UNIVERSITY OF CALGARY

ENGO 500: GIS and Land Tenure #2

Literature Review: Location Aware Smart Shelves

By:	Jeremy Steward
	Kathleen Ang
	Ben Trodd
	Harshini Nanduri
	Alexandra Cummins
Supervisor:	Dr. Steve Liang

DEPARTMENT OF GEOMATICS ENGINEERING

SCHULICH SCHOOL OF ENGINEERING

2013-11-08

Table of Contents

Introduction

Purpose

Information relating to customer frequency, interest, and product popularity is essential to many grocery store owners [1]. Analytics of this kind are useful to store owners as they make strategic marketing decisions, provided data is collected in an efficient and understandable manner. Issues that must be addressed comprise how many people come and go every day, their purchase decisions and history, and which products are sold at different times of year. Much of this data can be gathered by observing the areas of high and low customer traffic within the store, and correlating that information with product locations throughout the store, and to the objects which have been purchased. By gathering this data effectively, retail management can potentially arrange and analyze the store configuration over time, which can be an effective tool in maximizing profit and customer satisfaction [1].

In addition to knowing traffic patterns, store owners also value a well-stocked and orderly appearance. To satisfy this condition, grocery store owners will often ask their employees to *face* (or *block*) the shelves in their store's aisles. Facing a shelf involves moving all products on the shelf towards the front of the shelf (known as the face) and arranging them in a neat and orderly fashion. This allows the customers to easily see all available products and allows them to reach them without trouble as well. Therefore, keeping shelves faced is also important for grocery stores who wish to maintain their image.

This project will focus on consolidating the two above problems in the form of a "smart shelf" solution which can both track customer movement throughout the store, and likewise provide a means to report on the status of an item's stock (faced or not faced). To this end, a sensor-network in the form of the *Internet of Things* (IoT) will be employed as a novel solution to these problems. Thus, the purpose of this literature review is to not only understand the perspectives and needs of store managers, but also to study the current models which are being created to address the similar challenges, and to improve with regards to existing solutions. Upon understanding these methods and challenges, a specification for a "smart shelf" system is proposed. The proposed system will be designed to empower non-expert users from general grocers who stock shelves, to upper management, to understand the needs of individual retail chains as well as develop models and data that can be applied to both the store layout, and likewise to marketing tactics in order to maximize their profit and customer's satisfaction.

Background Information

The issues stated in the above section have been addressed before, by many different parties and in many different ways. All of them sit within the field known as the Internet of Things (IoT). The Internet of Things is defined as a relationship between tangible objects and their virtual counterparts, achieved by equipping objects with sensors. This provides them with the ability to communicate with the web without human interaction.

Internet of Things

There exists a broad spectrum of IoT applications, many of which are proprietary. To a user, each of these objects must be controlled by their respective application or interface individually. This becomes cumbersome as the number of internet-aware devices one might use increases. Two ways to approach this problem are: to create standards which are widely accepted and can be considered as IoT protocol, similar to how HTTP transformed the web, and the other way is a more decentralized approach, where many separate, unique libraries which accommodate for the diverse number of applications are written. With the latter approach, each object's library may be potentially written in multiple, various languages, with a unique API and programming style. This second approach has been adopted by an open source community called OpenRemote, which is designed to work with a variety of existing sensors already on the market. The idea is that users can design their own interface and integrate whichever sensors they want, regardless of that sensor's own protocol. Currently, OpenRemote support protocols from more than 25 different applications, such as PhilipsHue, FreeBox and Samsung Smart TV [2].

However, while this decentralized approach makes it simple to integrate various sensors for any single solution, it creates a problem whereupon programmers, hackers, and entrepreneurs must learn several different APIs for each of the devices they wish to use. This is in stark contrast to the former solution to integrating multiple sensors with the Internet of Things, where a common standard protocol can be used to access any and all devices. This means that regardless of the objects used, a single interface makes all of them commonly accessible. Such a standard has been put in draft by the Open Geospatial Consortium (OGC) [3], and is awaiting finalization as of this year. Since this method is vastly preferred for its simplicity over solutions such as OpenRemote, this project will attempt to serve as a means to proliferate and exemplify the use of the OGC IoT standard protocol.

Existing Smart Shelf Technology

When considering the problems stated previously, namely that of shelf facing and customer tracking, existing products that address the same issues we are trying to solve came to mind. Due to the requirements of our project scope and the limitations in budget and knowledge that comes with being students, our final product will certainly be designed differently from the existing products studied below. Nevertheless, they provided insight and guidance in laying out the concepts of our design, more notably with regards to software.

Many existing solutions rely on RFID sensors [4]. RFID, or radio frequency identification, is a form of automatic identification that falls under the same category as barcode scanning and voice recognition. RFID is becoming a key technology in vast computing networks, as it provides many advantages over traditional means, such as the aforementioned barcode scanning [5]. The most common implementation of RFID involves storing a serial number on a microchip which can then transmit this number to an automatic reader via an antenna. The reader can then convert the radio waves into digital information [6]. In this way, the reader is able to gather information that is unique to each separate item that has been tagged.

Shelves equipped with RFID technology have the ability to provide information to store owners about the products which are placed on it [4]. By scanning the product, the shelf is able to identify when an item is picked up or placed, and may also track an item through the store by recording which shelf it is placed upon. The RFID scan also provides data such as the expiry date, which the shelf can store and set up a notification if the product has not been bought before that date. Furthermore, it has likewise been shown that this concept is not strictly limited to supermarket retailers, but can be expanded into other industries such as construction or site management in tracking and maintaining appropriate stock for work to be completed [7]. RFID tracking has become very popular in the world of retail, and very few alternative approaches have been used in an attempt to solve similar problems.

Another solution proposed is the NeWave Sensor [8]. This item was developed to be used primarily by store owners; however, it does not focus on improving customer satisfaction, but rather on security and monitoring of product in-store. The technology involves an antenna that can detect change within a certain area, and in this way it is able to indirectly track the items placed within that area / on the shelf. Since it relies only on a sensory antenna, it can not tag items like the RFID can. It's functions allow data to be collected on merchandise availability, which may indicate high or low stock or even an emergency, but cannot directly produce positional information in regards to product placement on a shelf (i.e. facing).

The advantages of sensor-based solutions can primarily be seen in the failings of these existing solutions. While RFID tags dominate in popularity, especially in regards to retail product tracking, some distinct disadvantages associated with the technique are:

1. RFID signals can be potentially noisy and may require objects to perfectly oriented with respect to the sensor in order to improve the overall accuracy [9].
2. Unlike direct sensors, or remote sensors, direct information cannot be obtained without computational overhead. For example, it was shown that the tracking the storage of construction materials required additional computation such as the gradient descent method [7]. On the other hand, weight sensors and magnetometers can detect item positions (on or off the shelf) directly, which provides a much quicker solution.
3. RFID has been in use since the 1970s [10], but it has not been widely used until recently due to its high cost. Tags are now cheaper to produce, but in a supermarket context, individual products would require a tag on every item that was to be placed on the shelf. Some companies may be able to fund this, such as in the application mentioned above (RFID Arena Smart Shelf), and in return have access to interesting, product specific information, but the cost-barrier-to-entry may be too high for smaller competitors. Unlike RFID chips, which must be replaced per product, sensor networks provide a unique advantage in that they are a flat cost, and generally don't need to be replaced often.
4. Existing RFID based systems, in contrast to examples involving sensor networks, do not scale additively, and do not allow current solutions to be built upon and improved in an agile manner.
5. RFID tags provide little insight into customer motion throughout the store. At best they can identify product trends, but no spatial or temporal data can be garnered about customers using RFID chips, only about products themselves.

For the purposes of this project, the major concerns regarding RFID tags are numbers 2, 3, and 5 above. It is important to realise that while the above may appear to be beating a dead horse in terms of RFID applications, the selling point of our project is twofold: to determine the status of shelf facing, and providing analytics in the form of customer interest and time spent within each aisle (i.e. to determine the number of customers in a given aisle and how long they stay there). Moreover, as it would be difficult for the NeWave smart shelf to specify the position of the items on the shelf, it does not provide an efficient means to determine whether or not a shelf is faced. Finally, the NeWave solution is incapable of monitoring customer movements in detail, which makes it unsuitable for the problems we wish to solve.

Sensor networks, in contrast, offer a better solution to this problem. Unlike RFID sensors, sensor networks as used in IoT based applications can provide two primary advantages [11]:

1. Specialized sensors can be used to directly acquire information within given constraints. No additional computation is necessary for sensors to output data.
2. Sensor networks can be built in additively and are designed to scale [11], which reduces the economies-of-scale problem regarding the heavy cost of smart shelf implementations. Moreover, sensor networks provide a solution that can be additively upgraded as time progresses, reducing long-term cost as well.

For these reasons, and given that previous implementations of smart-shelves do not adequately address the problem we wish to solve, a multi-sensor approach is preferred for the implementation of our desired solution.

Retail Analytics

Currently, there are two main ways in which retailers collect data about traffic in stores: using video or using cell phone signals. These two ways have been researched and implemented because it typically requires little additional hardware setup. Many stores already have an existing set of closed circuit television (CCTV) cameras in their stores, and nowadays most customers carry cell phones with them while they shop. Leveraging the existing technology is a common strategy employed by several commercial companies and research groups. Some specific examples of both cases are discussed here.

As mentioned, many stores are already equipped video cameras throughout the store, in order to track suspicious behaviour and have resources for offline investigation if any theft occurs. However, having people monitoring surveillance cameras can be quite inefficient, as it requires a lot of time and some events (particularly in larger facilities) could go unnoticed. Hence, having some kind of automatic detection or monitoring systems can be useful. As developed and tested by [12], the video recordings can be augmented with various algorithms to help real-time monitoring of stores. For example, people can be tracked as they enter or leave the store, including what they carry with them when they enter. See Figure 5 below:

Figure 1: Colour density heat map of customer activity [12]

This is useful information to help with returns fraud, where people take new items and return them without having actually purchased them. In [13], video cameras are used to understand the shopping behaviour of customers. For example, they use various detection algorithms to determine if a shopper should be classified as goal-oriented or disoriented based on their walking speed. In addition, they detected facial expressions and interactions with objects.

Some commercial solutions are also available for gathering and communicating retail analytics. One commercial service is called ShopperTrak, which offers a variety of solutions for retailers to understand key questions about their consumers. Such questions include, “is my marketing effective?” and “which zones attract the most traffic?” [14]. This service again leverages video camera technology, which is undesirable, and uses mobile analytics from cellular phones, which creates privacy concerns amongst consumers [15].

Another commercial solution is Path Intelligence, which also has a variety services such as location usage analysis using interactive heat maps, signage effectiveness reports, flow management analysis and dwell time reports. All of these services help retailers understand their customers better. Path Intelligence uses the strength of cell phone signals in order to triangulate a location for each customer. While the locations are known, the identity of each consumer remains anonymous [16].

An example of Path Intelligence can be seen in Figure Error: Reference source not found, below, which shows how customers are tracked as they move throughout aisles within a store:

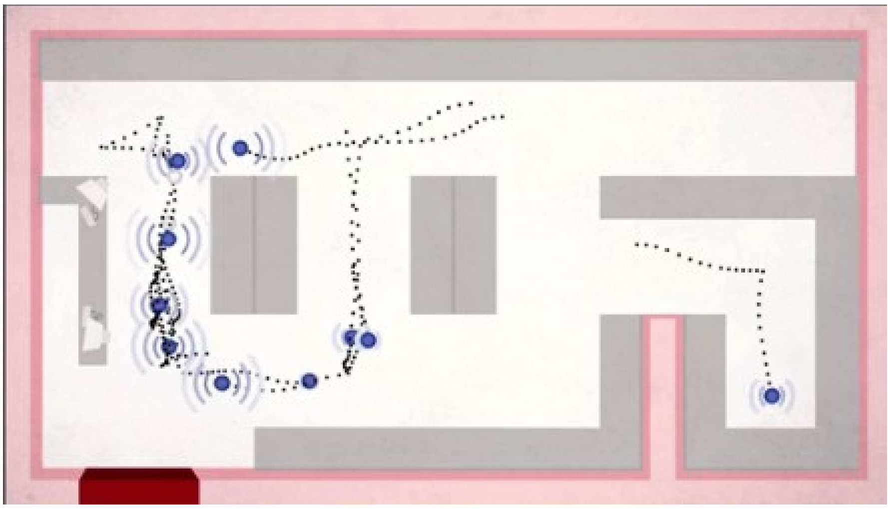

Figure 2: Illustration of path mapping using Path Intelligence [16]

Other retail analytics companies also use cell phones in order to track shopper movements, but they use MAC addresses instead of cell phone signals sent to cell towers. If a phone is Bluetooth or WiFi-enabled, it will broadcast its own MAC address in order to communicate with other devices. Based on detection of MAC addresses belonging to each device, their locations can be determined. Aggregate reports formed from this data provide insight into current waiting times at cash registers, optimize store location, and consumer shopping patterns [17].

Sensor Servers

In order to be able to communicate with different sensors simultaneously, a sensor server is typically used. There are several types of existing sensor servers which have been marketed commercially. Some of them are only able to interface with certain sensors, such as the PCW-SSRX Sensor Server made by SenSource [18]. This server communicates specifically with proprietary wireless sensors equipped with radio frequency transmitters, or wired sensors. From one or more servers, information can be communicated to a central PC, which can query the server using proprietary Server Manager Software or by simple ASCII commands.

Another sensor server has been created by Oracle, which is well-known for its database management systems. The Oracle Sensor Edge Server integrates data from different sensors, which includes RFID, temperature sensors, humidity sensors and others, as well as command/response-type equipment. It is the middle-tier component which communicates between these objects and applications [19].

Project Objectives

Aisle Management

Aisle management, a term which means changing the placement of products to increase traffic, sales, and profits is one method stores use to succeed in the competitive retail market. According to Larson [20], the value of moving a product's shelf space can be estimated by tracking customer traffic within a store. Working with this higher level statistic (treating the aisle as an entity) as opposed to a categorical approach, resource costs for analysis can be reduced. Dreze et al. [21] found that on average, a 4-6% sales increase could be found by optimizing product placement. Moving a product from the least ideal to the most ideal location could have an increase in sales of up to 60 percent. Additionally, creating a layout in which customers browse more isles is important to creating sales. A study by Coca-Cola [22] found shoppers travel and average of 41% of the aisles on a typical shopping trip. These findings prompted one of the goals of this project to be tracking customer movement through aisles. This would allow optimizations for store layout.

The aforementioned has shown that focusing on aisle management can be beneficial for improving a store's success. When products are grouped into different zones (aisles), these aisles can be monitored and analyzed in order to try to maximize the traffic in and profits from each of the zones. It was also found that if a greater number of people observed a certain product display, traffic increased in the aisle which had that product. Aisle length also has an impact, as it has been noted that consumers are less likely to go down an aisle that is too short or too long. While there are a number of principles which can be followed (for example, grouping products by type is more effective than grouping them alphabetically by their name) [20], stores may also find that some situations or trends are specific to the demographics and organization characteristic of their own store.

Further research, conducted by P. Chandon et al. (2009) [23], used eye tracking technology to measure attention towards items on supermarket shelves. The technology measured eye movements such as fixations and jumps. In complex settings, such as a supermarket shelf, eye fixations represent object identification, and the movement is a good indicator of the direction that the eye is looking. This study suggests that not all shelf locations attract equal attention, and that the items located near the center of two shelf displays were noted more often [23]. It is inferred from this that an increase in shelf space, or an increased number of facings, draws more attention. The report concludes that the effects of increasing the facing of the display area directly and positively impacts the profit made by the store, making it an important aspect for all store owners to consider.

Shelf Facing

The facing of a product refers to the location and amount of shelf space is allocated to it. According to research by Chandon et al. [23], eye movement studies, shopper surveys, and field experiments have all confirmed that large increases in shelf space increase sales even when the price and location of the product stay the same. A logical extension of this is that a product's facing will not be optimal after stocks begin to be depleted by shoppers and the product does not fill the front of the shelf properly. To combat this, a common task for store clerks is to pull stock from the back of the shelf forward to create the illusion that the shelf is full. This is essentially bringing the facing of the product back to its allocated space. By tracking which products need to be faced, management and store owners can see areas which need attention at a glance without needing to travel the store personally. This data would also indirectly show popularity and out of stock problems if a clerk had faced an aisle but products are shown as missing by the sensors. An example of shelf facing can be seen in Figure 3 below:

Figure 3: Shelf facing of different sizes according to brand [23]

Such data could be useful in other applications as well, such as one that makes use of detailed models to estimate stock levels. These statistical methods use historic and current data such as order/delivery numbers, sales averages, volatility, shelf space and point of sale monitoring [24]. This type of approach is seen in systems such as KSS Retail's Heartbeat [25].

Stores often arrange their stock as specified in a planogram, which acts as a map for their products on their shelves. However, planning out how to arrange the products is one matter – ensuring that products are in the correct locations and maintain an organized appearance is another matter. This task is often very labour-intensive and time-consuming. In order to address this problem, Intel has developed an intelligent shelf compliance solution. The solution is a robot named AndyVision, which navigates the aisles of the store using sonar technology and uses a Microsoft Kinect as eyes to check the shelves [26]. The collected images can wirelessly alert staff when stock is low and if items are disorganized or misplaced. These images, once processed, can also help staff identify product placement issues (e.g. if a product has been allocated too much or too little space). While the research project prototype was a robot, a typical retail deployment might consider using existing cameras or installing miniature cameras on the sides of shopping carts, as a robot might inconvenience shoppers during opening hours [26]. Unfortunately, both the cost and computational complexity of computer-vision based solutions do not fit into the desired model we wish to create.

The main goal of monitoring shelf facing is to be able to alert staff as to whether or not a shelf is faced, so that if it is not, an employee can remedy the situation. In order to reduce some of this time-consuming work, stores sometimes use shelves which have tracks, dividers and springs to automatically organize products, pushing them to the front of the shelf. These are available in different sizes to accommodate different types of stock [27]. An example of such a system can be seen in Figure 4, which shows a self-facing shelf using such a product:

Figure 4: Example of automatically-faced shelves [27]

Such systems, while automatically facing shelf products, do not work if there is no product in stock.

In such a scenario, one proposed solution is to integrate sensors in order to make the object *aware* of such situations. Such shelves simplify the facing problem significantly as they reduce the problem from “is the shelf faced?” to “does the shelf have any product at all?” This is important as it can contribute to a very large cost saving factor in developing a solution to this problem.

Hardware Specifications

The purpose of this project is to create a smart shelf where multiple sensors are connected to determine whether the shelf is faced or not, and to detect the behavior of a customer in the store. The sensors require a connection to a micro-controller, such as the Netduino [28], Arduino [29], or a RaspberryPi [30]. This is a necessary part of the project, as the multi-sensor network we wish to create to meet our objectives needs some way to interface with the internet, in order to join the Internet of Things. A summary of the technical specifications for these boards can be found in Tables 1 through 3 below:

Table 1: Netduino technical specifications [28]

Parameter	Value
Processor	Atmel 32-bit micro-controller
Speed	48 MHz, ARM7
Storage	128 kB
Memory	60 kB
Network	Ethernet
Other	Requires .NET Framework (Microsoft)

Table 2: Arduino technical specifications [29]

Parameter	Value
Processor	ATmega328 8-bit micro-controller
Speed	16 MHz
Memory (Including Storage)	32 kB
Network	No
Other	Requires additional components for internet connectivity

Table 3: RaspberryPi technical specifications [30]

Parameter	Value
Processor	Broadcom BCM2835 32-bit processor
Speed	700 MHz, ARM7
Storage	~4GB depending on SD card and expandable storage (recommended storage reqs.)
Memory	256 or 512 MiB
Network	Ethernet, supports USB Wifi
Other	Can support multiple programming languages, and a full GNU/Linux operating system.

Because of the vast difference in technical ability between the Raspberry Pi and it's competitors, it stands out above the rest. Moreover, the SensorWeb Lab at the University of Calgary owns a couple of these boards, which makes them ready and available for prototyping. Therefore, for the purposes of this project, the Raspberry Pi is vastly preferred.

One of the options considered for this project's implementation of a smart shelf are the aforementioned spring load racks (see Figure 4). Spring loads (also known as product pusher racks or shelf facing racks) are shelving racks that are spring loaded and they automatically push the product towards the front of the display, guaranteeing a full, well organized shelving system [31].

Advantages for using spring loads (product pusher racks) are [31]:

- Maintains consistent facing and organization of products
- Guarantees optimal use of shelf space and inventory
- Increases sales potential at the point of merchandise

For this project, we likewise want to use a multi-sensor setup that can interface with our micro-controller board in order to make our sensor network “smart” or “aware.” Some proposed sensors include light sensors, range sensors, and magnetometers amongst others. While the exact sensors required to solve the problem have not yet been decided upon, the following specifications are entirely necessary for the project to reach fruition.

Functional Specifications

The following functional specifications need to be met in order to deliver a product that allows our sensor network to collect data unobtrusively with respect to regular store activities:

1. The proposed shelf should be able to tell when a product is no longer faced. For example, in the case of spring load racks, magnetometers could be attached to the back of the spring loaded wall of the rack, and magnets at the face of the shelf could potentially be fixed such that the magnetometers have specific readings when the shelf is unfaced.
2. The proposed shelf should be able to determine if somebody is standing in-between the length-wise area of the shelves, such that it is possible to know whether a customer is standing between the two edges of the shelf-space. This makes it possible to know where in an aisle a customer might be. An example might be to use range or motion sensors at each edge of the shelf. If a customer crosses one side but not the other, it is possible to tell that they are in between both sensors.
3. The shelf should be sized similarly to existing store shelves, and should not provide any lower limits on the amount of stock that can be placed on a shelf. In this same way, the shelf should not obstruct or otherwise place limits on the amount of space between aisles compared to that of existing shelves.
4. The final device should not break without internet access, but should expect reliable internet infrastructure in place in order to perform.

Non-functional Specifications

The following non-functional specifications should be met in order for the hardware to perform as intended:

1. The final product should conform to the Open Geospatial Consortium standard for

IoT SWG [3]. Specifically, the data output from the hardware should be interoperable with a conformant database built from this standard.

2. Relatively stable internet access should be available (either through a wired or wireless connection) so that the shelf can update its status as frequently as possible.
3. Power to the shelf should be made available. For a single shelf consisting of one Raspberry Pi controller, a 700 mA power source is necessary to operate [30]

Software Specifications

For any project involving software, it is important to identify our goals and to define the specifications in advance. As mentioned in previous sections, we found that there is a clear advantage for store owners and retailers who wish to know more about how their store is faced, as well as the distribution and pattern of customer positions within the retail outlet.

Functional Specifications

The following functional specifications need to be met in order to deliver a product that allows non-professionals to interact with our proposed system:

1. The ability to view shelf facings in real time, and determine where products in the store need to be restocked.
2. The ability to track customers entering and exiting each respective aisle, and determine how long each customer stayed in that section of the store.
3. The ability to generate some basic analytics based on data from the sensors. This can include items such as determining hotspots or busy times around the store, analyzing which parts of the store are more popular (most shelves empty), and other organizational information, such as busy times within the store, or average time left unfaced.

These specifications, if achieved, will result in a product that produces value for non-professionals in a managerial or aisle-management context within a retail organization.

Non-functional Specifications

Although there are many things to consider when discussing the overall functionality of our end software, it is important to also note some of the non-functional requirements, as follows:

1. The system should be painless and simple to introduce. Specifically, it should not require the user to do anything beyond setting up the hardware. For this reason, a central website that distributes management of multiple entities is preferred to a more decentralized system, where entities would set up their own specialized hardware for their application.
2. The system should be secure and prevent unauthorized entry. This is to protect consumers from privacy concerns, as well as to protect both managers and the data server from malicious entities.
3. The system should be robust enough that events such as power loss or switching products between shelves should not produce any significant over head to the system.
4. The system should be easy to manage and should provide managers with the ability to readily pull up any information from a given store or branch that they might

choose.

5. The system should have received an extensive amount of testing, preferably upwards of 30 hours of actual use testing and bug-fixing. As the project expands the test requirements will also be expanded to maximize ease-of-use of the final product.
6. All of the software should operate using the OGC IoT SWG standard [3], and should be published as free, open-source software for all people to tinker with and improve upon.

These specifications are particularly necessary in that it will be important to establish ease-of-use in order to market the final product. If the system is painless and simple to introduce, there is a more likely chance that such a system will establish itself in the market. This is often referred to as the technological acceptance model [32], and makes a strong argument for ease-of-use over initial functionality. Specifically, it argues that system adoption is more likely when systems are designed in a fashion that emphasizes ease-of-use over sheer feature count. This also factors in to our 3rd, 4th, and 5th non-functional specifications, because we find that they represent a measure of *ease-of-use* in slightly different ways. Therefore, in order to follow the technological acceptance model, the initial product will focus more strictly on the ease-of-use problem before regarding feature addition.

With regards to information security, the website will enforce a strict client-server authentication model, where specific shelves will be attached to different accounts based on universally unique identifiers (UUIDs). The specific research for this is not fleshed out here, as many common development frameworks are bundled with a sufficient client-authentication system, and will be largely implementation dependent. However, it should be noted that for the majority of the website development proposed by this project, it is likely that client-side computations will be necessary, as the group has not been able to secure any hardware with the capability of performing server-side computations and storage. Therefore, modern technologies such as HTML5, CSS, and Javascript will be necessary in order to allow us to perform computation of database results and analytics, as well as allow us to use some form of local storage for result caching.

Methodology

A large amount of research has gone into studying ways to increase sales in retail stores. Within this large area of advertising, store layout, pricing, loyalty programs, and others, we have decided to focus on two main goals for the project. The first is aisle management, and the second is product facing. To accomplish the two metric tracking goals detailed above the group will be split into two teams. The first team will be focused on the hardware configuration and sensor interfacing while the second team will focus on developing the necessary software and website.

Moreover, the team has chosen an incremental approach in order to create a product capable of shelf face detection and customer-traffic tracking. This type of development is a combination of waterfall and iterative development methodologies [33]. These methodologies are combined by adding an iterative prototyping period after the initial investigation and requirements definition are completed in the waterfall fashion. An example of this type of work-flow can be seen in Figure 5, below:

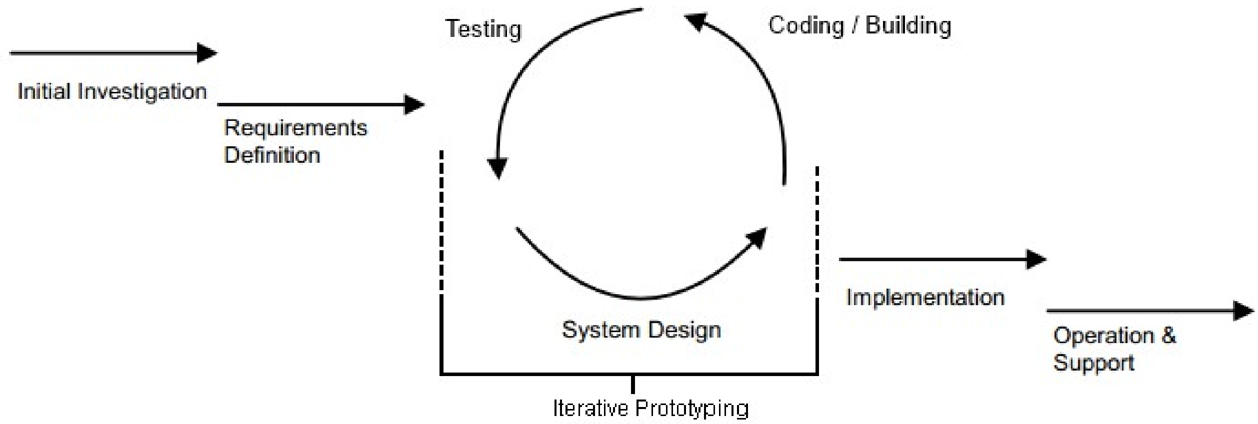

Figure 5: Flowchart diagram of iterative development process / work-flow (edited) [33]

This strengths of this type of development are [33]:

- The project is maintained with written documentation requiring approval/sign-off at designated milestones.
- Stakeholders can be given evidence of progress
- Gradual implementation allows evaluation of each incremental change
- Well suited to projects where requirements may change due to new knowledge, technology or expectations
- Well suited to leading-edge applications

However, despite the many advantages listed above, the following pitfalls that must be avoided with this methodology include:

- Difficult components may be ignored until late in the project time-frame
- Well defined interfaces are crucial since not all components of the project will progress at the same rate

Using the above team-based iterative methodologies, each of the two teams will attempt to complete the responsibilities outlined in the following sections.

Hardware based methodology

The hardware and interfacing team's responsibilities can be broken down into to the following tangible tasks:

1. Determine which sensors & micro-controller are most appropriate for the chosen specifications
2. Design the proposed sensor configuration & initial testing
3. Develop interface to send data to the server
4. Construction of prototype
5. Testing of prototype
6. Improvement to prototype (interfacing or hardware configuration)

Task 1: Determination of sensors & micro-controller

The sensor array being developed for this application is focused on embracing the IoT mantra as a main design influence. This means that the sensor network should include a number of sensors that are networked and accessible in a way that allows a full interoperability of interconnected devices..., providing them with an always higher degree of smartness by enabling their adaptation and autonomous behavior, while guaranteeing trust, privacy, and security" [34]. The required sensors will be decided upon based on IoT values, OGC standards, the technical specifications detailed by the team, and the goals of increasing sales in a retail environment.

The Sensor Web Research group at the University of Calgary [3] will be the first point of entry into deciding what sensors and micro-controllers will be used after a period of hands on experimentation with their resources. Some of these resources may be leveraged for the duration of the project. It is likely that not all of the required components for the project will be available in which case these will need to be acquired via an online vendor. A broad set of sensors may be tested for an initial investigation before the sensors for the final configuration is ordered.

Task 2: Design the proposed sensor configuration & initial testing

With the technical specifications in place, a sensor configuration will be developed to meet these design requirements. The main factors to be considered are cost, effectiveness, durability, robustness, and the protection of privacy.

To meet the cost requirements, different arrangements of sensors should be examined to find the best configuration of sensors and micro-controllers. With the cost of a micro-controller board being as much or more than most sensors, a design should emphasize on providing the needed detection methods for as much area as possible per micro-controller. Varying costs of sensors such as infrared ranging, magnetometers, luminance sensors, etc, may allow different combinations to achieve the same end goal.

To be effective, sensors need to be mounted in the optimal location. Logical mounting points for customer traffic tracking depend on the type of sensor used. A more naïve solution would be to track movement past certain gates, in which case sensors at the entrances and exits to aisles would count how many customers travel through the aisle, and potentially time spent in an aisle. Adding additional sensors in the middle of the aisle could also be beneficial to increase the resolution of the statistics gathered. Another option is image based soft-biometrics, which is more expensive both in terms of price and computationally. This type of tracking uses descriptors such as height and clothing color to track people in images, even between separate cameras [1]. Data from either method will be useful in correlating the length of the aisle, the products contained in it, and the time spent in a section. Shoppers may avoid going down aisles that are too short or too long [3]. In terms of facing detection, Christenfeld [23] found that when identical products are found side by side (an allocated product facing), that consumers tend to choose the middle product. This means that for products with greater than one facing, the sensors should be located in a position that they favor the middle of the display.

Durability is a great concern with any business purchase, so the sensors must be configured in a way that is not susceptible to major wear from normal customer behavior. This durability could be the result of mounting sensors in a way that customers will not interact with them, or in a way that the sensors are protected.

The system must be robust, so that the system can stay productive over long periods of time. In the software dealing with the sensors, functions to test and calibrate the sensors may be needed to detect and remedy errors in the system. The type of tests that each sensor will be subject to depends on the sensors used. Moreover, as with any IoT implementation, privacy is of primary concern to consumers [15][34]. The customer traffic tracking aspect of the project is the most likely to create privacy issues. To protect consumer privacy, the system will need to be developed in such a way that people are not identifiable from the data collected (i.e. data is anonymized in some fashion). Additionally, the data collected will only be available to authorized users, and should remain out of the public forum.

With the sensor design and set-up tentatively chosen, an intermediate step must be taken to verify that the product will work as envisioned. This will require the sensors to be connected to the micro-controller and data will need to be accessed to ensure that the intended behaviour is performed. At this stage the data read from sensors does not necessarily need to be transformed, analyzed, saved, or sent anywhere. Verifying that the sensor is giving the correct output will be a large but necessary step in the design process. This will likely require some programming to read the output of the sensors as well as circuitry design.

The final part of this task is drawing up schematics for the system as it would be installed on a retail shelf based on the knowledge gained through previously completed steps. This design will be a recommendation which can be adapted as new challenges arise. The format of this design should be a technical drawing or 3D model to ensure that the envisioned product can be created in 3D space. This may highlight some flaws in design choice which can be changed or adapted before sinking significant development time into interfacing sensors or building a prototype.

Task 3: Develop interface to send data to server

Once the configuration of the sensors has been chosen, the next step will be programming the software that handles the data. This will consist of the following steps: reading, interpreting, and temporarily storing data in a robust way; developing an interface between the server and micro-controller; testing and verification.

Depending on the sensors used, the data may come in analog or digital format. Useful data will be streaming from multiple sensors concurrently, and along with other metrics such as time of day will likely need to be included in what is sent to the database. Depending on the timing of incoming data, temporary storage will need to be allocated and managed. Due to the relatively low computing power of some micro-controllers, memory management will be an important consideration during this step.

Collected data must be packaged and sent in a way that is useful and efficient for transferring as well as storing in the database supplied by the Sensor Web Interface for IoT Standard working group [3]. The complexity of this task will very much depend on the micro-controller used and its supported languages. Although the prototype will see a very limited amount of data transfer in practice, the design should be created with a much larger scale of transactions in mind.

Once the interface has been developed, testing of the system as a whole can begin. This testing will ensure that the system can meet technical specifications before being built into a more permanent prototype. Testing will focus on correct results for test cases, uptime for the micro-controller, and reliability of the data being sent to the database. If any significant problems are encountered, they will be resolved before creating the system prototype.

Task 4: Construction of prototype

Once the product has a set of working sensors with a database connection, construction of the prototype can begin. This will be either a purpose built shelving unit or an adaptation of an existing shelve as described in the design completed in task 2. Sensors will be mounted and connected to the micro-controller, which will need its internet and power supplies implemented. Any non-permanent durability enhancements can be done at this point as it will be beneficial to test these adaptations.

Task 5: Testing of prototype

Having constructed the prototype, testing will commence. To begin, testing will be done against known cases which will have been developed as part of tasks 2 and 3. To supplement this, testing will also be done to verify the system performance in a real life scenario, which will involve team members adjusting stock over long periods of time. Data collected during this testing session could potentially be used for presentation purposes so it will be organized and saved. If bugs or improvements are encountered, they will also be recorded.

Task 6: Improvement of prototype

Any potential improvements or bug fixes derived from prototype testing can be added after they are verified by the team. Any changes to the hardware or software will have to be regression tested to ensure new bugs are not introduced and that the system is still functional. This will be an iterative process which will run until the project has reached completion.

Software based methodology

Collecting data from the sensors alone is not enough to achieve the goals of the project. The team also aims to interpret and present this data by means of a website. The software and website team's tasks are as follows:

1. Develop interface between website and database
2. Testing of database interoperability
3. Design website prototype
4. Testing of website prototype
5. Improvements to website prototype and website/database interface

Task 1: Develop interface between website and database

Similar to the hardware team's interfacing of the micro-controller to the server, the software team will need to interface the website and database. This should be done with security, efficiency, and robustness in mind. This task will require a small amount of website development in order to verify that the interface is working properly, but minimal time should be invested in an appealing graphical user interface (GUI) or layout issues.

Task 2: Testing of database interoperability

Once the interface has been developed, testing should be conducted to verify that the correct queries are being made, that errors can be identified, and that the interface is reliable. This task may be done in conjunction with task 4, after most of the website development is done.

Task 3: Design of website

Designing the website will be the major piece of work tackled by the software and website team. To begin with, the website framework must be chosen. Additionally, decisions pertaining to what browsers to support, what authentication model to use, update period, hosting solutions, and if the connection should be encrypted also need to be decided. Once the basic infrastructure has been planned, the team will focus on what kind of information is valuable to users, and what the best way to present this data is. This will cover things such as what statistics can be calculated, what kind of graphics can be generated to represent these data/statistics, what kind of navigation and structure should the website use, all the while maintaining a system that is easy to expand if new functionality is required. Information that can be derived from the data collected could include things such as stagnant products, traffic to aisle facing condition, historical vs. current trends, or a variety of others. The website will be the most visible part of the project, so the experience must be polished and easy to understand.

Task 4: Testing of website

The website should be tested on all supported browsers using a set of test cases as well as simulating real life conditions. Data and generated statistics need to be checked for accuracy as well as proper display. All navigation should be verified and authorization needs to be tested to ensure that user's data is secure. Feedback should also be gathered from user's outside the team to find potential improvements.

Task 5: Improvement of website and website/database interface

Any bugs or improvements noted during the testing tasks will be considered and implemented if the team agrees they are achievable and worthwhile. Outside opinions will also be considered. Again this will be an iterative process of updating and testing until the completion of the project.

Conclusion

In this document, various concepts and implementations regarding a smart-shelf product were explored. Compared to traditional methods involving RFID tracking [4], and alternative methods using custom solutions [8], it was determined that to solve the problems of shelf-facing and customer-traffic tracking, a “smart shelf” with a multi-sensor network built in is preferred, for both the scalability of sensor networks [11], and the speed and ease that direct acquisition of sensor readings provide. The proposed smart-shelf product will likewise be connected to the Internet of Things, which unlike alternative implementations, will provide management and supermarket workers the ability to react flexibly to the demand of the store in real-time.

In addition to examining the benefits of the proposed system over alternative implementations, both a hardware and software specification were laid out for the rem ainder of the project. In particular, it was decided that due to its power, versatility, and availability, the Raspberry Pi board is preferred to alternative micro-controllers proposed for the project. Lastly, a simple web-interface with an emphasis on ease-of-use as per the technological acceptance model was laid out for later implementation. Finally, a task model for the remainder of the project lifespan was projected, with an emphasis on iterative, agile development. Such a cycle encourages active testing during development with an emphasis on stability and a working product.

References

[1] S. Denman, A. Bialkowski, C. Fooks, and S. Sridharan, "Identifying Customer Behaviour and Dwell Time using Soft Biometrics," *Video Anal. Bus. Intell.*, vol. 409, pp. 199–238, 2012.

[2] OpenRemote, "OpenRemote: Open Source for Internet of Things," *OpenRemote: Open Source for Internet of Things*, 2012. [Online]. Available: http://openremote.com. [Accessed: 14-Oct-2013].

[3] Open Geospatial Consortium, "Sensor Web Interface for IoT SWG," *Sensor Web Interface for IoT SWG*, 2013. [Online]. Available: http://www.opengeospatial.org/projects/groups/sweiotswg. [Accessed: 22-Sep-2013].

[4] Sini Syrjala, "RFID Arena," *RFID Arena*, 13-Sep-2012. [Online]. Available: http://rfidarena.com/2012/9/13/%E2%80%9Csmart-shelves%E2%80%9D-the-store-shelf-of-the-future.aspx. [Accessed: 19-Oct-2013].

[5] Peter Russer and Uwe Siart, *Time Domain Methods in Electrodynamics*. Berlin Heidelberg: Springer, 2008.

[6] RFID Journal LLC, "RFID Journal FAQ 7638," *RFID Journal FAQ*, 2013. [Online]. Available: http://www.rfidjournal.com/site/faqs#Anchor-How-7638. [Accessed: 12-Nov-2013].

[7] Chien Ko, "3D-Web-GIS RFID Location Sensing System for Construction Objects," *Sci. World J.*, vol. 2013, pp. 1–8, Jun. 2013.

[8] NeWave Solutions Ltd., "NeWave," *NeWave*, 2013. [Online]. Available: http://newavesensors.com/products/smartshelf. [Accessed: 19-Oct-2013].

[9] Carla R. Medeiros, Jorge R. Costa, and Carlos A. Fernandes, "RFID Smart Shelf With Confined Detection Volume at UHF," in *RFID Smart Shelf With Confined Detection Volume at UHF*, 2008, vol. 7, pp. 773–776.

[10] RFID Journal LLC, "RFID Journal FAQ 36680," *RFID Journal FAQ*, 2013. [Online]. Available: http://www.rfidjournal.com/site/faqs#Anchor-If-36680. [Accessed: 12-Nov-2013].

[11] Steve H. Liang, Arie Croitoru, and C. Vincent Tao, "A distributed geospatial infrastructure for Sensor Web," *Comput. Geosci.*, vol. 31, pp. 221–231, 2005.

[12] A.W. Senior, L. Brown, A. Hampapur, C.-F. Shu, Y Zhai, R.S. Feris, Y-L. Tian, S. Borger, and C. Carlson, "Video analytics for retail," in *Video analytics for retail*, 2007, pp. 423–428.

[13] Mirela Popa, Leon Rothkrantz, Zhenke Yang, Pascal Wiggers, Ralph Braspenning, and Caifeng Shan, "Analysis of Shopping Behavior based on Surveillance System," in *Analysis of Shopping Behavior based on Surveillance System*, 2010, pp. 2512–2519.

[14] ShopperTrak, "ShopperTrak Solutions," *ShopperTrak Solutions*, 2013. [Online]. Available: http://www.shoppertrak.com/products. [Accessed: 01-Nov-2013].

[15] Katherine Poythress, "Is it OK for retailers to track shoppers with smartphones?," 06-Aug-2013. [Online]. Available: http://www.utsandiego.com/news/2013/aug/06/retailers-tracking-shoppers-privacy-concerns/. [Accessed: 17-Nov-2013].

[16] "Path Intelligence Ltd. Technology - Path Intelligence," *Path Intelligence Ltd. Technology - Path Intelligence*, 2013. [Online]. Available: http://www.pathintelligence.com/technology/. [Accessed: 01-Nov-2013].

[17] Future of Privacy Forum, "Mobile Location Analytics: Code of Conduct," *Mobile Location Analytics: Code of Conduct*, 22-Oct-2013. [Online]. Available: http://www.futureofprivacy.org/wp-

content/uploads/10.22.13-FINAL-MLA-Code.pdf. [Accessed: 01-Nov-2013].

[18] Sensource, “Sensor Server for Direct-to-PC or Network Integration,” *Sensor Server for Direct-to-PC or Network Integration*, 2013. [Online]. Available: http://www.sensourceinc.com/PCW-SSRX-Sensor-Servers.htm. [Accessed: 09-Nov-2013].

[19] Oracle, “Oracle Sensor Edge Server,” *Oracle Sensor Edge Server*, 06-Aug-2005. [Online]. Available: http://docs.oracle.com/cd/B14099_19/wireless.1012/b13819/rfid.htm. [Accessed: 09-Nov-2013].

[20] Ron Larsen, “Core Principles for Supermarket Aisle Management,” *J. Food Distrib. Res.*, vol. 37, pp. 107–111, 2006.

[21] Xavier Dreze, Stephen J. Hotch, and Mary E. Purk, “Shelf Management and Space Elasticity,” *J. Retail.*, vol. 70, no. 4, pp. 301–326, 1994.

[22] *Study Finds that Consumers Shop Less than Half the Store*, 4th ed., vol. 15. POPAI News, 1991.

[23] Pierre Chandon, J. Wesley Hutchinson, Eric T. Bradlow, and Scott H. Young, “Does In-Store Marketing Work? Effects of the Number and Position of Shelf Facings on Brand Attention and Evaluation at the Point of Purchase,” *J. Mark.*, vol. 73, pp. 1–17, 2009.

[24] Dimitrios A. Papakiriakopoulos, “Automatic Detection of Out-Of-Shelf Products in the Retail Sector Supply Chain,” Athens University of Economics and Business.

[25] Dunnhumby, “KSS Retail Heartbeat,” *KSS Retail Heartbeat*, 2013. [Online]. Available: http://kssretail.com/solutions/out-of-stock-detection/. [Accessed: 07-Nov-2013].

[26] Intel, “Intelligent Shelf Compliance Solution,” presented at the National Retail Federation Conference and Expo, New York City, MA, USA, 2013.

[27] Hubert, “Shelf Management Solutions and Shelf Facing Systems,” *Shelf Management Solutions and Shelf Facing Systems*, 2013. [Online]. Available: http://www.hubert.com/Display-Cases-Fixtures-0304/Shelf-Management-03040357.html. [Accessed: 02-Nov-2013].

[28] Netduino, “Netduino Hardware,” *Netduino Hardware*. [Online]. Available: http://netduino.com/hardware/. [Accessed: 10-Nov-2013].

[29] Arduino, “Arduino,” *Arduino*. [Online]. Available: http://arduino.cc/. [Accessed: 10-Nov-2013].

[30] Raspberry Pi, “Raspberry Pi | An ARM GNU/Linux box for $25. Take a byte!,” *RaspberryPi.org*, 2013. [Online]. Available: http://www.raspberrypi.org/. [Accessed: 17-Nov-2013].

[31] Full Steam Marketing & Design, “Merchandising / Spring Load,” *Merchandising / Spring Load*. [Online]. Available: http://www.fullsteam.com/merchandising/spring_loads. [Accessed: 11-Nov-2013].

[32] Albert L. Lederer, Donna J. Maupin, Mark P. Sena, and Youlong Zhuang, “Technological Acceptance Model and the World Wide Web,” *Decis. Support Syst.*, vol. 29, pp. 269–281, 2000.

[33] Department of Health & Human Services - USA, “Selecting a Development Approach.” Office of Information Services, 27-Mar-2008.

[34] Luigi Atzori, Antonio Iera, and Giacomo Morabito, “The Internet of Things: A survey,” *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010.