Location-based Responsive Space for Multimedia Environment Using Ultra-wideband Technology

Yu Sato

A Thesis
in
Special Individualized Program
of
School of Graduate Studies

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Arts (Special Individualized Program) at
Concordia University
Montreal, Quebec, Canada

March 2011

CONCORDIA UNIVERSITY

Special Individualized Program

School of Graduate Studies

This is to certify that the thesis prepared

By: **Yu Sato**

Entitled: **Location-based Responsive Space for Multimedia Environment Using Ultra-wideband Technology**

and submitted in partial fulfillment of the requirement for the degree of

Master of Arts (Special Individualized Program)

complied with the regulations of the University and meets with the accepted standards with respect to originality and quality.

Signed by the final examining committee:

________________________________Chair
Lynn Hughes

________________________________External Examiner
Daniel Forgues

________________________________Examiner
Nizar Bouguila

________________________________Examiner
Sha Xin Wei

________________________________Thesis Supervisor
Amin Hammad

Approved____
Graduate Program Director

_______________ 20___ ________________________________
Dr. Graham Carr
Dean of Graduate Studies

ABSTRACT

Location-based Responsive Space for Multimedia Environment Using Ultra-wideband Technology

Yu Sato

The rapid advances in Real-Time Location Systems (RTLSs), such as Ultra-Wideband (UWB) technology, provide opportunities to realize new types of multimedia environments. Taking advantage of the UWB technology, this research investigates how location information enhances multimedia environments by utilizing the relationship between the physical environment and its habitants. In this thesis, we present an approach, called UWB-based Responsive Multimedia Space (URMS), that combines near real-time location data with multimedia devices. This approach has three main parts: (1) the concept of URMS incorporating multimedia devices for the realization of a series of location-based events, (2) the requirements of the UWB system and multimedia devices for the development of URMS, and (3) the methods for the grid-based event selection, and continuous and smooth adjustment of location-based interaction based on distance. Furthermore, a touch user interface is proposed to define interactively multimedia events. In order to validate the proposed approach, a prototype system is developed and tested in two case studies.

ACNOWLEDGEMENT

My greatest appreciation goes to my supervisor, Dr. Amin Hammad for his intellectual and personal support, encouragement and patience. His advice and criticism was my most valuable asset during my studies. Overall, I feel very fortunate having the opportunity to know him and work with him.

I would like to acknowledge the contributions of Mr. Ali Motamedi, Ms. Sonia Rodriguez, Dr. Cheng Zhang and Mr. Ziad Nahas, and Mr. Michael Fortin for helping in the development of the application and performing tests.

Last but not least, I would like to thank Dr. Xin Wei Sha and Dr. Nizar Bouguila for spending their precious time in reading this thesis and for their valuable comments as members of my thesis committee.

DEDICATION

To my parents, Yoshimi Sato and Hiroko Sato who made all of this possible, for their endless encouragement and support.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Description
2D	Two-dimensional
3D	Three-dimensional
A-GPS	Assisted Global Positioning System
AOA	Angle of Arrival
API	Application Programming Interface
DLL	Dynamic-link Library
DMX	Digital Multiplex Signal
GHz	Giga Hertz
GIS	Geographic Information System
GPS	Global Positioning System
GUI	Graphical User Interface
HF	High Frequency
ID	Identification Data
IR	Infrared
ISO	International Organization for Standardization
IT	Information Technology
LAN	Local Area Network
LED	Light Emitting Diode
LF	Low Frequency
OSC	Open Sound Control
PoE	Power over Ethernet
RF	Radio Frequency

RFID	Radio Frequency Identification
RSS	Received Signal Strength
RTLS	Real-Time Location System
SNS	Social Networking Service
TDOA	Time Difference of Arrival
TML	Topological Media Laboratory
TV	Television
UHF	Ultra High Frequency
URL	Uniform Resource Locator
URMS	Ultra-wideband based Responsive Multimedia Space
USB	Universal Serial Bus
UWB	Ultra Wide Band
WLAN	Wireless Local Area Network

CHAPTER 1 INTRODUCTION

1.1 General Review

Over the last decade, digital multimedia devices have been increasingly a part of modern life. Our living space and life style are changing as they get blended with multimedia devices. These multimedia devices, such as displays, sound systems, lighting systems, etc., are becoming more computer-based, which makes them more manageable and interactive. The interactive multimedia environment is often described as ubiquitous, responsive, smart, or sensitive space. The term ubiquitous computing, as Mark Weiser proposed 20 years ago (Weiser, 1991), is particularly used to describe this trend of the Information Technology (IT) industry. This idea infers that modern information technologies aim to merge into our surroundings. As Weiser stated “For ubiquitous computing one of the ultimate goals is to design technology so pervasive that it disappears into the surrounding.” This perspective on ubiquitous computing evolved into new academic approaches, such as tangible user interfaces (Ishii and Ullmer, 1997), which focus on human interaction with physical objects. Researchers are now developing multimedia environments integrated with various sensor technologies, such as interactive wall installations (Sidor, 2008), interactive workspace (Bongers and Mery, 2007, Johanson et al., 2002) and interactive media arts (Aylward and Paradiso, 2006, Nagashima, 2003, Tanaka and Gemeinboeck, 2006).

With the current development of tracking technologies, location information has received much attention from both the scientific community and the industry. We assume that the use of location information can make multimedia environment interesting and useful. As the tracking technologies become prevalent, location information should expand the design possibility of multimedia environment for interaction designers. This trend is to change traditional ways of understanding the relationship between the environment and the users.

Past research on location-based services has described the idea for communication and sharing information between people using a single handheld device or a fixed computer. For example, in the multimedia system by Bacon et al. (1997), the location information was based on a room-by-room identification. Nowadays, new tracking technologies enable to detect a user's movement with a high degree of accuracy. This opens the door to the development of new range of multimedia environments for close interaction among users in the same space.

In this thesis, we will explore location interaction and build a responsive multimedia space supported by tracking technology to find new ways of approaching the location-based multimedia environment. A key feature of this thesis, distinguishing it from other researches in interactive environments, is a focus on environments incorporating the Ultra-Wideband (UWB) technology and multimedia apparatuses for real-life situations. Although interactive systems often rely on other techniques such as image and video processing, which enable them to detect a user's movement with a high degree of accuracy, they usually work only with a limited number of users or spaces. In contrast, the UWB system enables flexible event provisions based upon identification information management. Hence, the proposed system offers an innovative design approach for future multimedia environments.

1.2 Research Objectives

The objectives of this research are: (1) to describe the architecture of a UWB-based system for multimedia environments; (2) to define the requirements of such a system; (3) to define the methods of event processing based on locations; and (4) to implement a prototype system that can be used to validate the proposed approach through several case studies.

1.3 Thesis Organization

This study will be presented as follows:

Chapter 2 Literature Review: This chapter presents our review of the theory of interactivity from a phenomenological point of view, the notion of place and locative media, Real-Time Location Systems (RTLSs), the UWB technology and its methods, and multimedia environments.

Chapter 3 Proposed Approach: This chapter describes an overview of the research methodology, which includes the conceptual approach, the requirements for the tracking system and location-based events, the methods for grid-based event selection and continuous event control function, and the touch user interface.

Chapter 4 Case studies: This chapter describes the development of a prototype system and shows the applicability of the proposed approach in two case studies in order to verify the feasibility of the proposed approach for responsive space.

Chapter 5 Conclusions and Future Work: This chapter summarizes the present research work, highlights its contributions, and suggests recommendations for future research.

CHAPTER 2 LITERATURE REVIEW

2.1 Introduction literature review

The literature review focuses on topics related to the researches in the areas of human interaction, locative media, RTLSs, and multimedia environments. The purpose of this review is to understand the basic concepts and technologies necessary for this research.

2.2 Interactivity in Phenomenology

"The body is our general medium for having a world." (Merleau-Ponty, 1962)

The exploration of the relationship between humans and the world has deep roots in philosophy. According to an evolutionary standpoint, interaction is necessary for living things to survive. Looking at human history, on the other hand, there are always adaptations to our environment and relationships to others. Whether we consciously consider interaction or not, we use phones, Televisions (TVs), computers, Internet networks and services, all to communicate a message to others. Interaction is inherent in our body to make progress, innovate, improve, and adapt in certain environments because it is necessary in life to talk with others and exchange ideas. Our body is equipped to be interactive with the environment. This idea infers that modern information technologies do not aim to replace the real world with the virtual world. Instead, those technologies aim to merge into our surroundings, as Mark Weiser proposed with the term "ubiquitous computing" (Weiser, 1991).

Interactivity begins at the very moment we start to recognize. This moment is uniquely represented in a famous scene of the film 2001: A Space Odyssey (1968). Two groups of anthropoids are fighting each another. One of the anthropoids finds an animal bone, picks it up and plays with it. Using it as a weapon gives his group an advantage. The group with the power of the weapon beats down the other. The club is thrown up into the air, rotating slowly, and turns fluidly into a huge spaceship. This scene represents how objects become tools for us. The point is that it happens in an instant.

Paul Dourish, whose fields of expertise are in the domains of human-computer interfaces, ubiquitous computing and context-aware technology gave rise to the issue of interactivity as an important angle of pioneering technologies. Even though the definition of interactivity is the subject of long-standing debate spanning several decades, it is argued that interactivity may exist in our bodily reactions. Dourish states that phenomenology is one of the important philosophical movements that influenced a wide range of academic disciplines (Dourish, 1999, 2001). Phenomenology's main concern lies in exploring human reality as it appears to perception while negotiating between the subjective and objective aspects of human experience (Sokolowski, 1999). The phenomenological perspective provides specific methodological tools for observing phenomena and perception within a situation (Thompson, 2007). In its various theoretical aspects, phenomenology serves as a way into various forms of experience.

The study of phenomenology draws on the work of 20th century great philosophers such as Edmund Husserl and Merleau-Ponty (Kockelmans, 1976, Zahavi, 2002). Especially with Merleau-Ponty's theory, the body is an undivided unity, and it is meaningless to talk about the perceptual process of seeing without reference to all the senses, to the total physical environment in which the body is situated, and to the embodied intentionality that means we always interact with the world through bodily contact (Merleau-Ponty, 1962, Kockelmans, 1976). In other words, the body has the ability to adapt and extend itself through external devices. When we move our hand toward a cup of coffee in front of us, we relate to this object as a thing positioned outside of our body. When we grasp the object and move it to our mouth, the cup is no longer primarily an object external to us whose position we judge in relation to other objects, but it has entered into the space of our experienced body, thereby changing its structure. When we place the cup of coffee on the table again, it leaves our experienced body and becomes part of the space of objects. Through skill acquisition and tool use, we thus change our bodily space, and consequently our way of being in the world.

The study of interaction is used in product design that includes a deep consideration of bodily interaction. Any interaction with a product or service typically requires a cycle where the user perceives, thinks and acts; where for the most part, perceiving requires sensory capability, thinking requires cognitive capability, and acting requires motor capability (Clarkson, 2007). Figure 2-1 illustrates typical relations between human and product/service as an interaction process.

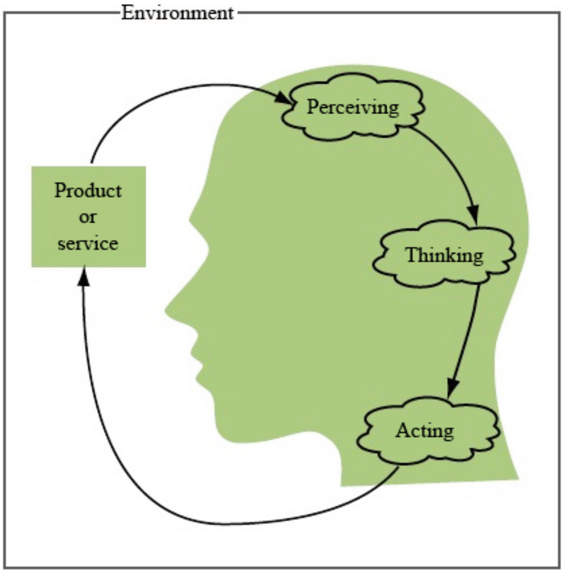


Figure 2-1: The concept of human interaction cycle (Clarkson, 2007)

This relationship prompted us to realize the importance of incorporating interaction in the development of information systems and led us to build a novel idea into existing systems. In this sense, an interdisciplinary approach helps us understand both how interactivity can be brought to us and how interactive design comes to be involved in philosophical and conceptual investigations of experience. This approach also helps us understand paradigms in modern technology. Experience is always open to the world because it is a mode of active engagement with the world (Russon, 2003). A phenomenological study of experience is not only an exercise in introspection method, but also an act of involvement to what one does in exploring the world. We direct our attention to the embodied and environmentally situated activity of exploring the environment.

2.3 Locative Media

"Place focuses on the realm of meaning and experience. Place is how we make the world meaningful and the way we experience the world." (Cresswell, 2004).

With the evolution of the information society, place decreases in importance as mobility increases. While in the past, place was used as a platform for information exchange, the Internet and mobile phones now have taken over these functions. In fact, people spend more time communicating with others through phones and social networking services (SNSs) than communicating face-to-face. Cresswell (2004) mentions that places, such as squares, shopping centers, schools, offices, hospitals, libraries, banks, etc., are changing with such informational networks and functions. In parallel with the evolution of information technology, changes in the functions of place have created a new concept of information infrastructure, called *Locative Media*.

The term locative media was coined by Karlis Kalnins. Lemos (2008) defines locative media as a combination of location-based technologies and LBSs. Locative media indicates that location information can provide many possibilities of using location reference in an application system. Lonthoff and Ortner's (2007) classification of LBSs opens up opportunities for us to address locative media using a variety of localization technologies and methods, as shown in Figure 2-2. These can be categorized into several groups: a variety of LBSs (maps, traffic data, and navigation), location-based information based on the profile of the user, location-based gaming, and so on. Additionally, locative media are also used for corporate and industrial applications (track material, consumers, suppliers and employees) (Schiller and Voisard, 2004), art projects (Hemment, 2006, Tanaka and Gemeinboeck, 2006), mapping, or accessing services and information (Bleecker and Knowlton, 2006).

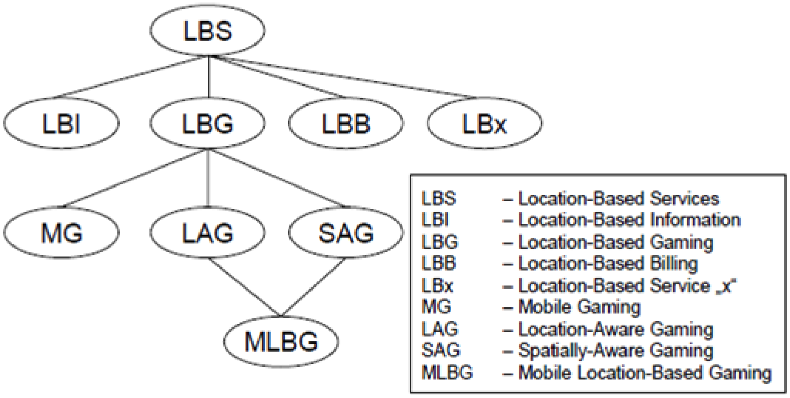


Figure 2-2: Taxonomy for LBSs (Lonthoff and Ortner, 2007)

All of those possibilities are essentially based on location-based technology. It comprises the set of digital devices, sensors and digital wireless networks that allows exchanges of information with the physical world. Thus, the LBS is a consequence of the information exchange generated by these devices and networks. The system requires that a node or tag be placed on the object or person being tracked, offering a combination of high data rate capability, security, multi-path immunity, and strong tag readability. Its implementation should be simple and low cost as well. For example, a Global Positioning System (GPS) receiver in a cell phone or a Radio Frequency Identification (RFID) tag on a book can be used to track those objects using systems such as GPS satellites or RFID receivers.

Locative media are examples of applications and services related to the notion of ubiquitous computing (Lemos, 2008). Locative media projects can help us to see places and space differently. Locative media do not point to a world of electronic cyberspace apart from the physical world. Instead, they insist that what they produce are "augmented realities" for playing on the street, in annotation, mapping and tagging real things (Lemos, 2008, Lonthoff and Ortner, 2007). What we are seeing now are several examples of integrated, mixed processes that merge electronic and physical zones, creating new forms and new senses of place. When we create tags and maps, use a GPS with a mobile phone to find a location, produce content and electronically annotate a place, play location-based mobile games or organize mobilization in public space by SMS, we are controlling the space, creating a new sense of place and new forms of physical zones (Lemos, 2008).

The new meaning of place can be a platform for media, where we have new dimensions of place as intersections of flows of information (Lemos, 2008). There are different uses of these spaces with different functions, including new forms of control, access, and surveillance. In the new meaning of place, mobility and flow come to be essential. Supposing place as flow, and mobility as a way to move around in physical and informational space, there is huge potential to improve our life with LBSs and technologies. Although the fixed place is still important to create memory and social meaning, informational territories, that relate real place and virtual space, create new functions for places and a redefinition of social and communication practices. Also, thinking about mobile social networks, collaborative maps, urban annotations, location-based games, and artworks, we can realize that relationships among locations can reinforce communities and the meaning of place to rebuild social bonds as a complement to physical contact (Lemos, 2008).

2.4 Real-Time Location Systems

"Real-time locating systems are wireless systems with the ability to locate the position of an item anywhere in a defined space (local/campus, wide area/regional, global) at a point in time that is, or is close to, real time" (International Organization for Standardization and International Electrotechnical Commission, 2006). A RTLS is used to identify and track the position of objects in real-time. RTLS is the convergence of several technologies including Geographic Information Systems (GIS), the GPS, RFID, and Wireless Local Area Network (WLAN).

A RTLS allows determining the location of assets or people at a specific time and frequency of update. Different technologies can be used in a RTLS such as sound, ultrasound, Bluetooth, Wi-Fi, RFID, ZigBee, UWB, GPS, cellular, camera vision (Bouguila and Ziou, 2005), infrared (IR) and light, among others (Malik, 2009). Location technologies estimate a location of a point in two-dimensional (2D) or three-dimensional (3D) space in relation to a coordinate system where some references are known. Distance and observable angles are in most of the cases the base for those findings and can be obtained from different sources such as arrival times, arrival time differences, and field strength. The main objective of a location system is to determine the exact position of people or assets by minimizing the errors in the measurements. By knowing the relative location of an object in relation to sufficient reference points, the absolute location of the object can be determined (Ward and

Webster, 2009). The sources of errors can be the instruments errors, measurement errors, noise, and inaccurate reference positions. Each location system has its intrinsic and cost limitations. Recently, location technologies are focused on improving the accuracy of the position of one system and data fusion of different systems. To get the maximum benefit of these technologies, different aspects must be taken into account when deciding which technology must be used. The accuracy, advantages and limitations are directly related to the final purpose of the location system. Therefore, the requirements of the location system contribute to the accomplishment of the final goal of the RTLSs. “Good applications are those that achieve an adequate equilibrium between system requirements, technological advantages, and associated costs.” (Muñoz et al., 2009).

According to (Malik, 2009, Di Benedetto et al., 2006), system requirements of RTLSs are described as follows: (1) Accuracy: less than 30 cm and update rates of at least 1 Hz were defined acceptable for many infrastructure applications such as automated material, work zone safety and location tracking and navigation. (2) Installation cost: the system should be affordable based on the application and the frequency of use. (3) Ease of use and operational cost: the hardware should be maintainable at a reasonable cost and in a simple way. (4) Size and weight: the sensors and tags should be of a size suitable for the object to be tagged and the place where they will be installed should ideally offer wireless operation. (5) Standards and regulations: each country has its own regulations that have to be followed. Low-powered commercially available UWB systems have no restrictions (Breed, 2005). (6) Interoperability: the system must interface with wireless communication technologies and needs to cohabit with other signals in the Radio Frequency (RF) spectrum. (7) Range: for indoor environments, the system must be usable between fixed and potentially mobile receivers and it needs to work in object-cluttered environments. (8) Interferences (multipath): the system must have a good performance in environments with obstructions that cause multipath signals.

2.5 UWB Technology

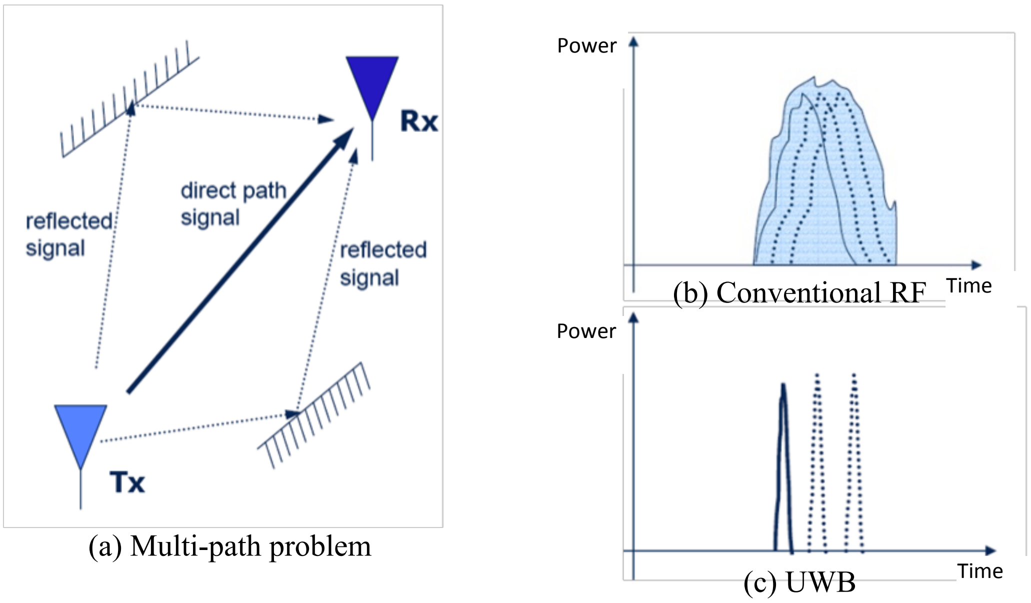
UWB is a radio technology that proposes a new way of using the radio spectrum: it works by generating a series of radio energy pulses, at specific time instants, simultaneously across a large band of frequencies, without interfering with existing narrowband frequencies (Di Benedetto et al., 2006). This relates to what is known as 'impulse', or short-pulse electromagnetic. Although the UWB ranges are relatively short (less than 100 meters), the UWB technique enables data to be transmitted at a very high rate (GB/s ranges); it was originally designed for commercial radar systems. However, the UWB technology has great potential in consumer electronics, personal area networks, and tracking applications. The advantages of using this technology are that it can carry high data rates with low power (less than 0.5 milliwatts) and little interference over a distance up to 70 meters. Table 2-1 lists the categories of applications approved by the Federal Communications Commission (FCC) for UWB (Malik, 2009).

Table 2-1: UWB applications (Malik, 2009)

Application/Class	Frequency
Communications and measurement systems	3.1-10.6 GHz
Imaging: ground-penetrating radar, wall, medical imaging	Less than 960 MHz or 3.1-10.6 GHz
Imaging: through walls	Less than 960 MHz or 3.1-10.6 GHz
Imaging: surveillance	1.99-10.6 GHz
Vehicular	24-29 GHz

UWB is used as a RTLS by different industries for manufacturing, logistic, transit and transportation, military, hazardous environments, immersive media and retail applications. As discussed in Section 2.4, RTLSs can track and identify the location of objects in near real-time using tags attached to objects, and sensors that receive the wireless signals from these tags to determine their locations. Due to the extremely low emission levels currently allowed by regulatory agencies, UWB systems tend to be short-range and used in indoor spaces. The components of a UWB RTLS are: (1) Tags: Electronic devices that send UWB pulses, which usually are short and have low repetition rates, typically 1-100 mega pulses per second. The tags have a battery and can be equipped with different interactive functions such as a buzzer, buttons or Light-emitting Diode (LED). The tags are usually small and are attached to the objects to be tracked; (2) Location sensors (UWB receivers): Devices which support two-way communication and use angles and signal timing to calculate the precise location of tags; (3) Location engine: The software that computes the location of the tags using the data provided by the sensors and using techniques such as Time of Arrival (TOA) and Time Difference of Arrival (TDOA); (4) Middleware: "The software that connects the disparate applications, allowing them to communicate with each other and to exchange data. It is the software that resides among the pure RTLS technology components (tags, sensors, and the location engine) and the business applications." (Malik, 2009); and (5) Application: "The software that interacts with the RTLS middleware and does the work users are directly interested in." (Malik, 2009).

UWB has the ability to carry signals through doors and other thin obstacles (e.g., interior walls) that tend to reflect signals at more limited bandwidths and a higher power. As shown in Figure 2-3, with conventional RF, the reflections in in-building environments distort the direct path signal, making accurate pulse timing difficult, whereas if UWB is used, the direct path signal can be distinguished from the reflections, making pulse timing easier. Thus, the accuracy of the UWB system can be up to 15 cm in good conditions. These conditions include the absence of multi-path problems because of the availability of direct line-of-sight signals from tags to sensors and the absence of metallic objects in the vicinity of the UWB system (Muthukrishnan and Hazas, 2009). In addition, UWB works better with metals than other RF devices do. These advantages make it possible to attach UWB tags to moving objects on site and collect accurate location data.



(a) Multi-path problem

(b) Conventional RF

(c) UWB

(a) Multi-path problem (b) Comparison between conventional RF and UWB

Figure 2-3: UWB vs. conventional RF (Ubisense, 2010)

2.5.1 Methods in UWB RTLSs

UWB technology is used as an indoor location system that measures distances or angles between known points and unknown position. There are commonly used methods to calculate those distances and angles, such as in Received Signal Strength (RSS), Angle of Arrival (AOA), TOA and TDOA (Correal et al., 2003). All these methods can be affected by the obstructions and reflections in indoor environments. A combination of TDOA and AOA can be used to obtain a high accuracy in the location measurement (Abdul-Latif et al., 2007).

In the TDOA method, the difference in time at which the signal from the tag to be positioned arrives at different receivers is measured. Each time difference is then converted into a hyperboloid with a constant distance difference between two receivers. The position is found by solving equations if the coordinates of the receivers are known (intersection of the corresponding hyperboloids). In 3D, at least four receivers are required and this technique requires synchronization of the receivers' clocks (Ghavami et al., 2004). Figure 2-4 illustrates this principle for 3 sensors.

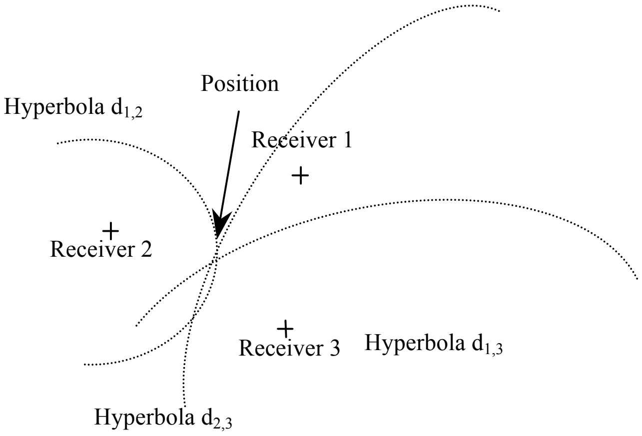


Figure 2-4: TDOA positioning principle (Ghavami et al., 2004)

In the AOA method, the angle of arrival of the signal sent by the tag to be positioned is measured at several stationary receivers. Each measurement forms a radial line from the receiver to the tag. In 2D positioning, the position of the tag is defined at the intersection of two directional lines of bearing. This method has the advantage of not requiring synchronization of the receivers nor an accurate timing reference. On the other hand, receivers require regular calibration in order to compensate for temperature variations and mismatches (Ghavami et al., 2004). The location estimate of the tag is calculated at the intersection of these lines. In theory, direction-finding systems require only two receiving sensors to locate a tag, but in practice, to improve accuracy and compensate for finite angular resolution, multipath and noise, more than two references are needed (Muñoz et al., 2009). Figure 2-5 illustrates this principle for 3 sensors.

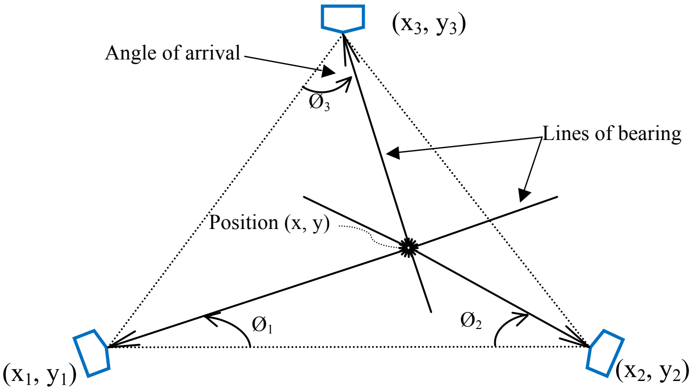


Figure 2-5: AOA positioning principle (adapted from Muñoz et al., 2009)

The multipath effect would affect the time and angle of an arrival signal decreasing the accuracy of estimated location. Another alternative is to estimate the distance of the mobile unit from a set of measuring units, using the attenuation of emitted signal strength. Signal attenuation-based methods attempt to calculate the signal strength loss due to propagation. AOA and TDOA accuracy can be improved by utilizing the premeasured RSS contours centered at the receiver or multiple measurements at several base stations (Liu et al., 2007).

In the specific UWB system used in this research, the location position techniques are combined TDOA and AOA. This combination provides greater system robustness. A sensor cell is constructed by several sensors connected together into a single operating unit that captures the location of tracked objects. The software component is a computer system that collects all the available TDOA and AOA data from the receivers that detected a tag's signal. Then, this data is used to compute a solution for the tag position best matching the input data. The system may attribute different weights to each item of data used in the position calculation (Ward, 2009). Sensors are synchronized using a timing signal (distributed by cables) from each sensor to the timing source. A master sensor is defined to receive and synchronize the timing data from the other sensors. Each tag registers with its containing sensor cell, and is inserted into the schedule for that cell. The schedule determines when the tag should emit UWB signals to be located by the cell. The schedule is optimized to give attention to each tag as close as possible to its requested quality of service, while maintaining enough space in the schedule for new tags to register. When a tag emits a signal, this signal is picked up by one or more sensors in the cell, as shown in Figure 2-6. The slave sensors decode the UWB signal and send the angle of arrival and timing information back to the master sensor through an Ethernet connection. The master sensor accumulates all sensed data and computes the location based on triangulation (Ubisense, 2009).

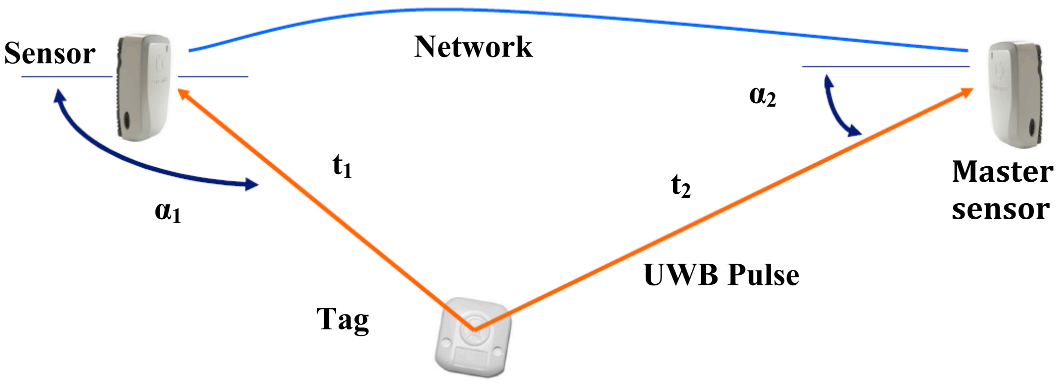


Figure 2-6: Signals sent to sensors are used to calculate the 3D position of a tag (Ubisense, 2009)

2.5.2 Range/Tag Readability

Figure 2-7 compares different location technologies, such as passive RFID, electromagnetic, laser, ultrasound, IR proximity, conventional Radio Frequency (RF) timing, UWB, WLAN, RSS, and assisted GPS (A-GPS). This comparison is done based on the accuracy and the coverage offered by each technology to identify the ideal technology as the technology that can achieve accuracy of less than 0.3 m and with coverage of more than 100 m (Ward, 2007).

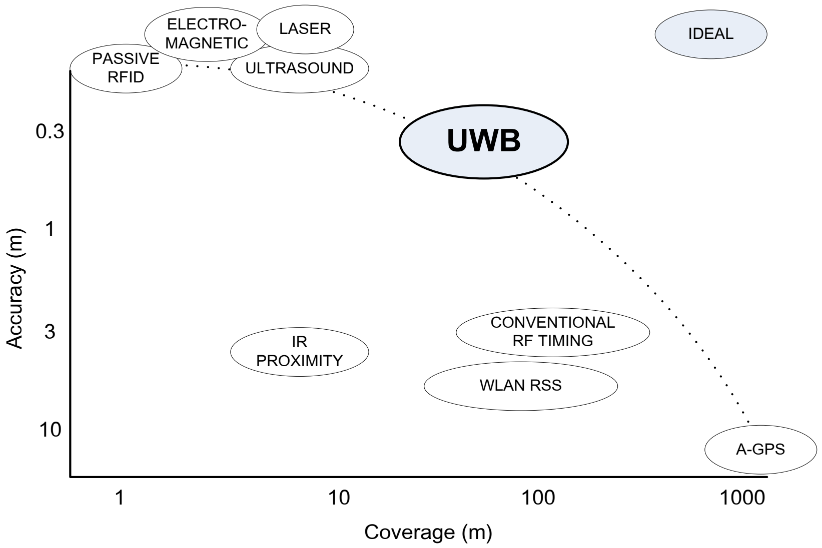
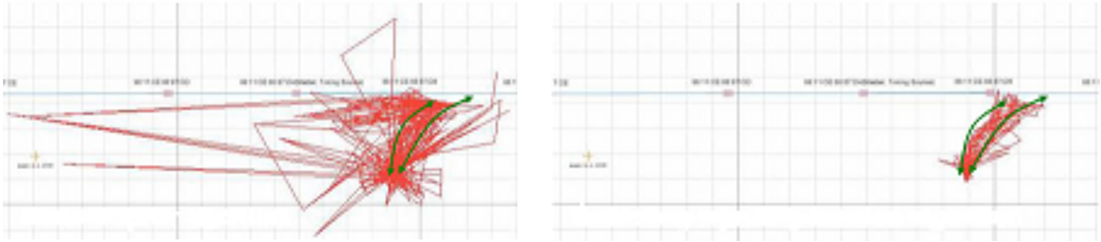


Figure 2-7: Comparison of location technologies (adapted from Ward, 2007)

2.5.3 Data Filtering

There are different methods to obtain the location in UWB such as TOA, TDOA, and AOA, as discussed in Section 2.5.1. All those methods, however, have problems determining the real position because of obstacles and reflections in the line of sight between tags and sensors. The basic location algorithms receive signals and reject all but the first signal to arrive. However, this first signal can be a reflection and affect the sensed position. To determine the true position, reflection-rejecting location algorithms are needed (Ward, 2007). Figure 2-8 show the difference between the basic location algorithm and the measurements computed after filtering algorithms are applied (proprietary from Ubisense). The real paths are shown in green and the measured ones in red. This presents the need to apply filtering and averaging techniques to determine the location of the objects to be tracked.



(a) Basic location algorithm **(b) Reflection-rejecting location algorithm**

Figure 2-8: Location algorithm comparison (Ward, 2007)

UWB RTLSs can provide a dynamic control of the tags by defining parameters such as scheduling and filtering by setting each tag to generate a UWB pulse depending on its profile. Those properties can be set either for a range of tags, or for individual tags. Data can be filtered in near real-time in order to reject individual measurements that may be reflections or corrupted by noise. For that purpose, different algorithms are available for estimating tag positions from sensor measurements. By defining some parameters, the behaviour of the algorithm can be controlled. In the case of Ubisense system, there are two options for collecting data: No filtering or filtering. In the first case, the readings are raw sensed positions and no filter is applied. In the information-filtering algorithm, the previous motion of the tag is used to predict its position at the time a new sighting is made. The information from AOA and TDOA measurements are individually validated against the predicted position, and then are applied to give a new estimate of position. The motion model for the filter has to be defined by specifying the constraints on the motion that the tracked object can undergo. The filtering options are: (1) Information filtering; in this case the tag is free to move in 3D and the motion model is of position and velocity with Gaussian noise on velocity. (2) Fixed height information filtering; the tag is constrained to move horizontally and the motion model is of position and velocity with gaussian noise on velocity. (3) Static information filtering; the tag is free to move in 3D and the motion model is of position with gaussian noise on position. (4) Static fixed height information filtering;

the tag is constrained to move horizontally and the motion model is of position with Gaussian noise on position. In addition to those filters, a filter can be created based on the purpose of the tracking system. Another point to consider is that those filters can be applied to individual tags or to a set of tags based on the objective of the tracking system (Ubisense, 2009).

2.6 Multimedia Environments

Multimedia can be defined in multiple ways depending on the perspective. A typical definition is "Multimedia is the combined use of several media, such as movies, slides, music, and lighting, especially for the purpose of education or entertainment." (Brooks, 1997). As the information is presented in various formats, multimedia enhances user experience and makes it easier and faster to grasp information. Multimedia devices play especially important roles in ubiquitous computing. They let people access the Internet and other information networks without time or place restrictions. For example, light has the obvious function of providing visibility of visual performance (Mania and Robinson, 2004). From the designer's point of view, with the pervasive use of LED, integrated lighting management units efficiently drive luminous colors of lights to conserve electricity. The shift from traditional lighting to LED has turned lighting units into multimedia devices. Especially in developing the lighting management solutions for commercial contexts, lighting designs may intentionally or unintentionally function more actively to selectively shift human visual experiences: focusing attention, guiding circulation and generally affecting impressions of a room or task situation (Mania and Robinson, 2004).

2.7 Previous Projects related to Responsive Space, RFID and UWB

2.7.1 Ozone - Continuous State-based Media Choreography System

Sha (2005) built a responsive environment in which wireless sensors beamed accelerometer data to OpenGL textures that were mapped onto a polygonal mesh. In his research, he noticed that the nearly zero latency synchrony resulted in participants engaging in the installation. This discovery subsequently motivated the Ozone project, a new media choreography system based on layered, continuous physical models, designed for building a diverse range of interactive spaces that coordinate arbitrary streams of video and audio synthesized in real-time response to continuous, concurrent activity by people in a live event. This project aimed to build rich responsive spaces that could sustain the free improvisation of collectively or individually meaningful, non-linguistic gestures. Ozone provides an expressive way to compose the potential "landscape" of an event evolving according to the designer's intent as well as contingent activities. It maps sensor data into fields of time-based media such as real-time video and sound derived from continuous states. Ozone is as rich as a custom built, singular interactive environment, yet is a general composition system that can yield a great variety of installation events. Figure 2-9 illustrates the architecture of the current Ozone system. The system consists of three primary engines: (1) a data routing, statistics, gesture tracking and media choreography engine, (2) a sound synthesis engine, and (3) video effects engine. These core components of the Ozone system enable to incorporate rich musical and visual imagery and dynamics. In addition, the system has wireless sensor platforms, with a sensor-board containing a magnetometer, accelerometer, photocell, sound meter, and analog and digital inputs for additional sensors.

Sha et al. (2005, 2009, 2010) explored continuous sensing of gestures to control audiovisual media. The research included an analysis of system requirements about the real-time synthesis and modulation of video and sound, by using wireless sensors. They introduced responsive media, media that evolve, up to the limit of human perception, concurrently and deterministically with human gestures and movements. The idea to realize that richness comes from corporeal experience, and is independent of combinatorial structure, which is a formal property not amenable to intuitive grasp. They argue for a phenomenological approach to complex visual interaction based on corporeal kinesthetic intuition, and provide an effective way to provide such texture-based interaction using computational physics. A motivating application is to create palpable, highly textured video that can be projected as structured light fields in responsive environments.

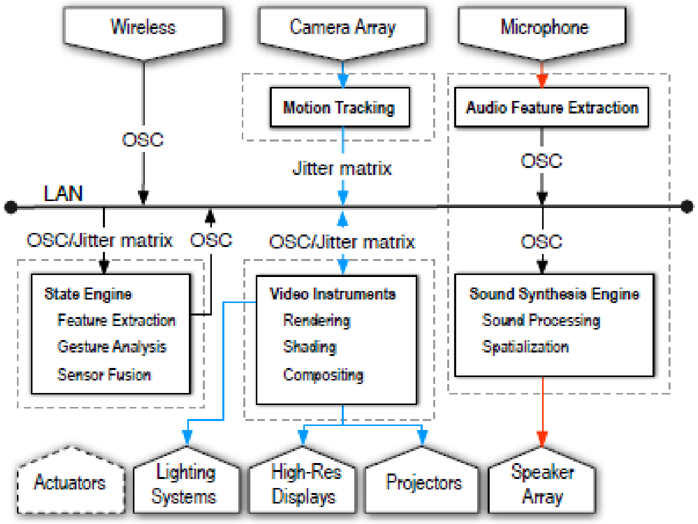


Figure 2-9: Ozone architecture: components and data paths (Sha et al., 2010)

2.7.2 Audio Graffiti

Audio Graffiti is a multi-user sound/music installation that explores new modes of sonic interaction, supported by the latest in locative technologies (Settel and Wozniewski, 2009). This installation took place at the 12th Biennial of Arts and Technology Symposium in the lobby of the Ammerman Center for Arts and

Technology. Being deployed in an indoor space, as shown in Figure 2-10 (a), participants who are wearing a wireless headset and tracking device can tag or spray sound on to a real physical wall, what they call "collage of audio graffiti". The system provides several small musical instruments, which can be used along with one's voice, to add sounds to the collaborative musical mix. The installation is equipped with some pre-existing sonic material, which allows participants to synchronize rhythmically, and maintain cohesion over time. All user-contributed sounds slowly fade away, resulting in an ever-evolving musical piece. As users moves around, they also experience a changing sonic perspective of the localized sounds, based on their particular location. Thus, users not only create the audio contents, but they also participate actively in the encounter (remixing) of sonic material. Participants who are waiting for their turn in the staging area may watch a real-time 3D visualization of the installation, which shows avatars of each player walking amongst virtual sound sources, as seen in Figure 2-10 (b).

(a) Audio Graffiti for an indoor space **(b) 3D visualization of the installation**

Figure 2-10: Demonstration of Audio Graffiti (adapted from Settel and Wozniewski, 2010)

2.7.3 eMotion - Mapping Museum Experience

eMotion - Mapping Museum Experience was designed to see how the perception of art can be measured at the Kunst Museum St. Gallen in Switzerland (eMotion, 2009).

Visitors who volunteered to take part in the experiment were outfitted with a special glove (Figure 2-11) containing a Ubisense active UWB tag that broadcasted the wearer's location four times per second via a 6-8.5 GHz RF signal. Each glove also contained sensors that tracked the electrical conductivity of the wearer's skin, as well as that person's pulse. The electrical conductivity information was used as a proxy measure for cognitive stimulation, while the heart-rate data served as an indicator of emotional excitement.

Figure 2-11: Glove containing UWB tag, electrical conductivity sensor and pulse sensor (eMotion, 2009)

The UWB tags transmitted the location data to Ubisense Series 7000 UWB readers, and then forwarded that information to Ubisense's Location Platform, which has a standard Application Programming Interface (API) that can export the data to other applications. The biometric data were transmitted via wireless LAN to a MySQL database, where they were merged with the location information. The location data were then used to determine which specific artworks the visitor had been viewing, and for how long. By pairing the location and biometric data, the researchers could measure a participant's biological and cognitive reaction to each artwork. Before entering the exhibition, the visitors were asked a series of questions, such as their age and nationality, whether they had heard of the research project and their level of interest in art. In order to calibrate the biometric sensors, the volunteers were also asked questions such as whether they smoke cigarettes or recently drank a cup of coffee. The visitors then made their way through the exhibition. Once they reached the end, they were offered the chance to view a computer screen displaying their tracks through the museum. That location data, merged with the biometric data, showed the visitors which artworks provoked cognitive or emotional responses (Figure 2-12). Each museum patron was then asked whether the recorded response matched his/her impression of how he/she responded.

Figure 2-12: Flow of participants' movements in the Museum (eMotion, 2009)

2.8 Summary

A wide range of literature in the field of interactivity, locative media, tracking technologies and RTLSs, and multimedia environment and devices are reviewed in this chapter. We explore human-location interaction design from both technical and philosophical point of view in Sections 2.2 and 2.3, respectively. What we investigated in Section 2.2 is how we experience the real world through interaction, and in Section 2.3, how places make the world meaningful when we experience the world. The interactivity could provide a key milestone for the study of tracking technologies. We found that numerous researchers have undertaken experiments in various technological approaches for human-location interaction. Their projects draw our attention to what we do when we experience our life and the world and to how we perceive the world through interaction. These researches concluded that there is a close relationship between experience and place, and brought the idea of combining location information and responsive space. We therefore envision a central theme of this interdisciplinary research where location information is used in the context of daily interaction.

Furthermore, the ongoing technological development of tracking technology and RTLSs are reviewed to understand the potential of the technology in Sections 2.4 and 2.5. This review helps envisioning the designs and manifestations of the tracking technology for indoor environments. Several RTLSs are investigated including RFID and UWB to track moving objects. The UWB technology can provide a relatively high accuracy and applying UWB in everyday situations is at preliminary stage. In Section 2.6, we presented an overview of the multimedia environments and devices. Finally, three case studies are reviewed for the collaboration of multimedia devices in Section 2.7.

It should be noticed that the definition of interaction design is very broad. The definition spans from small mobile devices to design of interaction environments, and even organizations. This points out that we need appropriate scenarios and frameworks based on observations and analyses of location interaction for the individual spaces. Given this definition, including the various focuses on users, spaces, devices, and systems, we propose our approach in Chapter 3.

CHAPTER 3   PROPOSED APPROACH

3.1 Introduction

In this chapter, using the characteristics of the UWB system, we aim to establish a framework of UWB-based Responsive Multimedia Space (URMS). Our research starts by identifying the concept of URMS. Then, based on the proposed concept and the literature review in Chapter 2, we define the requirements for URMS. Those requirements are categorized into two groups: (1) requirements for achieving an accurate, scalable and near real-time system, and maintaining visibility of UWB tags, and (2) requirements for achieving an acceptable latency of performance of the responsive space. Next, we discuss the methodology for realizing URMS, including the system design, the execution flowchart, and the event control techniques. The event control techniques cover grid-based zone detection, location parameters and event prioritization method. Those ideas are integrated in a core module for managing multimedia event outputs.

3.2 Concept and System Design Analysis for URMS

The concept of URMS aims to create a multimedia space incorporating UWB technology. The scope of the responsive space is vast and could include every space capable of sensing and responding accordingly to existing entities. URMS is thus expected to interact with those entities, which can be people or any sort of identifiable objects. Based on basic research in human interaction in Section 2.2 and locative media in Section 2.3, we define that URMS is able to find the presence of entities that reside in the space, get their location, and eventually utilize the location data to control multimedia outputs such as video and lighting.

The framework of URMS assumes that physical entities, including people and objects, are equipped with UWB tags so that they are automatically locatable, with the spatial relationships continuously changing. A significant aspect of the UWB system is its capability of accurate tracking of a large number of tags. Furthermore, the tags are relatively small so that they can be embedded in objects for user-centered interaction design. Thus, URMS is intended to encourage people to immerse themselves in media interaction at their living or work place with minimum constraints.

Figure 3-1 shows the system design of URMS, which represents the flow of data communication from tagged objects to multimedia devices. URMS allows us to track the locations of tags and communicate with a predefined list of events through its interface. The system consists of two main modules: (1) the monitoring module which manages tag information and transmits tags' IDs and the corresponding calculated location coordinates to the multimedia module, and (2) the multimedia module which executes the location-based events, such as lighting control and video streaming, according to the event list. This simple system design takes advantage of the high-performance UWB system for indoor localization, compared with other location tracking technologies reviewed in Section 2.4.

As discussed in Section 2.6, multimedia devices can provide various outputs, such as displaying videos, images and graphical objects, playing audio clips, and combining, mixing and distorting them based on aggregated tag information. As an example of graphical events, the tag location can be rendered with a dynamic representation of the instantaneous location data, such as spheres with varying sizes and colors. From interaction design point of view, the dynamic interaction continuously evolves user's experience in a seamless way. Therefore, the choices of media types and the form of outputs made by the designer play an important role for enriching users' experiences.

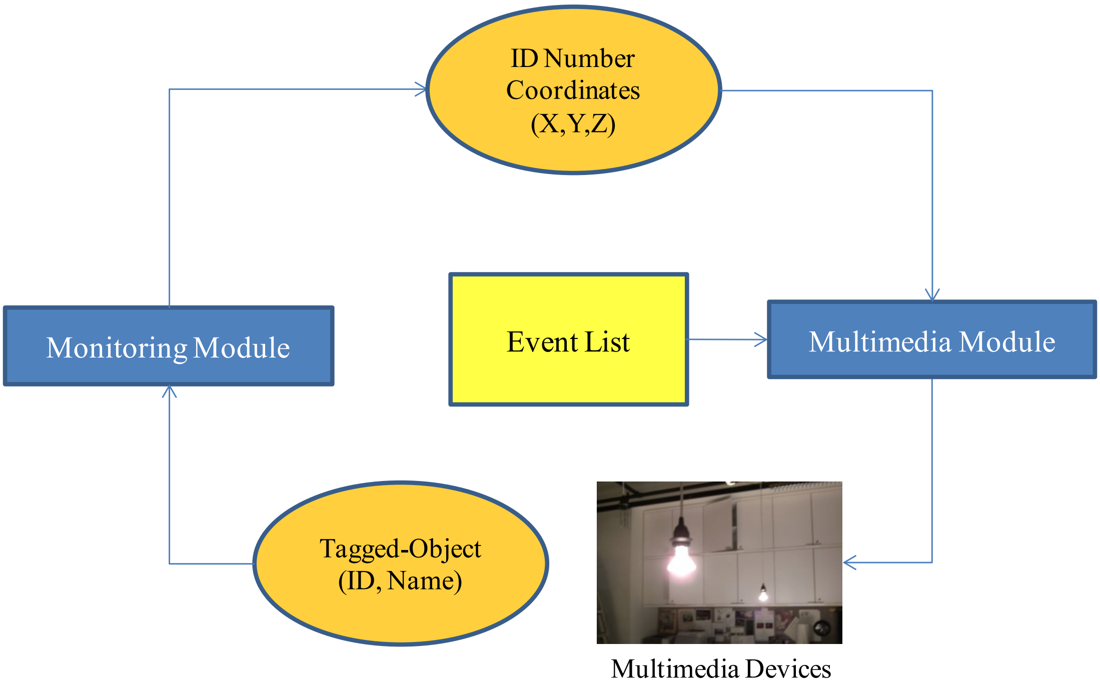


Figure 3-1: System design of URMS

Figure 3-2 illustrates a specific scenario of URMS that provides the location-based events at the right place and the right time, according to the situation. Tagged users/objects (e.g., a person, a cup of coffee, a chair, etc.) are located so that dynamic interactions in the physical space occur during activities of people in the space. In this figure, four UWB sensors are attached at the corners of the room. As an example of events, two lighting units above the table can be switched on or off, and the light intensity can be controlled based on the spatial relationships of the tagged objects and the users. The screen attached to the wall is used to display videos based on the location and preference of the user. Other interactions can be envisioned such as drawing visitors' attention by lighting effects.

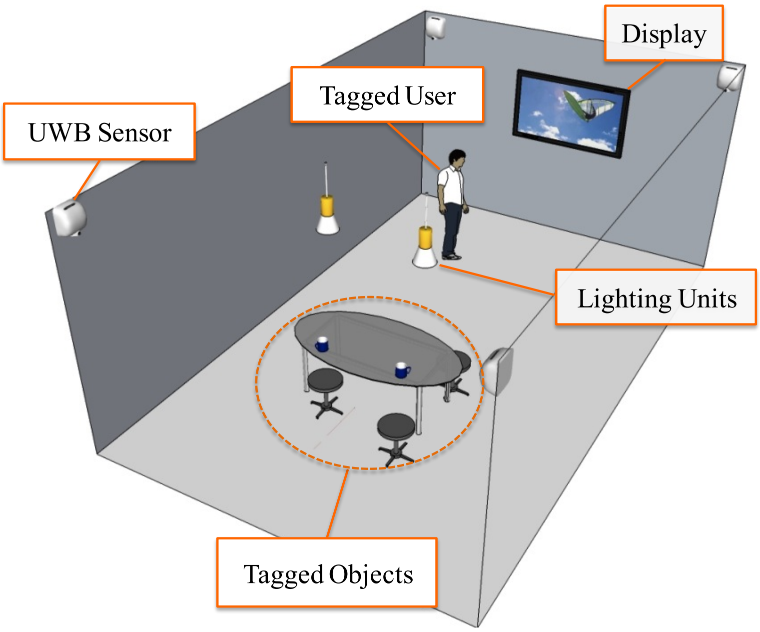


Figure 3-2: Example scenario of URMS

In this way, considering the interaction between objects and individuals, or groups of people, URMS can be applied to make the space more interesting (e.g., audiovisual events), friendly (e.g., turning on the light when a person approaches a desk lamp), or environmentally efficient (e.g., adjusting the light intensity). Close observation could yield valuable clues to establish a comfortable and user-friendly URMS. In the case studies in Chapter 4, we demonstrate some potential scenarios.

3.3 Requirement Analysis

3.3.1 High level requirements of URMS

By defining the requirements, a framework of URMS is designed as a general infrastructure for any space, enabling multimedia devices to be integrated with the UWB system. Based on the system design of URMS outlined in Section 3.2, we define the following specific requirements for the development of an ideal URMS system.

Firstly, we believe that URMS should not disturb the mobility of the user or the original purpose of using an object. For example, the appearance of an exposed tag attached to a cup may affect user's routine activities of drinking coffee. Therefore, the UWB tag attached to an object should be hidden so that the tagged object fits with its surrounding. Secondly, the location-based events should be linked with a natural course of action taken by the users. Based on the notion of human interaction, discussed in Section 2.2, any interaction with a tagged object/user requires a cycle of experience when people perceive, think and act. For example, if a coffee cup is used as a tagged object, URMS has to give feedback to the user in a timely fashion by taking advantage of physical routine activities corresponding to body movements of drinking coffee. Finally, the multimedia outputs should be designed to simply and quickly captivate the attention of the users involved in the spatial relationship with other tagged users/objects. For example, when some users begin a conversation around a table, the light gets brighter, or the volume of a TV is turned down gradually, based on their locations. If they cannot understand what is happening, then URMS ends up interfering with their conversation. In addition, the location-based events should be transparent to the user. To summarize, there are three primary requirements for URMS: (1) The system should support automatic location detection of users in a seamless way; (2) The system should respond to the spatial relationships as they arise, resulting from bodily actions; and (3) The location-based events should be easy to define and understand.

Based on these observations, the following subsections explore the technical requirements for the UWB system in Subsection 3.3.2, and for the location-based continuously changing multimedia events in Subsection 3.3.3.

3.3.2 Requirements for UWB system

Accuracy

Accuracy is an essential requirement of location-aware systems. As explained in Subsection 2.5.1, AOA and TDOA are used in UWB RTLSs to locate tags. If only AOA method is used, two sensors are theoretically required to locate a tag in 3D. However, to improve accuracy, more sensors are needed in practice to reduce the influence of multipath and noise (Muñoz et al., 2009). If only TDOA method is used, at least three sensors are required for 2D positioning and four sensors for 3D positioning (Ghavami et al., 2004). The combinations of the location method and the corresponding results are compared in Table 3-1. The combination of AOA and TDOA should be applied to get the highest accuracy. In practice, the more sensors are used, the higher the accuracy and availability, leading to a more robust solution.

Table 3-1: Combinations of the location methods and the results (adapted from Ubisense, 2010)

Location method	Number of sensors detecting tag	Other information required	Result
Single-sensor AOA	1	Known height of tag	2D horizontal position (+ known height)
AOA	2 or more	None	3D position
TDOA+AOA	2 or more	None	3D position (highest accuracy)
TDOA only	3	Known height of tag	2D horizontal position (+ known height)
TDOA only	4 or more	None	3D position

To gain accurate location data, accurate measurements of the location of sensors are required. A local coordinate system is defined, and the coordinates of each sensor should be measured precisely using surveying tools, such as total stations. With firmly fixed levelled sensors (zero roll angle), the calibration can be done so that the pitch and yaw angles of each sensor can be calculated and recorded in the system. Data filtering, as discussed in Subsection 2.5.5, is another important aspect of positioning accuracy, which reduces errors in near real-time and improves the accuracy.

We performed static tests in order to verify the accuracy of the system taking into account three variables: (1) Location and number of sensors (2 to 4), (2) UWB method, AOA only or combination of AOA and TDOA. Figure 3-3 shows the connections between four sensors and the floor plan of the lab where five points were chosen to compare the accuracy.

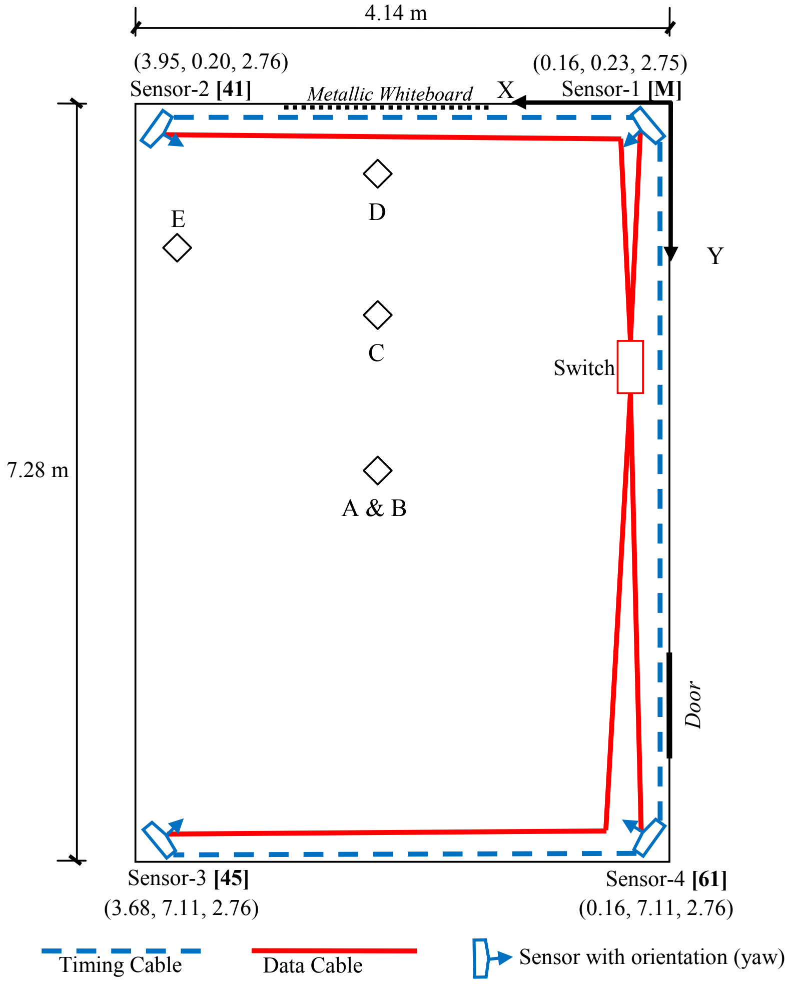


Figure 3-3: Lab setting

Figure 3-4 shows the laboratory setting in 3D. Tag A is at the center of the lab, tag B is lower in height, tag C is closer to the wall, tag D and tag E are near the walls where they are in proximity to metallic objects. The solid lines show the data cables connecting the sensors with the Power over Ethernet (PoE) switch, whereas the dotted lines show the timing cables. Sensor-1 acts as the master sensor.

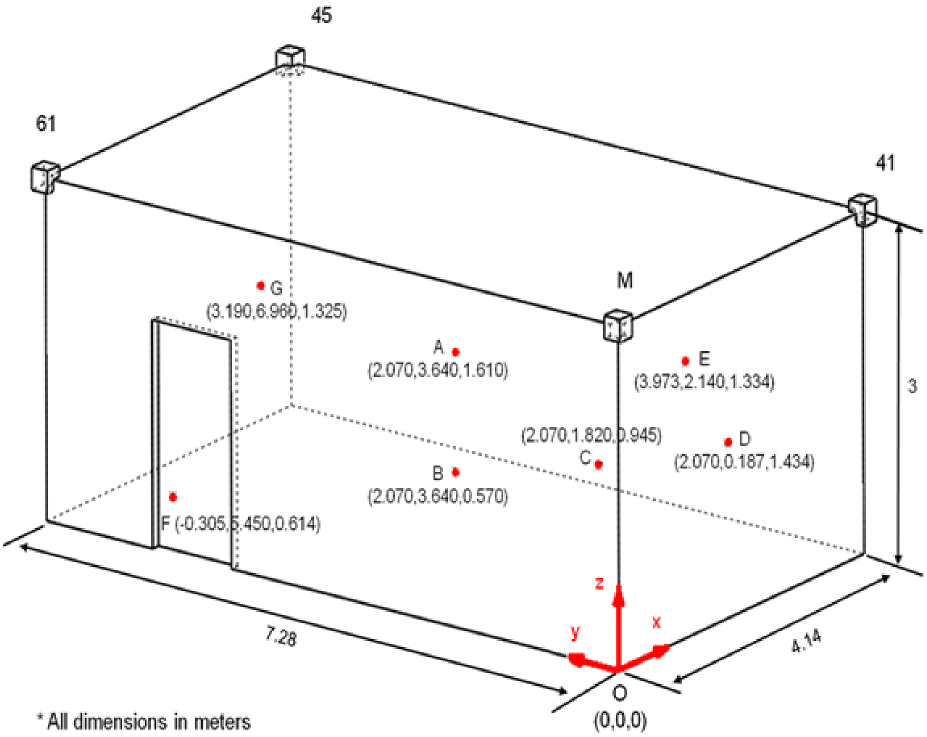


Figure 3-4: Laboratory setting in 3D (adapted from Kaushik, 2009)

Figure 3-5 shows the accuracy for each combination and point sensed. The naming convention of the results is explained as follows: number of sensors, methods used, and sensors used. For example, 3S_AOA_TDOA_M_41_45 means that 3 sensors were used (3S), both AOA and TDOA were used (AOA_TDOA), and the used sensors are the master sensor, and sensor 41 and 45 (M_41_45).

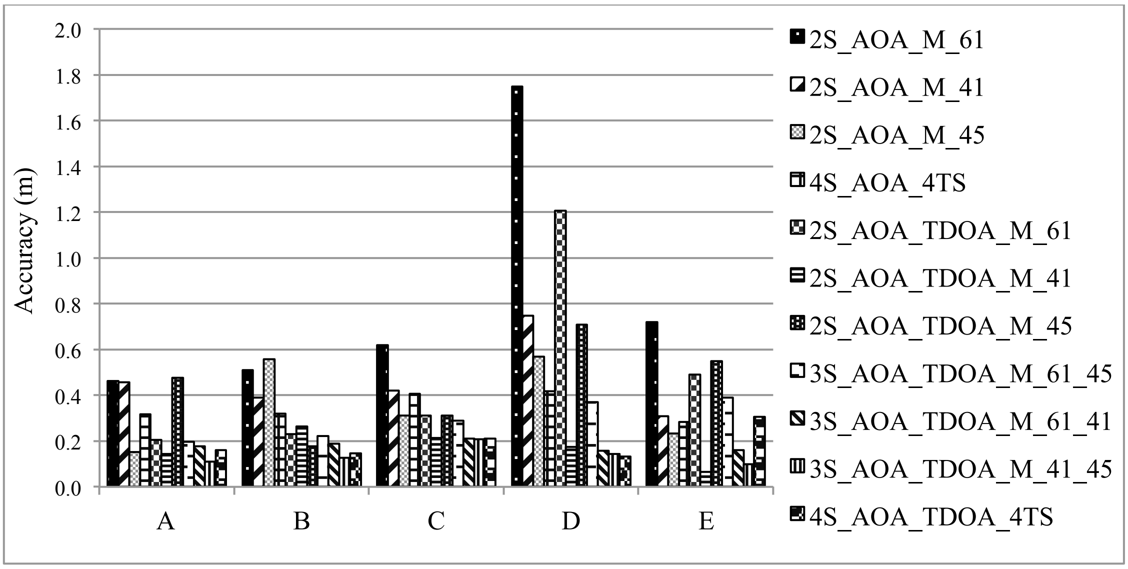
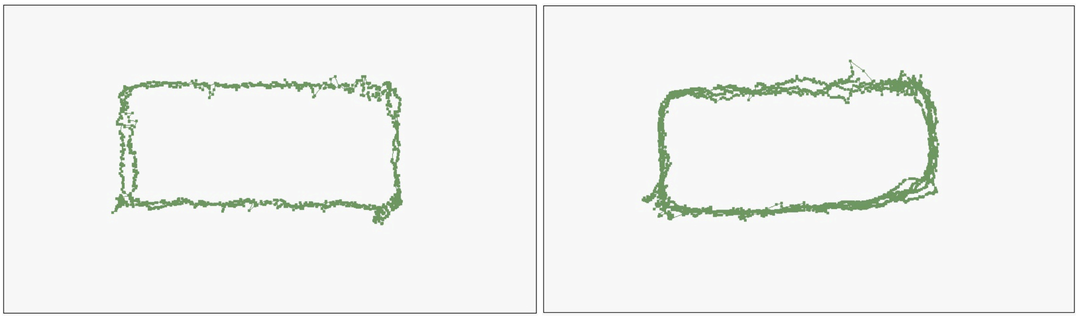


Figure 3-5: Accuracy comparison (Motamedi, 2010)

It was expected that with more sensors and by using both methods (AOA and TDOA) the accuracy would be better. This was true for almost all the points sensed. Four sensors using the combination of AOA and TDOA give the best results for accuracy followed by three sensors and two sensors. These results show that more sensors and a combination of both methods must be used to obtain accurate results. Another analysis is related with the position of the tags; in most of the cases position detection of tag A is more accurate than tag B and tag C. The results for tag D are the worst because of the reflection of RF from the walls and metallic objects. This means that the monitored objects near the center of the cell are located more accurately.

Another test was undertaken using three sensors to compare the performance of locating a mobile tag using: (a) the combination of TDOA and AOA, and (b) only AOA. The sensors in (a) were connected by both data and timing cables to the location server, whereas in (b) only data cables were used. Table 3-2 shows the update rate comparison between (a) and (b). The update rate of the system was set to the highest value of 38 Hz because only one tag was tracked in the monitoring area. Figures 3-6 (a) and (b) show the two cases, TDOA and AOA, and only AOA, respectively. It was expected that more missing readings would result when using AOA only. However, as can be seen in the table, the actual average update rate in both cases was almost the same (about 33 Hz). The figures show the path of a person wearing a hard hat with a tag attached to it. The person walked along the same rectangular path. By doing these tests, the location data of a moving tag on the same path were analysed to see if the accuracy of the AOA-only mode was acceptable. Although the two paths are similar, the path in Figure 3-6 (b) is noisier than that in (a). However, the difference is negligible for most URMS applications, which means that we can take advantage of the wireless setting that can be done only in the case of the AOA mode.



(a) TDOA+AOA **(b) AOA**

Figure 3-6: Comparison between TDOA+AOA and only AOA

Table 3-2: Update rate comparison between TDOA+AOA and only AOA

UWB method	**Readings**	**Time (sec)**	**Average Update Rate (Hz)**
TDOA + AOA	1748	53.384	32.7
AOA	2978	90.495	32.9

Scalability and Near Real-Time

Since in commercial UWB systems there is only a single channel used in time division mode, only one tag can be located at a time in each sensor cell (Zhang et al., 2009). A suitable number of tags used in the same area should be decided based on the frequency of the system and the size of the sensor cell. According to Zhang et al.

(2010), the number of time slots per second depends on the frequency of operation of the UWB system. For example, in Ubisense system with a nominal cell frequency of 160 Hz, one second is divided into 153 time slots, and each time slot has duration of 6.5 ms (Appendix A). Different slot intervals can be selected to determine how often the tags' locations are updated, and how often the system listens to data and schedule messages from the master sensor. The shortest slot interval can be set to 4 slots, which means the update interval is 26 ms, corresponding to a maximum update rate per tag of approximately 38 Hz.

The update rate of tags will decrease to allow the system to cover all tags with the fixed total number of time slots when there are a large number of tags in the cell. For example, if the time slot is set to 4 and only 4 tags are in the cell, the four tags are updated every 26 ms (38 Hz). When more tags are detected in the system, e.g., 8 tags, the update rate is decreased to 19 Hz. The more tags in the system, the bigger the slot interval should be selected, and the lower the update rate. A specific update rate can be set for an individual tag or a group of tags. One consideration when setting the update rate is the moving velocity of the object. Objects with high velocity need more frequent updates to accurately track their traces. Therefore, selecting a suitable number of tags with an appropriate update rate based on their velocity is essential for achieving the balance between the conflicting requirements of visibility and accuracy in near real-time.

Table 3-3 shows the update intervals and rates for different slot intervals for Ubisense 160 Hz system (Ubisense, 2009). Appendix A presents the complete update intervals for 160 Hz systems.

Table 3-3: Update intervals and rates for different slot intervals for Ubisense 160 Hz system (adapted from Ubisense, 2009)

Slot interval	Update interval (ms)	Nominal update rate for each tag (Hz)
4	26	38
8	52	19
16	104	10
32	208	5
64	416	2.4
128	832	1.2
…	…	…

To maximize the update rate, Zhang et al. (2010) introduced a heuristic rule as follows:

$$m \leq 4 \times 2^n \qquad (3.1)$$

Where m is the number of tags in the system, n is the minimum value that meets the inequality, and 4×2^n is the time slot interval that should be set. For example, if there are 10 tags in the system, the minimum value of n is 2; therefore, the time slot interval should be set to 16 (i.e., update rate of 9.6 Hz).

On the other hand, if the update rate is defined, another heuristic rule can be defined as follows:

$$R/r \geq 4 \times 2^n \qquad (3.2)$$

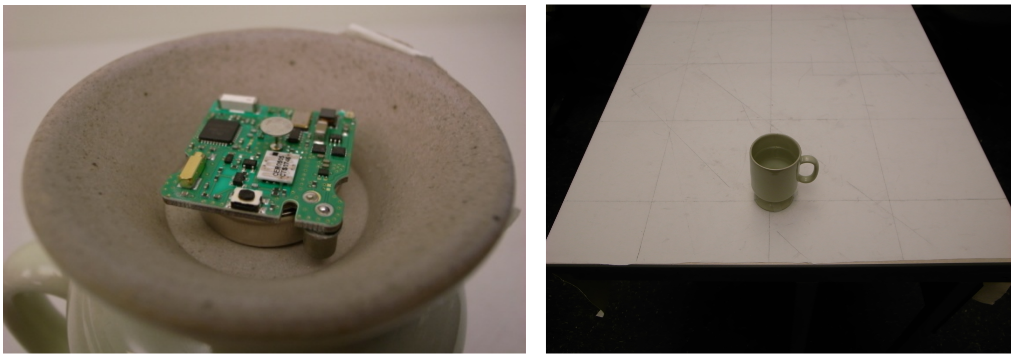
Where R and r are the update rates of the cell and the tags, respectively, n is the maximum value that meets the inequality, and 4×2^n is the time slot interval that should be set. For example, if an update rate of $r = 8$ Hz is required for the tags, the maximum value of n is 2, and the time slot interval can be set to 16. According to inequality (Equation 3.1), a maximum of 16 tags can be used in the system to obtain this update rate. Similar inequalities can be derived for other UWB systems.

Visibility

The location of the sensors' and tags' placement should be considered as a way to improve the visibility and thus to increase the probability of detecting the tags. Sensor installation is significantly constrained by the interior structure of the building. Basically, the sensors, whose field of view may be different from one UWB system to another, should be set in a way to utilize their antenna pattern both in the azimuth and the elevation. As explained in Subsection 2.5.2, the maximum range of sensors can be potentially up to 300 feet; thus a reasonable monitoring area should be defined considering the coverage of the sensor cell. If the area to be covered is big, more sensors are needed using one or more cells.

In commercial RFID systems, some tags are designed to be worn by a person as a badge, others are designed to be robust for industrial purposes. They are usually covered with a hard plastic case. Based on our high-level requirement that the UWB system should support users in a seamless way, the tags should be hidden as much as possible from the user but should be visible by the sensors. For example, a tag can be attached to the bottom of a cup. On the other hand, we should test whether the tags are properly detected because metals, liquids, and even human body decrease accuracy.

To verify the visibility of a tag embedded in a cup, we performed static tests using a cup made of ceramic in the same laboratory illustrated in Figure 3-3. The tests considered three different states of the tag: (1) exposed tag on a table, (2) tag attached to the bottom of the cup (Figure 3-7 (a)) and (3) tagged cup filled halfway with water (Figure 3-7 (b)). The sensor setting and the floor plan of the lab were the same as in Figure 3-3. Raw data were collected including the tag ID, date, time, and the x, y, z coordinates. Each set of data was collected three times each with 2,000 readings using a tag update rate of 38 Hz. The actual location of the tag measured with a laser meter is (2.00, 2.40, 2.75). Table 3-4 shows the results of the three tests, represented as Tag, Tag_Cup, Tag_Cup_Water, with the average of given location coordinates (x, y, z), the difference of the average and the actual location, the maximum and minimum values and the standard deviation. Figure 3-8 shows the results of each test represented as a series of points. According to the variations in each data, it can be seen that the ceramic cup reduced the precision, and the combination of the ceramic cup and water further reduced the precision. Thus, the designer has to consider where and how to place the tags in the space in terms of the characteristics of the UWB technology.

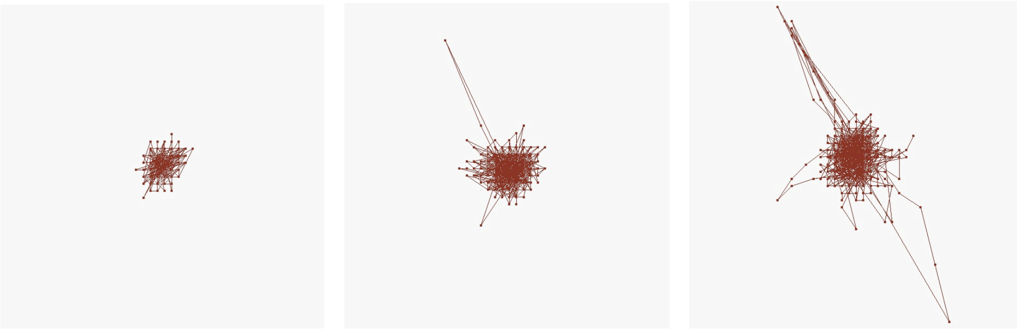


(a) Embedding a chip at the underside of the cup (b) Tagged cup of water placed on the table

Figure 3-7: Tagged cup used in the static tests

Table 3-4: Results of the static tests using a tagged cup

		Average (m)	Ave. - Ref. (m)	Max (m)	Min (m)	Standard Deviation (m)
Tag	x	2.003	-0.003	2.05	1.97	0.009
	y	2.415	-0.015	2.46	2.37	0.009
	z	0.761	-0.011	0.78	0.72	0.008
Tag_Cup	x	2.041	-0.041	2.09	1.95	0.013
	y	2.395	0.005	2.58	2.32	0.013
	z	0.747	0.003	0.81	0.65	0.012
Tag_Cup_Water	x	2.035	-0.035	2.17	1.93	0.016
	y	2.394	0.006	2.6	2.16	0.017
	z	0.786	-0.036	1.11	0.57	0.022



(a) Tag **(b) Tag_Cup** **(c) Tag_Cup_Water**

Figure 3-8: Results of static tests represented as a series of points

3.3.3 Requirements for Location-based Continuously Changing Multimedia Events

Latency

As explained in the high-level requirements, the user should be able to easily understand the multimedia events based on his movement. For example, the events can be graphical representation of objects based on the movements of tagged object or continuously changing light intensity based on the distance between tags. Based on the update rate of the UWB system, URMS has an important requirement for the latency consideration of location-based continuously changing multimedia events. The latency is used to evaluate the smooth responsiveness of interactive media, i.e., the time needed to sense changes actuated by the user, to modify its state, and to produce a feedback in the space (Sha et al., 2003).

Using the inequality expressions introduced in Subsection 3.3.1, we can calculate the update rate and the number of tags to maximize the performance of the UWB system. On the other hand, we have to evaluate the acceptable latency for URMS. For multimedia interaction through audio, visual and haptical mode, a late response from the system, relevant to frequency, could possibly ruin the smooth experience of the user. As a consequence of low responsiveness, the user will not be able to understand the intention of the designer or she/he will lose concentration on the events. Sha et al.

(2003) mentioned the correlation between frequency and latency; "Sustaining the illusion of continuous, direct interaction requires the system to support sensor readings with tight latency and update frequency bounds." Table 3-5 summarizes the relationship between the required update frequency of the sensor and latency, based on the human perceptual system.

Table 3-5: Human interaction bounds (Sha et al., 2003)

Type of feedback	Frequency (Hz)	Latency (ms)
Visual	15-25	50
Acoustic	5-10	10-50
Acoustic (melodic)	50	10
Haptic	1000	1

Based on this correlation, we define the minimum acceptable frequency by the following inequality:

$$F > F' \qquad (3.3)$$

Where F is the frequency of the system, F' is the value that meets the acceptable latency based on the bounds in Table 3-5. For example, if the designer uses a display with a video in Figure 3-2, both the UWB system and the display should have an update rate from 15 to 25 Hz. The Ubisense UWB system is able to achieve this rate only if the number of tags is less than 8, according to Table 3-3. Using the Ubisense system, the URMS meets the inequality (Equation 3.3) for visual and acoustic events because its maximum frequency is 38 Hz when the number of tags is 4 or less. If we have more tags, extrapolation methods can be used but with the side effect of decreased accuracy/reliability.

3.4 Grid-based URMS

A grid is defined as a spatial data structure that divides the space into a set of zones arranged in rows and columns. Figure 3-9 illustrates an example of the grid-based URMS, in which a tagged object/user location (x, y) is used for representing the zone (i, j) in two dimensions, as shown with the colored panels.

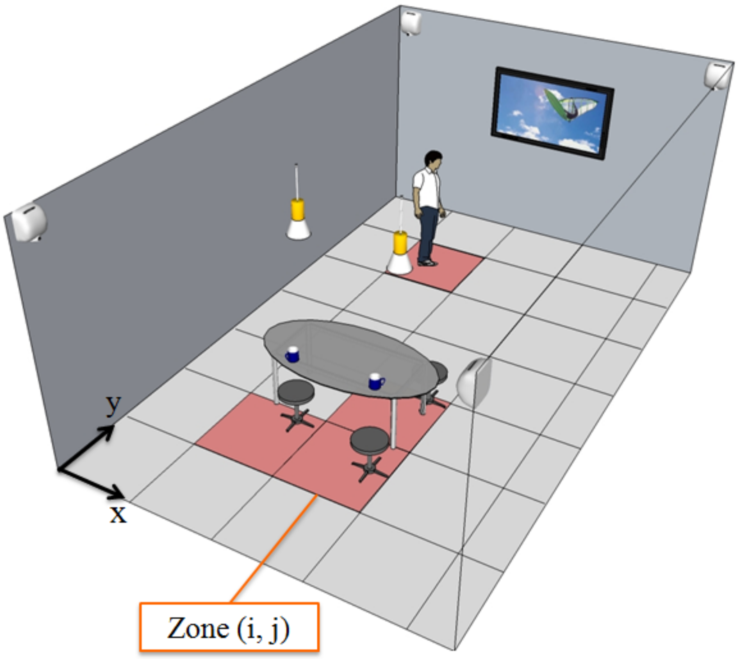


Figure 3-9: Grid-based URMS

Although the UWB system provides 3D coordinates (x, y, z), in this research, only 2D coordinates (x, y) are used. As Figure 3-10 shows, when a tag, *k*, being tracked moves into a location (x, y), its *zone* (i, j) is the zone including that location. The figure shows the case of zones of 1 × 1 m. From a perspective of interaction, the physical space is divided into a grid of 1 × 1 m because this size is considered suitable to trigger events when the user is in close interaction with other objects.

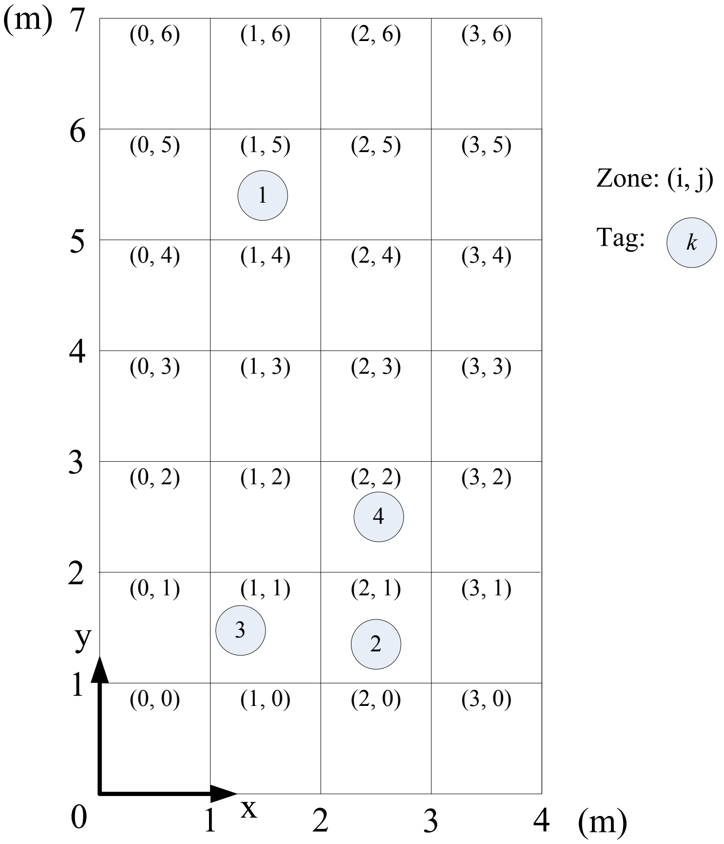


Figure 3-10: An example of grid-based physical space

3.4.1 Flowchart of URMS

The flowchart of the algorithm for URMS is illustrated in Figure 3-11. The algorithm has three modules: Event Definition Module, UWB Monitoring Module, and Multimedia Module. The Event Definition Module allows each user to define his/her events and corresponding regions using a touch display. The UWB Monitoring Module is used for collecting the readings of tags. Associating the tags' IDs with object names, this module acquires the location coordinates of the tags computed by the UWB system. Then, the tag data including the object name and the location coordinates are transmitted to the Multimedia Module. In the Multimedia Module, the tag data are processed to detect a zone by the zone detection function, and then the tag data (object name, location coordinates and zone) are broadcasted to each event-generating function of multimedia devices. The tag data are used to generate multimedia events such as continuously adjusting the light intensity based on the distance between the tags, or displaying a movie or an image. The details of this module are explained in the next subsection.

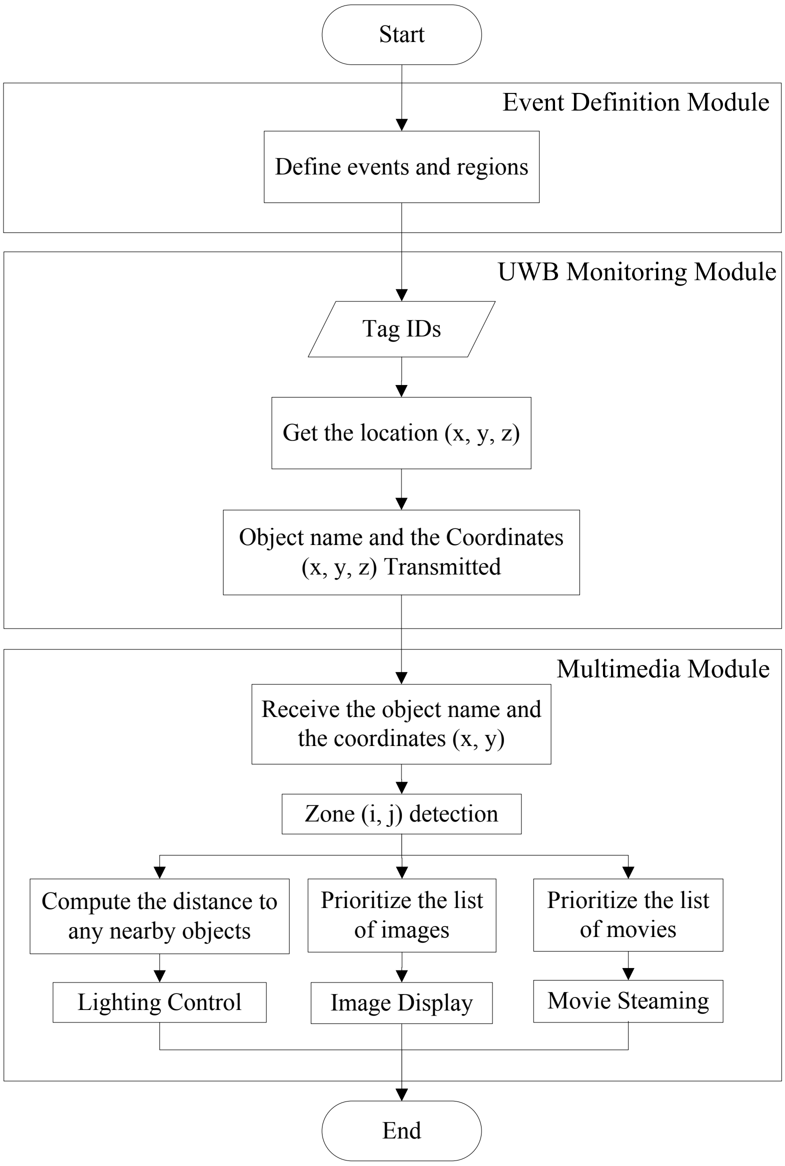


Figure 3-11: Flowchart of URMS

3.4.2 Grid-based Event Selection

Based on the grid of URMS, we aim to create a variety of location-based events with multimedia devices. In this subsection, we explain three steps of event selection methods in the following.

Step 1: Create Events Based on Location Only

As the simplest case, the designer can use the tag location for triggering an event. Figure 3-12 illustrates an example of the grid-based URMS, where there are two *event regions*, Lighting Region and Movie Region. By specifying the relationship between event and location, Table 3-6 simply explains how the event is triggered by having a tag in a zone (i, j). According to Figure 3-12, this URMS displays a movie when a tagged object/user enters the Movie Region, or it turns on the overhead light when the tag enters the Lighting Region.

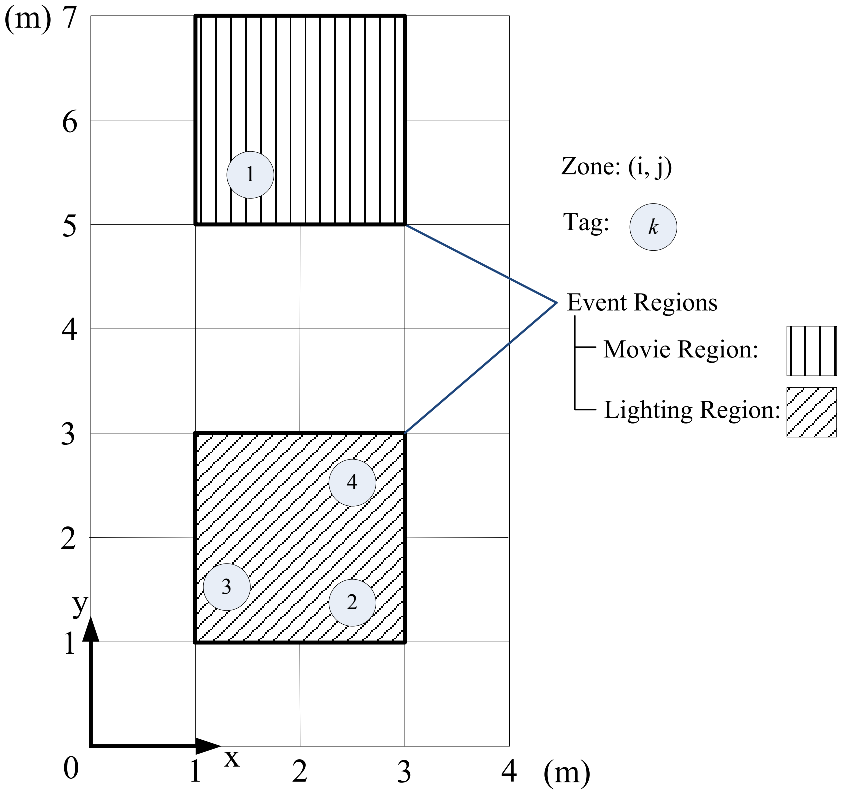


Figure 3-12: Grid-based event regions

Table 3-6: Event list by grid-based regions

Type of Event	**Zone (i, j)**
Lighting	(1, 1), (1, 2), (2, 1), (2, 2)
Movie	(1, 5), (1, 6), (2, 5), (2, 6)

Step 2: Create Custom Events Based on Location and Tag ID

To arrange various options for multiple events, a simple solution is to create an event table of possible combinations of event-triggering zones and tags. Table 3-7 shows how the events for tagged objects are associated with each tag based on location. According to this table, URMS considers the different regions, four tags and four events, including two different kinds of movies and two lighting effects registered in the event list. As can be seen in the table, each tag has different combinations of the events and the event-triggering zones. It should be noted that the event zones (light vs. movie) in this case are still consistent with the event regions shown in Figure 3-12.

Table 3-7: Examples of region definitions for combinations of tags and events

Tag *k* / **Event Output**	**1**	**2**	**3**	**4**
Lighting 1	(1, 1), (1, 2)	(2, 1), (2, 2)	(1, 1), (2, 1)	(1, 2), (2, 2)
Lighting 2	(2, 1), (2, 2)	(1, 1), (1, 2)	(1, 2), (2, 2)	(1, 1), (2, 1)
Movie 1	(1, 5), (1, 6)	(1, 5), (1, 6)	(2, 5), (2, 6)	(2, 5), (2, 6)
Movie 2	(2, 5), (2, 6)	(2, 5), (2, 6)	(1, 5), (1, 6)	(1, 5), (1, 6)

Step 3: Event Prioritization

So far in this subsection, we explained the relationship between tags and grid-based events. Next, we further explore the scope of the event selection method that prioritizes the events. As discussed in Steps 1 and 2, the outputs can be determined by the relationship between tag ID k and zone (i, j). However, URMS can create chaos in the space when many events are simultaneously executed by a large number of tags. Therefore, URMS should make decisions on prioritizing the events. From a practical standpoint, an event output corresponds to a media device. For example, a movie should be displayed on a screen, or a music clip should be played on a speaker, without mixing and overlaying visual and sound streams on the same media device.

Therefore, we investigate an event prioritization process based on the majority of triggered events. Once an event is triggered by a tag, other events are not triggered until the first event is finished. During a short period (e.g., 0.1 sec), URMS counts the number of occurrence of each event and sort the list of triggered events by the number of occurrence. Then, URMS triggers the event that has the largest number of occurrence as the most preferable event to the present situation. Figure 3-13 and Figure 3-14 illustrate an example where the event prioritization method is applied. Suppose Movie 1 is triggered by tag 1 in the Movie Region at time T_1, where the event selection process is based on the first-come-first-serve basis. During the movie, three more tags enter the region so that URMS counts the triggered events (1 for Movie 1 and 3 for Movie 2). As the result of sorting the events in the list, Movie 2 is selected for the group of tags at time T_2. This method is similar to majority voting.

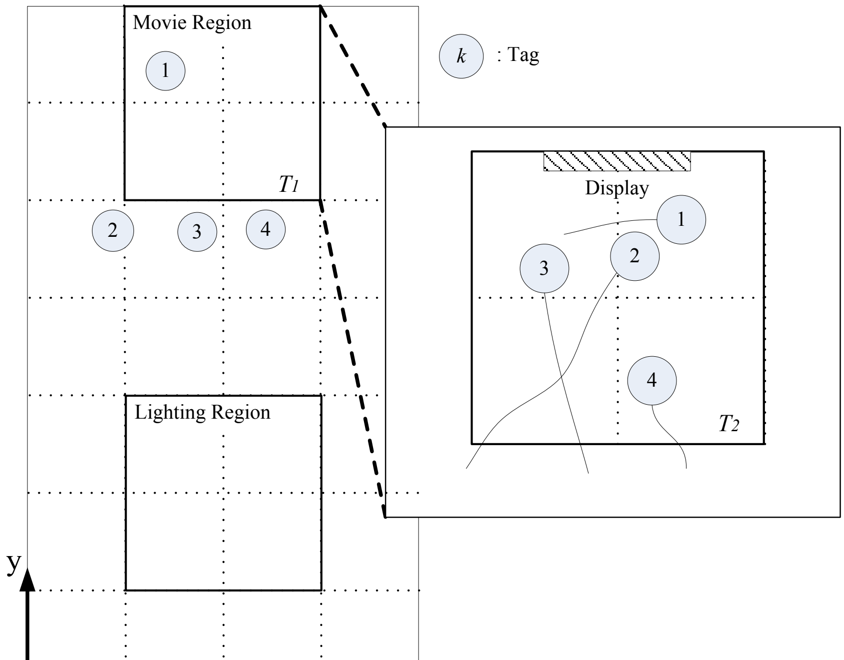


Figure 3-13: An example of event prioritization

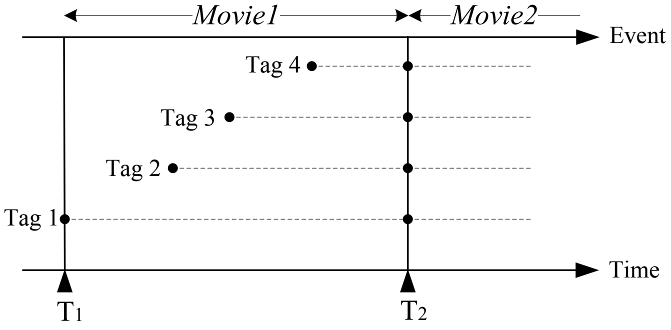


Figure 3-14: Real-time event selection

When the number of each occurrence is the same, URMS can randomly selects one of the triggered events. If several multiple media devices are installed in the space, parallel execution is possible for non-conflicting events. For example, a movie event and a lighting event can be triggered at the same time.

Smooth Transition Effect

For user perception of the events, smooth transition effect should be considered in URMS to avoid suddenly cutting short the event and triggering another. Fade transition effect can be used to gently transform the state of event output to be switched on or off. Fading is a primary time-based function, which gradually increases or decreases of the intensity of light, the brightness of display, or the volume of sound, as Figure 3-15 illustrates. The term fade-in refers to gradually changing the level from zero to a predetermined level. A fade-out refers to gradually decreasing the level to zero (Wikipedia, 2010). A cross-fade is when the levels are gradually altered from one setting to another. As an example of cross-fade effect in URMS, the brightness of screen gradually increases when a movie event is activated, and the light around the screen gradually turns down at the same time.

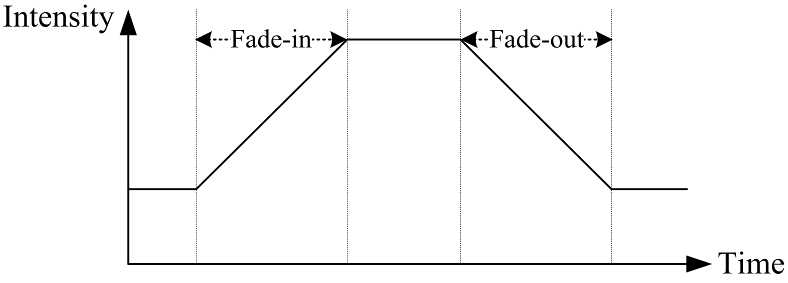


Figure 3-15: Time-Intensity plot for fade-in/out effect

3.4.3 Defining Event Parameters Using Touch Display

Touch interface is a significant medium for human-computer interaction to select and change the contents by simply pointing. Although the main focus of this thesis is not the touch user interface, it is one of the options for the selection method of event definition by the user of URMS in terms of easy control and management. The key to integrate the touch interface with URMS is to provide a simple and intuitive graphical user interface (GUI) for real-time visualization and operation, including event region controller. Using the controller each user can visually select his/her name/ID and then define the event regions associated with the event-output lists to be selected using the touch screen. Figure 3-16 illustrates the prototype design of the touch user interface for URMS. By pointing the controller on the interface, the user can define the dimensions of his/her own event region and set the type of event, categorized as movie clips, music clips and lighting modes, and the specific output. The video and music clips include genres and titles, and the lighting modes include modes of energy saving, spotlight and so on.

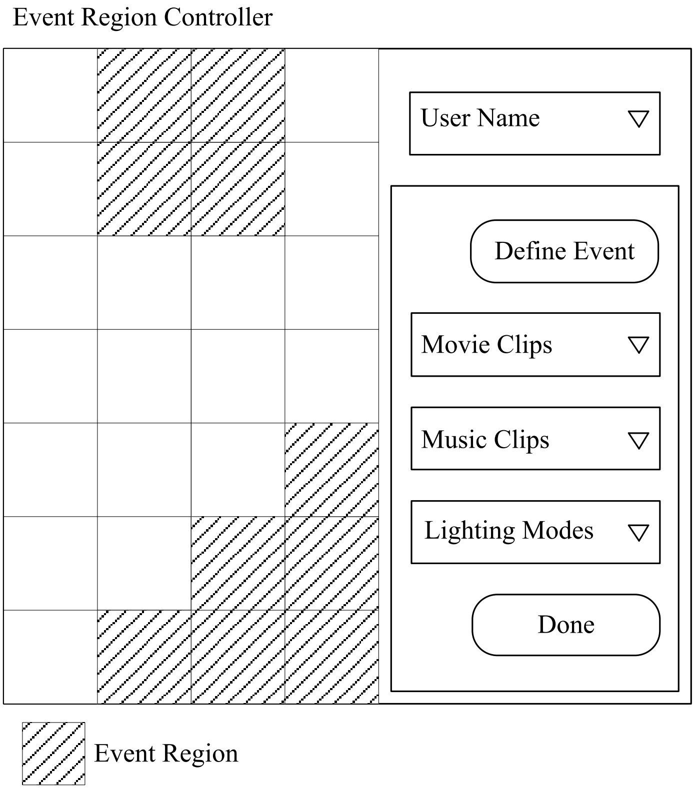


Figure 3-16: Design of touch user interface for URMS

Figure 3-17 shows the flowchart of event input. After selecting the user name, each user can choose his/her own events associated with at specific regions in the space.

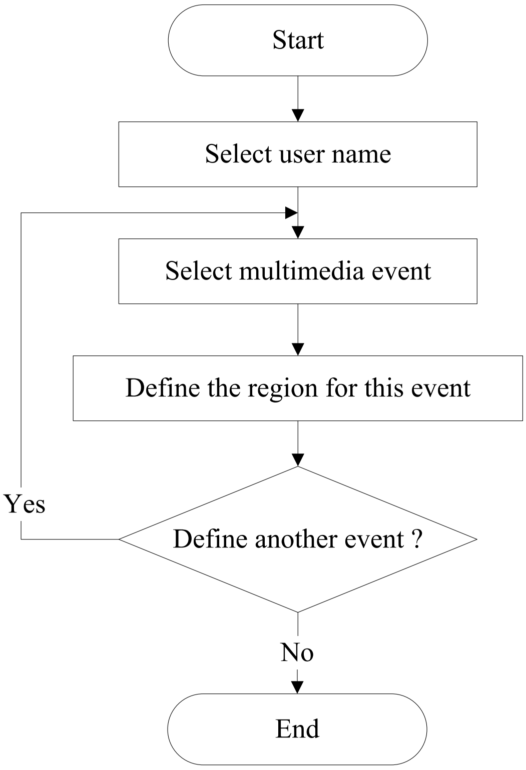


Figure 3-17: Flowchart of event definition

3.5 Continuous Event Control based on Distance

In this section, we explore a methodology for smoothly and continuously adjusting the attribute of multimedia events of the same media type based on distance. As discussed in Section 3.2, continuous interaction is an important element of URMS that aims to make the space more interesting, friendly, or environmentally efficient. To achieve the continuous event control, URMS uses distance between the tags detected in the same event region. Figure 3-18 illustrates a possible situation, in which there are two event regions, Lighting Region and Movie Region. We explain how URMS achieve the continuous event control based on distance in the following.

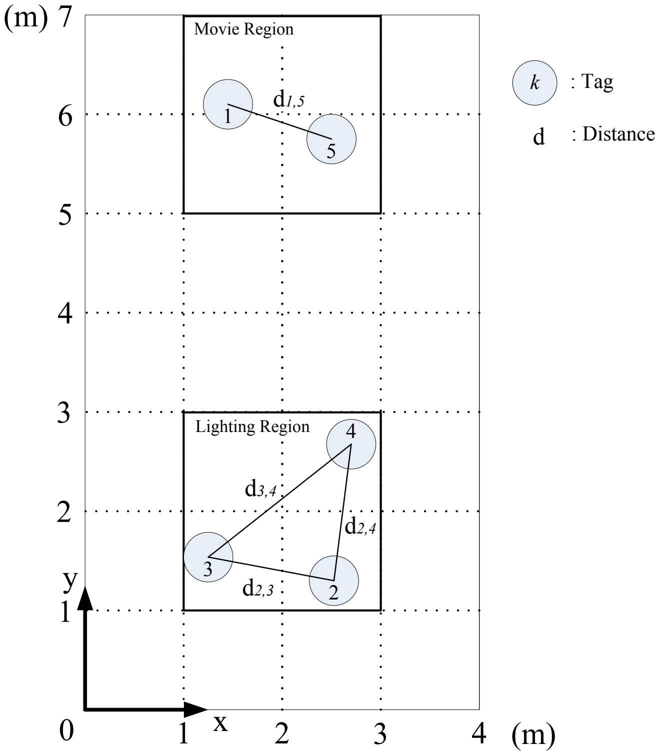


Figure 3-18: An example of the URMS for continuous interaction

URMS can continuously increase or decrease the intensity of light, the brightness of display, or the volume of sound with respect to the distance. This proximity effect engages with the actions the users take in a natural manner. When there are two tags in the region, the proximity value is calculated as the distance $d_{a,b}$ between two tags in the Cartesian coordinate system (Equation 3.4). Then, the distance is dynamically converted to the value of each event output. When there are more than two tags in the region, URMS averages the proximity based on all the combinations of all tags in the region. The calculation process is described by Equation 3.5 for the number of combinations and Equation 3.6 for the averaging of the distances.

$$d_{a,b} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \quad (3.4)$$

$$C = \binom{n}{2} = \frac{n!}{(n-2)!2!} \quad (3.5)$$

$$P = \frac{\sum d_{a,b}}{C} \quad (3.6)$$

The number of possible combinations *C* as pairs from a set of *n* objects is defined by Equation 3.5. $\sum d_{a,b}$ is as the sum of all the distances of all tag pairs, and *P* is the average distance of all combinations of tags.

3.6 Summary and Conclusions

This chapter presented the proposed approach of URMS design. The approach started by defining the concepts of responsive space, and then the requirements of both the technical and conceptual aspects of URMS were discussed. The methods for the grid-based event selection and the corresponding continuous event control function were discussed. As explained in Section 3.2, one of significant aspects of URMS is the ability to support dozens of tags although there is a trade-off between the number of tags and the UWB system performance.

The conclusions of this chapter are:

(1) The concept of UWB-based Responsive Multimedia Space (URMS) has been proposed where UWB RTLS is used for tracking users and objects to trigger

location-based multimedia events. URMS encourages people to immerse themselves in multimedia interaction at their living or work place with minimum constraints.

(2) The high-level requirements for URMS have been defined: (1) The system should support automatic location detection of users in a seamless way; (2) The system should respond to the spatial relationships as they arise, resulting from bodily actions; and (3) The location-based events should be easy to define and understand.

(3) The requirements for the deployment of the system and the location-based events have been investigated. They have been categorized into two groups: (1) requirements for achieving an accurate, scalable and near real-time system, and maintaining visibility of UWB tags, and (2) requirements for achieving an acceptable latency of performance of the responsive space. Furthermore, the system design for satisfying the requirements has been proposed.

(4) The grid-based event selection methods based on tags' locations have been proposed. To support a variety of situations of URMS, we proposed an event prioritization method based on the number of occurrence of each event. Furthermore, a touch user interface was proposed that allows the user to define the events easily and directly.

(5) A method for continuously and smoothly adjusting the multimedia events based on distance was proposed to sustain location interaction with tag locations and movements. The events can give the user an interactive experience such as switching the light on and off, or gradually controlling the light intensity with respect to his/her position in the space or with respect to other objects.

CHAPTER 4 IMPLEMENTATION AND CASE STUDIES

4.1 Introduction

This chapter describes an overall URMS prototype system implementation, and demonstrates two case studies of the developed URMS system. In the first case study, the URMS prototype system is integrated with Ozone system developed at the Topological Media Lab (TML) for realizing location-based events controlled by the UWB system. In the second case study, we integrated the URMS prototype system with a touch display. The following sections describe the details of the implementations and experimentations.

4.2 Software Tools

4.2.1 Ubisense UWB System

To build the URMS system, we used Ubisense RTLS including sensors and tags, whose specifications are shown in Appendix B and Appendix C, respectively. The Ubisense platform is designed according to the client-server architecture, which enables to communicate information between the server machine and client machines, as illustrated in Figure 4-1. By utilizing the Ubisense platform, the URMS system can be integrated with multimedia devices easily and reliably.

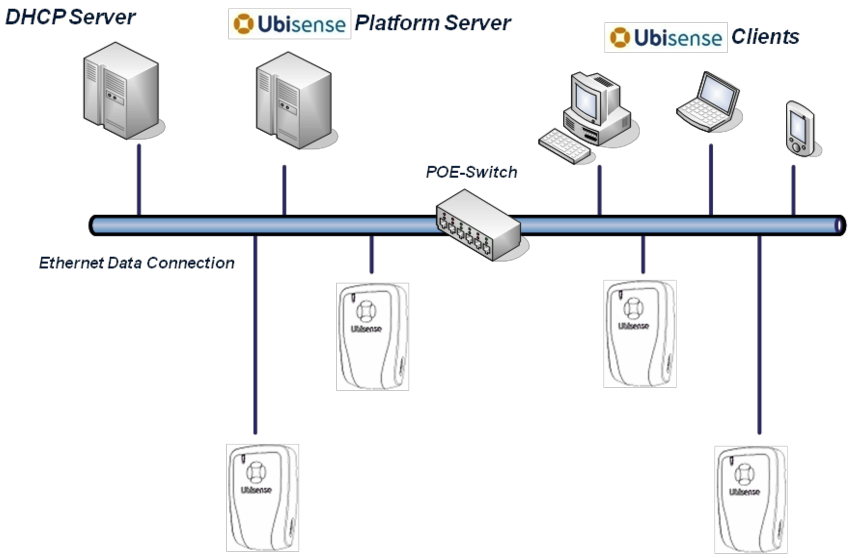
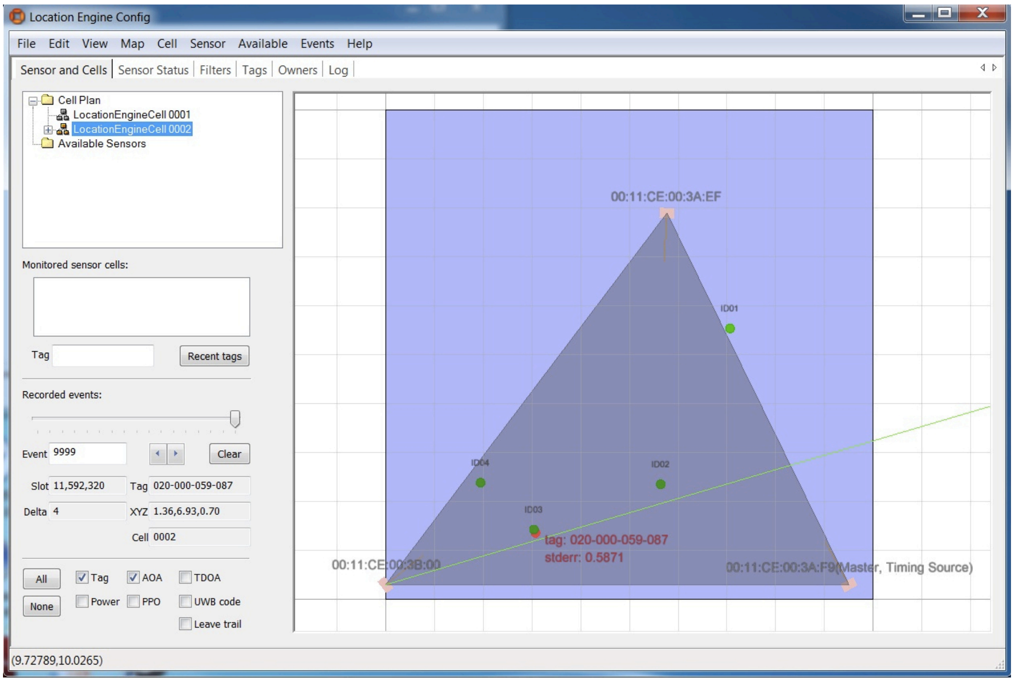


Figure 4-1: Ubisense platform (Ubisense, 2009)

The Ubisense system detects and calculates positions of moving tags. Figure 4-2 shows one of Ubisense applications, *Location Engine*, which monitors the positions in 2D or 3D. Tag locations are represented by green spheres in the figure. The Location Engine also manages and controls sensors, cells, tag status, filtering and so on.



Event 9999

Slot 11,592,320

Tag 020-000-059-087

Delta 4

XYZ 1.36,6.93,0.70

Cell 0002

☑ Tag ☑ AOA ☐ TDOA

☐ Power ☐ PPO ☐ UWB code

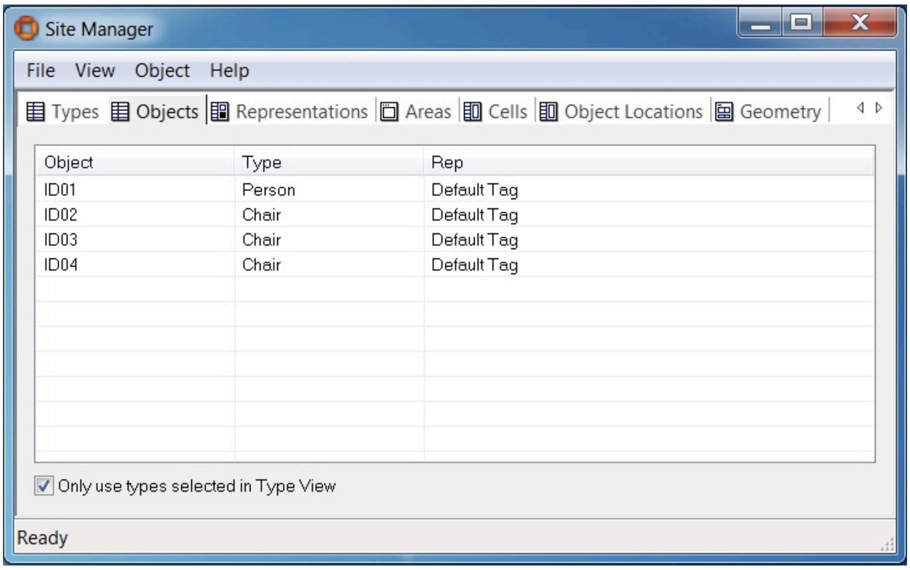
☐ Leave trail

tag: 020-000-059-087
stderr: 0.5871

(9.72789,10.0265)

Figure 4-2: Ubisense Location Engine

Figure 4-3 shows another Ubisense application, *Site Manager*, which manages object names and types, creates representations of virtual objects and spaces, and defines detection cells.



Object	Type	Rep
ID01	Person	Default Tag
ID02	Chair	Default Tag
ID03	Chair	Default Tag
ID04	Chair	Default Tag

☑ Only use types selected in Type View

Ready

Figure 4-3: Ubisense Site Manager

As Figure 4-4 shows Ubisense software applications are fully integrated within Microsoft .NET 3.5 Framework. Ubisense provide .NET API that makes it possible to control every system function. The location platform plays a key role in implementing all the system functions based on the API. The API consists of four libraries: UbisesneLocationEngine.dll, UbisenseLocationServices.dll, UbisensePlatform.dll and UbisenseVisualization.dll. These libraries can be included in any Microsoft Visual Studio .NET projects, and thus the integration of URMS and the Ubisense system is developed by Microsoft .NET programming languages such as C# and Visual Basic .Net.

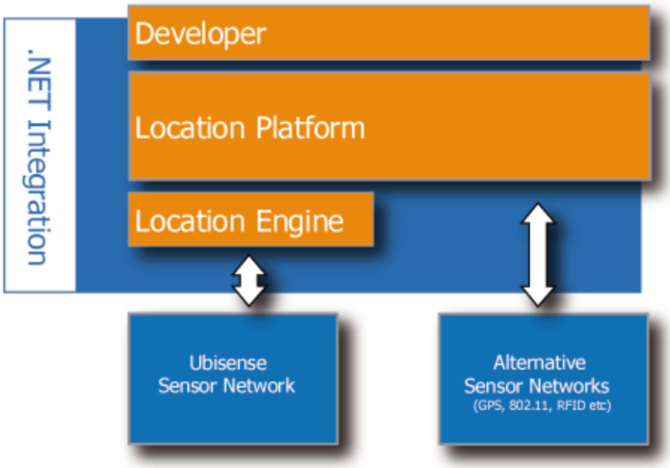


Figure 4-4: Ubisense System Architecture (Ubisense, 2009)

4.2.2 Max/MSP Jitter

Max/MSP Jitter (Max) is a graphical programming environment (Cycling74, 2010). The Max environment is an effective way to work with digital media, in applications ranging from prototypes to finished products (Randy, 2009). Using a flexible message-passing architecture, MSP and Jitter are groups of Max objects, dealing with signals (e.g., sound, light) and matrix data (e.g., video), respectively. Max simplifies integration of different types of computer-based multimedia devices in URMS.

Max programs are constructed by creating intercommunicating groups of objects (MakingThings, 2010). Each object is passed information, does a small portion of the overall work and sends the results. This model is called dataflow. A simple Max program, called a Max patch, is presented in Figure 4-5.

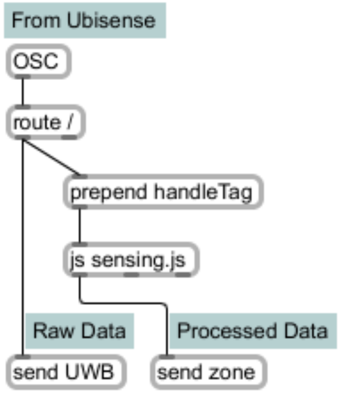


Figure 4-5: A sample Max patch

As shown in Figure 4-5, some objects are connected to each other. In this patch, all process flows are from top to bottom. First of all, the "OSC" object receives and sends tag data as a message. The "route" object passes the message starting with "/" to the next objects, and then the symbol "/" is eliminated from the message. The "prepend" object passes a message adding the following message *handleTag*. Receiving the message with "handleTag", the "js" object executes a JavaScript-based program named *sensing.js*, which executes a specific function named "handleTag". At the end, the "send" object transmits the message to "receive" objects of the same name, which enables to communicate the messages between other patches. To summarize, receiving tag data in the "route" object from "OSC" object, the sample patch sends the unprocessed data to "receive" objects named by *UWB*, and the processed data by the "sensing.js" program to "receive" objects named by *zone*.

By laying out objects and linking them up, users can easily implement a program with Max-based user interfaces and other Max programs, taking advantage of about 500 objects available to users (Cycling74, 2010).

4.3 Prototype System Implementation

The URMS software is a middleware that supports the integration of the UWB system with multimedia devices. The main functionalities of the software are acquiring location data streams, processing them and providing the processed data to computer-based multimedia control modules. As shown in Figure 3-11, the following two modules have been designed in the prototype system:

- ***UWB Monitoring module*:** includes the visualization of the scene and the transmitting of the tag data captured from the sensors.

- ***Multimedia module*:** includes the processing of the tag data and the mapping of the tag data for multimedia device.

By decoupling the UWB Monitoring module and the Multimedia module, multimedia devices connected to URMS are not dependent on a specific UWB system. The Multimedia module is based on Max, which is compatible with Mac and Windows PC, and therefore URMS is a cross-platform system.

4.3.1 UWB Monitoring Module

Figure 4-6 shows a GUI of the UWB Monitoring module, which runs on top of the commercial Ubisense software. The module is written as a set of components using different Ubisense API functionalities, written in C# using Visual Studio 2008. By referencing the Ubisense Dynamic Link Library (DLL) files, the module uses the Ubisense API to create several client schemas, which connect to the Ubisense core server and query the server schema for information about active tags in the selected area.

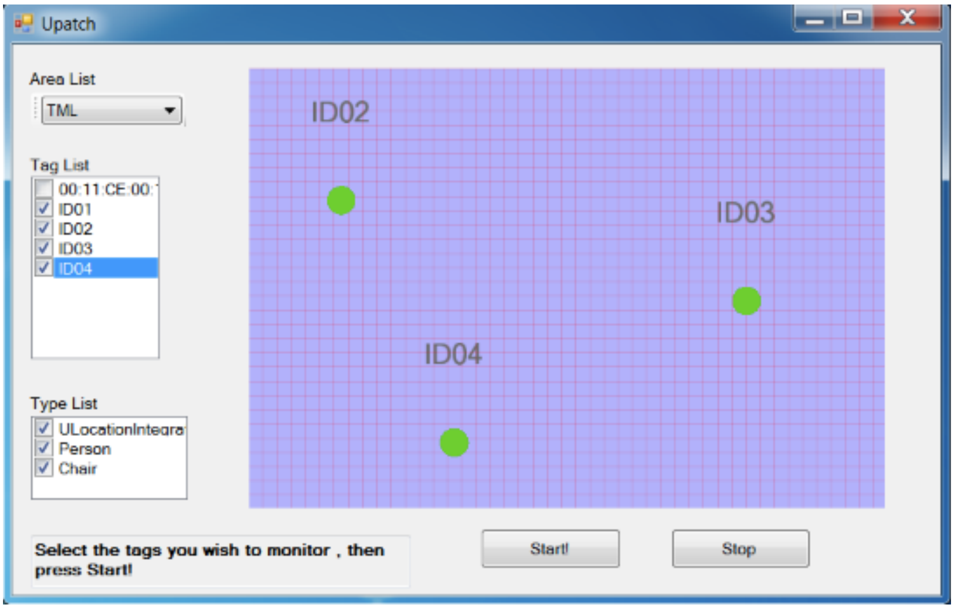
The UWB Monitoring module supports communication among computers that are natively not working together. In fact, the Ubisense system and the Multimedia module are implemented in different programming languages. The transmission between the UWB Monitoring module and the Multimedia module is performed with Open Sound Control (OSC) protocol originally developed by the UC Berkeley Center for New Music and Audio Technology (Open Sound Control, 2010). The OSC protocol forms the basis for communication among the various systems. The OSC protocol has several advantages, such as naming of individual resources, ease of use, availability in various development environments, and segregating from network transport software modules, which interact with each other using OSC message format. The OSC message format is a hierarchical Uniform Resource Locator (URL) style name beginning with a forward-slash ("/") character. The messages in URMS are formatted as follows:

/ control/ object-name xPosition yPosition zPosition

The first string, *control*, is a symbolic address that represents the object name, *k*, and coordinates (*x*, *y*, *z*), respectively.

The functionalities of the UWB Monitoring module include the following:

- Switching between areas
- Ability to select the tags for which to capture information
- Ability to filter the tags by type
- Live 2D and 3D visualization of the location engine
- Capturing tag location data
- Sending tag data over OSC to the Multimedia module



Area List: TML

Tag List:
☐ 00:11:CE:00:
☑ ID01
☑ ID02
☑ ID03
☑ ID04

Type List:
☑ ULocationIntegra
☑ Person
☑ Chair

Select the tags you wish to monitor , then press Start!

Figure 4-6: GUI of UWB Monitoring module

Ubisense API and several third-party libraries are used to facilitate the development of the UWB Monitoring module with the integrated OSC format. Readers can use the code samples described in Appendix D to understand how to create schemas, how to connect schemas to the server and query for information, and how to format the OSC message.

4.3.2 Multimedia Module

The Multimedia module is responsible for computing values of tag data governing the event selection and continuous interaction control, as discussed in Subsections 3.4.2 and 3.4.3, respectively. This module is organized into three sub-modules (patches): *Receiving Tag Location – Zone Detection – Event Control*. The mechanism of each stage is explained in the following.

Receiving Tag Location Patch

Figure 4-7 shows the Receiving Tag Location patch. Specifying a port, this patch receives the message of tag data in a stream transmitted from the UWB Monitoring module. The "udpreceive" object receives any data transmitted over a network using the User Datagram Protocol (UDP). In this patch, the OSC library is used over a UDP Port to communicate between the two modules. The "osc-route" object provided by the OSC library supports OSC address pattern matching functionality. Passed through the osc-route object, the OSC message of "/ ID xPotition yPotition zPotition " is sent to the next object.

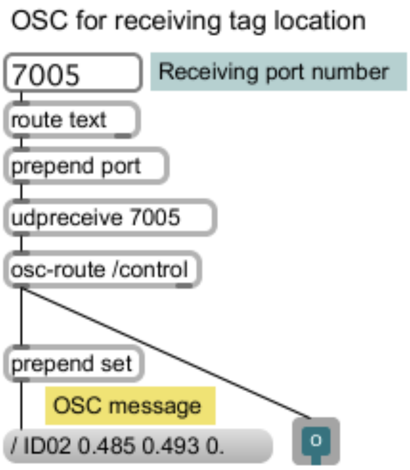


Figure 4-7: Receiving Tag Location patch

Zone Detection Patch

Figure 4-8 shows the Zone Detection patch as the core of the Multimedia module. As explained in Subsection 4.2.2, the OSC message is processed by the JavaScript-based program, *sensing.js*, to have a set of location parameters: zone name, (i, j), the number of tags for the current zone, t, the sum of tags for the entire space, s, and object name, k. As shown at the bottom of the patch, the message is converted to a form of four discrete numbers in the order of i, j, n, s, and k. In addition, the sensing.js program identifies a specific tag existing in a specific event region. If a tag enters a region, the "js" object sends 1 as an enter symbol, and if the tag leaves, it sends 0 as an exit symbol. Finally, this patch transmits the raw data from the UWB Monitoring module, the processed data and enter/exit symbol to the next objects.

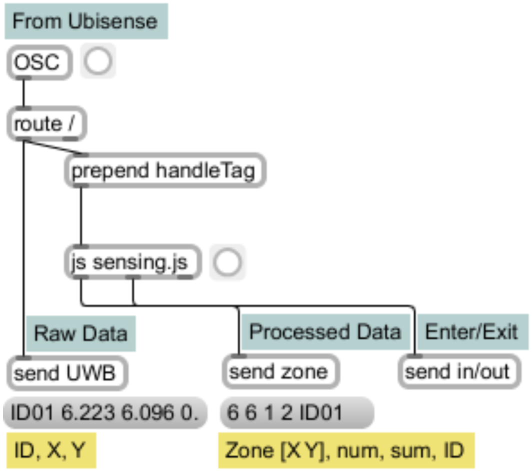


Figure 4-8: Zone detection patch

Lighting Control Patch

As an example of the multimedia event control (lights, images, videos, audio clips, etc.), Figure 4-9 shows the *Lighting Control* patch that executes the lighting event process, offering users an intuitive adjustment of the light intensity. This patch initially activates the lighting event process when it receives a notice that a specific tag enters the lighting region. Receiving the raw data from the Zone Detection patch, the patch then processes the location data to control the light intensity based on at least two tags in the light event region.

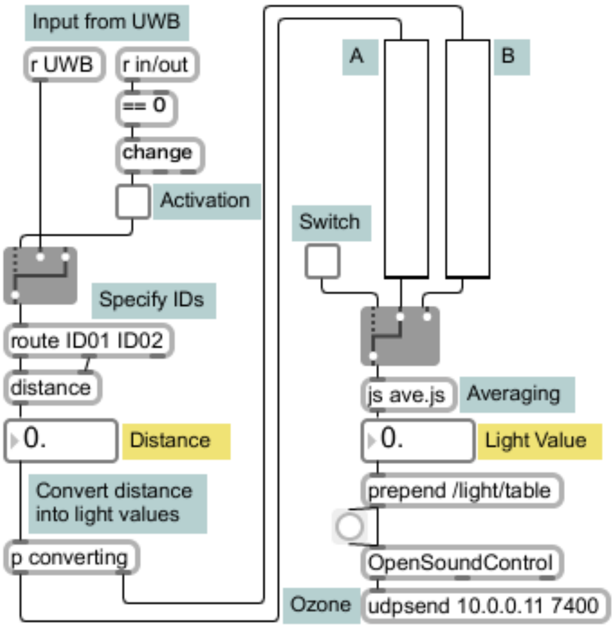


Figure 4-9: Lighting Control patch

In this patch, there are three main functions: (1) calculating the distance between the tags, (2) converting the location data into the value of the light intensity, and (3) averaging the values of the light intensities to smooth data transformation. In the "converting" object, a linear interpolation process is applied to normalize the value in the range of the light intensity. Figure 4-10 shows the patch inside the converting object, which allows user to visually change the interpolation function for the speed of the transformation of light. This patch performs mapping operations using two Max objects categorized as *Data* objects. In the figure, there are the two objects, *function* and *itable*, which work in a similar way but support different controls. X is the distance value, and Y is a value to be converted, correlated to the interpolation function. Depending on the interpolation function, the speed of transformation of the light intensity varies with distance between the tags.

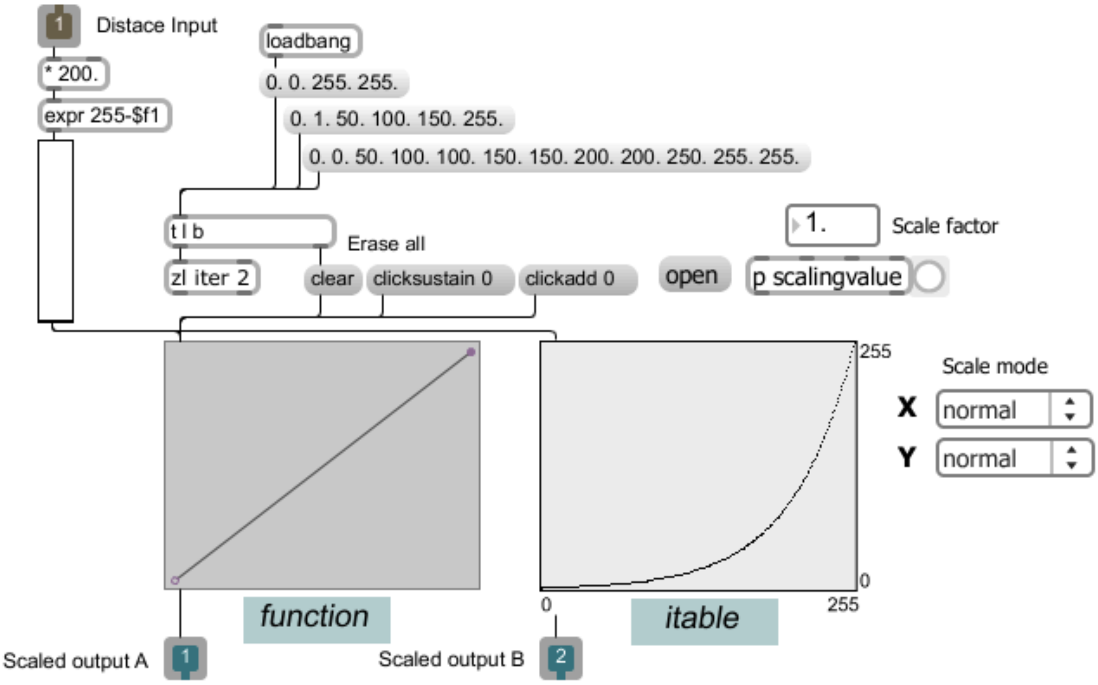


Figure 4-10: Converting patch

Finally, the "js ave.js" object averages the converted values for the light intensity to make the transformation of light smooth. As discussed in Subsection 3.3.2, the UWB technology is prone to occasional accuracy fluctuations caused by radio wave interference. To reduce the impact of these fluctuations on the smooth transformation, the values are smoothed with a moving average filter of length of 100 data. Then, the value is put together in the form /light/table/value to be sent directly to the lighting system using the "OpenSoundControl" object that interfaces directly with the *udpsend* and *udepreceive* objects.

4.3.3 Touch User Interface

We used an HP 42-inch touch display (HP, 2010) to develop the touch user interface. The specifications of the display are available in Appendix E. Figure 4-11 shows the prototype of URMS interface for the touch display developed using Max. Max supports receiving the identifiers for touch points and their x and y coordinates through a Universal Serial Bus (USB) port to define the zones of events as follow: (1)

reading the stream of the parameters from USB connected to the display, (2) extracting touch events with x and y values, (3) scaling these values to fit the definition of the grid graphical interface, (4) detecting the zones.

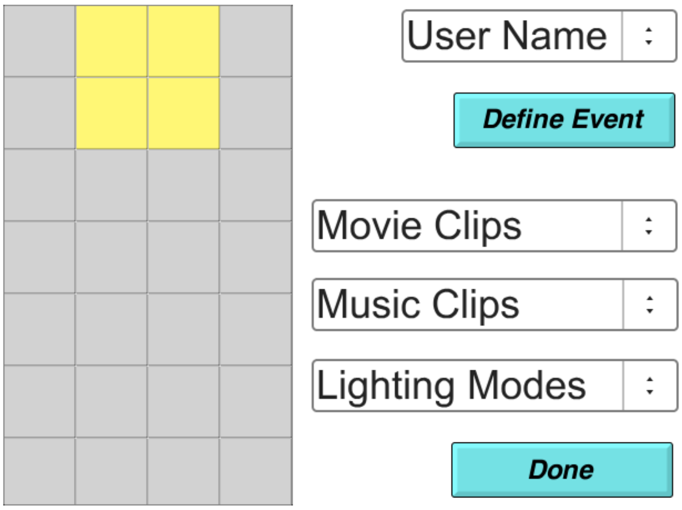


Figure 4-11: A prototype of URMS touch user interface

As discussed in Subsection 3.4.3, the interface supports the definition of event regions on the screen using a controller aligned according to the same grid partition of lines and columns of the real space. Following the GUI design in Subsection 3.4.3, we have implemented the *event region controller*, and three list menus beside the controller to select one of event outputs for movie clips, music clips and lighting modes.

4.4 Case Study 1: Experimental Design at Topological Media Lab

The first case study was motivated by the TML's project scope for multimedia environment, and performed at the TML laboratory at the EV Building of Concordia University. TML is a studio-laboratory for experimental studies of gesture and materiality from computational and phenomenological perspectives. TML has developed their own interactive media system called Ozone, as introduce in Subsection 2.7.1. The interactive media system presents studies in which sensing technology and multimedia apparatuses are used to capture individual behavior and translate behavioral features of TML members to audiovisual interaction. The focus is on dynamic physical activity and speech activity, as measured by camera tracking, noise sound and gyro sensors.

4.4.1 Scenario

For URMS scenario, we started by monitoring people in TML's laboratory under ordinary circumstances. There are two types of participants in the laboratory: TML members, as students, researchers or artists, working on their own projects, and visitors who have different backgrounds. Most of them are frequently moving around the space for information exchanging, work scheduling with colleagues, sharing resources, and other social reasons. As discussed in Subsection 2.3, the meaningful physical space, denoted as *place*, is an important node for human activities, socialization and informal knowledge sharing. Especially among TML members and visitors, a crucial connection creating a touch point with someone is around an oval table (Figure 4-12). This space is a focal place for people to meet others spontaneously and talk about everything from movies to issues of their works.

Figure 4-12: TML members frequently have a conversation around the oval table

After this observation, we noticed that the place at the oval table would be more interesting if it attracts people into an interactive experience. In fact, TML's Ozone system has similarities of the concept of URMS, and thus URMS and the Ozone system are ideal combination for locative media. Then, we decided to integrate the Ozone's lighting system with URMS. As discussed in Subsection 2.6.1, light is a simple but rich medium that supports functional and aesthetic concepts such as illuminating a room, drawing attention and affecting impressions of a situation. As a medium, light can help in our case study to demonstrate a potential of URMS.

4.4.2 Location-based Events Using Tagged Objects

Ubisense supports two types of UWB tags: compact tags and slim tags as shown in Figure 4-13 (a) and (b), respectively. The tags are equipped with battery, buttons, and LED (slim tag only). In this research, compact tags are attached to cups and chairs. These objects were chosen because TML members are accustomed to see around the oval table, as shown in Figure 4-14. For the tagged cup, in order to embed the tag, we took the tag out of the plastic cover and placed the chip underneath the cup so that the appearance of the cup does not change. A test was done to verify that the hidden tag was still detected by the UWB system as explained in Subsection 3.3.2.

(a) Compact Tag **(b) Slim Tag**

Figure 4-13: Tag forms (adapted from Ubisense, 2009)

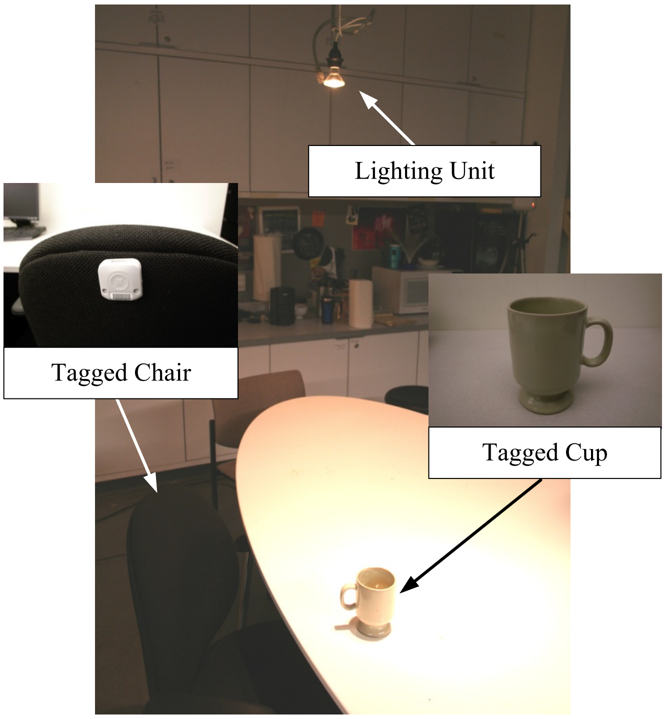


Figure 4-14: Tagged objects for TML's studio

4.4.3 Deployment

On January 20th, 2011, the deployment was carried out in the studio of TML at Concordia University. The equipment consisted of three sensors, five CAT5e Ethernet cables, a Laptop running Windows XP, a PoE switch, four UWB tags, and three tripods. Figure 4-15 shows the setting of the sensor cell with the timing and data cables, and the locations and yaw orientations of the three sensors. The locations were measured using a laser meter. The orientation of the sensors was done considering the layout of the TML studio and the aspects of the UWB system requirements discussed in Subsection 3.3.2. The sensor cell network was installed and calibrated properly following the procedures defined in Appendix F. Figure 4-16 (a) shows the angles of the sensors to be considered during calibration in terms of visibility requirement, and Figure 4-16 (b) shows a part of the site with a sensor fixed on a tripod.

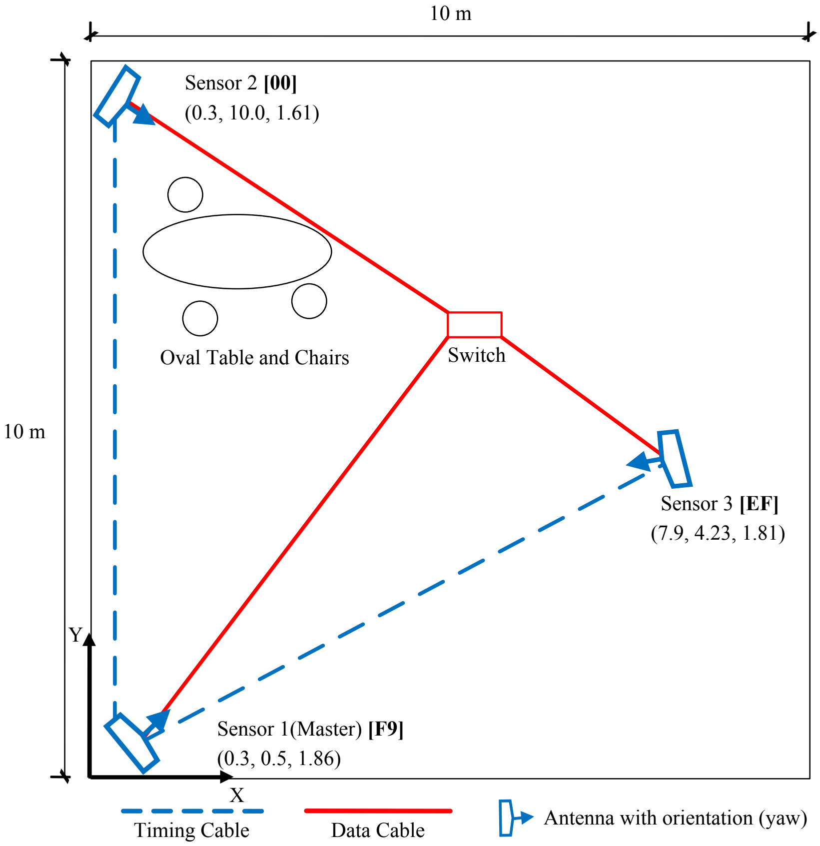
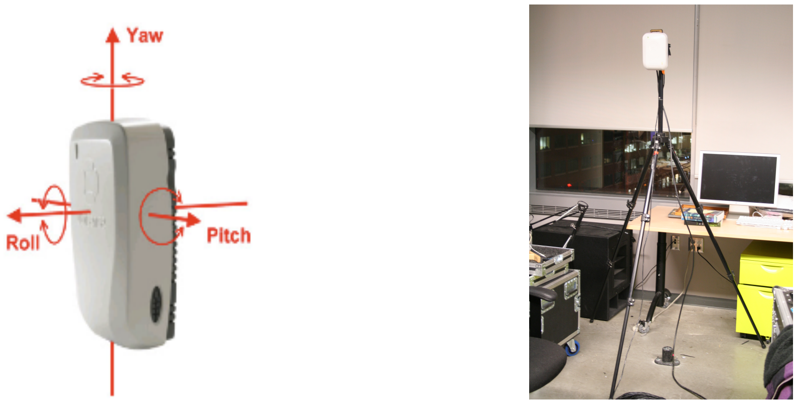


Figure 4-15: The layout of system deployment at TML



(a) Yaw, roll and pitch of a sensor (Ubisense, 2009) (b) A sensor fixed on a tripod

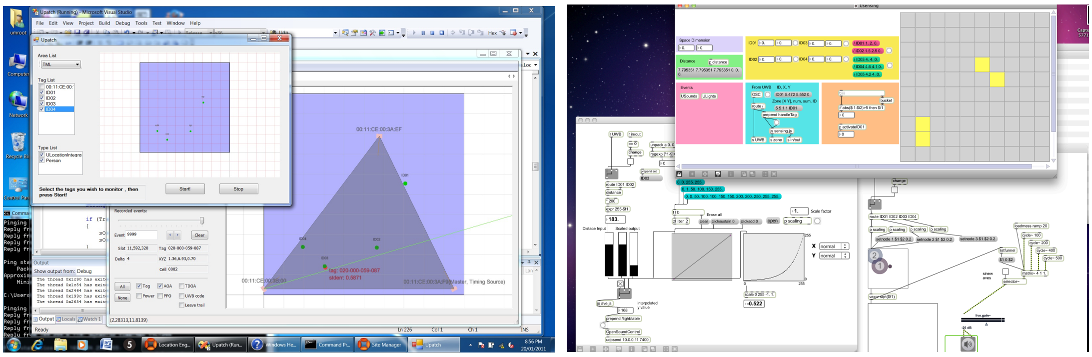
Figure 4-16: Orientation of a sensor (Ubisense, 2009)

On January 21th, 2011, we demonstrated an experiment of URMS. The pre-test activation of sensors and cables was already completed before running the URMS system. The calibration of sensors was done using a reference tag approximately at the center of the space. A work sheet for organizing the data of the system calibration is shown in Appendix G, where the data of the system setting are recorded.

For an optimal URMS setting, we defined system requirements based on Section 3.3. The acceptable latency of lighting interaction, as a visual interaction, is 50 ms based on Table 3-5. In this condition, the UWB system needs to keep the frequency between 15-25 Hz. We decided to set the frequency of the UWB system to 38 Hz because we used only four tags in the cell. In addition, an information data filter provided by Ubisense was used to improve accuracy with a motion model of position and Gaussian noise on position (Ubisense, 2009). The same scheduling and filtering properties were configured for all the tags.

4.4.4 URMS Software for TML

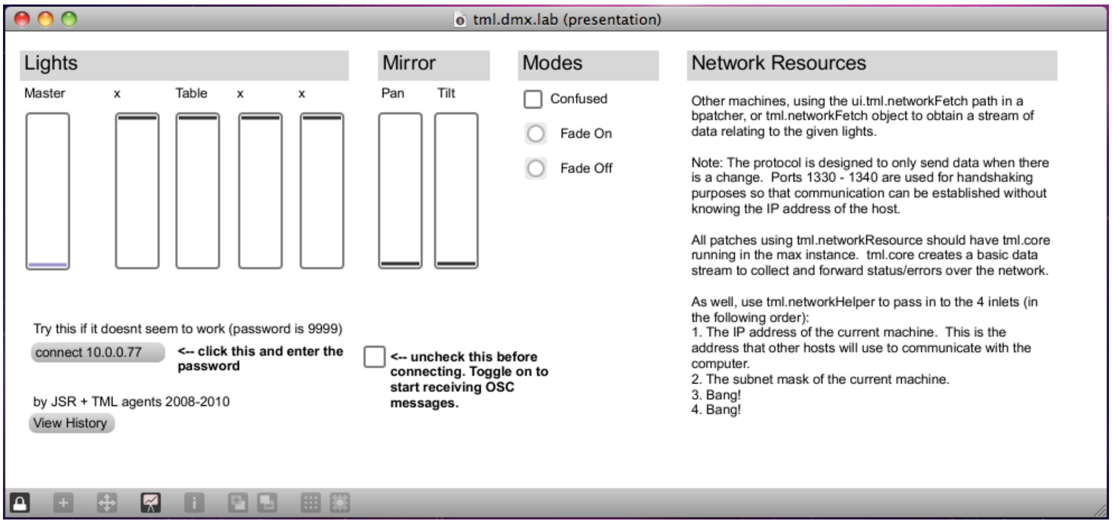
Figure 4-17 show URMS software modules that consist of the UWB Monitoring module (Subsection 4.3.1) and the Multimedia module (Subsection 4.3.2), configured for use with the TML's Ozone system depicted in Figure 2-9. In order to build a connection between the Ubisense system and the Ozone system, we use Windows PC (Windows 7) and Mac PC (OSX). The Windows PC runs the Ubisense software and the UWB Monitoring module, while the Mac runs the Multimedia module including Max-based Ozone's Lighting system parameterized Digital Multiplex Signal (DMX) (OpenDMX.net, 2010).



(a) UWB Monitoring module on Windows PC **(b) Multimedia module Mac PC**

Figure 4-17: URMS software modules

Figure 4-18 shows the lighting control module of the Ozone system, called "tml.dmx.lab" patch, which allows URMS to control the DMX system, in which the range of the light intensity is between 0 and 255. This patch acts as a coordination engine that orchestrates communication between the Multimedia module and the lighting unit.



Other machines, using the ui.tml.networkFetch path in a bpatcher, or tml.networkFetch object to obtain a stream of data relating to the given lights.

Note: The protocol is designed to only send data when there is a change. Ports 1330 - 1340 are used for handshaking purposes so that communication can be established without knowing the IP address of the host.

All patches using tml.networkResource should have tml.core running in the max instance. tml.core creates a basic data stream to collect and forward status/errors over the network.

As well, use tml.networkHelper to pass in to the 4 inlets (in the following order):
1. The IP address of the current machine. This is the address that other hosts will use to communicate with the computer.
2. The subnet mask of the current machine.
3. Bang!
4. Bang!

Try this if it doesnt seem to work (password is 9999)

<-- click this and enter the password

☐ **<-- uncheck this before connecting. Toggle on to start receiving OSC messages.**

by JSR + TML agents 2008-2010

Figure 4-18: Lighting control module by TML

4.4.5 Demonstration

Figure 4-19 show the demonstration of URMS at TML. We have successfully tested our system on region-based event selection in near real-time and proximity-based continuous adjustment of the light intensity. In this demonstration we tested two particular actions under the light based on the following scenarios: (a) two persons moving closer to each other, represented by the distance $d_{1,2}$ in Figure 4-20, and (b) one is heading for the table pulling a chair, represented $d_{3,q}$ in Figure 4-20. For scenario (a), a user carried two tags to provide proximity event. For scenario (b), the light was turned on, and the light intensity was smoothly increased and decreased with respect to the distance between the oval table and the tagged chair. Here our experiments found that the user was able to move around the space with a tagged cup or a tagged chair to explore the location-based lighting event associated with lighting event region.

(a) A test of proximity-based lighting event **(b) A test of location-based lighting event**

Figure 4-19: Demonstration at the TML studio

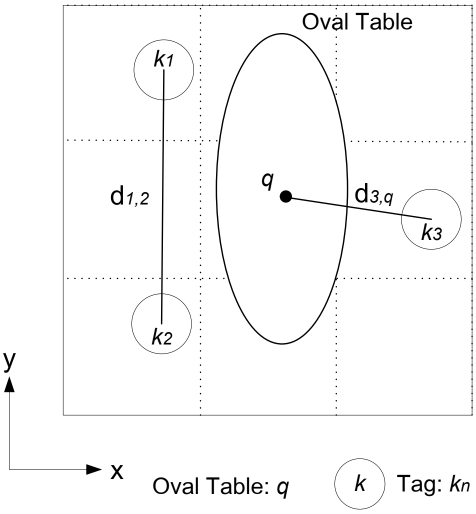


Figure 4-20: Location interaction around oval table based on distance

4.5 Case Study 2: Multimedia Space with Touch Display

In the second case study, we implemented the URMS prototype system that prioritizes the events as well as a prototype of touch user interface to explore how this interface can be integrated with URMS. The advantage of the touch user interface is a more engaging interaction that allows the users to define the multimedia events by themselves according to their preferences.

4.5.1 Near Real-Time Event Prioritization

As explained in Subsection 3.4.2, the first event is triggered based on the arrival time, and then the system determines the next event based on the majority of triggered events during a short period (0.1 sec.). In this case study, we used four tags, one display and two different movies, Movie 1 and Movie 2, as event outputs. For the event list, we used the same region definitions for combination of tags and events defined in Table 3-7. When one or more tags are detected in the region, the prototype system counts the number of occurrence of each event associated with tags' names within 0.1 sec. By sorting the occurrence of two movies based on the triggered events, the prototype system finds which movie should be played (Movie 1 or Movie 2).

4.5.2 Deployment

For this case study, we deployed the UWB system with the touch display in the Infrastructure Management Systems laboratory at Concordia University. The floor plan and the Movie Region for the touch display are shown in Figure 4-21. As shown in Figure 4-22, the grid and the region follow the same setting we explained in Subsection 3.4.2. The dimension and partitions of the grid are associated with the laboratory, and the controller of the GUI shown in Figure 4-11 features a 4×7 grid of gray zones, and the color of a zone turns to yellow when the user touches any point within that zone. The host computer (Windows 7) is connected with the HP 42-inch touch display.

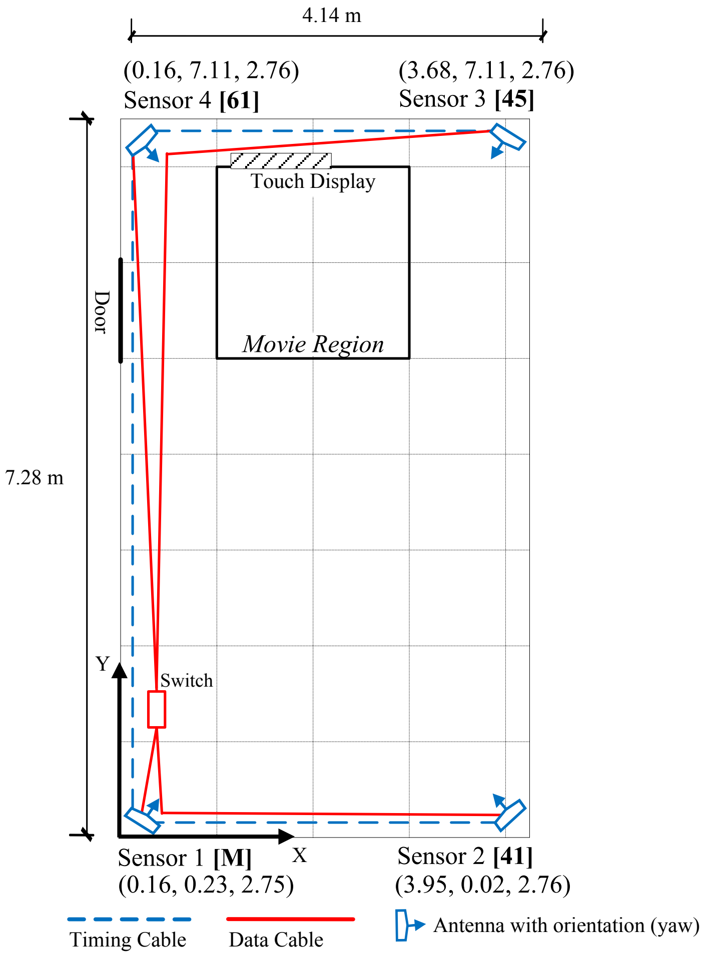


Figure 4-21: Ubisense setup with touch display

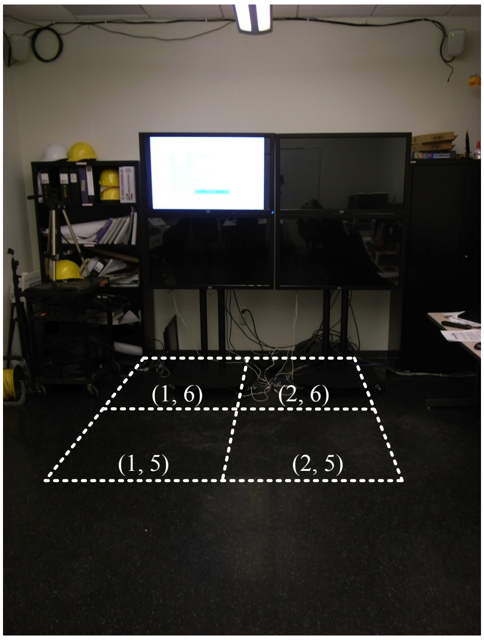
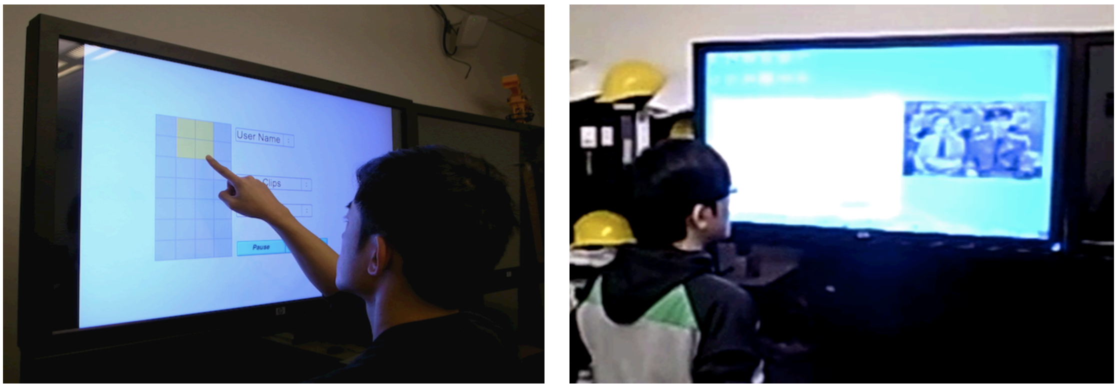


Figure 4-22: Touch display and the movie region in the laboratory

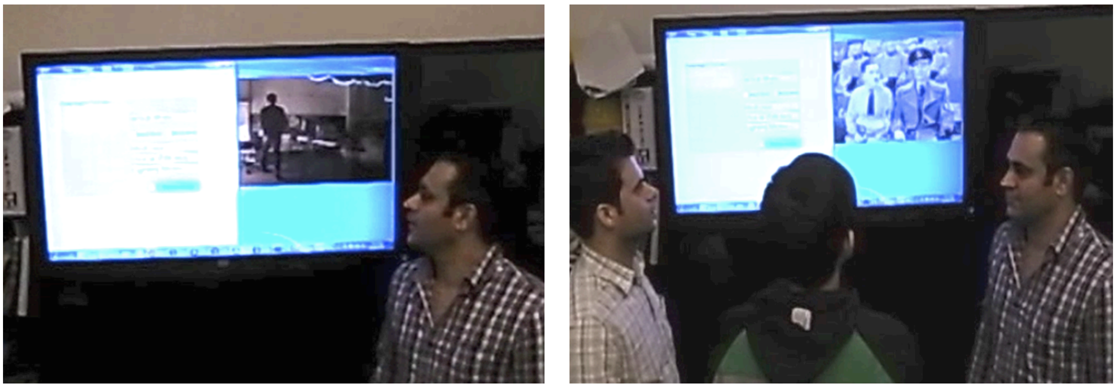
4.5.3 Result of the Demonstration

Following the same setting of Figure 3-13, four users were able to define their own movie events and the corresponding regions using the touch screen. The details of the movie events are given in Table 3-7. Figure 4-23(a) shows a scene of one of the users defining events using the touch user interface. Figure 4-23(b) and Figure 4-23(c) show the movies displayed to different users based on their defined events. Figure 4-23(d) shows that Movie-1 is displayed based on the majority of the triggered events. During the demonstration of the touch user interface, the users were able to explore the event options for movie and music clips and to design events interactively. The test showed that the grid-based URMS could establish a clear correlation between the tags' locations and the events. The test also demonstrated the applicability of the event prioritization method proposed in Subsection 3.4.2.



(a) User-1 defining events using the touch interface

(b) Movie-1 is displayed to User-1



(c) Movie-2 is displayed to User-2

(d) Movie-1 is displayed based on the majority of triggered events

Figure 4-23: Results of the second case study

4.6 Summary

The present chapter describes the development of a prototype system and shows the applicability of the proposed approach. The selection of the development tools was based on an investigation of the availability, the integration possibility, and the functionalities of several software systems. The Ubisense system was selected as the UWB system for the implementation to take advantage of its reliable and accurate tracking capability and its rich API. The software tools were linked with a multimedia device using the OSC protocol, Max programs and Ubisense API. Two modules were designed for the prototype system including UWB data capturing, communication, and data processing for continuous and smooth adjustment of the location-based events. The case studies demonstrated the feasibility of the proposed approach for responsive space.

CHAPTER 5 CONCLUSIONS AND FUTURE WORK

5.1 Summary of Research

The present research has explored a responsive multimedia space using a UWB RTLS. This research has proposed an innovative approach to integrate the UWB system with multimedia devices focusing on real-time location-based events. Our approach focuses on: (1) A grid-based event selection method in near real-time, (2) the prioritization of the events triggered by multiple tags, and (3) the continuous and smooth event interaction control based on distance. The resulting integrated system can engage users to location interaction with multimedia devices by associating the location of tags with a predefined list of multimedia events.

5.2 Conclusions

The conclusions of this research are:

(1) The concept of UWB-based Responsive Multimedia Space (URMS) has been proposed where UWB RTLS is used for tracking users and objects to trigger location-based multimedia events. URMS encourages people to immerse themselves in multimedia interaction at their living or work place with minimum constraints.

(2) The high-level requirements for URMS have been defined: (1) The system should support automatic location detection of users in a seamless way; (2) The system should respond to the spatial relationships as they arise, resulting from bodily actions; and (3) The location-based events should be easy to define and understand.

(3) The requirements for the deployment of the system and the location-based events

have been investigated. They have been categorized into two groups: (1) requirements for achieving an accurate, scalable and near real-time system, and maintaining visibility of UWB tags, and (2) requirements for achieving an acceptable latency of performance of the responsive space. Furthermore, the system design for satisfying the requirements has been proposed.

(4) The grid-based event selection methods based on tags' locations have been proposed. To support a variety of situations of URMS, we proposed an event prioritization method based on the number of occurrence of each event. Furthermore, a touch user interface was proposed that allows the user to define the events easily and directly.

(5) A method for continuously and smoothly adjusting the multimedia events based on distance has been proposed to sustain location interaction with tag locations and movements. The events can give the user an interactive experience such as switching the light on and off, or gradually controlling the light intensity with respect to his/her position in the space or with respect to other objects.

(6) A prototype system has been developed to verify the applicability of the proposed approach. Two case studies have demonstrated the feasibility of the proposed approach for responsive space and the potential of URMS for building user-centered interaction design in a multimedia environment.

5.3 Future Work

In this section, we outline some potential directions for the future work concerning the applications of URMS in entertainment and social networking. Using digital 2D or 3D models, sounds, or light, URMS can provide a game-like interactive space where users are encouraged to respond to the feedback from the media devices. For example, an augmented reality game can be designed based on URMS. It allows users in a virtual space to interact with virtual objects through a display. For entertainment purposes, URMS can be integrated with moving spotlights that have pan and tilt controls. The spotlights illuminate a person, object, or group of them following their locations. Furthermore, selected multimedia events (e.g. sounds, movies or colored lights) can be superimposed or blended to create an artistic flavor. As an extension to the priority setting of multimedia environments, future work will consider the possibility of customized commercial provisioning in public spaces. For example, URMS can use the locations and background information of several persons to display the most relevant commercial on the nearest digital signage display. With respect to the definition of events using touch screen, future work is needed to explore the possibilities of using smart phones for the collection and analysis of users' preferences that could be used to refine the definition of events.

We believe that URMS becomes more useful and interesting if all multimedia devices existing in the space are in synchronization with each other for adjusting the states of outputs ongoing according to social interaction. Moreover, it should be noted that URMS needs to adapt its events to the surrounding environment. For example, the glowing light does not always match an exhibition space where the atmosphere is often kept quiet and calm. Thus, the form of expression of multimedia events for URMS should be considered based on the environment.

REFERENCE

Abdul-Latif, O., Shepherd, P., and Pennock, S. (2007).
TDOA/AOA Data Fusion for Enhancing Positioning in an Ultra Wideband System. In IEEE International Conference on Signal Processing and Communications (ICSPC), Dubai, United Arab Emirates, pp.1531-1534.

Aylward, R., and Paradiso, J. A. (2006).
Sensemble: A Wireless, Compact, Multi-User Sensor System for Interactive Dance. In Proceedings of the 2006 International Conference on New Interfaces for Musical Expression (NIME06), Paris, France, pp. 134-139.

Bacon, J., Bates, J., and Halls, D. (1997).
Location-Oriented Multimedia. IEEE Personal Communications, October, pp. 48- 57.

Bleecker, J. and Knowlton, J. (2006).
Locative Media: A Brief Bibliography And Taxonomy Of Gps-Enabled Locative Media. In Leonardo Electornic Almanac, Vol. 14. No. 3.

Bongers, B., and Mery, A. (2007).
Interactivated Reading Table. University of Technology Sydney, Cairns, QLD, Australia. Retrieved Apr 06, 2010 from <http://www.educ.dab.uts.edu.au/interactivation/ReadingTable.pdf>

Bouguila, N, and Ziou, D. (2005).
Using unsupervised learning of a finite Dirichlet mixture model to improve pattern recognition applications, Pattern Recognition Letters, vol. 26, pp. 1916-1925.

Breed, G. (2005).
A summary of FCC rules for Ultra wideband Communications. Retrieved Nov 18, 2010 from <http://www.highfrequencyelectronics.com/Archives/Jan05/HFE0105_Tutorial.pdf>

Brooks, D. W. (1997).
Web-teaching: A guide to designing interactive teaching for the World Wide Web. New York: Plenum.

Clarkson, J. (2007).

Designing a More Inclusive Design. Retrieved Nov 23, 2010 from http://www.universaldesign.ie/exploreampdiscover/conferenceproceedings/universallearningforthe21stcentury/designingamoreinclusiveworld

Correal, N. S., Kyperountas, S., Shi, Q., and Welborn, M. (2003).

An UWB Relative Location System. In: IEEE Conference on Ultra Wideband Systems and Technologies. Reston, Virginia, U.S, pp.394-397.

Cresswell, T. (2004).

Place: a short introduction. Malden/Oxford/Victoria: Blackwell Publishing.

Cycling74. (2010).

Retrieved Apr 12, 2011 from

<http://cycling74.com/>

Di Benedetto, M., Kaiser, T., Porcino, D., Molisch, A., and Opperman, I. (2006).

UWB Communication Systems A Comprehensive Overview. Hindawi Publising.

Dourish, P. (1999).

Embodied Interaction: Exploring the Foundations of a New Approach to HCI, Xerox Palo Alto Research Center.

Dourish, P. (2001).

Where the Action Is: The Foundations of Embodied Interaction. MIT Press.

eMotion. (2010).

Retrieved Apr 06, 2010 from

<http://www.mapping-museum-experience.com/en>

Ghavami, M., Michael, L. B., and Kohno, R. (2004).

Ultra Wideband Signals and Systems in Communication Engineering. John Wiley & Sons, Chichester.

Hemment, D. (2006).

Locative Arts," Leonardo, Vol. 39, No. 4, pp. 348-355.

HP LD4200tm (2010).

Retrieved Apr 06, 2010 from

<http://h10010.www1.hp.com/wwpc/us/en/sm/WF05a/382087-382087-

64283-72270-3915216-4032279.html>

International Organization for Standardization, and International Electrotechnical Commission (2006).
ISO/IEC 24730-1:2006 Information Technology - Real-Time Locating Systems (RTLS) - Part 1: Application Program Interface (API). International Organization for Standardization, Genève, Switzerland.

Ishii, H., and Ullmer, B. (1997).
Tangible Bits: Towards Seamless Interfaces between People, Bits, and Atoms. In *Proceedings of CHI 1997*, Atlanta, GA, ACM, pp. 234-241.

Johanson, B., Fox A., and Winograd, T. (2002).
The Interactive Workspaces Project: Experiences with Ubiquitous Computing Rooms. IEEE Pervasive Computing Magazine, 1(2).

Kaushik, A. (2009).
Testing and Analysis of Ubisense Ultrawide Band System. Internal Report. Concordia University. Montreal, Canada.

Kockelmans, J. J. (1976).
Merleau-Ponty on Space Perception and Space. In Phenomenology and the Natural Sciences: Essays and Translations, edited by Joseph J. Kockelmans and Theodore J. Kisiel. Evanston, IL: Northwestern University Press.

Lemos, A. (2008).
Locative Media and Informational Territories. In L. Santaella & P. Arantes (Eds.), *Estéticas Tecnológicas. Novos Modos de Sentir* (pp. 207-230). São Paulo: EDUC. Retrieved Nov 10, 2010 from <http://medialab-prado.es/mmedia/1836>

Liu, H., Darabi, H., Banerjee, P., and Liu, J. (2007).
Survey of Wireless Indoor Positioning Techniques and Systems. In IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews. 37 (6), pp. 1067-1080.

Lonthoff, J., and Ortner, E. (2007).

Mobile Location-Based Gaming as Driver for Location-Based Services (LBS) – Exemplified by Mobile Hunters. In Informatics 31, 2007, pp. 183-190.

MakingThings. (2010).
Retrieved Dec 10, 2011 from
<http://www.makingthings.com/teleo/products/documentation/teleo_user_guide/max_msp.html>

Malik, A. (2009).
RTLS for Dummies. Wiley, Hoboken, N.J.

Mania, K., and Robinson, A. (2004).
The Effect of Quality of Rendering on User Lighting Impressions and Presence in Virtual Environments, In ACM Siggraph Int'l Conf. Virtual Reality Continuum and its Applications in Industry, ACM Press, pp. 200-205.

Merleau-Ponty, M. (1962).
Phenomenology of Perception. Translated by Colin Smith. Atlantic Highlands, New Jersey: The Humanities Press, "Space."

Motamedi, A. (2010).
Accuracy Analysis of UWB Tracking in a Laboratory Static Test. Internal Report. Concordia University. Montreal, Canada.

Muñoz, D., Bouchereau, F., Vargas, C., and Enriquez-Caldera, R. (2009).
Position Location Techniques and Applications. Academic Press, Burligton, MA.

Muthukrishnan, K., and Hazas, M. (2009).
Position Estimation from UWB Pseudorange and Angle-of-Arrival: A Comparison of Non-Linear Regression and Kalman Filtering. In Proceedings of the 4th International Symposium on Location and Context Awareness. Tokyo, Japan. pp. 222-239.

Nagashima, Y. (2003).
Bio-Sensing Systems and Bio-Feedback Systems for Interactive Media Arts, IN 2003 Conference on New Interfaces for Musical Expression (NIME-03), Montreal, Canada. Retrieved Sep 02, 2010 from

<http://nagasm.suac.net/ASL/NIME03/index.html>

OpenDMX.net. (2010).
Retrieved Apr 06, 2010 from
<http://www.opendmx.net/index.php/OpenDMX.net>

Open Sound Control. (2010).
Retrieved Apr 06, 2010 from
<http://opensoundcontrol.org/>

Randy, J. (2009).
Max MSP/Jitter Start. Retrieved Apr 06, 2010 from
<http://2uptech.com/papers/MSP_Jitter_overview.pdf>

Russon, J. (2003).
Human Experience: Philosophy, Neurosis, and the Elements of Everyday Life. Albany: State University of New York Press. Introduction and Part 1, "The form of human experience", chapters on interpretation, embodiment, memory, pp. 1-47.

Schiller, J. H., and Voisard, A. (2004).
Location-Based Services, Morgan Kaufmann Publishers, San Francisco, CA.

Settel, Z., and Wozniewski, M. (2009).
Audio Graffiti: A Location Based Audio-Tagging and Remixing Environment, In Proceedings of ICMC2009, Montreal, Quebec, Canada. Retrieved Sep 02, 2010 from
<http://cim.mcgill.ca/~nicolas/ICMC2009-demo.pdf>

Sha, X. W. (2005).
The TGarden Performance Research Project. Retrieved Sep 28, 2010 from
<http://muse.jhu.edu/journals/modern_drama/v048/48.3wei.html>

Sha, X. W., Fortin, M., Navab, N., and Sutton, T. (2010).
Ozone: Continuous State-based Media Choreography System for Live Performance, In Proceedings of the international conference on Multimedia ACM Multimedia, October 2010, Firenze, Italy.

Sha, X. W., Fortin, M., and Rousseau, J. S. (2009).

Calligraphic Video: A Phenomenological Approach to Dense Visual Interaction, In Proceedings of the seventeen ACM international conference on Multimedia, October 2009, Beijing, China.

Sha, X. W., Iachello, G., Dow, S., Serita, Y., St. Julien, T., and Fistre, J. (2003). *Continuous Sensing of Gesture for Control of Audio-Visual Media*, In Proceedings of the Seventh IEEE International Symposium on Wearable Computers.

Sidor, R. T. (2008).
Interaction Design and Information Visualization for Wall-Size Displays with User Tracking. Retrieved Sep 02, 2010 from
<http://www.thomassidor.com/files/MasterThesisLQ.pdf>

Sintoris, C., Raptis, D., Stoica, A., and Avouris N. (2007).
Delivering Multimedia Content in Enabled Cultural Spaces. In Proceedings of the 3rd international conference on Mobile multimedia communications, Rio Patras, Greece.

Sokolowski, R. (1999).
Introduction to Phenomenology, Cambridge University Press.

Tanaka, A., and Gemeinboeck, P. (2006).
A framework for spatial interaction in locative media. In Proceedings of New Interfaces for Musical Expression (NIME). France, Paris: IRCAM, pp. 26-30.

Thompson, E. (2007).
Mind in life: Biology, phenomenology, and the sciences of mind. Cambridge, MA: Harvard University Press.

Ubisense. (2009).
Location Engine Configuration Manual. Ubisense, Cambridge, U.K.

Ubisense. (2010).
Retrieved Apr 06, 2010 from
<http://www.ubisense.net>

Ward, A. M. R. (2007).

In-Building Location Systems. In the Institution of Engineering and Technology Seminar on Location Technologies. London, U.K. pp. 1-18.

Ward, A. M. R. (2009).
Calibration of a Location System. *Patent US 2009/0079633 A1.* Date of patent: Mar. 26, 2009.

Ward, A. M. R., and Webster, P. M. (2009).
Location Device and System using UWB and Non-UWB Signals. Patent US 7,636,062 B2. Date of patent: Dec. 22, 2009.

Weiser, M. (1991).
The Computer for the 21st Century. In Scientific American, 265(3), pp. 94-104.

Wikipedia. (2010).
Fade out. Retrieved Nov 16, 2010 from
< http://en.wikipedia.org/wiki/Fade_out >

Zahavi, D. (2002).
Husserl's Phenomenology, Stanford University Press.

Zhang, C., Hammad, A., and AlBahnassi, H. (2009).
Path Re-Planning of Cranes using Real-Time Location System. In: The 26th International Symposium on Automation and Robotics in Construction (ISARC). Austin, Texas, U.S. pp. 412-419

Zhang, C., Hammad, A., and AlBahnassi, H. (2010).
Improving construction safety through real-time motion planning of cranes. In Computing in Civil and Building Engineering, Proceedings of the International Conference, W. TIZANI (Editor), 30 June-2 July, Nottingham, UK, Nottingham University Press, Paper 53.

APPENDIX A. Timeslot Length for a 160Hz system (Ubisense, 2009)

Slot interval	Update interval (seconds)	Listen interval (seconds)	Reschedule response time
4	0.026	0.026	fast
8	0.052	0.052	fast
16	0.104	0.104	fast
32	0.208	0.208	fast
64	0.416	0.416	fast
128	0.832	0.832	fast
256	1.66	0.832	fast
512	3.32	0.832	fast
1024	6.66	0.832	fast
2048	13.3	0.832	fast
4096	26.6	26.6	slow
8192	53.2	53.2	slow
16384	106	106	slow
32768	106	106	slow

APPENDIX B. Ubisense Series 7000 Sensor Specifications (Ubisense, 2010)

Size and Weight	
Dimensions	20cm x 13cm x 6cm (8” x 5” x 2.5”)
Weight	650g (23 oz)
Operating Conditions	
Temperature	0°C to 60°C (32°F to 140°F)
Humidity	0 to 95%, non-condensing
Enclosure	IP30
Location Performance	
Operating Range	Up to 160m (520ft) in open field conditions
Achievable Accuracy	Better than 30cm (12”) in 3D
Radio Frequencies	
Ultra-wideband	6GHz – 8GHz
Telemetry channel	2.4GHz
Certifications	
FCC Part 15 (FCC ID SEASENSOR20)	
EU CE	
Power Supply	
Power-over-Ethernet IEEE 802.3af compatible 12V DC @ 10W (optional)	
Mounting Options	
Adjusting mounting bracket (supplied)	
Ubisense Part Codes	
UBISENSOR7000, UBISENSPS (optional 12V power supply)	

APPENDIX C. Ubisense Series 7000 Compact Tag Specifications (Ubisense, 2010)

Size and Weight	
Dimensions	38mm x 39mm x 16.5mm (1.50” x 1.53” x 0.65”)
Weight	25g (0.88 oz)
Operating Conditions	
Standard	-20°C to 60°C (-4°F to 140°F)
Extended	-30°C to 70°C (-22°F to 158°F)
Humidity	0 to 95% non-condensing
Enclosure	
Standard	IP63
Extended	IP65
Location Performance	
Update Rate	0.00225Hz to 33.75Hz (can be varied dynamically under software control)
Peripherals	
LED (application controllable)	
Push button (application controllable)	
Motion detector	
Radio Frequencies	
Ultra-wideband	6GHz – 8GHz
Telemetry channel	2.4GHz
Certifications	
FCC Part 15 (FCC ID SEATAG22, SEATAG22HH)	
EU CE	
Power Supply	
3V coin cell (CR2477)	
Mounting Options	
Industrial adhesive pas (supplied)	
Industrial velcro	
Magnetic mountings	
Screw mountings	
Ubisense Part Codes	
Standard	UBITAG70022
Extended	UBITAG7022 X-65

APPENDIX D. Ubisense Application Configuration

Adding References to the Project

It is crucial to remember adding the Ubisense DLLs as reference to the C# project, in order to be able to use the Ubisense API and OSC. We use Bespoke to manage connection between .Net frameworks and OSC.

Schema Declarations

‘Naming’ will be used to get the name of tags being tracked by the sensors; ‘Monitor’ will be used for tracking information about tags in certain monitored areas; ‘Building’ can be used querying information about different building areas and cells; finally ‘Vis’ will be used for displaying a graphical representation of the location engine.

```
using Naming = Ubisense.UName.Naming;

using Vis = Ubisense.UVis.Representation;

using Monitor = Ubisense.USpatial.Monitor;

using Building = Ubisense.UBuilding;
```

We then define some schemas, and connect them in order to be able to query for information. Again we use `naming_schema` to query for object/tag names; `buildSchema` provides controls for the building area; `vis_schema` is the visualization schema; and `cellData` is used to query locations.

```
static Naming.Schema naming_schema = new Naming.Schema(false);

static Ubisense.UBuilding.Building.Contents.Schema buildSchema = new
Ubisense.UBuilding.Contents.Schema(false);

static Ubisense.UVis.Representation.Schema vis_schema = new
Ubisense.UVis.Representation.Schema(false);

static Ubisense.ULocation.CellData.Schema cellData = new
Ubisense.ULocation.CellData.Schema(false);
```

GetTags and GetTypes Method

These two methods are called to complete the loading of the user interface form. There are two list views (checkedBoxList) that we want to populate with the list of tag names, and their corresponding types respectively. This would enable the user to select types and narrow down the list of displayed tag names.

The first method, `getTypes`, loops through the rows in the naming schema, and retrieves the corresponding type. If the type is not already in the checked list box, it is added. Then, another loop goes through the added types to make sure the check boxes are all ‘checked’. The second method, getTags, also loops the naming schema, and retrieves the name associated with each row. Before adding the name to the list, we check that the corresponding type is currently checked in the types checkedListBox.

Message Transmission as OSC Format

We will describe some messages used in tracking works. Ubisense location engine compute the tag's location. The monitor module sends the location message to the multimedia module including object's name, and keep Ubisense Location Engine inform the location of all tags on the display. The following data is the structure of the message.

```
//OSCServer

private static readonly TransportType TransportType =   TransportType.Udp;


private static readonly IPAddress MulticastAddress =
IPAddress.Parse("10.211.55.2");


private static readonly int Port = 7005;


private static readonly IPEndPoint Destination = new
IPEndPoint(MulticastAddress, Port);


private static readonly string AliveMethod = "/control/";


private static OscMessage sMessage;


 private static OscClient sOscClient;
```

Update Handlers

Two update handlers are used in the application: one of them monitors each tag movement, keeping track of the position coordinates, and the other monitors tag button presses. The update handlers are declared in the following way:

```
        Ubisense.ULocation.CellData.Location.AddUpdateHandler(cellData,
positionUpdateHandler);

        Ubisense.UData.Data.ObjectButtonPressed.AddUpdateHandler(dataSchema,
buttonPressedHandler);
```

Where `celldata` and `dataSchema` are the corresponding schemas that are queried and `positionUpdateHandler` and `buttonPressedHandler` are the methods that implement the code handling the updates. These methods will thus be called each time there is an update, i.e., each time a tag movement is detected.

The position update handler needs to retrieve and log the tag positions (x y z coordinates) of a text file on each occurrence. The corresponding code follows below; first, we obtain the position coordinates and format them to our desired degree of precision.

```
   string x = row.position_.P.X.ToString();
```

```
    string y = row.position_.P.Y.ToString();
    string z = row.position_.P.Z.ToString();

    // Get position of decimal point and specify precision
    int indexX = x.IndexOf('.') + 4;
    int indexY = x.IndexOf('.') + 4;
    int indexZ = x.IndexOf('.') + 4;

    // truncate coordinates
    string truncX = x.Substring(0, indexX);
    string truncY = y.Substring(0, indexY);
    string truncZ = z.Substring(0, indexZ);

    // Get timestamp
    string time = row.time_.ToString();
```

Before writing to the output file, we verify that the tag in question has been selected by the user in the corresponding list box:

```
        if (tagsList.CheckedItems.Contains(getName(row.object_)))
```

Finally we write the results to both the console and the output file:

```
Console.WriteLine(getName(row.object_) + ":" + x + ", " + y + ", " + z);
writer.WriteLine(getName(row.object_) + ": " + truncX + ", " + truncY + ", " + truncZ);
```

APPENDIX E. HP LD4200tm Specifications (adapted from HP, 2010)

Resolution:	1920 x 1080
Viewing Angles:	178/178
Brightness:	500 cd/m2
Contrast Ratio:	1000:1
Response Time:	9ms
Color Gamut:	72% NTSC
Connectivity:	VGA, HDMI, Component (2), Composite, S-Video, VGA (for tiling), USB (for touch enablement)
Dimensions:	40.3 x 24.2 x 4.7-inch (102.3 x 61.5 x 11.9 cm)
Weight:	67.5-pounds (30.7 kg, w/o stand)
Operating Temperature:	41 to 95F (5 to 35C)

APPENDIX F. Instructions for Ubisense system setup

(1) Needed equipments:

Sensors

Mounting equipments

Cables (4 regular network cables and 3 thick timing cable)

Computer

Software: Ubisense package, DHCP server, solver

Switch and power: PoE switch plus its power cables, power generator for outdoor scenario

Level

Cutter

(2) **Layout design steps**

Conceptual connectivity design (daisy chain, star, extended start)

Decide on where to put the sensors in the yard

1. Decide on where to put one tag for calibrating the sensors
2. Decide on two, easy to measure, points on the yard
3. Decide on how to run the cables and protect them
4. Decide on where to put the switch
5. Draw the connectivity map
6. Decide on the reference point (0,0,0)
7. Fill out table (Calibration tag ID info, Sensors: x,y,z and mac)

(3) Site preparation

Note: activities with letter "P" could be done in parallel

P1: Fix the sensors on the mounting device on the designated place

P1: Put the switch on its designated place and attach to the power

P1: Run the network cables from switch to sensors and fix the cables

P1: Run the timing cables based on the connectivity map (most often between the sensors)

(4) Measurements

P2-0: Measure the “area”: W,L,H

P2-1: Measure x, y, z of the sensors (one by one or using the solver)

-Decide on two points and measure the x, y of them, (preferably we set the (0,0) on the corner of two walls in the area)

-Measure the distance from the points to the sensors

-Enter them in the solver

-Get the xyz of the sensors

P2-2: Measure the reference point for calibration (x y z) (could be the same points in the last step)

(5) Basic configuration

Note: sensors are now turned off because they are not attached to the switch

All cables should be connected properly (check if they are loos/ timing cables are connected to upper left)

1. Restart the pc (switch is powered, sensors are not connected, pc is connected to switch)
2. Attach the computer to the switch
3. Start DHCP server
4. Open platform control
5. Make sure the services are running (no prefix, not in standalone mode)
6. Open location engine
7. Open log tab
8. Connect the sensor cables
9. Looking at logs to see if we receive logs
10. Move sensors to cell
11. Define master (checkboxes)
12. Look at the LEDs

(6) Software configuration

Note: So far the sensor MAC addresses should be under “available sensors”

1. Open “site manager”
2. Area tab
3. Open note pad
4. Type the coordinates of the area the file (P2-0)
5. Save in .dat file

6. Create walls in the area tab >load walls> load .dat
7. Go to the cell tab and load the area
8. Extend the cell
9. Open location engine
10. Load area and cell
11. Drag the sensors to the area
12. Select the master and check: “master”, “timing source” and “disable sleep”
13. Check the RF power of the cell (must be 250)
14. Check the LEDs (should be solid green)
15. Enter XYZ of the sensors (using P2-1)
16. Put the tag on the calibration point (using P2-2)
17. Do the dual calibration

(7) Assign tags to objects

1. Open “Site Manager”
2. Go to tab “Objects”
3. Click “Objects” on the menu, and select “New”
4. Create new object and type
5. Open “Location Engine Config”
6. Go to tab “Owners”
7. Click “Ownership” on the menu, and select “New” and assign tags to objects

(8) Set update time slots

1. Open “Location Engine Config”
2. Go to tab “tags”
3. Double click on lines, and select the Slower QOS, the Faster QOS, and the Threshold.

(9) Using Logger to record data

1. Run “\ubisense\joshua Materials\Logger\V4\bin\Debug\UbisenseLogger.exe”
2. Click “Record/Playback file…”
3. Create a new file with “.txt” under a specified folder
4. Select objects need to be monitored
5. Go to tab “Record” and click on the red button
6. Captured events will be shown in the frame

APPENDIX G. A work sheet for organizing the data of the system calibration

Date:		Feb. 20, 2011	**Place:**	TML studio
Sensor name		Sensor-1	Sensor-2	Sensor-3
MAC address		00:11:CE:00:1C:F9	00:11:CE:00:1C:00	00:11:CE:01:1C:EF
Sensor position	**X**	0.5	0.5	7.9
	Y	0.5	10.0	4.23
	Z	1.86	1.6	1.81
Ref. tag name		020-000-059-016		
Ref. tag position	**X**	3.81		
	Y	5.05		
	Z	1.28		