors can be paired with audio sensors, since the infrared sensor does not detect audio within a given space. The goal of the circuit design is to improve detector accuracy by implementing secondary checks in the network to better account for environmental errors in each space.

The use of multiple sensors can greatly improve the quality of a security system. The two main methods used to implement multiple sensors into a system are the addition of sensors in a series to each other or in parallel with each other (Romeu, 2004). Each of these methods has its own implementation methods to improve detection accuracy.

2.3.2.1 Series sensors. Sensors used in series require only one sensor in the group to be triggered for the whole network to register an alarm. Figure 4 shows a typical series design for three sensors.

Figure 4: Series sensor configuration: the arrows in the figure represent the connection between the sensors. These could be wired or wireless connections. Each of the three sensors is in a single loop, which connects back to the detection system. If any of the three sensors are triggered, the whole loop breaks and goes into alarm.

For example, if Sensor A is a front-entry door contact, Sensor B is a motion sensor, and Sensor C is a back-entry door sensor, then when any of the contacts are opened, the whole alarm triggers due to any one of the series contacts opening to detect an intruder.

However, sensors used in a series have a major flaw: if any fails, all fail. This flaw can be represented mathematically. Given a Sensor A that has a 90% reliability rate, adding a second Sensor B in series to the circuit with the same accuracy will now decrease reliability to 81%. The series system's reliability calculation is shown in Equation 1 (Romeu, 2004):

$$Rs = R1 \times R2 \times \ldots.\ Rn \quad (1)$$

$$Rs = 0.9 * 0.9 = 0.81 \text{ or } 81\%$$

One way to overcome this series flaw is to add sensors in a parallel configuration.

2.3.2.2 Parallel sensors. Sensors can also be linked in a parallel configuration as in the Figure 5.

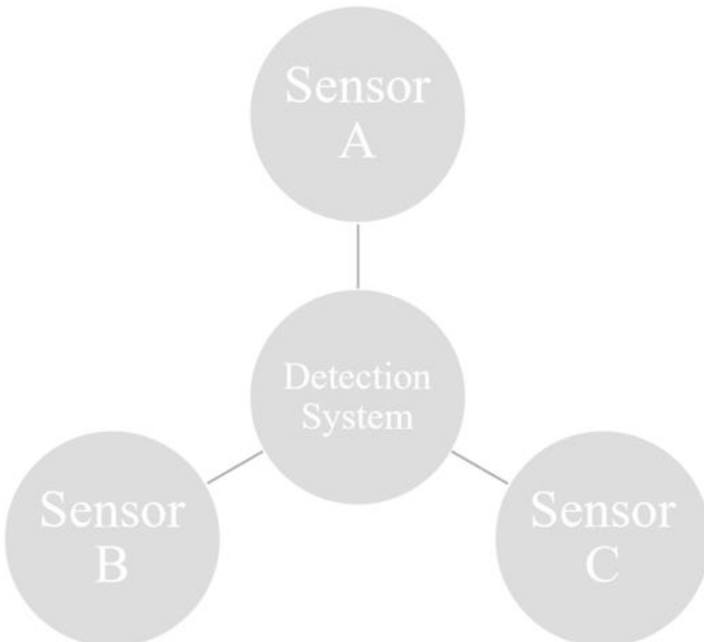

Figure 5: Sensor parallel configuration shows each of three sensors separately connected back to the detection system in the network. If any of the three sensors are missing, or triggered, unlike the series sensors, the whole loop does not go into alarm, as it would have in the series arrangement.

An algorithm determines the type and how many sensors need to activate for the system to identify the event. For example, Sensor A is a front-entry door contact, Sensor B is a motion sensor, and Sensor C is a back-entry door sensor. When both door sensors are open then a motion sensor is activated and the whole alarm triggers an event. This helps prevent false alarms since it takes two events at the same time to generate a positive event.

This example demonstrates how parallel sensors can overcome the major flaw within a series system. Individual sensors are not connected into a chain that links them. Rather, they are connected into a star shape in which each of the inputs are independent of the other. Demonstrated mathematically, if Sensor A has a 90% reliability, due to its specific design, adding a second Sensor B in parallel to the circuit with the same accuracy will increase the reliability to 99% (RAC). The parallel system's reliability calculation is shown in Equation 2:

$$Rp = 1 - (1 - R1) \text{ x } (1 - R2) \text{ x } \ldots. (1 - Rn) \quad (2)$$

$$Rp = 1 - (1 - 0.9) *(1 - 0.9)$$

$$Rp= 1 - (0.1) * (0.1)$$

$$Rp = 1 - 0.01 = 0.99 \text{ or } 99\%$$

The parallel system shows improvement from 81% to 99% when sensors are added. One common way to add sensors in both series and parallel systems is by programming an algorithm in which each of the sensors are considered. The parallel configuration clearly guarantees superior accuracy over the series configuration. Other analytical approaches may increase accuracy.

2.3.3 Analytical approach

Advancements of computers and microcontrollers have brought significant improvements to the use of real-time analytical approaches for determining occupancy. The basic principle of these methods is to use a collection of past data to calculate a predictive algorithm to better determine the state of an area and the occupancy within an area. Le presents common ways to determine these predictive equations (Le, 2018): linear regression, logistic regression, linear discriminate analysis, classification and regression

trees, Bayesian network, *K*-nearest neighbors learning vector quantization, support vector machines, bagging and random forest, and boosting and AdaBoost.

A linear regression uses one dependent and one independent variable to determine the slope of the line, which can be used to determine other predictions such as estimated time before an event occurs or when there will be an occupant within the space. Logistic regression uses a binary variable called a “dichotomous dependent variable.” These dichotomous variables are normally zero or one. This method is used to explain the relationship between one dependent binary variable and one or more independent variables.

Unlike logistic regression, linear discriminate analysis can handle more than two-class classification problems. A discriminate value for each class can then be calculated and a prediction can be made for the class with the largest value. Classification or regression trees are a method of decision trees, which are basic flowcharts driven by binary answers to specific questions. This method is more simplistic for programming and the initial configuration of the predictive analysis.

The remaining models are more capable and have several advantages over the simpler models. Bayesian network models use two forms of probability, one driven from a current history data set and another, a conditional probability, driven from assumptions. The Bayesian method assumes that all the inputs are independent of each other. Another method is the *K*-nearest neighbor method, which can be used for either regression or for classification. However, it is normally used for classification. One benefit of the *K*-nearest neighbor method is that it does not make any assumptions about the data. However, all the training data must be stored. With the learning vector quantization

method, the training data does not have to be saved, since it is an artificial neural network algorithm (Brownlee, 2016). This model is a small codebook of vectors that are constructed from the training data, which is used to make a classification to predict data. Support vector machines deploy a method used to transform a problem using linear algebra to simplify the results. This simplification is done by looking at the inner product between the points for classification. The random forest method is derived from a learning algorithm called "bootstrap aggregation" or "bagging." This method takes many samples and adds a random variable to improve the accuracy of the prediction. Boosting and AdaBoost are techniques that attempt to create a strong classifier from a group of weak classifiers. This is done by creating shorter decision trees and viewing their performance and weights to create stronger classes. The following sections review common methods used for detecting occupancy.

2.3.3.1 Bootstrap aggregation. **"**Random forest" and "bagging" are alternative terms for the ensemble machine learning algorithm called "bootstrap aggregation." This method can be used in various applications. One of these applications is energy management for building occupancy. Few studies use this method to identify whether there is an occupant within a certain place, for example, in a building for the scheduling of operations such as work on HVAC systems. There are many advantages of bootstrap aggregation due to the ensemble prediction model. The ensemble prediction model is defined as a "set of individually-trained based models … whose outputs are combined to make a prediction" (Wang, Wang, & Srinivasan, 2017, p. 110). Similarly, the random forest method is created by introducing a component of randomness into the bagging or classification tree. Classification tree is explained as follows: "each tree is constructed

using a different bootstrap sample, and each node is divided based on a randomly selected set of predictors specific to that node" (Kontokosta & Tull, 2017, p. 307). Bagging is a process of sequentially combining the weak learners to reduce prediction errors by voting over results from a substantial number of bootstrap data points (Wu et al., 2018). Once these weak learners are defined and the random parameters are tested, the results become good predictors.

There have been several ways that bootstrapping has been used to solve problems. Wang, Wang, and Srinivasan worked with 11 input features, including occupancy and temporally related data collected hourly (Wang et al., 2018). This study by Wu et al. used environmental, microclimatic, and demographic parameters, including building type, degree of thermal environment control, and other values recorded from a field study for their inputs into the bagging model. Once the data sources were determined, each research group created a plan around the methodology they desired. This methodology exhibited the relationship between the different inputs such as the building, floor area, number of floors, presence of an attached lot and year built, and so on.

After the data were collected, Wang et al. then partitioned the data to help organize their logic behind the data for the bootstrap method. This required the bootstrap be trained through a process in which the common threads were created to be used for predictions. This was done by reviewing the data from the location, the local weather, the class schedules, and the building operation to check for any visible patterns that needed to be considered (Wang et al., 2018). Once the patterns were reviewed, they created three modules to be trained and tested independently for the data set. At the end of the study, all the modules had high prediction of accuracy, with average mean absolute percentage

of errors (MAPEs) of 2.97%, 4.62%, and 4.63% of prediction. One of the major limitations of this method was the amount of time it took to complete the predictions. This time requirement was due to the training required over the iterations necessary to derive accurate predictions (Wang et al., 2018).

Kontokosta and Tull used a similar method to train their models. However, in their case, they trained the model to the point that the predictions matched the historical data before trying to predict the future data. Their use of a wider dataset enabled this method of training. Training was conducted on a few regions within New York City. The information was used to predict other regions within the city data set, as well as across different categories within the same set. Kontokosta and Tull found their results could successfully predict the utility usage by utilizing the PLUTO data. They did not address the amount of time calculations took, since they were not trying to find live data. With the successful prediction of bootstrap aggregation, a large amount of training is required for each data set used. This extra training requires more computing and a more hands-on approach than would be needed for live-occupancy detection.

These several methods show how data modeling can be used to predict certain outcomes. This dissertation uses similar methods to model occupancy data collected from a previous timeframe and applies a method to better enhance occupancy predictions.

2.3.3.2 AdaBoost model. Similar to the Bootstrap algorithm, the AdaBoost model still uses classification of selected weak features to build a cascade-structured detector (Wu & Nevatia, 2005). AdaBoost uses an algorithm to learn a set of classifiers also known as “weak learners.” Weak learners are obtained sequentially, using re-weighted versions of the training data, with the weights depending on the accuracy of the previous

classifiers. These weak leaners are then ordered to produce the final stronger classifier (Vafeiadis et al., 2017, p. 3). Training is normally done twice to build the weaker learner, and from there the algorithm can support prediction. Once there is an understanding of what the prediction node requirements are for AdaBoost, data can then be collected for specific applications.

Vafeiadis et al. have presented a different method of gathering information. They used three different systems to gather raw information from energy consumption, water consumption, and occupancy in specific locations for one month. This data was recorded using sensors for all three systems at one-minute intervals to measure the variables for their study. Rather than using sensors and data to detect building occupancy, as did Vafeiadis et al., Wu and Nevatia used camera detection to detect occupants (Wu & Nevatia, 2005). The basic principles used by the two research groups are identical. Wu and Nevatia constructed training sample sets with photos, not numbers. They use a process like data training with a dataset. From there, Wu and Nevatia ran the training method for detecting occupancy, and this result revealed an accuracy detection occupancy rate in the range of 80% to 91.2% with the images used compared with occupants within the space (Wu & Nevatia, 2005). Vafeiadis et al. had a result of 83.2% accuracy in detection of occupancy from the modeling that used trained data compared to the occupants' ground-truth.

Similar to the bootstrap algorithm, the AdaBoost model still has computing requirements, though they are significantly lower. Results are accurate when predicting outcome and sorting information using weaker learners if real-time performance is not required. An occupancy system requires real-time triggers, these systems may not be able

to process the information that is required in time or may require higher computing capabilities to get desired results.

2.3.3.3 Bayesian model. Bayesian method uses conditional probability for independent variables. The goal of using the Bayesian model is to determine the probability of each cell to be occupied and the state variable associated with the given variables (Ribo & Pinz, 2001). A cell is explained as a location defined within a given criteria with known previous states. In this application, a cell would be a space that has a previously known occupancy probability. This makes the Bayesian model useful when there are no strong dependent and independent relationships between the variables. As stated by Yeonsook Heo, “A Bayesian approach is used for calibrating uncertain parameters in the normative model and quantifying uncertainties in the parameters” (Heo, 2011, p. 7). Another strength when using the Bayesian method for predictions is that it allows for the component forecasting models to come from any trained forecaster with well-defined distribution of the forecaster’s mis-forecasts (Howard & Hoff, 2013). As such, one can use various history or training backgrounds for each segment or node of the Bayesian network.

In summary to section 2.3.3, the graphical comparison between the three analytical approaches can be seen in Figure 6. Each of the three methods have their advantages, but due to the Bayesian model’s usefulness when there are no strong dependent or independent relationships this dissertation will be using the Bayesian model to determine occupancy.

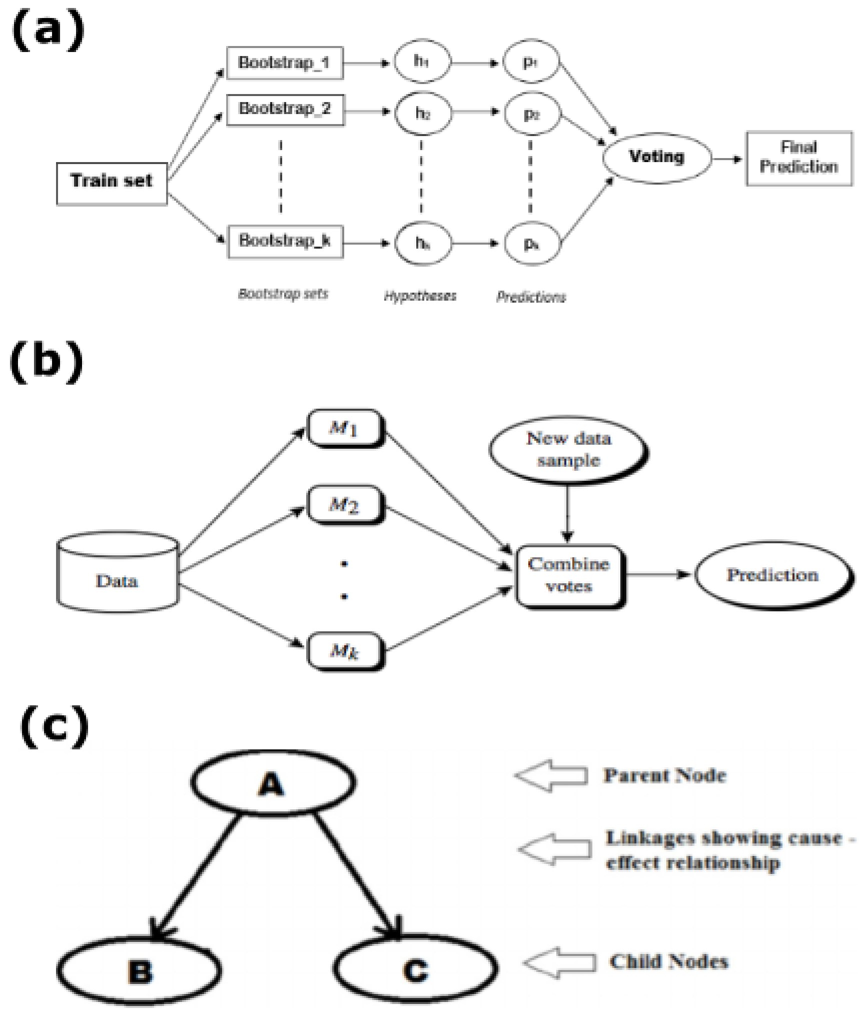

Figure 6: Comparison of the models from the analytical approaches. (a)Bootstrap Aggregation (Khan et al., 2019); (b)AdaBoost Model (Corporate Finance Institute, n.d.); (c)Bayesian Model (McKee & Anddriyas, 2015).

2.4 Bayesian Model Basics

Named after Thomas Bayes, who studied binomial distribution, Bayes' model has been more commonly used in probability and risk assessments due to the increasing ease of computer calculations. The theorem was not published until Richard Price published it as the Bayesian network method. Section 2.4 addresses how Bayesian networks can be used to minimize risk of error in security system installations for detecting home occupancy. It focuses on the basic background information required to understand

Bayesian networks and of the model for calculations and predictions, as well as the methods and terms necessary to express the relationship between each of the nodes with their corresponding relationships.

Section 2.4.1 describes the foundation of Bayesian networks and the ability to use conditional probability with assumptions gathered from previous results. Nodes are used to express the probability of certain outcomes which is explained in Section 2.4.2. Section 2.4.3 discusses the relationship between the nodes and arcs. Research examples of Bayesian modeling can be found in Section 2.4.4. The conclusion in Section 2.4.5 will summarize the basic principles of Bayesian modeling.

2.4.1 Conditional Probability

Fenton and Neil explain that at the heart of the Bayesian approach is the role of conditional probability (Fenton & Neil, Risk Assessment and Decision Analysis with Bayesian Networks, 2013). Conditional probability is defined as interest in a given event after observing a different related event. An event is defined as "some unknown entity," where probability is used to quantify uncertainty about that event. Some examples of events are as follows: 1) the home will be occupied, 2) there is motion in the home, and 3) the next flip of a coin will be tails. Conditional probability can be applied to decide what the chances are that a space is occupied due to the monitoring detected. Conditional probability statements can help determine whether detected motion is due to curtains moving or the room being occupied. If the curtain moves, there is a lower probability that the living room is occupied.

- Event: Motion was detected.
- Event: Curtain is moving.

- Event: Someone is home.
- Reasoning 1: P(motion was detected | curtain is moving) = room is unoccupied. Given that motion was detected, and the curtain is moving, there is a low probability there is an occupant.
- Reasoning 2: P(motion was detected | curtain is not moving) = room is occupied. Given that motion was detected and there is no curtain moving, there is a higher probability there is an occupant.

This is the base principle of conditional probability in Bayes' modeling. The known linked (arc) events (nodes) are used to predict the probability of other unknown events, referred to as "determining the belief" of those events (Charniak, 1991).

2.4.2 Node

There are two types of nodes that make up a Bayesian network: root nodes and non-root nodes. Root nodes have no parent nodes. From Figure 7, "motion was detected" and "curtain is moving" are both root nodes. Root nodes must use prior probabilities. Non-root nodes have parent nodes. From Figure 7, "there is an occupant" is a non-root node. Non-root nodes are given all possible combinations of their direct parents to create their conditional probability tables, as shown in the following example:

- "Motion was detected" and "curtain is moving" are root nodes (i.e. parent nodes).
- "Someone is home" is a non-root node i.e. a child node to "motion was detected" and "curtain is moving."
- Prior probabilities for all root nodes and all the conditional probabilities for the non-root nodes are required.

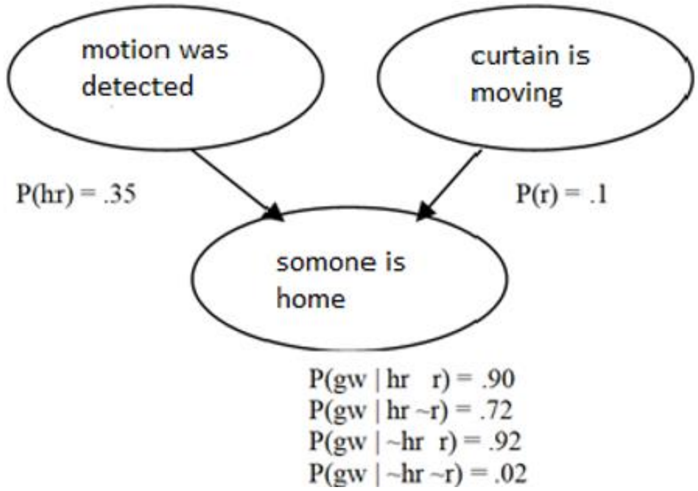

Figure 7: Example: Node – is it occupied?

Figure 7 shows a 35% chance that motion was detected and a 10% chance it is due to the curtains moving; these are the prior probabilities. Prior probabilities are normally calculated from historical data. The following statements present conditional probabilities (normally an expert opinion, subjective data, or instance information): 1) if there is motion and the curtains are moving, there is a 90% chance that there is no occupant; 2) if motion is detected and no curtains are moving, there is a 72% chance there is an occupant; 3) if there is no motion detected and the curtains are moving, there is a 92% chance there is no occupant; and 4) if there is no motion detected and no curtains are moving, then there is only a 2% chance there is an occupant. This method requires that a conditional probability be created for all the conditions present from the parent nodes.

Using the nodes, the arcs, and the conditional probability, one can construct a Bayesian network to create beliefs that would not normally have been possible to calculate with only probability data and historical data. This possibility is the key benefit of Bayesian networks; one can develop a belief based on a wide range of events, which

can then be used to make expert assumptions concerning events or predictions in real time.

2.4.3 Arc

Charniak explains that arcs in the "Bayesian network specify the independent assumptions that hold between the random variables" (Charniak, 1991, p. 51). Fenton and Neil explain that an arc from node A to node B denotes a direct causal or influential dependence of node A on node B, with A being the parent of B (Fenton & Neil, Risk Assessment and Decision Analysis with Bayesian Networks, 2013). For example:

- There is an arc between "motion was detected" and "someone is home."
- There is an arc between "curtain is moving" and "someone is home."
- That means the parent nodes of "someone is home" are "motion was detected" and "curtain is moving."

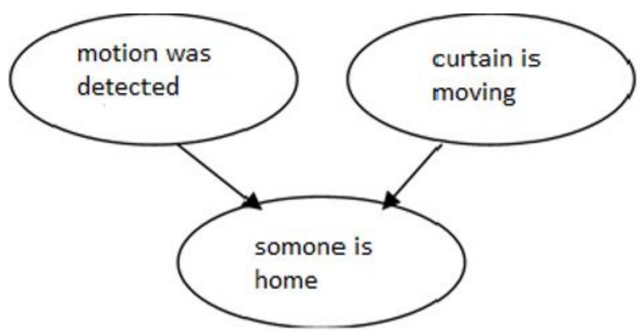

Figure 8: Example: Arc relationship – is it occupied? Arcs are indicated by arrows and point from the parent node to the child node.

The definition of an arc indicates a causal dependency of the "someone is home" when a "motion was detected" and when a "curtain is moving" on the event that "there is an occupant." Notably, no assumptions or arcs are made to determine whether motion was detected because the curtains are moving or rather motion was detected because someone is home. Arcs express the relationship within the Bayesian network's nodes and the educated assumptions or the known behaviors.

2.4.4 Research Examples

Bayesian networks have been used in the development of artificial intelligence and predictive risk analysis. Weber et al. have reviewed a collection of over 200 articles describing the application of Bayesian models that directly relate to the dependability, risk analysis, and maintenance of the model (Weber, Medina-Oliva, Smon, & Benoît, 2012). Within those 200 articles, 61% were referenced as dependability analysis and 26% were referenced as risk analysis. The aim of dependability analysis is to provide a prediction of a parameter (remaining time to fail, mean time to fail, reliability, etc.) that is an input to the data for the decision step (Weber et al., 2012). Risk analysis identifies, characterizes, quantifies, and evaluates critical event occurrences. Tijani et al. present a case in which they used Bayesian network to simulate occupant behaviors in office buildings to "determine the belief" in air quality.

Tijani et al.'s model considers the following: 1) permanent calendar, 2) intermittent calendar, 3) professor calendar, 4) guest calendar, and 5) CO_2 concentration. After the five root nodes were defined, they determined which events would be used as nodes to help create a belief from defined assumptions. Their certain known assumptions were determined by a video feed located in a professor's office for a determined amount of time, which was combined with "expert knowledge" to generate the dynamic Bayesian network (Tijani, Ngo, Ploix, Haas, & Dugdale, 2016). This network was used to predict the CO_2 level in the office from the root nodes and the other node states. Inputs such as calendar events help predict occupancy. This research used past historical information to build a template of predictions for the behaviors of the occupants. Using historical

information is fundamental to analytical methods. The information is used to build the historical probability for the methods.

Mulia et al. used building occupancy to assess Bayesian networks and other methods to compare the accuracy of several algorithm approaches, such as moving averages, decision trees, conditional random fields, and random forests. They reviewed a range of research for occupancy with a range of data collected from passive infrared sensors, pressure sensors, CO_2 sensors, and depth image cameras. Among the many methods reviewed, the Bayesian approached reached one of the highest rates of accuracy at 82%, as compared to other analytical methods such as moving average, learning machines, and the Markov models (Mulia, Supangkat, & Hariyanto, 2017). Mulia et al. concluded that algorithm learning methods and artificial intelligence were most accurate when processing data using the Bayesian network. Both the learning methods and artificial intelligence used the Bayesian network to determine information about events as they were happening. The Bayesian network is simple to implement because the data-driven estimation strategy does not require prior knowledge. Any lack of information can be addressed through subjective judgment, a notion explained as an expression of a rational agent's beliefs about uncertain propositions. In this sense, a rational agent is generally considered a subject expert (Fenton & Neil, Risk Assessment and Decision Analysis with Bayesian Networks, 2013).

2.5 Conclusion

The added need for more accurate occupancy detection requires more advancement in detection methods. As discussed in this chapter, there are ways other researchers have improved the accuracy of detections which were: 1) detectors, 2) circuit

design, and 3) analytical approaches. Analytical approach is the common method of improving the occupancy detection. In the research that was reviewed, Bayesian method showed to be increasingly common practice when being implemented into systems due to the simplicity of applying the required application logic. Two gaps that were discovered within the literature review were: 1) that the use of multiple sensors were not very common within the research, the researchers relied on each sensor in a series configuration; and 2) the researchers used historical information over a relatively short period of time. This dissertation will bridge these two common gaps by applying sensors in a parallel configuration and use data over a one-month trial.

CHAPTER 3 METHODOLOGY

This chapter defines how the dissertation method is implemented within this paper. The independent, multiple-parallel sensor approach is described in this chapter, along with how the Bayesian modeling will be applied to the multiplied-parallel sensors approach.

Section 3.1 discusses the flow of the research project. Section 3.2 defines the project layout and Section 3.3 demonstrates the data collection process as it is defined. Section 3.4 expresses the requirements for cleaning the data that were collected from the sensors. Section 3.5 will review the three distinct methods; 1) Independent sensor method; 2) Grouped sensor method; and 3) trained Bayesian modeling, that are used.

3.1 Research Flowchart

This dissertation investigates the application of Bayesian modeling to a group of PIR sensors to improve the accuracy of the overall detection system, allowing for a more accurate determination of ‘presence’ within a given area. Figure 9 outlines the conceptual flowchart for this project.

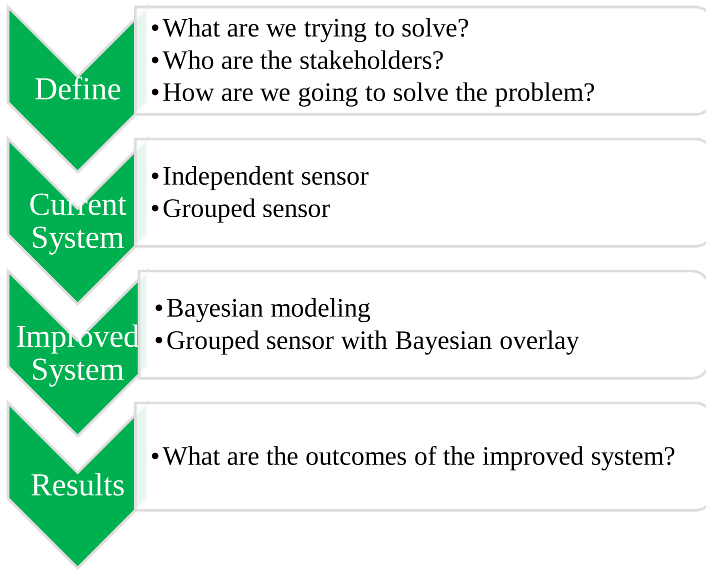

Figure 9: Research flow chart for the design of this project and the testing for Bayesian method implementation.

3.2 Defining the Project

A Bayesian network analysis paradigm using multiple sensors is proposed to help prevent many common false positives in occupancy-detection systems, ultimately improving the accuracy of occupancy detection compared to traditional detection systems and methods.

Data were collected from three locations, as follows: The first location was a condominium with two occupants residing in the home. The condominium had a total of six PIR sensors within the space, comprising three in the living room and three within the bedroom. Figure 10 shows sensor placements within the condominium.

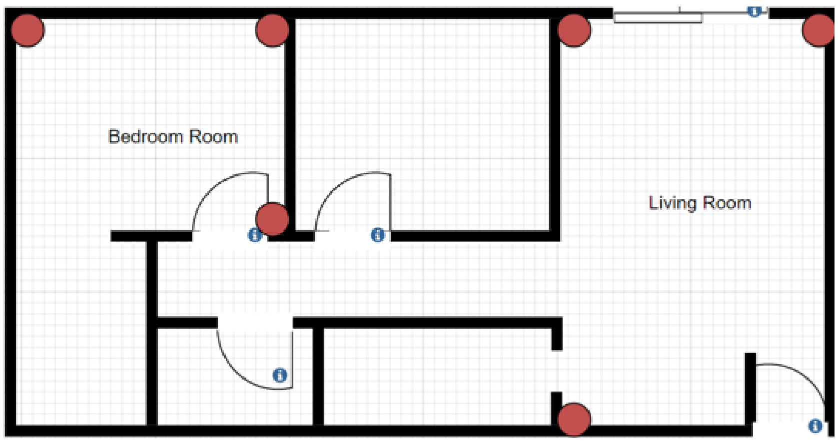

Figure 10: PIR sensor placement for the condominium.

The bedroom and living rooms were chosen because they are the rooms with the most activity or events throughout the day and evening. Data were collected from the location for four weeks.

The second location was a detached house with one occupant. Data were collected from the location over two separate time periods, once in 2018 and another in 2019. Figure 11 shows the sensor locations and placements in 2018. Data were collected at this location for four weeks.

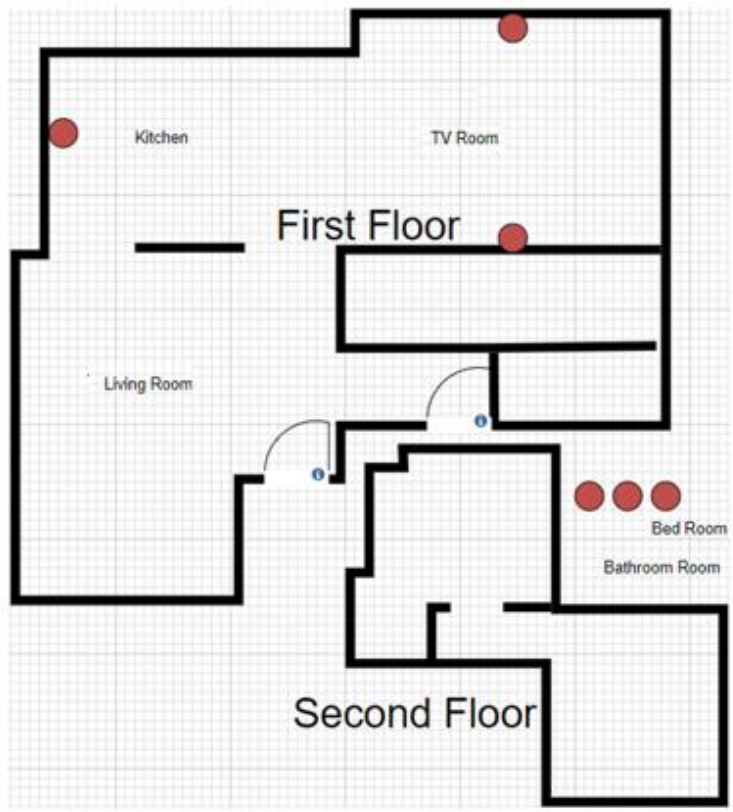

Figure 11: PIR sensor placement for the one occupant in 2018.

More data were collected at this same location for a four-week period in 2019. Different sensor placements were selected, as depicted in Figure 12.

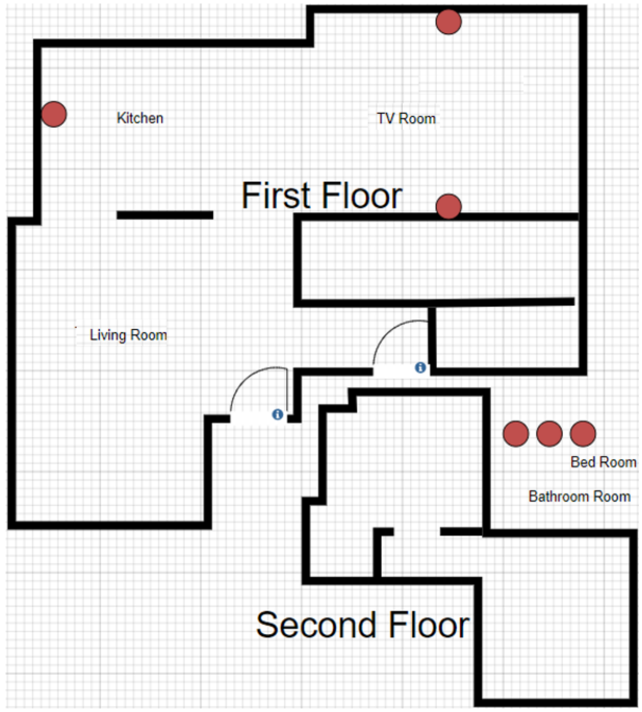

Figure 12: PIR sensor placement for the one occupant in 2019.

The third location had residents that were home for longer periods during the day. Two retirees lived there. Three of the six PIR sensors were placed in the living room, the other three in the bedroom. Data were collected from the location in 2019 over a four-week period.

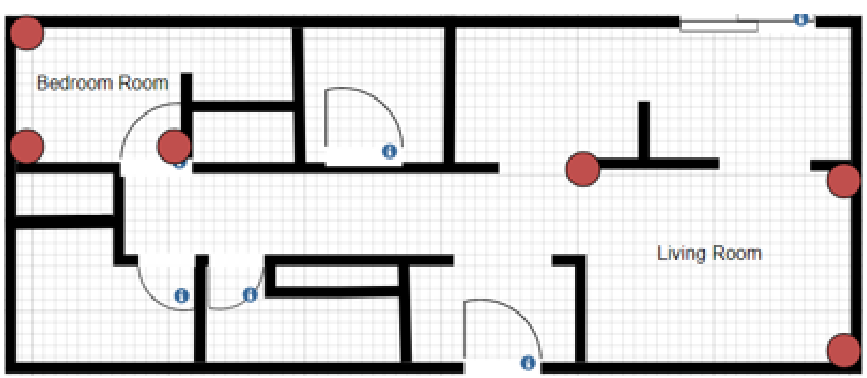

Figure 13: PIR sensor placement for the two retirees' in 2019.

With the six PIR sensors placed in the designated locations, the data for motion events were then collected. Once the data were collected, analysis methods were applied which were then used to predict the occupancy within the selected homes.

The raw sensor data collected at each location were complemented by ground-truth data gathered from a smartphone GPS signal, and a Bluetooth key fob. The sensors and workflow are described in the next section.

3.3 Data Collection

This section describes the data collection method, explaining where the data was collected from, and how the data was collected. Section 3.3.1 discusses the method on how the sensor data was collected. Section 3.3.2 details how the information from the sensors was stored and Section 3.3.3 explains how the stored data was exported for the project.

3.3.1 Data sensor collection

The design of this project utilizes several PIR sensors, global positioning system GPS-based or Bluetooth-based presence detectors, and a central hub that was used to gather the information and read the sensor states. The PIR sensors, smartphone GPS signal, and Bluetooth key fob presence detectors sent their state changes back to the central hub. Every five minutes, these data were uploaded into Google Drive for long-term storage. Each of these links are explained in this section. Figure 14 shows the relationship between the devices.

Figure 14: Occupancy detector collection process and device relationship flow.

The top layer of the data collection included the PIR sensors and cellular phones that acted as the two devices used to measure ground truth. The PIR sensors, BOSCH Zigbee PIR ISW-ZPR1-WP13 (BOSCH, n.d.) and the Samsung SmartThings F-IRM-US-2, 3305-S (SmartThings, n.d.) were the sensor models used in this project. These sensors are inexpensive and reliable, and they offer a wireless ZigBee communication protocol. The BOSCH PIR and the Samsung PIR sensors have a 90° angle of operation to allow the sensors to pick up motion within the space. They include eight detection patterns, allowing for detailed movement monitoring. Three sensors were placed in each area monitored.

The second group of sensors in the top layer of the data collection, as depicted in Figure 14, were the cellular phones acting as the smartphone GPS signal presence and the Bluetooth key fob presence detectors. The GPS provided ground-truth through a smart

phone application. For this project, the radius was 517 feet. The research set the center point of the GPS radius as the center point of the location, as depicted in Figure 15.

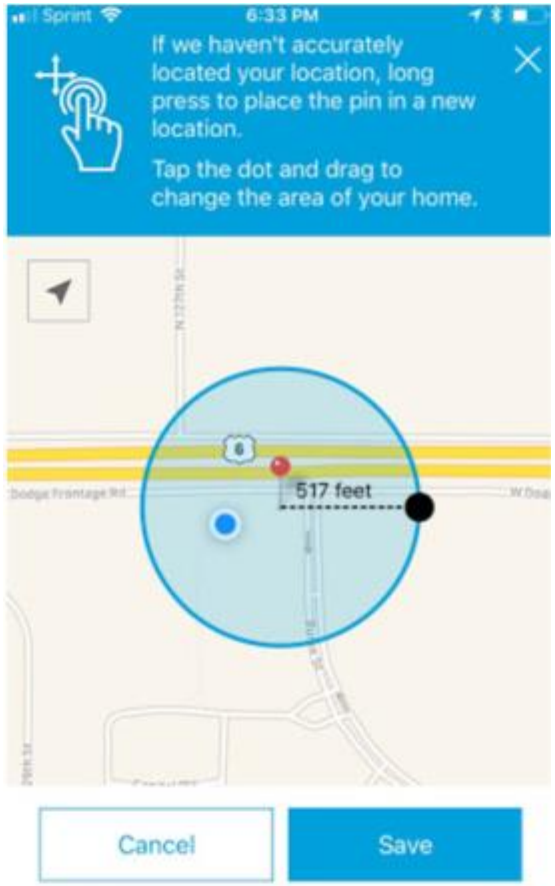

Figure 15: SmartThings application location and presence area.

Once the phone has entered the given space, the device then shows the presence of an occupant. The presence establishes the ground-truth. Once the phone is outside of the given area, the phone state then shows the occupant to be away. The Bluetooth key fob detector provides ground-truth by a similar method. When it is detected within the range of the SmartThings hub, it registers the occupant as present. When the presence detector is outside of the range, it shows the occupant as not present.

All the sensors within the top layer report to the second layer, the Samsung SmartThings hub. The SmartThings hub was selected for several key features. This

device can connect wirelessly to a wide range of sensors. With the intent of commercial application, the device allows hardware monitoring and control of connected devices in your home using a single smartphone application. The SmartThings hub device facilitated the collection and management of PIR sensors and other sensor data (SmartThings, n.d.).

3.3.2 Information storage

Signals from individual devices were stored at the SmartThings hub using the application Simple Event Logger. Simple Event Logger, designed by Kevin LaFramboise, captures event logs within the SmartThings hub and stores them in a Google Sheets spreadsheet (LaFramboise, n.d.). LaFramboise recommends using the application to accurately log all device activity. Each event is stored on a separate row to show the time and the details of the events. The logger records the event time, device, event name, event value, and event description.

Since the Simple Event Logger can be easily configured, it creates a data source when working with the information and the configuration parameters selected for this project. Configurations are listed as follows: 1) motion sensors selected, in this case six PIR sensors; 2) presence sensors selected, in this case two cellular phones acting as a smartphone GPS signal presence detector; 3) events logged, in this case activity, motion, and presence; 4) logging options, in this case event logging every 5 minutes; 5) maximum catch-up interval, in this case up to one hour; 6) maximum number of events to log for each device per execution set, in this case up to 200; 7) log event description on or off, in this case on; 8) use value and unit for description on or off, in this case on; 9) additional columns for short date and hour on or off, in this case on; 10) delete extra

columns turned on or off, in this case on; and 11) set archive type, in this case set as events.

These specific configurations allow the Simple Event Logger to collect the desired data from the occupants. The above configurations allow each of the six PIR sensors and the cellular phones to create a log entry in the Simple Event Logger. Once the events are collected, the event logger can then pass the logs onto Google Sheets every five minutes, where they can then be exported, as was required for the subsequent step in the present study. The spreadsheet was exported for use within the algorithm and cleaned for errors that might have occurred during the logging process, as explained in Section 3.3.3.

3.3.3 Information export

Once the sensor states have been collected within Google Sheets Spreadsheets, information can be exported. Data can be exported from Google Sheets in two main ways: through the export function within Google Sheets Spreadsheets or through the code used to calculate instant predictions from the sensor states. These two methods are detailed below.

Google Sheets has a built-in export feature that was used for exporting the sensor state data from online to a local computer for review. Assume one has an active account with Google, where the document is being stored. The first step in exporting the information from Google Sheets is to log into Google Sheets. Once a user is logged in, the sheet where the Simple Event Logger transfers and stores data can be selected. When the desired sheet is selected, the document can be opened, and the “File” menu can be selected. The file can then be downloaded (to export the data in the desired format). The

researchers cleaned the information the data exported as a Microsoft Excel (.xlsx) document.

3.4 Data Cleaning

Data cleaning is vital for data sets, as it is used to exclude information that the system does not know how to process. For example, when utilizing a learning algorithm from previous occupancy information, excess data recorded are more susceptible to errors, due to undefined data. The undefined data that are collected can include temperature, or any other logged entries created for testing. Such extra information can cause errors within the learning algorithm. The undefined data must therefore be removed from the dataset. In this case, another example of cleaning information is that the battery percentage data were removed from the dataset, since these data are not vital to detecting motion but were used for debugging to ensure the battery levels were not contributing to false signals. Other events and their subgroups removed from the data included temperature-related information. The temperature event was removed because it has no effect on occupancy within the given space. Transaction data were also removed since they have no significant effect on the event and provides only a health check for when the device is connected to the SmartThings hub.

Only recorded pulses from the PIR sensors and signals from the smartphone presence detectors and key fob presence detector were retained. The events and subgroups not deleted were the motion events, including the subgroups of “active” and “inactive.” The “active” status, signaling occupancy, indicates a motion event was detected from the PIR sensor, while “inactive” status means the device does not detect a motion event.

After the data were cleaned, they were then uploaded in Google Sheets so that the code could pull them when applying the Bayesian model. This step completed the integration of the analysis and the hardware into a system to improve the detection of presence.

3.5 Analysis Methods

This dissertation reviews three methods that are used to determine occupancy detection. Section 3.5.1 discusses the independent method and how it is applied to the research. Section 3.5.2 demonstrates the grouped sensor method and Section 3.5.3 describes how the Bayesian modeling is applied.

3.5.1 Independent sensors

The first analysis method applied uses data from only the PIR sensors. This method relies on each of the six sensors in each of the locations to trigger an event independently. Even if a single sensor triggers due to a movement, the system will assume there is an occupant within the space. There is no redundancy or verification within this independent method. This method was implemented in all the test locations. The equation pseudocode for the independent sensor method is as follows:

If (LivingSensor1 OR LivingSensor2 OR LivingSensor3 OR BedroomSensor1 OR BedroomSensor2 OR BedroomSensor3 = Motion Detected)

Then (there is occupancy)

Else (there is no occupancy)

For an example from the condominium location in Table 2, observe line one. There were no PIR sensors triggered, so the results showed there was no occupant within the location. From the second line, there was one motion event trigger, which is the

minimum requirement for a location to register an occupant, and so occupancy is registered.

Table 2: Condominium analysis method for independent method example

	Date/Time	LivingRoom Sensor1	LivingRoom Sensor2	LivingRoom Sensor3	Bedroom Sensor1	Bedroom Sensor2	Bedroom Sensor3	Output
1	2018-09-27 20:16:00	0	0	0	0	0	0	0.0
2	2018-09-27 20:17:00	0	0	0	1	0	0	1.0
3	2018-09-27 20:18:00	1	1	1	0	1	0	1.0

The independent sensor arrangement shows the degree of accuracy for each of the sensors, acting on their own. The accuracy was calculated using data collected over a typical month and viewing a specific week within that same month. A typical month was defined as each of the four location's households behaving according to their normal routines of going to work and coming back home to their house. Each of the four locations had sensors located in the high-traffic areas to best detect any motion that may occur.

The independent sensor arrangement is used in home automation and detection systems. It incorporates a motion-detection device that engages an event when triggered. For example, when there is motion in a room, a light may turn on, then turn off when motion is no longer detected. The theoretical segment of ϕ correlation equation represents the presence data for each of the time segments. The ϕ correlation coefficient is expressed by taking into account both the correct and incorrect results to compare how close the occupancy measured by each method and its sensors were to the event time of each space (Guo, 2007). The researchers calculated ϕ correlation following Guo for each sensor and node at the one-minute level over the length of the four-week study. The PIR sensor data

were resolved to a one-minute interval and compared with the smartphone and key fob ground-truth, using ϕ correlation. This resolution can be seen in Table 3.

Table 3: Correlation coefficient cross-table format (Guo, 2007).

Truth \ Measured	1	0	
1	N_{11}	N_{10}	$r_1 = N_{11} + N_{10}$
0	N_{01}	N_{00}	$r_2 = N_{01} + N_{00}$
	$c_1 = N_{11} + N_{10}$	$c_2 = N_{10} + N_{00}$	

$$\varphi = \frac{N_{11}N_{00} - N_{10}N_{01}}{\sqrt{r_1 r_2 c_1 c_2}} \qquad (3)$$

For each of the six-independent sensors at each of the four locations, the ϕ correlation was calculated as against the ground-truth. These six correlations were then averaged together to calculate the error for each of the one-minute time slots. An example of the correlation method described would be that if there were 100 time slots of one minute recorded, the PIR sensor data would show that 25 of the 100 time slots measured motion. Out of those same 100 time slots, the ground-truth showed there were 50 time slots that had an occupant. Of those 50 time slots, 20 had correct ground-truth and PIR sensor detection. The ϕ correlation calculations variables are $N_{11} = 20$, $N_{10} = 30$, $N_{01} = 5$, and $N_{00} = 45$.

$$\Phi = \frac{(20*45) - (30*5)}{\sqrt{50}*50*25*75} \qquad (4)$$

$\Phi = 0.4087$

3.5.2 Grouped sensors

This section discusses the principle of implementation of the grouped method with the six sensors at each of the four locations. Unlike the independent sensor method, the grouped method requires that two or more of the six sensors must be triggered to initiate an event. This method is like the “parallel” sensor configuration described earlier. At least two sensors must register a movement for the system to assume there is an occupant within the space. If only one sensor pulses, then that single sensor event is ignored. This requirement adds a layer of redundancy and verification. The equation pseudocode for the grouped sensor method is as follows:

If (LivingSensor1 + LivingSensor2 + LivingSensor3 > 1) OR

(BedroomSensor1 + BedroomSensor2 + BedroomSensor3 > 1)

Then (there is occupancy)

Else (there is no occupancy)

For an example from the condominium location in Table 4, observe line one. There were two PIR sensors triggered, so the results showed there was an occupant within the location. From the third line, there was only one motion event trigger, which does not meet the minimum requirement for a location to register an occupant hence the result is no occupant.

Table 4: Condominium grouped sensor example

	Date/Time	LivingRoom Sensor1	LivingRoom Sensor2	LivingRoom Sensor3	Bedroom Sensor1	Bedroom Sensor1	Bedroom Sensor1	Output
1	2018-09-25 19:38:00	0	1	1	0	0	0	1.0
2	2018-09-25 19:39:00	0	0	0	0	0	0	0.0
3	2018-09-25 19:40:00	1	0	0	0	0	0	0.0

The motivation for grouping sensors is that one PIR sensor is not accurate enough to detect motion. Detection errors are minimized by ensuring more than one PIR sensor has pulsed before the system registers an event.

The grouped sensors method is used to improve the accuracy of event data. There are several options for grouped sensors that can be implemented. Grouped methods incorporate several motion devices that must be triggered before an event is registered. The network must contain at least two sensors in the same room space that trigger before an event is registered. For example, LivingRoomSensor1 and LivingRoomSensor2 must detect motion within the same one-minute time slot before an occupant is considered to be in the room. If only one sensor detected a motion event, the logic would consider the room unoccupied. Only one of the rooms needed to detect motion for this logic to consider there was an occupant within the entire space or to have an occupant in either of the four locations.

Multiple room assumptions were taken into account to conclude that if any motion triggering event (i.e., if two or more sensors detect motion) occurred within the given space, an occupant was assumed present in that space. If “MotionLivingGroup”

detected a motion event and "MotionBedroomGroup" did not detect the same event, an occupant was still registered within the given location.

3.5.3 Bayesian modeling

This section demonstrates the Bayesian modeling implemented for the data from the six sensors at each of the four locations. This method is the principle of applying a past probability of occupancy to generate a prediction that the location is occupied. The Bayesian modeling used two historical data sets that were collected. The first historical data set is the PIR sensor signals. This historical data set is a collection of all motion pulses that were collected within each of the four locations. This collection allows the calculation of the prior probability that was used in the "trained PIR sensor hours". The second data set is the collection of the smartphone GPS signal and the Bluetooth key fob location occupancy data. This set was the collection of occupancy within each of the four same locations. The smartphone GPS signal and Bluetooth key fob history was used to calculate the prior probability of the occupancy, this is referred to as the "trained ground-truth".

Using each of the two historical data sets, the prior probabilities for the Bayesian modeling was calculated over a four-week period. This four-week period is called "training" since the model is learning from the historical data points. The four-week training period for the four locations are: 1) the timeframe for the condominium from 09/19/2019 through 10/18/2019; 2) the house with one occupant (2018), from 03/03/2018 through 04/03/2018; 3) the house with two retirees, from 05/05/2019 through 06/05/2019; and 4) the house with one occupant (2019) from 06/12/2019 through 07/02/2019.

Training took each of these four-weeks' worth of historical information and categorized the events into one-minute time slots. Each of these minute time slots accounted for the day of the week. That means the training accounted for each 1,400 minutes of each day of the week. Once the training categorized the events, it then could calculate the probability that an event occurred for that minute time slot for each day. The probability was calculated based off how many times an event occurred within the specific minute and day.

For example, at the condominium on Thursday 09/19/2019 between 12:00:00 – 12:00:59 and Thursday 10/03/2019 between 12:00:00 – 12:00:59 there were PIR sensor pulses, but the other two Thursdays 09/26/2019 and 10/10/2019 between 12:00:00 – 12:00:59 there were none. Two out of the four Thursdays' minute time slots between 12:00:00 – 12:00:59 had a pulse so the trained data would have resulted in a prior probability of 0.5, or half of the time there would be a pulse generated during the specific time slot. This same training using the days of the week and minute time slots were used for each of the six PIR sensors and the smartphone GPS signals and Bluetooth key fobs to generate their corresponding prior probability for each time slot.

CHAPTER 4: BAYESIAN MODELING FOR OCCUPANTS

This chapter explains the Bayesian modeling method to be used to detect occupants within each of their locations using the sensor data collected at each home. The same Bayesian network was applied to the specific datasets from each location. This chapter defines the network that will interpret the sensors' events and convert the results into a percentage of probability that the occupant will be home. The purpose of this chapter is to provide more project orientated explanation of how the Bayesian network was designed and how each of the network nodes function.

Section 4.1 identifies each of the seven steps that were used to develop the Bayesian network. Section 4.2 presents each of the nodes that are in the network and their corresponding conditional or discrete truth tables. Section 4.3 provides a walk-through example of the Bayesian network that is used in this dissertation.

4.1 Bayesian Modeling Design

Fenton and Neil have explained that at the heart of the Bayesian approach is the role of conditional probability (Fenton & Neil, Risk Assessment and Decision Analysis with Bayesian Networks, 2013). Conditional probability is explained as the likelihood of a given event after a different event is observed. An event is defined as "some unknown entity" where probability is used to quantify uncertainty about that event. The uncertain event in this study is the determination of occupancy within a location. The principle used to construct the conditional probability is historical data collected in the location and gathered using the six PIR sensors, the smartphone GPS signal, and the Bluetooth key fob. The conditional probability is inserted into the Bayesian network and then used to generate a prediction of whether some space is occupied, given real time sensor inputs

and the occupancy timeframe. The breakdown of each of the main parts used to apply the Bayesian method is discussed in the following sections.

Fenton and Neil (Fenton & Neil, Risk Assessment and Decision Analysis with Bayesian Networks, 2013) have defined seven main steps in building a Bayesian network model. These seven steps were used as guidelines to construct the model for this work: 1) identify the set of variables relevant to the problem; 2) create a node corresponding to each of the variables identified; 3) identify the set of states for each variable; 4) specify the states for each node; 5) identify the variables that require direct links; 6) create the identified links; and 7) for each node in the Bayesian network, specify the node probability table.

The inputs that applied to the Bayesian network are motion sensor states, the specified room in the house, the time of day, and the state of the smartphone and Bluetooth fob. When the Bayesian model has processed the historical data (i.e. training from four weeks data), a prediction is generated for that time slot and is compared with the occupied state values inferred from the individual sensors, the grouped sensors, and the presence overlay. Section 4.1.2 describes the process.

4.1.1 Identify the set of variables that are relevant for the problem

The main variables used to determine occupancy are as follows: PIR sensor signal, GPS/Bluetooth signal, time slots, and location. The variable of motion is defined as a Boolean variable that is true when a PIR sensor pulses or false when no motion is detected within the given timeframe. The motion variable is parsed with the three motion sensors in the living room and the three motion sensors in the bedroom, giving a total of six variables required with motion. For each of the six sensors, only one pulse is required

for the detector to register motion. Ground-truth is determined via the smartphone GPS signal and Bluetooth key fob. Each is a Boolean variable, true when the device is present within the given area and false when the devices are away or not within the given space. Data for the two devices are compiled as a single variable, since the model is not concerned with who is home, but rather, whether someone is present.

Time is used as a label variable and as a labeled node. Fenton and Neil explain a node as a variable whose set of states is simply a set of labels (Fenton & Neil, Risk Assessment and Decision Analysis with Bayesian Networks, 2013). Time is a labeled node, since the model is not using time as a ranking or a percentage but as a slot of time during which the researchers look for positive events to happen. Each of the one-minute time slots were used over a one-week time slot and is generated as a variable for use in the probability table. The last variable is a label variable that tells in what room the event occurred. This information generates the creation of the conditional tables. The next step is to incorporate these variables into nodes.

4.1.2 Create a node corresponding to each of the variables identified

The nodes were created within a python script using the open-source Bayesian pomegranate library (Schreiber, 2018). Using the pomegranate library, the model can add each of the variables as corresponding nodes: LivingSensor1, LivingSensor2, LivingSensor3, BedroomSensor1, BedroomSensor2, BedroomSensor3, MotionLivingGroup, MotionBedroomGroup, and Ground-Truth.

In the nodes, the model added two extra groups that were not part of the original variables: “MotionLivingGroup” and “MotionBedroomGroup.” These groups are used to

help reduce the conditional probability table complexity by reducing the direct arc nodes to the final node.

4.1.3 Identify the set of states for each variable

These nodes are used throughout the Bayesian network modeling, allowing the predictions of each of the node states to be followed. The final state of occupancy is then predicted. Each of the states for the defined variables and nodes must be defined. The list of states is as follows:

- LivingSensor1 (0 – No motion detected, 1 – Motion was detected)
- LivingSensor2 (0 – No motion detected, 1 – Motion was detected)
- LivingSensor3 (0 – No motion detected, 1 – Motion was detected)
- BedroomSensor1 (0 – No motion detected, 1 – Motion was detected)
- BedroomSensor2 (0 – No motion detected, 1 – Motion was detected)
- BedroomSensor3 (0 – No motion detected, 1 – Motion was detected)
- MotionLivingGroup (0 – No motion detected, 1 – Motion was detected)
- MotionBedroomGroup (0 – No motion detected, 1 – Motion was detected)
- Ground-Truth (0 – Not present, 1 – Present)
- HomeHours (one-minute time slots over the one-week timeframe)

Each room required two PIR sensors to trigger a signal event. For this reason, only two states exist for each of the six PIR sensors, as well as for the "MotionLivingGroup" and "MotionBedroomGroup," since these are derived from the six PIR sensors split evenly between the rooms.

The "Ground-Truth" has the two states of "Not present" and "Present." These two states are from the GPS-enabled or Bluetooth-enabled devices that establish ground-truth.

If either sensor is located at home, then the state will be present, or if neither are home at the location, then the state will show not present. The last node in the list is the "HomeHours" node. This node is defined as a list node, since more than two states are used. Each state of the nodes uses a one-minute time increment over the selected week. The Bayesian predictions of the network are compared over the one-minute time slot throughout the whole selected week.

4.1.4 Specify the states for each node

In this step, the two states that are used with the nodes in the network are either discrete or continuous. The pomegranate library website (https://pomegranate.readthedocs.io/en/latest/index.html) describes two types of states: 1) a discrete distribution made up of characters and their probabilities, whose probabilities sum to 1.0; and 2) a conditional probability table that depends on values from at least one previous distribution and can have as many distributions as needed to encode for each node within the model.

Each of the six sensor nodes are defined as a discrete variable since they are not driven by any other nodes and are independent of each other. The "MotionLivingGroup" and the "MotionBedroomGroup" are created as conditional nodes. These two node results are generated by sensor signal nodes for their particular room. The Ground-Truth node is conditional since it is from the conditions of the HomeHours nodes. The HomeHours nodes are discrete since they are used to influence the other nodes within the network. These network links are identified in the next section.

4.1.5 Identify the variables that require direct links

Direct links are arcs in which each of the nodes reacts. The links connect with each of the sensors in the rooms to drive the conditional tables for their corresponding rooms. The Ground-Truth and HomeHours decide whether an occupant is in the room groups within that timeframe.

Main links are used to help build the conditional tables for each of the conditional nodes, as shown in Figure 16. For example, either LivingSensor1, LivingSensor2, or LivingSensor3 must be triggered for the "MotionLivingGroup" node state to change. State changes depend on the node directly linked to their previous nodes. Since BedroomSensor1, BedroomSensor2, and BedroomSensor3 are not directly linked to the "MotionLivingGroup" node, there is no interaction between those nodes, so there is no state change.

4.1.6 Create the identified links

The direct links are arcs and are shown in the Figure 16.

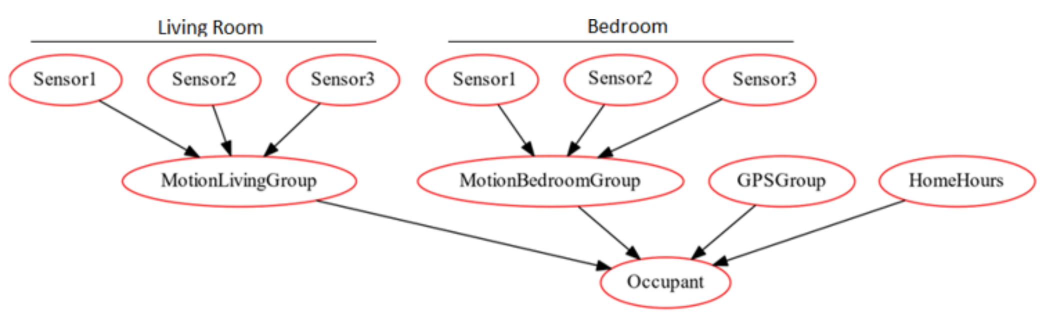

Figure 16: Bayesian Networks: Direct links for Bayesian network with HomeHours parallel with Ground-Truth.

The figure identifies the direct links by the arrows between the nodes. The discrete variables are the root nodes or the parent nodes. They are the drivers of the conditional table nodes, the non-root nodes, and the child nodes. The root nodes or the parent nodes "LivingSensor1," "LivingSensor2," and "LivingSensor3" are directly linked to the

conditional non-root node “MotionLivingGroup”; “BedroomSensor1,” “BedroomSensor2,” and “BedroomSensor3” are directly linked to the “MotionBedroomGroup.” The non-root nodes “MotionLivingGroup” and “MotionBedroomGroup” and the two root nodes “Ground-Truth” and “HomeHours” are directly linked to the final node “Occupant.” Occupant is the final node indicating whether someone is home during a specific timeframe from the “HomeHours” node. This data will be used to test the result of the Bayesian network against the other methods.

4.1.7 Bayesian network node probability tables

The root nodes LivingSensor1, LivingSensor2, LivingSensor3, BedroomSensor1, BedroomSensor2, BedroomSensor3, and Ground-Truth are the discrete distribution nodes in the network. These discrete distribution nodes have only two states, and the probability of each state for the PIR sensors is the same. The probability for each of the six sensors is as follows:

- P(LivingSensor1 | No Motion) = 0.5 and P(LivingSensor1 | Motion) = 0.5
- P(LivingSensor2 | No Motion) = 0.5 and P(LivingSensor2 | Motion) = 0.5
- P(LivingSensor3 | No Motion) = 0.5 and P(LivingSensor3 | Motion) = 0.5
- P(BedroomSensor1 | No Motion) = 0.5 and P(BedroomSensor1 | Motion) = 0.5
- P(BedroomSensor2 | No Motion) = 0.5 and P(BedroomSensor2 | Motion) = 0.5

- P(BedroomSensor3 | No Motion) = 0.5 and P(BedroomSensor3 | Motion) = 0.5
- P(Ground-Truth | Not Present) = 0.5 and P(Ground-Truth | Present) = 0.5

Each of the six sensors has the same probability of detecting the same motion event because the sensors monitor the same space. Hence, conditional probability tables for “MotionLivingGroup” and “MotionBedroomGroup” are as follows:

- P(MotionLivingGroup | LivingSensor1,LivingSensor2,LivingSensor3) = 0.33
- P(MotionBedroomGroup | LivingSensor1,LivingSensor2,LivingSensor3) = 0.33

Each of the three sensors has an equal probability assigned in the Bayesian model, meaning that each of the three sensors has 0.33% of the overall probability of the motion grouped sensors.

4.1.8 Conditional probability

“HomeHours” is the next table to be constructed. This action is discussed in this section, where the historical data are applied to construct the conditional probability tables and more accurately express the predictions over each time slot for the week tested. The Bayesian method allows for the option to have conditional probability tables that can drive predictive results from the network. The conditional probability table is constructed from past results or historical data. This historical data is comprised of the collection of events over a certain timeframe from the location. The test uses a four-week period in which events are gathered. These events are due to each of the six PIR sensors,

plus the two devices used to measure ground truth. Each of the events are paired with the one-minute increment time slots from each day of the week.

For example, all recorded events were gathered for a four-week period during the time slot of a Monday at 12:00:00–12:00:59 for “MotionLivingGroup.” Then the probability for that time increment of Monday at 12:00:00–12:00:59 for “MotionLivingGroup” is calculated as shown below:

Triggering Events: 2 of 3 sensors

Device: “MotionLivingGroup”

Time Slot: Monday at 12:00:00 – 12:00:59

Probability Equation: P() = Events / (Total Event)

P(LivingSensor1 | Monday at 12:00:00 – 12:00:59) = 2 pulses / 4 weeks = 0.50

On Monday between 12:00:00–12:00:59, there were two events over a four-week period; hence, there was a probability of 0.50, or a 50% chance of an event on Mondays at 12:00:00–12:00:59 for “MotionLivingGroup.”

The calculation for each of the six sensors and the two devices used to measure ground truth are used to show the combined probabilities. These four-week probabilities drive the conditional tables for the Bayesian nodes in the predictive network. The period of four weeks allows for a greater defined probability for the grouped sensors. When the timeframe is increased to more than a four-week period, the probability for the grouped sensors levels out. This leveling would limit the conditional probability table and the effect on the rest of the Bayesian network. The grouped sensors were defined to register an event when they detected two or more motion events within the same one-minute time slot.

To verify whether the occupants were or were not present, it was necessary to calculate the historical presence data from the ground-truth detectors over the same four-week time period. This process is similar to the probability calculations for each of the grouped sensors. Since the data were collected over four weeks, the probability for the data is as follows:

Triggering Events: 1 of 2 sensors

Device: “Ground-Truth”

Time Slot: Friday at 09:00:59 - 09:30:00

Probability Equation: P() = Events / (Total Event)

P(Present | Friday at 09:00:59 - 09:30:00) = 1 pulse / 4 weeks = 0.25

A plot for each day of the week was created, showing the probability for the combined devices used to measure ground truth. The probabilities for the combined devices are vital to help improve the accuracy of the Bayesian network because they help apply the historical information. Having the ground-truth data applied to the Bayesian network allows the network to draw upon information from occupant routines, which then can be applied during the sleep cycles, since the PIR motion sensors can’t detect the movements of the occupants while they sleep.

4.2 Bayesian Method Truth Tables

Once the nodes have been defined, the network defines the parameters for each node. These parameters can be configured for either a discrete distribution or a conditional probability table. The network, the parent nodes, or the starter nodes were defined as discrete distribution nodes. The remainder of the nodes were constructed as conditional probability tables, relying on the results from the other nodes within the

occupancy network. Each node has its own outputs depending on the beliefs applied to the parent nodes, as explained in Sections 4.2.1 to 4.2.13.

4.2.1 LivingSensor1

LivingSensor1 is a discrete distribution node in which a belief is applied. When beliefs are applied to a particular node, these beliefs are considered states. Hence, the LivingSensor1 node could be either in a state of “Motion” or a state of “No Motion.” No differences between the input and output of this node exist.

Table 5: LivingSensor1 node inputs and outputs comparison

State	Input	Output
LivingSensor1 – Motion	1	1
LivingSensor1 – No Motion	0	0

Table 5 demonstrates that the inputs and outputs are identical for the LivingSensor1 node, which is due to the application of a belief statement to the node. The belief signifies “Motion” when the column shows a value of 1, and 0 when no motion was detected

4.2.2 LivingSensor2

LivingSensor2 is another discrete distribution node in which a belief can be applied to the node. Inputs, outputs, and all other details are the same as described in 4.2.1.

Table 6: LivingSensor2 node inputs and outputs comparison

State	Input	Output
LivingSensor2 – Motion	1	1
LivingSensor2 – No Motion	0	0

4.2.3 LivingSensor3

LivingSensor3 is also a discrete distribution node. Details are the same as described in sections 4.2.1 and 4.2.2.

Table 7: LivingSensor3 node inputs and outputs comparison

State	Input	Output
LivingSensor3 – Motion	1	1
LivingSensor3 – No Motion	0	0

4.2.4 MotionLivingGroup

The “MotionLivingGroup” is a conditional probability table in which the results for this node are generated by its input arcs that connect to other nodes. This node applies the inputs of LivingSensor1, LivingSensor2, and LivingSensor3 to the conditional table to generate the outputs that are shown in the Table 8.

Table 8: MotionLivingGroup node inputs and outputs comparison

	State	LivingSensor1	LivingSensor2	LivingSensor3	Output
1	Motion Living Room	0	0	0	0.0
2	Motion Living Room	0	0	1	0.0
3	Motion Living Room	0	1	0	0.0
4	Motion Living Room	0	1	1	1.0
5	Motion Living Room	1	0	0	0.0
6	Motion Living Room	1	0	1	1.0
7	Motion Living Room	1	1	0	1.0
8	Motion Living Room	1	1	1	1.0

If more than two sensors detect the same event, then the output is true. A true output means motion was detected within the group.

4.2.5 BedroomSensor1

BedroomSensor1 is a discrete distribution node in which a belief would be applied to the node. When beliefs are applied to a particular node, these beliefs are considered states. The BedroomSensor1 node could therefore be either in a state of “Motion” or a state of “No Motion.” There are no differences between the input and output of this node, as indicated in the results columns of Table 9.

Table 9: BedroomSensor1 node inputs and outputs comparison

State	Input	Output
BedroomSensor1 – Motion	1	1
BedroomSensor1 – No Motion	0	0

The inputs and outputs are identical for BedroomSensor1 node, because of the application of a belief statement to the node. In Table 9, the belief is “Motion” when there is a one located in the column. If no value of one appears, then a zero is substituted for all other results.

4.2.6 BedroomSensor2

BedroomSensor2 is another discrete distribution node in which a belief would be applied to the node. When beliefs are applied to a particular node, these beliefs are considered states. Hence, the BedroomSensor2 node could be either in a state of “Motion” or a state of “No Motion.” There are no differences between the input and output of this node, as Table 10’s results columns demonstrate, for BedroomSensor2 node.

Table 10: BedroomSensor2 node inputs and outputs comparison

State	Input	Output
BedroomSensor2 – Motion	1	1
BedroomSensor2 – No Motion	0	0

The inputs and outputs are identical for the BedroomSensor2 node, because of the application of a belief statement to the node. In Table 10, the belief is “Motion” when there is a one located in the column. If there was not a one, then there was a zero substituted in for all other results.

4.2.7 BedroomSensor3

BedroomSensor3 is a discrete distribution node in which a belief would be applied to the node. When beliefs are applied to a particular node, these beliefs are considered states. The BedroomSensor3 node could be either in a state of "Motion" or a state of "No Motion." There are no differences between the input and output of this node.

Table 11: BedroomSensor3 node inputs and outputs comparison

State	Input	Output
BedroomSensor3 – Motion	1	1
BedroomSensor3 – No Motion	0	0

Outputs follow the inputs, as is similar to the other five sensors shown previously.

4.2.8 MotionBedroomGroup

The "MotionBedroomGroup" is a conditional probability table in which the results of this node are generated by its input arcs, which connect to other nodes. This node used the inputs BedroomSensor1, BedroomSensor2, and BedroomSensor3 applied to the conditional table to generate the outputs shown in Table 12.

Table 12: MotionBedroomGroup node inputs and outputs comparison

	State	Bedroom Sensor1	Bedroom Sensor2	Bedroom Sensor3	Output
1	Motion Bedroom Room	0	0	0	0.0
2	Motion Bedroom Room	0	0	1	0.0
3	Motion Bedroom Room	0	1	0	0.0
4	Motion Bedroom Room	0	1	1	1.0
5	Motion Bedroom Room	1	0	0	0.0
6	Motion Bedroom Room	1	0	1	1.0
7	Motion Bedroom Room	1	1	0	1.0
8	Motion Bedroom Room	1	1	1	1.0

If more than one sensor detects the same event, then, the output is true. True output means motion was detected within the group.

4.2.9 Occupant

The “Occupant” is another conditional probability table in which the results of this node are generated by the input reviewed from the inputs “MotionLivingGroup” and “MotionBedroomGroup,” which are applied to the conditional table to generate the outputs shown Table 13.

Table 13: Occupant node inputs and outputs comparison

	State	MotionLivingGroup	MotionBedroomGroup	Output
1	Occupant	0.0	0.0	0.0
2	Occupant	0.0	1.0	1.0
3	Occupant	1.0	0.0	1.0
4	Occupant	1.0	1.0	1.0

The conditional table of the “Occupant” node output occurs when there is at least one input that has detected motion, signifying that the condition is true, indicating an occupant within the space. Table 13 shows that if one input detects an event, the output is true. True output means was motion detected within the group. This logic stems from the fact that if either of the two rooms had two or more sensors detect the same event, someone must be in that room. Row three of Table 13 shows an occupant in the “MotionLivingGroup,” which caused the output to be one. For the last line in the table, there were two rooms that detected motion or had an occupant, so again, the table output was true, indicating motion was detected within the location.

4.2.10 HomeHours

“HomeHours” is a discrete distribution node in which a belief would be applied to the node, but in a manner that differs from that applied for LivingSensor1,

LivingSensor2, LivingSensor3, BedroomSensor1, BedroomSensor2, and BedroomSensor3. For the "HomeHours" node, it did not apply a belief as a state of "Motion" or "No Motion," but rather a probability regarding a motion event within that certain timeframe. Hence, the "HomeHours" node applied the probability that there was a "Motion" or "No Motion" event from the four weeks of training data. This training probability was applied to the discrete distribution input. The node probability "Motion" applied to the network is the same as the output of this node and can be seen in the results columns of Table 14, for the "HomeHours" node.

Table 14: HomeHours node inputs and outputs comparison

	State	Motion	No Motion	Output
1	Home Hours	1	0	1
2	Home Hours	0.75	0.25	0.75
3	Home Hours	0.5	0.5	0.5
4	Home Hours	0.25	0.75	0.25
5	Home Hours	0	1	1

The "Motion" probabilities and the outputs are identical for the node because they applied a belief statement to the node. Notably, the inputs to the discrete distribution equation must equal one, this can be seen in the "Motion" and "No Motion" columns for each of the time slots where the sum of the two columns equal a value of one.

4.2.11 OccupantIncludingHours

The "OccupantIncludingHours" is a conditional probability table in which the results are generated from the two prior nodes: 1) Occupant and 2) HomeHours. These two inputs are applied to the Bayesian network, which generates the outputs. The possible combinations of inputs and outputs are shown in Table 15.

Table 15: OccupantIncludingHours node inputs and outputs comparison

	State	Occupant	HomeHours	Output
1	Occupant Including Hours	1.0	1.0	1.0
2	Occupant Including Hours	0.0	0.75	0.37
3	Occupant Including Hours	1.0	0.5	0.87
4	Occupant Including Hours	0.0	0.25	0.12
5	Occupant Including Hours	0.0	0.0	0.0
6	Occupant Including Hours	1.0	0.75	0.93
7	Occupant Including Hours	0.0	0.5	0.25
8	Occupant Including Hours	1.0	0.25	0.12

4.2.12 Ground-Truth

The “Ground-Truth” node applies a belief as a state of “Present” or “Not-Present” from historical probability. This application allows the “Ground-Truth” node to implement the probability that there was a “Present” or “Not-Present” event from the four weeks of trained data as a prior probability to the Bayesian network. Table 16 shows the prior probability combinations of the Ground-Truth node.

Table 16: Ground-Truth node inputs and outputs comparison

	State	Present	Not Present	Output
1	Ground-Truth	1	0	1.0
2	Ground-Truth	0	0	0.0
3	Ground-Truth	0.75	0.25	0.75
4	Ground-Truth	0.25	0.75	0.25
5	Ground-Truth	0.5	0.5	0.5

The “Present” prior probabilities and the outputs are identical for the node, because of the belief statement applied to the node. The inputs to the discrete distribution equation must equal 1, as shown in the “Present” and “Not-Present” columns for each of the time slots, where the sum of the two columns equal a value of one.

4.2.13 OccupantIncludingGroundTruth

The "OccupantIncludingGroundTruth" is a conditional probability table in which the output for this node is generated by the OccupantIncludingHours probability and the output of the Ground-Truth node. The OccupantIncludingHours is the prior probability from the historical sensor signal data. The output of this node is displayed in Table 17.

Table 17: OccupantIncludingGroundTruth node inputs and outputs comparison

	State	OccupantIncludingHours	Ground-Truth	Output
1	Occupant Added Hours + Ground-Truth	0	1	0.74
2	Occupant Added Hours + Ground -Truth	0	0.2	0.18
3	Occupant Added Hours + Ground -Truth	0.1	0.2	0.26
4	Occupant Added Hours + Ground -Truth	0.1	0.5	0.43
5	Occupant Added Hours + Ground -Truth	0.1	0.7	0.6
6	Occupant Added Hours + Ground -Truth	0.1	1	0.78
7	Occupant Added Hours + Ground -Truth	0.2	1	0.81
8	Occupant Added Hours + Ground -Truth	0.2	0.7	0.65
9	Occupant Added Hours + Ground -Truth	0.3	0.7	0.7
10	Occupant Added Hours + Ground -Truth	0.8	0.2	0.73
11	Occupant Added Hours + Ground -Truth	0.8	0.2	0.69
12	Occupant Added Hours + Ground -Truth	0.8	1	0.96
13	Occupant Added Hours + Ground -Truth	0.9	1	0.98
14	Occupant Added Hours + Ground -Truth	0.9	0.7	0.91
15	Occupant Added Hours + Ground -Truth	1	1	1

4.3 Bayesian Example for the Condominium

The first level of nodes in the Bayesian network include the six individual PIR sensor nodes. Each of these nodes is a binary state. The binary states are either zero or one, where zero indicates no motion was detected by the PIR sensor and where one indicates motion detected within the area. These binary states are the same for the six sensor nodes. The inputs and outputs of these sensor nodes directly follow the motion activity within the location. The condominium input and the output comparison for the sensor parent nodes for the Bayesian network defined in the previous section is shown in Table 18.

Table 18: Bayesian sensor node truth table template for each of the six PIR sensor nodes

	Sensor Node States	Output
1	0	0
2	1	1

The next level of the Bayesian network contains the grouped room nodes. These are the two nodes interpreting their PIR sensor nodes to ensure at least two motion events. Given two motion events, the group nodes become true. For example, as depicted in Table 19, the condominium had three sensors in the living room that detected motion. This scenario resulted in the output of the living room grouped node to be one. There was motion within the room. However, the bedroom group only had one PIR sensor that detected motion, so the bedroom group node remained a zero; in other words, no motion was detected.

Table 19: Living room and bedroom group node input and output example

Date/Time	Node Input			Node Output
2018-09-24 17:09:00	LivingRoom Sensor1	LivingRoom Sensor2	LivingRoom Sensor2	LivingGroup Results
	1	1	1	1
	Bedroom Sensor1	Bedroom Sensor2	Bedroom Sensor3	BedroomGroup Results
	1	0	0	0

Once the grouped nodes were calculated, the output of the living room and bedroom group nodes were then sent to the occupant node. The occupant node checked the output of the grouped nodes and determined whether any occupants were in either of the rooms. If any occupant was present—meaning if either of the two group's outputs had a value of one—then the condominium had occupants. This situation is expressed by the group node, as depicted Table 20.

Table 20: Condominium occupant grouped node input and output example

Date/Time	Node Input	Node Output
2018-09-24 17:09:00	LivingGroup Results 1 BedroomGroup Results 0	Occupant 1

An occupant was determined to be within the condominium. If both the living room and the bedroom nodes were zero, then the output of the occupant node would have been zero. The output of the occupant node is then combined with the PIR sensor historical data, or prior probability, and is applied to the Bayesian network to produce the

trained sensor output. The Bayesian network generates a value, P(E), which is the initial degree of belief based off the whole network. The trained sensor output can be expressed as the likelihood of an occupant being at the condominium based on historical information combined with the current PIR sensor states. Table 21 shows the occupant node result and the prior probability that produces the train sensor node occupancy probability.

Table 21: Condominium trained sensor node example and calculations

Date/Time	**Occupant Node Results P(E|H)**	**Prior Probability P(H)**	**Bayesian Network Value P(E)**	**Trained PIR Sensor Node (output) P(H|E)**
2018-09-24 21:00:00	1	.75	0.8108	0.925
2018-09-24 17:09:00	1	0.5	0.5747	0.87
2018-09-24 21:01:00	1	0.25	0.3086	0.81

The Trained PIR Sensor Node is calculated using the Bayesian equation (Fenton & Neil, 2013, p. 116):

$$\mathrm{P(H|E)} = \frac{P(E|H) * P(H)}{P(E)} \tag{5}$$

The higher the prior probability, or the more often the occupant was home during training time, the higher the output probability generated. A high output probability indicates a greater chance of the occupant being home if they are “normally” home during the four weeks of the trained hours at that time.

The next step in this analysis was to apply the presence detector prior probability to the Bayesian network. This step is called the “trained with presence detector hours” method. An example of this calculation with the condominium location is like the trained

hours calculation previously demonstrated. Table 22 shows the trained hours node result and the prior probability that produces the train sensor node occupancy probability.

Table 22: Condominium trained Ground-Truth node example and calculations

Date/Time	**Trained PIR Sensor Node P(E|H)**	**Prior Probability P(H)**	**Bayesian Network Value P(E)**	**Trained Ground-Truth (output) P(H|E)**
2018-09-24 21:00:00	0.925	1.0	0.9439	0.98
2018-09-24 18:40:00	0.87	0.75	0.7331	0.89
2018-09-24 17:09:00	0.81	1.0	0.8526	0.95

The Trained PIR Sensor Node is calculated using the Bayesian equation (Fenton & Neil, 2013, p. 116):

$$\mathrm{P(H|E)} = \frac{P(E|H) * P(H)}{P(E)} \tag{6}$$

The impact of the presence sensor on the trained sensor hours in the example is shown in Table 22. The top row in the table demonstrates a probability of 0.925 with a prior probability of the presence sensor of 1. This strong prior probability caused the output of the Bayesian network to increase to 0.98. Additionally, this can be seen on the bottom row in Table 22, even with a 0.81 probability, the prior probability remains the same at one, and the occupancy probability still increases to 0.95, since the presence probability has a strong influence on occupancy in the Bayesian network.

In conclusion, the Bayesian network is constructed using Fenton and Neil’s seven suggestions: 1) identify the set of variables relevant to the problem; 2) create a node corresponding to each of the variables identified; 3) identify the set of states for each

variable; 4) specify the states for each node; 5) identify the variables that require direct links; 6) create the identified links; and 7) for each node in the Bayesian network, specify the node probability table (Fenton & Neil, Risk Assessment and Decision Analysis with Bayesian Networks, 2013). The Bayesian network has a node for each of the main decision points to be used to detect the occupancy. The truth table behind each of the decisions used has also been presented. This allows the Bayesian network to be constructed, which will generate an occupancy prediction.

CHAPTER 5: COMPARISON

This chapter compares the Bayesian network results with the independent sensors, grouped sensors, trained hours, and GPS signal and Bluetooth key fob presence data for each of the four locations. The comparison groups include the following: 1) individual sensor results compared with the smartphone GPS signal and Bluetooth key fob presence data, 2) grouped sensor results compared with the smartphone GPS signal and Bluetooth key fob presence data, 3) trained PIR sensor hours data, and 4) trained ground-truth information compared with the GPS signal and Bluetooth key fob data over several simulations to determine the reliability of the Bayesian network when certain types of sensor failures are injected into the network.

5.1 Independent Sensor

In the independent sensor method, one of the six sensors at a location must trigger for the location to show an event. A correlation between the ground-truth is calculated for each of the six sensors. Then, an overall calculation is based on any one sensor triggering. This overall calculation is the correlation of the independent sensor method. Calculations are completed for each of the four locations.

The correlation is calculated from the independent sensor events compared with the ground-truth. The higher the correlations are, closer to 1.0, the stronger the relationship is considered. If the correlation is 1.0, then the PIR sensor events triggered at the same time, and the signal from either of the devices used to measure ground truth showed occupancy. The way the correlation is calculated for LivingSensor1 is shown in Equation 7.

$$\phi \text{ correlation } = \frac{(296)(3{,}637) - (6{,}140)(7)}{\sqrt{(6{,}436)(3{,}644)(303)(9{,}777)}} \quad (7)$$

$$\phi \text{ correlation } = 0.12$$

From equation 7, the ground-truth and the PIR sensor pulse data detected motion at the same time 296 times, and neither the PIR sensor nor the ground-truth detected motion 3,637 times. Over the one-week timeframe, there was a false positive seven times, and 6,140 times no motion was detected when there was an occupant. The end correlation is thus 0.12. This low correlation stems from the 6,140 times there was no motion when there was an occupant, of which 4,015 of these times occurred between 07:00 PM and 07:00 AM.

Using the cross-table correlation below, the coefficient of the correlation can be calculated.

Table 23: Condominium sensor 1 correlation to ground-truth over one-week timeframe.

Truth\Measured	1	0	
1	296	6140	r1=6436
0	7	3637	r2=3644
	c1=303	c2=9777	

The other six PIR sensor correlations at each location are calculated in the similar manner. The results indicate the strength of the relationship. Correlations range from "−1" to "1," with positive 1.0 being strongly related and negative 1.0 being not related. In the correlation above, the Condo-LivingSensor1 has a weak relationship to ground-truth with a value of 0.12. Table 24 shows the values of the ϕ correlation with ground-truth for all sensors at all locations. These values are generally low as expressed because the pulse rate of individual sensors is low relative to the time any space is occupied.

Table 24: Independent correlation for the six PIR sensors for the individual locations

	Living Sensor1	Living Sensor2	Living Sensor3	Bedroom Sensor1	Bedroom Sensor2	Bedroom Sensor3
Condominium	0.12	0.12	0.12	0.14	0.12	0.10
One Occupant (2018)	0.18	0.21	0.19	0.13	0.18	0.12
Two Retirees	0.68	0.69	0.69	0.71	0.81	0.72
One Occupant (2019)	0.16	0.18	0.23	0.16	0.21	0.36

The independent method takes into account that if any of the PIR sensors register an event, then it is assumed there is an occupant. The correlation for the independent method is then calculated across a one-week time period. Similarly, the correlations for each of the methods are calculated to compare the predicted events with the ground-truth, which is demonstrated by using Microsoft Excel equation, CORREL. CORREL is used to simplify the process of calculating the correlation of the occupant data with ground-truth over the large data set, see Table 25 (MAT, n.d.).

Table 25: Independent method correlation calculation variables. Where n = *10,080 minutes per week,* x = *Independent method detection and* y = *ground-truth*

∑(x)	∑(y)	∑(xy)	∑(x^2)	∑(y^2)
871.0	6436.0	857.0	871.0	6436.0
		Corelation		**0.22**

In a comparison of all the four locations (see Table 26), the location with two retirees has the highest correlation. Figure 17 shows the weak relationship between the individual sensors and ground-truth. The blue lines show where the individual sensors detected occupancy, and the orange bar shows the ground-truth of an occupant present.

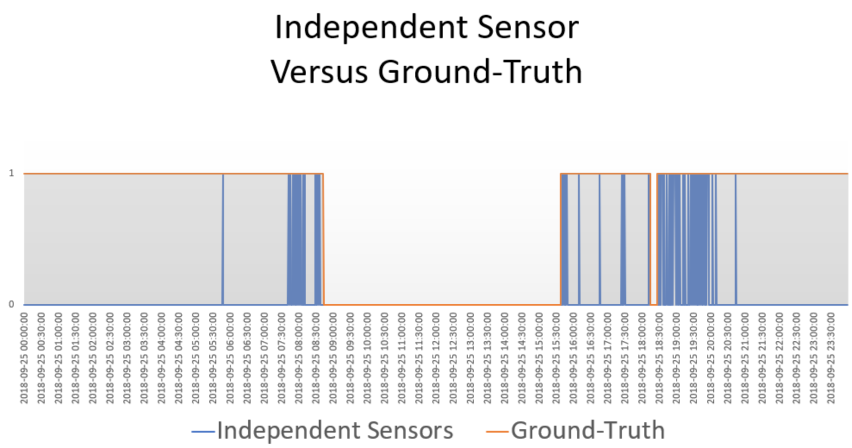

Figure 17- Independent Sensor correlation to Ground-Truth at the Condominium for 09-25-2018

The main difference between this location and the other three locations is the addition of movement during sleeping hours. The two retirees had much more movement during these hours.

Table 26: Independent sensor summary

	Independent Sensor Method
Condominium	0.22
One Occupant (2018)	0.30
Two Retirees	0.90
One Occupant (2019)	0.41

In conclusion, the independent sensor method improves the results over a single PIR sensor within a given location because it considers all the six sensors and any of the sensors can trigger an event. The main factor that greatly affects the results is sleep time in which the occupants are present, but no motion event can be detected. This factor is due to the PIR sensors not being sensitive enough to detect slight movement within a space.

5.2 Grouped Sensors

The grouped sensor method requires at least two sensors in an area to trigger before an event would record an occupant. This method reduces the desired correlations in all four locations since the sensors in each of the locations had fewer motion triggers throughout the timeframes and particularly during the sleeping hours. The correlation calculation for the condominium is shown in Table 27.

Table 27: Grouped method correlation calculation variables. Where n = 10,080 minutes per week, x = grouped method detection and y = ground-truth

∑(x)	∑(y)	∑(xy)	∑(x^2)	∑(y^2)
461.0	6436.0	450.0	461.0	6436.0
		Corelation		**0.15**

The ϕ correlation coefficient at the condominium, one occupant (2018), the two retirees' location, and one occupant (2019) equal: 0.15, 0.23, 0.83, and 0.27, respectively. These correlations decrease when compared with the independent detection method. This decrease is due to some of the motion events being filtered out, since with the grouped method at least two PIR sensor events must occur for the system to indicate an occupant. Figure 18 presents the group sensor occupancy compared with ground-truth occupancy that occurred at the Condominium.

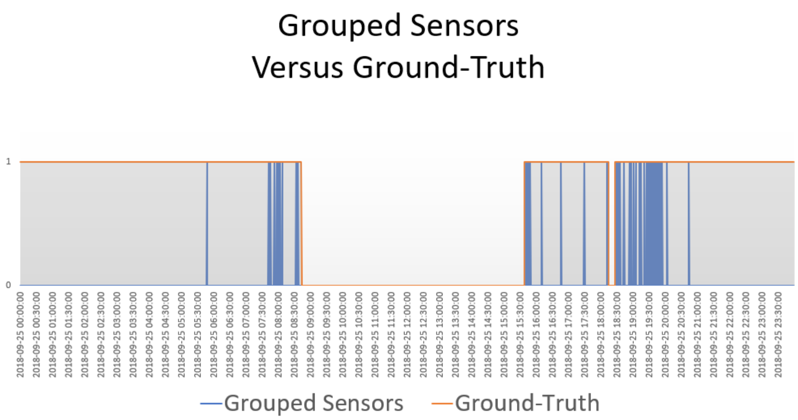

Figure 18- Grouped Sensor correlation to Ground-Truth at the Condominium for 09-25-2018

This added requirement decreases the ϕ correlation for each of the locations.

Table 28: Grouped sensor summary

	Grouped Sensor Method
Condominium	0.15
One Occupant (2018)	0.23
Two Retirees	0.83
One Occupant (2019)	0.27

At the two retirees' location, there is still a higher correlation over the other locations. This higher correlation is due to the extra motion events detected by the PIR sensors during the sleep time. This method decreases the accuracy of events detected to lower than the independent method above. This decrease can be seen in Table 29.

Table 29: Comparison between independent and grouped sensor method

	Independent Sensor Method	Grouped Sensor Method	Compared
Condominium	0.22	0.15	−0.07
One Occupant (2018)	0.30	0.23	−0.08
Two Retirees	0.90	0.83	−0.06
One Occupant (2019)	0.41	0.27	−0.14

For the first three locations in the table above, it is interesting to note that the decrease in the correlation is about the same at each of the locations. This rough equivalence does not hold for the one occupant (2019) location. The assumption is that at the one occupant (2019) location, the sensors are more evenly spread throughout the house. It was unlikely that two sensors would detect the same motion event in the same timeframe.

5.3 Trained PIR Sensor Hours

The trained sensor hours' method, using Bayesian method, first takes the previous sensor data over a month, also known as prior probabilities, from the specific sensors and then applies them to the analysis to improve predictions. The calculations for the condominium consider the historical information to create a more accurate prediction. This increase in accuracy can be seen in the condominium calculation in Table 30.

Table 30: Trained sensor method correlation calculation variables. Where n = 10,080 minutes per week, x = trained method detection and y = ground-truth

∑(x)	∑(y)	∑(xy)	∑(x^2)	∑(y^2)
513.4	6436.0	464.0	329.1	6436.0
		Corelation		**0.16**

The ϕ correlation for the condominium location using the trained or historical sensor data method was a ϕ of 0.16, a slight decrease from the independent method and slight increase over the grouped method. For the one occupant (2018) location, there is a correlation of 0.24, and for the other one occupant location (2019), a correlation of 0.34. The sensor location placements do demonstrate an effect on the accuracy of detecting an occupant within the given locations.

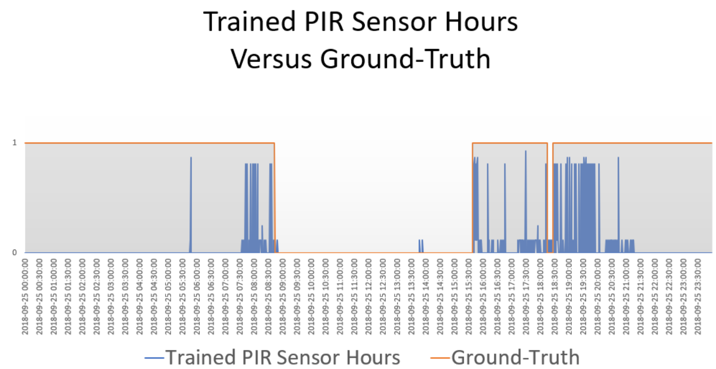

Figure 19- Trained PIR Sensor Hours correlation to Ground-Truth at the Condominium for 09-25-2018

In Table 31, the two retirees' location shows the greatest correlation due to their increase in activity during nighttime hours and the detection of this during the sleeping hours.

Table 31: Trained PIR sensors with historical sensor summary

	Trained PIR Sensor Hours Method
Condominium	0.16
One Occupant (2018)	0.24
Two Retirees	0.84
One Occupant (2019)	0.34

The correlation for four locations presents a decrease from the independent sensor method, as can be seen in Table 32. The trained Bayesian hours do not provide the same correlation results as the independent sensor method due to the lack of sensors providing event detection. The trained Bayesian hours requires two sensor pulses to detect an event which ultimately decreases the correlation in the results, though increases the accuracy.

Table 32: Comparison between independent and Trained PIR sensor hour history method

	Independent Sensor Method	Trained PIR Sensor Hours Method	Compared
Condominium	0.22	0.16	−0.06
One Occupant (2018)	0.30	0.24	−0.07
Two Retirees	0.90	0.84	−0.06
One Occupant (2019)	0.41	0.34	−0.07

A comparison of the Bayesian trained hours results with the grouped sensor method results offers a more accurate correlation. The Bayesian trained hours method does improve the accuracy of event detection within specific scenarios. The grouped method and the Bayesian method are more closely related than either are to the independent sensor system because the grouped method and the Bayesian method use the logic of having at least two or more sensors detected in order to determine occupancy.

Table 33: Comparison between grouped and trained PIR sensor hour history method

	Grouped Sensor Method	Trained PIR Sensor Hours Method	Compared
Condominium	0.15	0.16	0.01
One Occupant (2018)	0.23	0.24	0.01
Two Retirees	0.83	0.84	0.01
One Occupant (2019)	0.27	0.34	0.07

A review of the results at the location with two retirees indicates that the correlations scored much higher at this location. This higher score is due to the occupants not leaving the location as regularly as the occupants in the other three locations, allowing less time for the sensors to detect that no motion has occurred. The other three locations had more no motion events, which negatively affects their correlations.

5.4 Trained Ground-Truth

When comparing the historical probability for the hours with the ground-truth, as seen in Section 3.5.3, the addition of ground-truth information in the correlation for the condominium improves the accuracy of the results significantly over those of other methods. Table 34 shows the steps required to calculate the correlation between the trained presence method and ground-truth:

Table 34:Trained with presence sensor method correlation calculation variables. Where n = 10,080 minutes per week, x = trained presence method detection and y = ground-truth

∑(x)	∑(y)	∑(xy)	∑(x^2)	∑(y^2)
5499.1	6436.0	4412.4	3719.0	6436.0
		Corelation		**0.70**

The correlation for the condominium using the trained presence-detection method sees an improvement to 0.70, and for one occupant (2018) sees an improvement to 0.71. Figure 20 shows the trained ground-truth occupancy results after applying the previous known occupancy. The trained ground-truth and the actual ground-truth are more similar than the other methods tested.

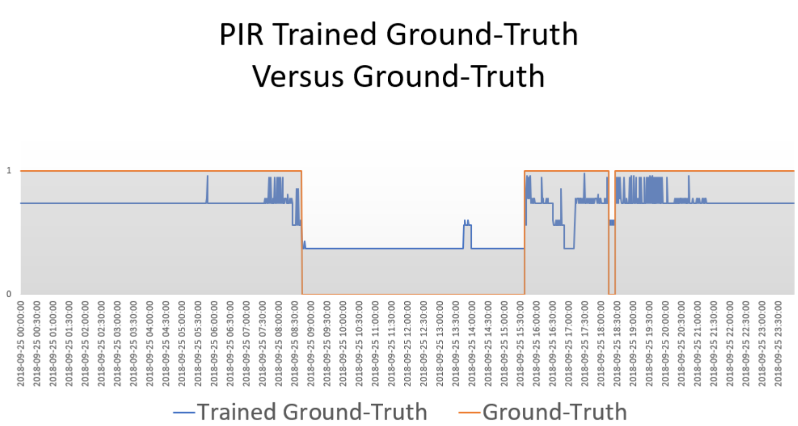

Figure 20- Trained Ground-Truth correlation to Ground-Truth at the Condominium for 09-25-2018

The location with the retiree's correlation shows an increased correlation of 0.94, and the final location of one occupant (2019) demonstrates an improved correlation of 0.82, as shown in Table 35.

Table 35: Trained ground-truth summary

	Trained Ground-Truth Method
Condominium	0.70
One Occupant (2018)	0.71
Two Retirees	0.94
One Occupant (2019)	0.82

These results show greater than double the performance of the independent, grouped, and trained hours' methods. This improvement is predominately due to the GPS or key fob at the location during the sleeping hours still prompting the detection of an occupant in the locations. Improvement of the correlation can be seen across all the other methods in Tables 36, 37, and 38.

Table 36: Independent sensors compared with trained ground-truth summary

	Independent Sensor Method	Trained Ground-Truth Method	Compared
Condominium	0.22	0.70	0.48
One Occupant	0.30	0.71	0.41
Two Retirees	0.90	0.94	0.05
One Occupant	0.41	0.82	0.41

The comparison of the two retirees' correlation for the independent sensors and the trained sensors with the presence sensors is of particular interest. The retirees were at home most of the time. When the occupants are home for a longer duration, the need for a GPS or Bluetooth presence detector is reduced. The greatest improvement amongst the locations is the condominium location. The residents were not home as often, meaning

that the need for a GPS or Bluetooth presence detector was required to improve the accuracy of the occupancy.

The grouped method compared to the trained ground-truth shows similar results. The two retirees' location shows the least improvement, due to the relatively low reliance on the ground-truth as the occupants are home more. The overall correlation does improve with the added presence-detection history.

Table 37: Grouped sensors compared with trained ground-truth summary

	Grouped Sensor Method	Trained Ground-truth Method	Compared
Condominium	0.15	0.70	0.54
One Occupant (2018)	0.23	0.71	0.48
Two Retirees	0.83	0.94	0.11
One Occupant (2019)	0.27	0.82	0.55

Both Bayesian methods (the trained PIR sensor hours method and the trained ground-truth method) increase the correlation in each of the four locations. The largest increase is observed when ground-truth detectors were applied at each of the locations.

Table 38: Bayesian sensor history compared with trained ground-truth summary

	Trained PIR Sensor Hours Method	Trained Ground-truth Method	Compared
Condominium	0.16	0.70	0.53
One Occupant (2018)	0.24	0.71	0.47
Two Retirees	0.84	0.94	0.10
One Occupant (2019)	0.34	0.82	0.48

5.5 Comparison Conclusion

This section presents the final comparison of the different methods. The condominium has a 0.22 correlation value with occupancy and ground-truth when each of

the six PIR sensors acted independently of each other. This higher result over the grouped and Bayesian trained hours and those trained with the ground-truth is due to the lack of the required motion that must be detected for an event to register an occupant.

The grouped sensors method exhibited a poorer relationship with ground-truth than did the independent sensor method. This lower performance is due to the grouped sensors method's requirement of more than one PIR sensor to detect the same motion from the occupants. When applying the training for the Bayesian model for each of the six PIR sensors, the correlation value increases from 0.15 to 0.16 for the grouped method. The increase is due to the network knowledge for when the event was "most likely" to be present. In the condominium, the greatest improvement came when the trained sensor hours with presence detector was applied and found a correlation of 0.70.

Table 39: Comparison of the correlation for each of the four methods for the condominium location

Method	**Correlation**
Independent Sensors	0.22
Grouped Sensors	0.15
Trained PIR Sensor Hours	0.16
Trained Ground-Truth	0.70

Table 39 demonstrates that the trained presence detector has a much stronger relationship with ground-truth compared to any of the other methods. The main reason for this improvement was that the ground-truth detector knew when the occupants were home and not active, unlike the PIR sensors, which required movement across the detection zones.

The single occupant location shows a trend like that of the condominium location. The independent sensors have a stronger relationship with ground-truth than do the grouped sensors, with a 0.30 correlation value for the independent method and 0.23 for

the grouped method. Applying the trained sensor hours and the current PIR sensor data to the Bayesian network, an increased correlation value of 0.24 was noted. Finally, the strongest correlation between the four methods tested is the method where the ground-truth detection history was applied, with a correlation of 0.71. The comparison results for the one occupant (2018) location in are shown in Table 40.

Table 40: Comparison of the correlation for each of the four methods for the one occupant (2018) location

Method	**Correlation**
Independent Sensors	0.30
Grouped Sensors	0.23
Trained PIR Sensor Hours	0.24
Trained Ground-Truth	0.71

The next location, with the two retirees, revealed some extra insight into how different locations have different activities that can affect results. In this location, the correlation values were much higher than in any of the other locations. The main factor driving these stronger correlation values was the two occupants being at their home for most of the time. This tendency to be at home did not apply to the three other locations, where the occupants were normally away. The retiree’s location had the strongest correlation with ground-truth of all the four methods.

Notably, however, the improvement from the grouped method to the trained hour method again increased by a correlation value of 0.01. This increase, presented in Table 41, did not hold in other previous locations.

Table 41: Comparison of the correlation for each of the four methods for the two retirees' location

Method	**Correlation**
Independent Sensors	0.90
Grouped Sensors	0.83
Trained PIR Sensor Hours	0.84
Trained Ground-Truth	0.94

The last location to compare is the one occupant (2019) location. This is the same location as the one occupant (2018) location but with different sensor placements and in a different year. The location has the second-strongest correlation values between the four locations, due to the improved PIR sensor location placement. Unlike the individual study at this location where the six PIR sensors were placed only within two rooms, this second individual study had the sensors placed throughout the whole house. The PIR sensor had more surface to have a larger coverage for detecting motion within the location.

Notably, Table 42 shows that the Bayesian modeling affects the difference between the grouped method and the trained hours' method more substantially than in the other three locations. Unlike the other three locations, this location had an increase of 0.16 in the correlation value. This increase could be due to the different sensor placements or to different activities registered by the single occupant. The second point of interest for this location is the stronger correlation for the independent method, likely due to the increased PIR sensor coverage. However, with the improved coverage, the performance of the grouped sensor method was about the same in this particular location.

Table 42: Comparison of the correlation for each of the four methods for the one occupant (2019) location

Method	**Correlation**
Independent Sensors	0.41
Grouped Sensors	0.27
Trained PIR Sensor Hours	0.34
Trained Ground-Truth	0.82

The review of the results shows that the four locations indicate an increase in correlation when the ground-truth is applied to the Bayesian network. All the other methods were within the same lower ranges of correlation. The locations with one occupant, overall, have a greater correlation with ground-truth than does the condominium location. This greater reliability is due to the increased motion events detected during the evening hours, as seen with the retiree’s location. This location has better results overall, due to the detection during the normal sleeping periods. These overall results can be seen in Table 43.

Table 43: Overall summary results of each method and locations

	Independent	**Grouped Sensors**	**Trained PIR Sensor Hours**	**Trained Ground-Truth**
Condominium	0.22	0.15	0.16	0.70
One Occupant (2018)	0.30	0.23	0.24	0.71
Two Retirees	0.90	0.83	0.84	0.94
One Occupant (2019)	0.41	0.27	0.34	0.82

The correlation results for the four methods shown in Table 43, are low compared to previous research. This lower correlation is due to the minimal PIR sensor pulses that occur throughout the nighttime. This can be seen when comparing the correlation for a 24-hour period versus a 07:00AM – 07:00PM period where nighttime is not included, see

Figure 21. Table 44 shows the results for a 24-hour period for each of the four methods.

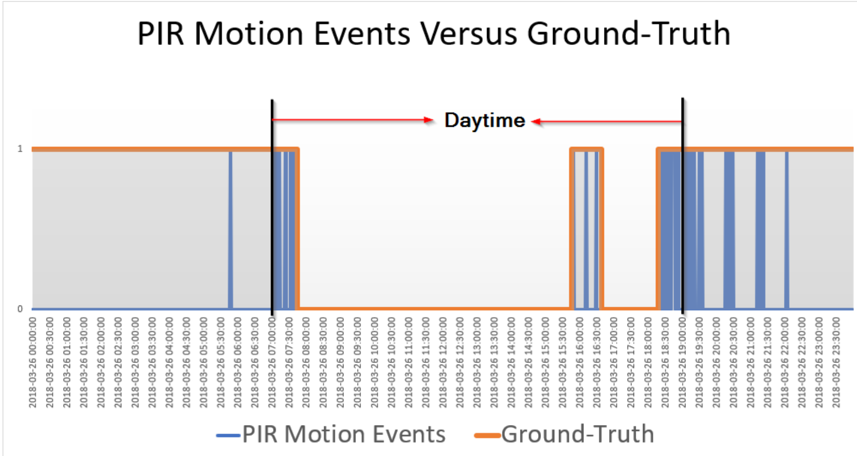

Figure 21- Condominium daytime example of low correlation

Table 44: Comparison of the correlation for each of the four methods for the condominium location in a 24-hour period and a daytime period (07:00AM – 07:00PM)

Method	24-hour	Daytime
Independent Sensors	0.22	0.44
Grouped Sensors	0.15	0.30
Trained PIR Sensor Hours	0.16	0.34
Trained Ground-Truth	0.70	0.69

By removing the nighttime hours, the correlations for the condominium location improved. This shows that throughout the nighttime, the PIR sensors are unable to detect enough movement when measuring occupancy. However, the Trained Ground-Truth method results were nearly the same between the 24-hour period and the daytime correlation. This is due to the added ground-truth that helps overcome the PIR sensor detection issue, since the ground-truth are using GPS and Bluetooth detection devices and do not rely on motion to be detected. The other three locations had similar results as seen in Table 45 through Table 47.

Table 45: Comparison of the correlation for each of the four methods for the one occupant (2018) in a 24-hour period and a daytime period (07:00AM – 07:00PM)

Method	**24-hour**	**Daytime**
Independent Sensors	0.30	0.47
Grouped Sensors	0.23	0.34
Trained PIR Sensor Hours	0.24	0.37
Trained Ground-Truth	0.71	0.67

Table 46: Comparison of the correlation for each of the four methods for the two retirees' location in a 24-hour period and a daytime period (07:00AM – 07:00PM)

Method	**24-hour**	**Daytime**
Independent Sensors	0.90	0.89
Grouped Sensors	0.83	0.82
Trained PIR Sensor Hours	0.84	0.83
Trained Ground-Truth	0.94	0.92

Table 47: Comparison of the correlation for each of the four methods for the one occupant (2019) in a 24-hour period and a daytime period (07:00AM – 07:00PM)

Method	**24-hour**	**Daytime**
Independent Sensors	0.41	0.46
Grouped Sensors	0.27	0.31
Trained PIR Sensor Hours	0.34	0.36
Trained Ground-Truth	0.82	0.82

After reviewing the PIR sensor, Table 48 through Table 51, each sensor correlation for the four locations had a strong relationship. However, none of the six sensors in each location had a correlation of 1.0. The strongest correlation was found to be at the Condominium between sensor2 and sensor3 with a correlation of 0.86. The weakest correlations were found at one occupant (2019) between sensor5 and sensor2, and sensor5 and sensor3 with a correlation of -0.02.

Table 48: Comparison of the correlation for each of the six sensors for the Condominium

	Sensor1	Sensor2	Sensor3	Sensor4	Sensor5	Sensor6
Sensor1	-	0.81	0.82	0.18	0.07	0.12
Sensor2	0.81	-	0.86	0.19	0.08	0.11
Sensor3	0.82	0.86	-	0.16	0.07	0.11
Sensor4	0.17	0.19	0.16	-	0.32	0.26
Sensor5	0.07	0.08	0.07	0.32	-	0.26
Sensor6	0.12	0.11	0.11	0.30	0.26	-

Table 49: Comparison of the correlation for each of the six sensors for the one occupant (2018)

	Sensor1	Sensor2	Sensor3	Sensor4	Sensor5	Sensor6
Sensor1	-	0.68	0.71	0.03	0.03	0.02
Sensor2	0.68	-	0.75	0.06	0.05	0.04
Sensor3	0.71	0.75	-	0.06	0.07	0.05
Sensor4	0.03	0.06	0.06	-	0.54	0.81
Sensor5	0.03	0.05	0.07	0.54	-	0.50
Sensor6	0.02	0.04	0.05	0.81	0.50	-

Table 50: Comparison of the correlation for each of the six sensors for the two retirees' location

	Sensor1	Sensor2	Sensor3	Sensor4	Sensor5	Sensor6
Sensor1	-	0.74	0.74	0.14	0.11	0.14
Sensor2	0.74	-	0.84	0.21	0.17	0.21
Sensor3	0.74	0.84	-	0.22	0.18	0.22
Sensor4	0.14	0.21	0.22	-	0.78	0.92
Sensor5	0.11	0.17	0.18	0.78	-	0.78
Sensor6	0.14	0.21	0.22	0.92	0.78	-

Table 51: Comparison of the correlation for each of the six sensors for the one occupant (2019)

	Sensor1	Sensor2	Sensor3	Sensor4	Sensor5	Sensor6
Sensor1	-	0.17	0.47	0.28	0.02	0.22
Sensor2	0.17	-	0.44	0.53	-0.02	0.00
Sensor3	0.47	0.44	-	0.64	-0.02	0.06
Sensor4	0.28	0.53	0.64	-	-0.01	0.01
Sensor5	0.07	0.08	0.07	0.32	-	0.47
Sensor6	0.22	0.00	0.06	0.01	0.47	-

In Table 52 through Table 55 the amount of time each PIR motion sensor and ground-truth detectors measured an occupant within each of the four locations can be seen.

Table 52: Comparison of the occupant time for each of the six PIR motion sensors and each ground-truth detector for the Condominium

Senor/Detector	**Time(min)**
Sensor1	**303**
Sensor2	**306**
Sensor3	**311**
Sensor4	**372**
Sensor5	**258**
Sensor6	**188**
Ground-Truth1	**6,436**
Ground-Truth2	**6,421**

Table 53: Comparison of the occupant time for each of the six PIR motion sensors and each ground-truth detector for the occupant (2018)

Senor/Detector	**Time(min)**
Sensor1	**313**
Sensor2	**417**
Sensor3	**355**
Sensor4	**159**
Sensor5	**313**
Sensor6	**138**
Ground-Truth1	**4,970**
Ground-Truth2	**4,958**

Table 54: Comparison of the occupant time for each of the six PIR motion sensors and each ground-truth detector for the two occupants in two retirees'

Senor/Detector	**Time(min)**
Sensor1	**313**
Sensor2	**417**
Sensor3	**355**
Sensor4	**159**
Sensor5	**313**
Sensor6	**138**
Ground-Truth1	**8,471**
Ground-Truth2	**4,369**

Table 55: Comparison of the occupant time for each of the six PIR motion sensors and each ground-truth detector for the occupant (2019)

Senor/Detector	Time(min)
Sensor1	**134**
Sensor2	**155**
Sensor3	**415**
Sensor4	**277**
Sensor5	**246**
Sensor6	**754**
Ground-Truth1	**7,415**
Ground-Truth2	**7,394**

The correlation between the two ground-truth detectors is 0.98 for three of the four locations. The two retirees' location has a lower correlation. This lower correlation between the two ground-truth devices seemed to occur when one of the two devices stopped registering its' presence. The correlation between the ground-truth devices at each of the locations can be seen in Table 56.

Table 56: Comparison of the correlation for each of the four-locations ground-truth devices.

Method	Correlation
Condominium	0.9812
One Occupant (2018)	0.9777
Two Retirees	0.2833
One Occupant (2019)	0.9759

CHAPTER 6 SUMMARY AND FUTURE RESEARCH

This chapter discusses how each of the project stakeholders is affected by the results of this research paper and the way they can implement Bayesian modeling to their applications. This research is a foundation on how a practical application could be constructed. This chapter discusses how other future research may apply the practice of multiple sensors and Bayesian modeling to other applications.

Section 6.1 discusses the stakeholders for the project and how this paper can benefit them. Section 6.2 suggests other ways to implement the Bayesian modeling that may improve on this research. Section 6.3 summarizes the paper and the results.

6.1 Project Stakeholders

The stakeholders for this project are those who need to enhance their occupancy detection within their locations. The main stakeholders are homeowners who desire to improve their occupancy detection in a residential environment, doctors and patients who want to monitor movement and occupancy within a given space, people involved with elder care safety, and those who deal in utility automation. These stakeholders will all have an advantage when utilizing the Bayesian network within their area of expertise.

6.1.1 Homeowners

Homeowners would benefit from a Bayesian network when looking for ways to automate various procedures within their home. Some of these procedures might include operating lights, controlling an appliance such as a coffee maker, setting alarms or alerts, setting a thermostat, or feeding pets, to name a few. With the combination of sensors and the Bayesian network modeling, homeowners can improve their lifestyles and implement better functionality within their homes, in conjunction with current devices. Currently,

most devices like coffee makers, lights, and alarms are set manually, where “set” means defined parameters configurable by the user. Homeowners set the time the coffee maker turns on, for example at 6:00 AM every weekday. Homeowners might also set yard lights to turn on at 7:00 PM and turn off at 7:00 AM. Bayesian modeling could help improve set time configurations.

With the implementation of computing configurations on the owner’s devices, consumers can apply some logic to their device’s operations. From the research conducted in this paper, an improvement in the detection of an occupant in the home was discovered by gathering the information from previous events and applying that information to a monitoring network to better improve the outcome. This practice can be applied to the owner’s devices, after a training period, to gather historical data. This historical data would then be added to the proper logic. The logic would then be applied to the owner’s devices.

For example, at what time does the homeowner turn on the coffee maker? What is the likelihood that the time is the same on the weekend or a weekday? Do they have a confidence level range when they want to turn on the coffee maker? They can answer these questions in a manner similar to questions answered about whether there was motion within a given room. They can then apply it to a network, which will create a prediction and operate the coffee machines or an alternative task. One of the advantages of the Bayesian network is its capability to apply other sensors within the network to help create a prediction. One example is the application of this prediction to the coffee maker to trigger an event. Predictive analysis could be applied with a motion sensor, allowing the coffee maker to operate based on defined parameters.

The use of Bayesian modeling and the advancement in smart home device hubs, which can monitor events and apply logic behind them, can allow individuals to further automate their lifestyles.

6.1.2 Doctors and Patients

Doctors and patients can both benefit from the Bayesian modeling. Doctors could implement behavioral data into their monitoring systems. Bayesian modeling must be carefully applied when used in healthcare, since the network depends on reliable data. However, with proper testing, a Bayesian model can help doctors create personal, complex logic for each patient. The use of several types of sensors could be implemented to monitor movements within the patient's location and alert the monitoring station of information that does not fit the patient's normal routine. These routines could be built from the motion sensors or cabinet sensors, within the patient's home.

These personal routines will be patient-dependent since the logic would be constructed from the patient movements. An example of such routines is a patient advised to start being active at 9:00 AM each day and open the door to head to the kitchen for their morning medication. If the patient has completed the appropriate training time period for the stated task, then the implementation of Bayesian modeling can help ensure the task is completed within its parameters. If the task is not completed, a notification method could be implemented.

In this system, the network may be designed to detect occupancy within the location. The opposite of occupancy is no occupancy. That sounds simple enough; however, what is meant by this is that if during a certain timeframe the occupants have a 25% chance of being home, or of completing a set task, then one can say with 75%

certainty that they are also not expected to be home or not to have completed the set task. This example shows that this same method can be used to detect patients' behavior probabilities within their location and to monitor activities within the healthcare system to ensure strong, effective healthcare and improved patient care outcomes.

6.1.3 Elder Care

With elder care stakeholders, the Bayesian modeling could help enhance the monitoring of the elderly. Depending on the types of sensors and the placement of sensors within the monitored space, several practical applications could be used. Some of these applications might include observing motion within the space if the elderly person is still mobile, monitoring location occupancy, and watching for specific events such as the opening of cabinet doors, lighting changes, temperature changes, and audio changes. These sensors can help monitor specific triggering events, which can help register the normal activities of the elderly.

One example of this application is to use cabinet or drawer sensors to register triggering events, if this is the place for medications that need to be taken. The network can be constructed to check for normalcy for when the cabinet door is typically opening and closing. Then, the monitoring system could move to the next step to determine whether an event is normal; if not, the system could send a notification to the elder, the caregiver and/or the family to advise them of the atypical action.

The growing elderly population desires more options to allow them to remain in their homes and age in their current places of residence. This phenomenon is driving the need for more adaptable monitoring systems. This Bayesian modeling, in elder care, supports peoples' desires to remain in their own homes and lead safer lives as they age.

6.1.4 Utility Automation

Utility automation is becoming a popular trend within the smart home scenario. There are devices that customers can use to set air conditioning and heating units to certain temperatures within their homes. The temperatures for their utilities can be set and programmed to change over various timeframes throughout the day. In order for utility automation stakeholders to continue to create benefits for their customers, they will need to improve their computational power, and the Bayesian network can allow for some of this improvement by including probabilities in the processing.

Most of these learning devices and smart devices still require the consumer to set the base on their devices, and they then rely on this information to continue forward through time. However, in this research, multiple types of sensors are needed to have more accurate implementation. This principle of applying more than one type of sensor can significantly improve outcomes. This method could be used for utility automation such as heating or air conditioning, operating water supplies, and electrical usage monitoring, rather than having the user continue to manually set the devices for every application.

The results of the applied Bayesian modeling can help reduce manual input into many types of systems. Once the manual application is removed, their reliability could be improved, since the use of the Bayesian method can be adaptive for a forward-rolling timeframe and based on updated historical events. Applying a rolling timeframe means having the historical data in only four-week increments that are constantly updated, like how activities were monitored within this study research. This solution is more practical than setting system constant values, due to ever-changing behavior over time. It is

important to keep up-to-date probability within the modeling to provide the best environment for consumers.

The desire to detect occupancy accurately is a main goal for many stakeholders: 1) homeowners, in order to improve the efficiency and function in their homes; 2) healthcare professionals, to help provide better care at home for patients; 3) eldercare, to enable them to safely live longer within their homes; and 4) security industry, allowing them to detect events more precisely. These goals are the core motivators that will push the development of home occupancy systems into the next stages of predictive analysis, where there will be a need to use several types of detection methods. Some of these methods will use additional sensors, while others will use additional algorithms to improve the sensor data generated into valuable event occurrences, such as occupancy.

6.2 Future Research

Two key points should be investigated to identify areas to address in attempts to improve the effectiveness of the Bayesian method. These points include changing the conditional tables and trying to use other forms of sensors. This study created the conditional tables by determining the values using experience and previous studies. It would be interesting to develop a method where the conditional tables could be generated by other means. The new conditional table values might be a combination of additional relationships or be generated by running a string of simulations to determine the best outcome. Future research could also involve experiments that use another form of sensor to help predict occupancy. This study was based on the use of PIR sensors, smartphone GPS signals, and a Bluetooth key fob presence data that was applied to improve the results. Additional forms of motion sensors were not tested. Two other forms of sensors

that might enhance the gathering of information are pixilated and infrared cameras. This study did not include these two types of cameras because the cost is greater, the PIR sensors were readily available, and primarily because this study aimed to focus on Bayesian modeling, not sensor technology. Additionally, there are several other types of presence sensors that might be acceptable to replace or enhance the GPS devices, such as door contacts and other Bluetooth-connected devices.

Our research of the Bayesian network could enhance the outcome of a simple PIR sensor network. This dissertation did not study additional forms of sensors or other methods to improve the conditional tables. It will take additional research to enhance and improve the Bayesian modeling for home occupancy detection. Such research is vital for the enhancement of the smart home prediction methods currently in use.

6.3 Summary

In summary, this study was able to produce the answer to its research question: "When using a set of PIR sensors and applying the Bayesian method as an overlay, is this study able to improve the accuracy of presence detection within the system?" There was an improvement in the accuracy of prediction when using a set of PIR sensors and applying the Bayesian method. In Chapter 1 the hypothesis stated:

> H_1: When applying Bayesian modeling in conjunction with PIR sensors, the accuracy of presence detection will improve within a given location.
>
> H_0: When applying Bayesian modeling in conjunction with PIR sensors, the accuracy of presence detection will have no effect on the improvement within a given location.

This study was able to accept the hypothesis and reject the null hypothesis when applying the Bayesian modeling in conjunction with the PIR sensors. The accuracy of the presence-detection results did improve within the given location and for the current occupants. For the condominium, the correlation improved from 0.22 to 0.70; for one occupant (2018), the correlation improved from 0.30 to 0.71; for the two retirees', the correlation improved from 0.90 to 0.94; and for the one occupant (2019), the correlation improved from 0.41 to 0.82. The improvement in the correlation was due to the implementation of trained sensor data into a Bayesian network.

The results of the independent sensors showed that the ϕ correlation between detection and ground-truth for each location were the condominium at 0.22, one occupant (2018) at 0.30, two retirees' at 0.90, and one occupant (2019) at 0.41. These results show two interesting comparisons: 1) comparing the one occupant (2018) and one occupant (2019) showed that sensors spread throughout the space differently improved the correlation in the one occupant (2019) location, and 2) the two retirees who are home most of the day have a higher ϕ correlation compared to the other four locations. These higher correlation results should be studied further because the independent sensor method may be more practical in specific locations, as seen in the two retirees' location results where the occupants are at home longer. The limit to the independent sensors is their lack of confirmation to each of the other sensors where there is no second sensor that is required to confirm the sensor signal. Further investigation will need to be done to better understand how false signals can affect the results when reviewing the correlation over a week period.

The grouped sensors have this secondary confirmation by design. For each location, the correlation at the condominium is 0.15, one occupant (2018) is 0.23, two retirees' is 0.83, and one occupant (2019) is 0.27. These showed the same two interesting results as in the independent sensor method; the one occupant (2019) location had slight improvement over the one occupant (2018) and the two retirees' location had a higher correlation then the other locations. Grouped sensor has the limitation of sensor placement. This research was designed to place three sensors within a given space, however each location had several different room shapes and sizes which lead to some rooms having more sensor area coverage than others. This can be viewed with the one occupant (2019) and one occupant (2018), where there is an increase in correlation due to sensor placements.

The trained Bayesian hours' method has a slight correlation improvement over the grouped method. The results showed that the correlation for condominium is 0.16, one occupant (2018) is 0.24, two retirees' is 0.84, and one occupant (2019) is 0.34. The improved results are from the historical probability that was applied to the Bayesian network. The limitation is the premade assumptions that were used for the Bayesian network conditional tables. The assumptions may be able to be enhanced by running several scenarios within each location to identify the ideal conditional table assumptions.

The Bayesian presence method shows the largest improvement over all the other methods. The correlation for each location is, condominium is 0.70, one occupant (2018) is 0.71, two retirees' is 0.94, and one occupant (2019) is 0.82. Applying the location history probability to the Bayesian network increases the correlation to the condominium, one occupant (2018), two retirees' and the one occupant (2019). It is important to present

the two retirees' location did not improve as much, since they are normally home. The limitation to this method was having to rely on smartphone or Bluetooth presence detector connectivity during the research period. If the connectivity is poor, the occupancy data and the ground-truth data were not available.

The main limitation that affects all the methods discussed is the PIR sensor's lack of ability to detect occupants during the sleep cycles. The methods could not detect motion during that timeframe while the occupants were asleep, motionless, due to the construction and placement of the devices. The condominium had nearly zero movement during the evening; however, the other three locations had a fair amount of movement during the sleeping hours. This extra movement at the retiree's location helped increase the overall correlation, since movement was detected, allowing the system to help predict there was an occupant home during those hours. The use of the PIR sensors can be helpful for specific types of purposes, but for the detection of occupants, they were not very effective without presence detectors.

Adding another form of detection, namely of detecting the occupants during their sleep cycle, improves prediction results overall. This improvement with the presence detectors was seen in the results for all locations after adding the presence detector history. The primary added benefit was that the presence detectors had the ability to detect the occupant's presence during the sleeping cycle. In future studies, additional methods of detection could be investigated to detect occupancy movements during the sleep cycle.

The use of the Bayesian modeling adds significant capabilities when one uses sensors to predict events, and the accuracy of its use improves with a second form of

detection added to the system. Accurate and appropriate sensors are needed to detect all the events required for system design. Without these sensors, lower correlations will occur due to the construction of certain types of sensors. PIR sensors were unable to detect motion within the space during the sleep cycle of the occupants. Once this issue was overcome with Bayesian modeling and the presence-detection events applied, the network became more accurate and was more immune to the limitations in the test.

WORKS CITED

Al-Shaqi, R., Mourshed, M., & Rezgui, Y. (2016). Progress in ambient assisted systems for independent living by the elderly. *SpringerPlus*, *5*(1), 1-20.

Berger, M., & Armitage, A. (2010). *Room Occupancy Measurement Using Low-Resolution Infrared Cameras*. Retrieved from http://www5.informatik.uni-erlangen.de/Forschung/Publikationen/2010/Berger10-ROM.pdf

BOSCH. (n.d.). ISW-ZPR1-WP13 Zigbee PIR w/Pet Immunity. Retrieved September 17, 2018, from https://us.boschsecurity.com/en/products/intrusionalarmsystems/premiseswireless/zigbee/zigbee_products_56458

Brownlee, J. (2016, April 18). Learning Vector Quantization for Machine Learning. Retrieved August 1, 2018, from https://machinelearningmastery.com/learning-vector-quantization-for-machine-learning/

Charniak, E. (1991). Bayesian networks without tears. *AI magazine*, *12*(4), 50-63.

Co2meter. (n.d.). K30 10,000ppm CO2 Sensor. Retrieved from https://www.co2meter.com/collections/co2-sensors/products/k-30-co2-sensor-module

Corporate Finance Institute. (n.d.). Bagging (Bootstrap Aggregation). Retrieved from https://corporatefinanceinstitute.com/resources/knowledge/other/bagging-bootstrap-aggregation/

Demir, E., Köseoğlu, E., Sokullu, R., & Şeker, B. (2017). Smart Home Assistant for Ambient Assisted Living of Elderly People with Dementia. *Procedia Computer Science*, *113*, 609-614.

Digikey. (2012). Zilog ZEPIR0AAS02MODG. Retrieved from https://www.digikey.com/product-detail/en/ZEPIR0AAS02MODG/269-2513-ND/2471949

Eisa, S., & Moreira, A. (2017). A behaviour monitoring system (BMS) for ambient assisted living. *Sensors*, *17*(9), 1-24.

Fenton, N., & Neil, M. (2013). *Risk Assessment and Decision Analysis with Bayesian Networks.* Boca Raton, FL: CRC.

Gokalp, H., & Clarke, M. (2013). Monitoring activities of daily living of the elderly and the potential for its use in telecare and telehealth: A review. *TELEMEDICINE and e-HEALTH*, *19*(12), 910-923.

Gross, K. (2018, February 19). Ultrasonic Sensors: Advantages and Limitations. Retrieved from https://www.maxbotix.com/articles/advantages-limitations-ultrasonic-sensors.htm/

Guo, X. (2007). *Occupancy sensor networks for improved lighting system control*. Retrieved from https://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=11561&context=dissertations

Guo, X., Tiller, D. K., Henze, G. P., & Waters, C. E. (2010). The performance of occupancy-based lighting control systems: A review. *Lighting Research & Technology*, *42*(4), 415-431.

Hammoud, A., Deriaz, M., & Konstantas, D. (2017). UltraSense: A Self-Calibrating Ultrasound-Based Room Occupancy Sensing System. *Procedia Computer Science, 109*, 75-83.

Heo, Y. (2011). *Bayesian Calibration of Building Energy Models for Retrofit Decision-Making under Uncertainty* (Dissertation). Retrieved from https://smartech.gatech.edu/bitstream/handle/1853/42878/Heo_Yeonsook_201112_phd.pdf

Howard, J., & Hoff, W. (2013). Forecasting building occupancy using sensor network data. In *Proceedings of the 2nd International Workshop on Big Data, Streams and*

Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications (pp. 87-94). New York, NY: ACM.

Hutchins, J., Ihler, A., & Smyth, P. (2007). Modeling count data from multiple sensors: a building occupancy model. In *2007 2nd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing* (pp. 241-244). New York, NY; IEEE. doi:10.1109/CAMSAP.2007.4498010

Huynh, T., Fritz, M., & Schiele, B. (2008). Discovery of activity patterns using topic models. In *Proceedings of the 10th international conference on Ubiquitous computing* (pp. 10-19). Berlin, Germany: Springer.

Jin, M., Bekiaris-Liberis, N., Weekly, K., Spanos, C., & Bayen, A. (2015). *Sensing by Proxy: Occupancy Detection Based on Indoor CO2 Concentration*. Retrieved from https://www.researchgate.net/profile/Carlos_Westphall/publication/280125651_UBICOMM_2015_The_Ninth_International_Conference_on_Mobile_Ubiquitous_Computing_Systems_Services_and_Technologies/links/55ab3d0408ae481aa7fbd294.pdf#page=15

Keller, H. J. (2000). 30 Years of Passive Infrared Motion Detectors: A Technology Review. Retrieved from https://pdfs.semanticscholar.org/f395/ffc5cda453d7aea8b36d79ce1bf3daa852d2.pdf

Khan, G., Siddiqi, A., Ghani, U., & Qayyum, S. (2019). Geometric positions and Optical Flow based on Emotion Detection using MLP and Reduced Dimensions. In *The Institution of Engineering and Technology*.

Kim, S. H., Moon, H. J., & Yoon, Y. R. (2017). Improved occupancy detection accuracy using PIR and door sensors. In *Building Simulations* (pp. 2753-2758). Las Cruces, NM: IBPSA.

Kontokosta, C. E., & Tull, C. (2017). A data-driven predictive model of city-scale energy use in buildings. *Applied energy*, *197*, 303-317. http://dx.doi.org/10.1016/j.apenergy.2017.04.005

Labonnote, N., & Høyland, K. (2017). Smart home technologies that support independent living: Challenges and opportunities for the building industry–a systematic mapping study. *Intelligent Buildings International*, *9*(1), 40-63.

LaFramboise, K. (n.d.). Krlaframboise/SmartThings [Software]. Retrieved September 21, 2018, from https://github.com/krlaframboise/SmartThings#simple-event-logger

Le, J. (2018, January 20). A Tour of The Top 10 Algorithms for Machine Learning Newbies. Retrieved August 1, 2018, from https://builtin.com/data-science/tour-top-10-algorithms-machine-learning-newbies

MAT, S. D. (n.d.). Correlation Coefficient. Retrieved from https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/correlation-coefficient-formula/

Mathas, C. (2012, June 28). Sensing Motion with Passive Infrared (PIR) Sensor. Retrieved from https://www.digikey.com/en/articles/techzone/2012/jun/sensing-motion-with-passive-infrared-pir-sensors

McKee, M., & Anddriyas, A (2015). Development of a Bayesian Belief Network Model Framework for Analyzing Farmer's Irrigation Behavior. In 2015 *Journal of Agricultural Science*, *7(7)*

Mulia, M. T., Supangkat, S. H., & Hariyanto, N. (2017, September). A review on building occupancy estimation methods. In *2017 International Conference on ICT For Smart Society (ICISS)* (pp. 1-7). New York, NY: IEEE.

Murata. (2008). *Ultrasonic Sensor.* Retrieved from 2018 http://www.symmetron.ru/suppliers/murata/files/pdf/murata/ultrasonic-sensors.pdf

Occupancy. (n.d.). In *Merriam-Webster*. Retrieved from https://www.merriam-webster.com/dictionary/occupancy

Ribo, M., & Pinz, A. (2001). A comparison of three uncertainty calculi for building sonar-based occupancy grids. *Robotics and autonomous systems*, *35*(3-4), 201-209.

Romeu, J. L. (2004). Understanding Series and Parallel Systems Reliability. *Selected Topics in Assurance Related Technologies, 11*(5), 1-8. Retrieved from https://www.dsiac.org/sites/default/files/reference-documents/s%26psysrel.pdf

Schreiber, J. (2018). Pomegranate: Fast and flexible probabilistic modeling in python. *The Journal of Machine Learning Research*, *18*(1), 5992-5997.

SmartThings. (n.d.). Home. Retrieved September 20, 2018, from https://www.smartthings.com/

Tarzia, S. P., Dick, R. P., Dinda, P. A., & Memik, G. (2009). Sonar-based measurement of user presence and attention. In *Proceedings of the 11th international conference on Ubiquitous computing* (pp. 89-92). New York, NY: ACM.

Tijani, K., Ngo, D. Q., Ploix, S., Haas, B., & Dugdale, J. (2016). *Dynamic Bayesian networks to simulate occupant behaviors in office buildings related to indoor air quality*. Geraadpleegd van https://pdfs.semanticscholar.org/7c05/d4ca3ae5f6f8a04f9ca216225b9997e33206.pdf

Vafeiadis, T., Zikos, S., Stavropoulos, G., Ioannidis, D., Krinidis, S., Tzovaras, D., & Moustakas, K. (2017, October). Machine learning based occupancy detection via the use of smart meters. In *2017 International Symposium on Computer Science and Intelligent Controls (ISCSIC)* (pp. 6-12). New York, NY: IEEE.

Wang, Z., Wang, Y., & Srinivasan, R. S. (2018). A novel ensemble learning approach to support building energy use prediction. *Energy and Buildings*, *159*, 109-122.

Weber, P., Medina-Oliva, G., Simon, C., & Iung, B. (2012). Overview on Bayesian networks applications for dependability, risk analysis and maintenance areas. *Engineering Applications of Artificial Intelligence*, *25*(4), 671-682.

Wu, B., & Nevatia, R. (2005). Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *Tenth IEEE International Conference on Computer Vision* (Vol. 1, pp. 90-97). New York, NY: IEEE .

Wu, Z., Li, N., Peng, J., Cui, H., Liu, P., Li, H., & Li, X. (2018). Using an ensemble machine learning methodology-Bagging to predict occupants' thermal comfort in buildings. *Energy and Buildings*, *173*, 117-127.