

Towards Internet of Things Interaction Framework Using Geometric Annotated Multimedia Objects

by

Abu Saleh Md Ma Rahman

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the Ph.D. degree in
Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

© Abu Saleh Md Ma Rahman, Ottawa, Canada, 2017

Abstract

The prevalent visions of ambient intelligence leverage natural interactions between users and available services in a smart space. In recent years, we have seen a huge interest from industry and academia in using handheld devices to interact with things, places and people in the real world. To facilitate such interactions, things are usually annotated with RFID tags or visual markers. These tags or markers are read by a handheld device equipped with an integrated RFID reader or a camera, in order to fetch related information and initiate further actions. Interacting with the Internet of Things (IoT) in a real environment has become increasingly desirable and feasible. This thesis contributes to the domain of physical interactions with IoT; however, we use a spatial-geometric approach instead of RFID or marker based solutions. Using this approach, for example, a user can point his/her handheld device to an annotated thing, from a distance, for the purpose of interaction. The pointing direction and location is determined based on the fusion of the mobile position and of the accelerometer data of the handheld device. To annotate things, their geometric coordinates are specified and related information or services are associated to them. In this thesis, we present a comprehensive and extensible framework to integrate various physical interactions with IoT into multimedia applications. The framework supports the implementations of pointMe, touchMe, and context-aware based interactions with geometrically annotated IoT. We define specific methods and practices that can be incorporated in order to build the interactions. We realize smart home, atlas learning, presentation interaction, smart haptic interaction, and learning based video interaction game prototypes in order to perform experiments and demonstrate the applicability and potential of the proposed geometric based annotation approach. In the analysis of the interaction techniques of the prototypes, we present the advantages and disadvantages of the geometric based annotation of IoT as seen by potential users, in comparison to RFID tags or visual markers based approaches.

Acknowledgements

This thesis would not have been possible without the mercy of Almighty, Omnipotent and Omniscient, to whom I express my gratitude. Thereafter, I would like to thank my supervisor Professor Abdulmotaleb El Saddik, for providing me with the opportunity to pursue a PhD as a member of the Multimedia Communication Research Laboratory (MCRLab) and Distributed & Collaborative Virtual Environments Research Laboratory (DISCOVER), EECS, at the University of Ottawa. His tips and encouragement enabled me to complete this thesis. Furthermore, he introduced me to many researchers that influenced my point of view, my work and my research in many different ways.

My thanks go to my colleague Anwar Hossain for his guidance regarding my work. I am particularly grateful to all the fantastic colleagues who did their project and thesis within the scope of this work. Among them Sk Alamgir Hossain, Kazi Masudul Alam, and Ali Karime had the largest influence on this thesis. I would like to thank my colleagues at the MCRLab and DISCOVER laboratory, especially Anisur Rahman, Shamim Hossain, Fawaz A. Alsulaiman, Mohamad Hoda, Hawazin Badawi, and Basim Hafidh.

I would like to thank my parents and my brother for their motivation, support, prayers and love throughout my whole life and especially during the years of my education. I extend my deep gratitude to both my father, Md Saleh-ur-Rahman and my father-in-law, M Ashraful Islam, whose constant prayers, and guidance made all of this possible.

Last but not least, I would like to express my deepest gratitude to my wife Ashrofee. This thesis would have never happened without the love, patience and unending support of her during the most challenging periods of my PhD work. Thanks to my dear children Mahdi Rahman (Ameer) and Majdi Rahman (Omaar) who were born during my studies.

Thanks to the numerous people who contributed to my life and work during the past years but whose names have not found their way into the acknowledgements.

Contents

1	Introduction	1
1.1	Background and Motivation	1
1.2	Existing Problems	3
1.3	Research Objectives	6
1.4	Thesis Contributions	7
1.5	Scholarly Output	8
1.5.1	Papers in Referred Journals	8
1.5.2	Papers in Referred Conferences	9
1.5.3	Chapter in Book	11
1.6	Thesis Outline	11
2	Background and Related Work	13
2.1	Background	13
2.1.1	Pervasive and Ubiquitous Computing	13
2.1.2	Mobile Spatial Interaction	14
2.1.3	Bridge between Virtual and Real	16
2.1.4	Context-aware Smart Interactions	17
2.2	Types of smart mobile interaction	18
2.2.1	TouchMe Interaction	19
2.2.2	PointMe Interaction	20
2.2.3	ScanMe Interaction	22
2.2.4	Location Aware Interaction	23
2.2.5	Summary	24
3	Smart Mobile Interaction Framework	27
3.1	Overall Architecture	29
3.1.1	Smart Environment Server	29

3.1.2	Interaction Controller	31
3.1.3	Mobile Device Client	32
3.1.4	User Client	33
3.1.5	Smart Multimedia Objects	33
3.2	Multimedia Renderer	34
3.2.1	Physical Device Renderer	35
3.2.2	Home Audio Video Renderer	35
3.2.3	Haptic Renderer	36
3.3	User's Mobile Position Determination	37
3.3.1	IR Camera Based Approach	37
3.3.2	3D Camera Based Approach	45
3.4	Annotation Scheme	46
3.4.1	Annotation Requirements	46
3.4.2	Annotation Language	47
3.4.3	3D Annotation Scheme	48
3.4.4	2D Annotation Scheme	51
3.5	Interaction with Geo-Annotation	52
3.5.1	Annotation to pointMe Mapping	52
3.5.2	Annotation to touchMe Mapping	57
3.5.3	Annotation to scanMe Mapping	61
3.5.4	Annotation to Context Mapping	65
3.6	Summary and Conclusion	71
4	Annotated IoT in Smart Home	72
4.1	Introduction	72
4.2	PointMe Interactions with Home Things	72
4.2.1	Prototype Description	73
4.2.2	Annotation of IoT	74
4.2.3	Realization and Measurements	75
4.3	TouchMe Interactions with Home Things	86
4.3.1	Prototype Description	86
4.3.2	Annotation of IoT	88
4.3.3	Realization and Measurements	90
4.4	Context-aware Interactions with Home Things	92
4.4.1	Prototype Description	94

4.4.2	Annotation of IoT and Scene Programming Editor	98
4.4.3	Realization and Measurements	100
4.5	Summary and Conclusion	111
5	Haptic Interactions with Annotated IoT	112
5.1	Introduction	112
5.2	TouchMe Based Social Haptic Interaction	112
5.2.1	Prototype Description	113
5.2.2	Annotation of IoT	116
5.2.3	Realization and Measurements	117
5.3	TouchMe Based Haptic Interaction with EBooks	120
5.3.1	Prototype Description	121
5.3.2	Annotation of IoT	123
5.3.3	Realization and Measurements	124
5.4	Summary and Conclusion	128
6	Annotated IoT in Edutainment	129
6.1	Introduction	129
6.2	PointMe Interactions with a Smart Multimedia Screen	129
6.2.1	Prototype Description	130
6.2.2	Annotation of IoT	134
6.2.3	Realization and Measurements	134
6.3	PointMe based Interactions with Physical Atlas	141
6.3.1	Prototype Description	142
6.3.2	Annotation of IoT	147
6.3.3	Realization and Measurements	148
6.4	PointMe, TouchMe, and ScanMe interactions with an Annotated Movie System	153
6.4.1	Prototype Description	154
6.4.2	Spatial Television Game	155
6.4.3	Wii based PointME Selection	156
6.4.4	Kinect based Selection Approach	157
6.4.5	Annotation of IoT	158
6.4.6	Realization and Measurements	159
6.4.7	Comparison between various Interactions	163
6.5	Summary and Conclusion	168

7	Conclusion and Future Works	169
7.1	Conclusion	169
7.2	Future Work	171

List of Tables

2.1	Common mobile interaction techniques	19
2.2	Comparison of RFID and NFC interaction schemes	20
2.3	Comparison of various visual code based mobile interaction approaches.	21
2.4	Related work comparison in pointing interaction techniques.	22
2.5	Comparison between different annotation schemes	25
2.6	Overview of smart mobile interaction methodologies	26
3.1	LOM categories used to package geo-spatial data.	48
3.2	Context parameters used in the framework	66
4.1	Accelerometer raw data based target analysis	80
4.2	User satisfaction on the overall evaluation in Likert scale.	85
4.3	Usability test questions to the user for smart home interaction.	91
4.4	User satisfaction on the overall evaluation in Likert scale.	92
4.5	Location calculation accuracy with respect to the geometric annotation maps.	101
4.6	User posture accuracy.	108
4.7	User stand posture classification accuracy.	109
4.8	User sit posture classification accuracy.	109
4.9	User recline posture classification accuracy.	110
4.10	User lie-down posture classification accuracy.	110
5.1	Usability test questions to the user.	118
5.2	User satisfaction on the overall evaluation in Likert scale.	120
6.1	Usability test questions for the parents.	162
6.2	Parent’s satisfaction in Likert scale.	163

List of Figures

2.1	Point to discover system components. [50]	15
3.1	Geometric annotation based interaction model development in an IoT system.	27
3.2	Overview of the framework.	28
3.3	Interaction channels of the system.	29
3.4	Smart Mobile Interaction framework.	30
3.5	Interaction packages of the system.	32
3.6	Heterogeneous device types are added to the Multimedia Renderer as they each implement a specific Multimedia Interface.	34
3.7	The 2D IR position capturing system of the IR camera. Here the IR emitter node $P(x, y, z)$ is activated in the virtual $ax + by + cz + d = 0$ plane facing the camera.	38
3.8	IR Camera positions shown in the indoor environment that are used to calculate the 3D location and orientation information of the user using two 2D IR positions.	40
3.9	Geometric orientation, IR camera frustum, FOV angle and distance based relation of the IR emitter point P, with respect to the two IR cameras.	42
3.10	Streaming 3D depth images of a 3D camera.	49
3.11	2D layout based physical IoTs annotation approach by using 3D image data.	49
3.12	3D virtual IoT annotation.	51
3.13	Movie objects annotation with respect to timeline.	52
3.14	Manual annotation process of a map.	53
3.15	Mobile pointMe based object selection approach.	54
3.16	FOV angle variation and physical target object selection. Narrow FOV angle selects only the target objects but user maneuvering becomes difficult and results in poor accuracy.	55

3.17	Closed FOV based physical object selection approach. On the left we graphically show the horizontal spatial query process in overhead view. On the right we graphically show the 3D vertical spatial query process. Here, point p is the mobile phone position, F is the fan distance, V is the video device distance and L is the distance of the Lamp source.	57
3.18	PointMe scenarios with nearby geo-annotated objects	58
3.19	Mobile thin client based touchMe interaction.	58
3.20	Mobile touchMe app examples.	59
3.21	Mobile touchMe app for Haptic Book.	60
3.22	Mobile scanMe client for map selection.	61
3.23	User position with respect to geo-spatially annotated objects.	62
3.24	3D skeletal gestures.	63
3.25	Skeletal 3D standing postures of two users.	63
3.26	List of user gaits used in the implementation of context-aware interaction.	65
3.27	Context analysis system.	67
3.28	Difference between user postures based on 3D skeletal joint ratios.	68
3.29	Context rules incorporated into the framework.	69
4.1	PointMe interaction with annotated smart home objects.	74
4.2	3D physical objects annotation approach.	75
4.3	Annotation process of an object (Lamp).	76
4.4	The overall system components that works by fusing the 2D IR emitter locations into a 3D point and then applying accelerometer parameters in order to determine the user's point of interest.	77
4.5	Triangular IR path sampling and rotation correction using the cartesian distances.	79
4.6	Number of accurate object selections of the user at varying distance (6 – 42) feet as depicted in [A] and at different FOV angle ranging from (5 – 180), shown in [B]. The 50 voxel based target objects were placed in random order in the environment. Each of the sampling points represents a trial by the respective users at the said arrangements. E.g., the blue triangles in [A] shows the selection accuracy achieved by the subject 4 who was standing 6, 8, 10, 14, 22, 32, and 42 feet away from the randomly arranged targets in the said evaluation.	82

4.7	Number of accurate object selections of the user at varying distance (6 – 42) feet as depicted in [A] and at different FOV angle ranging from (5 – 180), shown in [B]. The 50 voxel based target objects were arranged in Row Fashion. Each of the sampling points represents a trial by the respective users at the said arrangements and the curve represents their averages. e.g., the blue triangles in [A] shows the selection accuracy achieved by the subject 4 who was standing 6, 8, 10, 14, 22, 32, and 42 feet away from the targets in the said evaluation.	83
4.8	Number of accurate object selections of the user at varying distance (6 – 42) feet as depicted in [A] and at different FOV angle ranging from (5 – 180), shown in [B]. The 50 voxel based target objects were arranged in columnar fashion. Each of the sampling points represents a trial by the respective users at the said arrangements and the curve represents their averages. e.g., the blue triangles in [A] shows the selection accuracy achieved by the subject 4 who was standing 6, 8, 10, 14, 22, 32, and 42 feet away from the targets in the said evaluation.	84
4.9	Usability study with the mobile interaction system. The participating users responded to the listed assertions and rated those in Likert scale. The average of their ratings are shown in percentage.	85
4.10	TouchMe interaction with annotated smart home things.	87
4.11	The flexible object annotation scheme allows the user to annotate objects and to set rules that create events to render real world services.	88
4.12	An overview of the target object specific interaction rules stored (and could be share) in an <i>XML</i> file.	89
4.13	User response for the usability test questions that are listed in Table 4.3.	91
4.14	Comparison between the responses of users from different (a) age groups and (b) technical backgrounds.	93
4.15	Context aware interaction with annotated smart home things.	95
4.16	Programming the television to pause when kids are very close to it. . . .	96
4.17	Programming lights that responds to a user gesture when s/he is nearby.	97
4.18	Programming to encourage breaks in a long television watching session. .	98
4.19	Annotating smart space elements to allow programmable physical interaction.	99
4.20	Programming smart space by using the Interaction Rule Editor.	100
4.21	Geometric annotation of large objects and its advantages.	103

4.22	Usability comparisons of various interactions with an annotated light. . .	104
4.23	Usability study of a context-aware healthy posture awareness application.	105
4.24	Usability study of a context-aware applications to monitor baby locations in front of a multimedia screen.	106
4.25	Ease of use of the context editor application.	107
5.1	TouchMe based inter-personal social haptic interactions, such as hug, kiss, and tickle in Second Life.	114
5.2	The Haptic jacket controller and its hardware components. Array of vibro- tactile motors are placed in the gaiter like wearable cloth in order to wirelessly stimulate haptic interaction.	116
5.3	The flexible avatar annotation scheme allows the user to annotate any part of the virtual avatar body with haptic and animation properties. When interacted by the other party, the user receives those haptic rendering on his/her haptic jacket and views the animation rendering on the screen. .	117
5.4	Usability study of the Second Life haptic interaction system.	119
5.5	Comparison between the responses of users from different (a) gender, (b) age groups, and (c) technical background.	119
5.6	TouchMe based haptic interactions with an annotated eBook.	122
5.7	Annotation editor for the e-book document (i.e. PDF)	124
5.8	User response to various haptic feedbacks in Likert Scale	125
5.9	Usability study of the HE-Book system. a) Likert scale presentation of the attention, retention of content and entertainment metrics in a traditional text reading experience, b) The same study with text-visual rendering added in the reading process, c) Incorporating audio-visual simultane- ously to determine the changes in reading process d) Visual-haptic feed- backs added in the reading process, e) Combining the hapto-audio-visual feedbacks in the paragraph reading experience to report the user scores. .	126
5.10	Knowledge acquisition scores of the users in different multimodal aug- mentation, where every user had to answer two questions related to each multimodality.	127
6.1	PointMe based interactions with annotated multimedia screen contents. .	131
6.2	A description of the steps of the system after the learner activates the pointMe based searching and retrieval of the vLOM elements.	132
6.3	System architecture of the LOM viewing client.	133

6.4	Notice board based mobile metaphor application for viewing and organizing LOMs.	133
6.5	Presentation media authoring application. Any annotation of the media transforms along a path with the timeline of the presentation (if applicable).	135
6.6	A learner pointing the targeted screen to select a vLOM using his handheld device. The IR camera, mobile and IR emitters are also shown.	135
6.7	Comparison of the accuracy of the system with changing inter-distance of the vLOM elements.	137
6.8	Accuracy of the system with varying position of the learner from the screen.	137
6.9	Searching and retrieval time of the different size vLOM elements.	139
6.10	Usability study when the user is pointing in order to retrieve the vLOM elements.	140
6.11	Usability study with automatic retrieval of the vLOM elements.	140
6.12	PointMe and TouchMe based interactions with annotated physical atlas.	142
6.13	2D IR point clouds and spatial selection query of the annotation map. . .	143
6.14	Selection feedback from the game module in the mobile client.	145
6.15	Map annotation and authoring scheme.	147
6.16	PointMe selection accuracy of the map LOMs.	149
6.17	Selection and retrieval time of the map LOMs.	150
6.18	Average of the responses (in percentage) from Young and Adult users of the physical interaction system.	151
6.19	Likert scale response of the users of the given assertions.	152
6.20	Usability study of pointMe and RFID based interactions with world map.	153
6.21	PointMe, TouchMe, and ScnaMe interactions with an Authored Movie System.	155
6.22	Textual and geometrical visual answer selection approaches.	156
6.23	Presentation of media authoring application. Any annotation of the media transforms along a path with the timeline of the presentation.	159
6.24	Comparison of the learning scores of two user groups with and without interaction options.	160
6.25	Comparison of the number of accurate selections by the users.	161
6.26	Parent's response on Likert scale for the given assertions [Q1~Q3] as listed in Table 6.1.	162
6.27	Comparison of the accuracies of various physical mobile interaction schemes.	164

6.28	Comparison of the interaction times of various physical mobile interaction schemes.	165
6.29	Comparison of the average interaction times of various physical mobile interaction schemes.	165
6.30	Usability comparison of perceived ease of use of various physical mobile interaction schemes.	166
6.31	Usability comparison of perceived application and entertainment values of various physical mobile interaction schemes.	167
6.32	Usability comparison of intention of use of various physical mobile interaction schemes to solve real world problems.	167
6.33	Usability comparison of perceived awareness of computer systems working in the middle (determining transparency of the computation in the user interaction)	168

Chapter 1

Introduction

1.1 Background and Motivation

The prevalent visions of ambient intelligence [60][18] leverage natural interactions between users and available services in a smart space[144][82][37]. In recent years, we have seen a huge interest from industry and academia in leveraging user mediated natural interactions with things, places and people in the real world[130][19]. The goal of Internet of Things (IoT) is to bring computing elements into the real physical world and to allow people to interact with them in a natural way. In a typical IoT environment the user should be able to interact with computing elements by gesturing, pointing, or moving [30], or by using a combination of these interactions. An important step towards implementing the vision of IoT computing is the use of mobile devices, which are the first pervasively available computer and interaction devices. The increasing prevalence of mobile devices has led to the idea of using them as intuitive interaction devices [20][8][83][116]. People are conducting new experiments to interact with their physical environment by using mobile handheld devices in order to obtain digital information about the surrounding objects/things and initiate further actions relevant to those entities [93]. To facilitate such interactions, devices or things are usually annotated with RFID tags[143][138][111][69] or visual markers[114][128][115]. These annotations are read by a handheld device equipped with an integrated RFID reader or a camera, in order to fetch related information about the objects and initiate further actions.

These interactions are commonly known as smart mobile interactions[20][3] or as physical browsing [139]. Such smart mobile interactions are the focus of this dissertation. They enable access to ubiquitous services through the use of multimedia objects such

as a physical object, a geo spatial location, a virtual object etc. The use of smart mobile interactions simplifies the discovery and use of ubiquitous services, and enables the development of context aware and naturally interactive applications [103][69][102] [106]. The most important smart mobile interaction techniques are identified as touching [104], pointing [107], scanning [85], and implicit or explicit multimedia selection [148][122] etc. In mobile interaction research these interactions are referred to as touchMe, pointMe, and scanMe respectively.

In the last decade, IoT research has progressed with regards to the question of how to combine virtual information with a users physical location and its surroundings. Users have the ability to access digital information that has been attached to physical places called points of interest (POI). PointMe based natural interactions can be used to point and direct people to places of interest or act as virtual geographic pointers for the selection of surrounding objects [36]. For example, the Geo-Wand [51] is a research concept that enables tourists to point their mobile devices at a historic building and obtain information about that building. The RELATE system presented in [52] shows a list of available target devices in the user's display with respect to their spatial arrangement. As these target devices use magnetometer/accelerometer and GPS data, the RELATE system is only suitable for outdoor-applications.

Similarly, pointing a handheld device towards a home appliance and interacting with it seems appealing. For example, a user may point his/her handheld device towards the television screen to interact with the content of the screen, s/he may point towards a home appliance to switch it "ON", or a reading light is automatically turned "ON" when s/he sits on a reading chair, etc. The natural selection based service acquisition from surrounding objects leverages the familiar relationship between a user and these objects. A smart IoT system could benefit from this relationship and utilize it for context based service delivery. In turn, the approach may help lower the complexity of the user interface and may make the computing service delivery mechanism more appealing to users [59][154].

IoT envisions a world, where millions of sensors are connected to the Internet and where it is not impractical to process all the data collected by those sensors. Therefore, context-awareness and intelligent mobile interactions[24] will play a critical role [92] in the user's interaction with IoT. The emphasis should be on making the user interactions smart, convenient, and friendly. Smart and intuitive user interaction is especially important in pervasive applications, where people collaborate on both virtual and physical worlds [143]. We focus on real world based physical mobile interaction collaboration and

address smart interaction possibilities with physical and virtual world objects. We further enhance the typical collaboration between virtual environments and the real world in order to create more realistic interactive applications. More specifically, we attempt to bring forth new modes of communication that help the user interact with not only virtual objects, but also real world objects to obtain relevant information or invoke associated services. In order to bridge the gap between virtual and real, and provide real world services, we attempt to identify smart mobile interactions with virtual multimedia objects [64][100][4].

The goal of this thesis is to augment everyday things with computing services in the IoT paradigm. The services can be triggered when a user interacts with the things in a particular context [91]. In this process, instead of requiring a user to modify his/her behaviour with a computer or a mobile device, the computing devices integrate with things to accommodate the user's habits and work patterns. This presents a huge opportunity for the researchers to learn about and to leverage our natural interactions with things and develop appealing and useful services/applications.

1.2 Existing Problems

IoT anticipates a world where our surrounding things are connected with millions of sensors and communicate with each other transparently in an automatic fashion [91]. Understandably, it is not feasible to process all the data collected by those sensors. Also, in such a sensory environment, the traditional application based approach used to connect sensors to applications individually and interact with them manually becomes impractical[92]. Therefore, a greater focus on context-awareness[91] and intelligent mobile interaction[24][70] will play a critical role in IoT research [92].

In physical mobile interaction research, IoT multimedia objects are usually annotated with virtual information. Existing mobile interaction applications have adopted RFID/NFC tags [57][85][123] or visual markers [128][134][90] based on physical annotation solutions. In these schemes, physical chips or markers are attached to an object in order to allow a particular type of interaction with that object. Typically, a handheld device equipped with an RFID reader or a camera is used to either touch/point/scan the tagged objects or take pictures of the visual markers to obtain relevant information or invoke associated services [20][3]. However, there are issues related to RFID and visual marker based approaches, which impose some restrictions while interacting with IoT. For example, the limited range (10-20cm) of a passive RFID tag can only facilitate interac-

tion with an object from a close distance [118]. Besides, a large object requires many tags to be placed on its surface so that a handheld device can locate any of those tags and read its associated information. This requires extra effort and additional cost to tag the physical objects. Moreover, in cases where multiple RFID readers are actively used, there is an area between them where an RFID tag cannot communicate with either reader. This phenomenon is called Reader-to-tag interference or tag jamming[127]. In situations, where numerous RFID tags are to be read by a single RFID reader, the tags can collide and cancel each other out, leading to retransmission of tag IDs that results in a wastage of bandwidth and an increase in the total delay [146]. NFC based mobile interactions also have their limitations. An NFC device can only communicate with one object at a time. Unlike RFID, NFC devices do not use broadcast methods to communicate with multiple devices [127]. The NFC devices send information to a target and further can only accept a response from that target. Other NFC devices will ignore the communication and wait for their turns. These limitations add restrictions when designing interactions by using NFC tags in IoT applications.

On the other hand, visual markers or Quick Response (QR) codes are placed on the visible surface of the objects so that the handheld device can easily read them and fetch the associated virtual contents [9]. There are 40 different versions of QR Codes and each of these versions is accompanied by four levels of error corrections [81]. QR codes are widely used to save time and effort as information (e.g. addresses, url, etc) does not have to be entered manually into handheld devices. They appear in advertising, in print media, on business cards, on products, on websites and on vending machines. However, these optical barcodes suffer from several drawbacks. First, human intervention is required to scan them and objects must be physically manipulated to align the barcodes with the scanner[127]. Unlike an RFID tag, which can be placed in the background, the marker-based approach eventually suffers from visual obtrusiveness as they take up visible surface space [145].

In mobile spatial interactions it is common to use magnetometer/accelerometer and GPS data to allow access to digital information that has been attached to physical Points Of Interests (POI) [36][51][87]. PointMe based natural interaction can be used to point and direct people to places of interest or act as virtual geographic pointers for the selection of surrounding objects.

Pointing a handheld device towards a home appliance and interacting with it seems appealing; however, as the user location data and POI map are unavailable, it is challenging to bring such interaction paradigms into indoor spaces[149]. There are expensive

solutions that utilize pulsed DC magnetic fields ¹ to obtain live user location data. This technology is used in the computer animation industry, but it is quite expensive and its range is limited. Researchers have used RFID[69] [152], RF [7], and IR emitters [142, 149, 6][103] to track users in indoor spaces and produce live user location data that can be of importance in many location aware IoT system. Unlike an outdoor GPS, these indoor solutions suffer from many drawbacks. They scale poorly due to limited range (IR, RFID), they are susceptible to magnetic interference such as tag jamming (RF, RFID), and they necessitate installation and maintenance costs.

Researchers are currently exploring mobile application based IoT in indoor spaces. For example, SmartThings ² connects household appliances, door locks, thermostats, etc to a central router, which can then be controlled with a smartphone app. The quality of mobile app based IoT is limited by its User Interface (UI) design. In most cases, app based interaction becomes tedious and repetitive for simple tasks [72]. They also fail to consider the inter-relationship of the users with the things. Mobile app based solutions [8] for IoT interactions often require more interactions to complete a simple task. For example, unlocking a car door using a mobile app may not always be feasible [72].

It should be noted that the interaction action realization suffers from the technical limitations imposed by the choice of an annotation mechanism. The choice of the annotation mechanism is also affected by the selection of the type of interaction technique, the location and the type of the things (virtual or real), the distance between the things and the user, the service related to the object, the hardware capabilities of the hand-held reader device etc. For example, it is not practical to use RFID tags to annotate a virtual object. Furthermore, the design, development and implementation of such techniques and the applications based on them are often time-consuming tasks. In order to implement each type of interaction, a number of different hardware units and setups are usually required. For example, NFC tags only support touchMe type interactions and NFC chip augmented smart objects can be read with NFC read capable handheld devices. Also, changing the type of interactions for a smart mobile application requires changing the hardware settings and may not be practical or cost effective. Moreover, with these individual hardware and setup requirements it may be challenging to develop applications with deeper and easier smart mobile interaction capabilities.

In this work, we aim to overcome some of these restrictions in the IoT paradigm with a novel geometric spatial annotation approach. Geometric spatial annotation operates

¹Ascension Technology Corp., "<http://ascension-tech.com>"; last accessed on June 2013

²Samsung Smartthings, "<http://www.smartthings.com/how-it-works>"; last accessed on April 2015

on the application level and supports different interaction techniques. The scheme uses a logical annotation of IoT that describes position and service end points for that object and may allow an application to implement different type of interaction paradigms easily.

We propose an application framework that supports not only traditional pointMe, touchMe, and scanMe interactions with things but also provides support to build many context-aware interaction [91] paradigms by incorporating the inter-relationship between the things and the users in the IoT paradigm. For example, when a user sits on a reading chair, the reading light is automatically turned "ON" etc.

1.3 Research Objectives

The main focus of this thesis lies in smart mobile interactions, in which mobile interactions are used as pervasive mediators for interactions with real and virtual world multimedia things. Our proposed mobile interaction approach requires the annotation of physical objects or sub-regions of interest of the objects with their geometric coordinates. Unlike RFID chip, NFC chip, or visual marker based annotation approaches that operate at the physical or device level, the geometric annotation approach incorporates logical annotation and operates at the application level. In order to facilitate pointing to an object using the handheld devices, the proposed approach adopts a sensor fusion mechanism that combines the data from multiple sensors, including a handheld accelerometer, to calculate the position of the mobile user, the location of his/her handheld device and the target direction in which it is pointing. In the case of a pointMe interaction, if the geometric coordinates of the pointing direction matches the pre-annotated geometric boundary, the information or service endpoint stored for that boundary is shown or activated with the help of the handheld device. Alternately, the functionalities available for the target objects will appear in the handheld device for invocation.

In the domain of mobile spatial interactions [48][51][58], researchers have demonstrated the use of handheld devices to interact with remote buildings or objects. However, those approaches utilize GPS data from handheld device along with a map to provide geospatial selection of interesting outdoor things [129]. In this thesis, we attempt to provide a similar remote object pointing facilities to the user in the convenience of an indoor environment.

RFID [69], visual marker [23], or laser based solutions may not permit a deeper integration and more natural interaction possibilities with the application as they operate in the physical layer. Whereas, the spatial geometric annotation approach operates in

the application layer and allows the customization of the interaction layer, event triggers, context parameters, etc. Some of these customization features are leveraged in this thesis to show the capabilities of the spatial geometric annotation in a number of different prototype applications that are challenging to achieve with other solutions such as RFID based approaches.

The objectives of our research are:

- To find a way to facilitate communication between virtual and real world.
- To investigate an intuitive, user-friendly and flexible 2D/3D annotation mechanism to actuate the communication between virtual/real things, people, and computing elements in a transparent and seamless manner.
- To design and implement intuitive touchMe, pointMe, scanMe, and context-aware interaction scenarios with IoT objects by using the spatial geometric annotation approach that provides seamless and transparent integration with many multimedia applications.

1.4 Thesis Contributions

The major contributions of this thesis are listed in the following:

- Design and development of geometric annotations (geo-annotations) to logically describe virtual and real world multimedia objects. The geo-annotations provide logical spatial mapping of the multimedia objects. One or more geo-annotations can be processed concurrently at the application level instead of at the physical level. The geo-annotations are described by using standards for better interoperability and portability.
- Design of a physical mobile interaction framework to implement applications by leveraging geo-annotations. The framework supports the implementation of commonly used natural interaction paradigms such as touchMe, pointMe, and scanMe. Additionally, user mediated selections of the geo-annotations in a context-aware settings are also developed. As the framework adopts geo-annotations to describe the IoT, additional interaction mechanisms can easily be added to the application.
- Design and development of a sensor fusion algorithm to combine the data from multiple IR sensors and handheld device's accelerometer, gyroscope, and compass

data to calculate the position of the mobile user and the target geo-annotation pointed at by the handheld device.

- Design and development of algorithms and tools to allow user-programmable context scenario definitions by incorporating the geo-annotation maps, a user's relative position, body posture, body gait, and other sensory parameters. The context scenario based smart home interactions make the intermediate computing elements transparent to the user and reduces the number of interactions needed to activate certain services.
- Design and development of virtual geo-annotation of screen objects including 2D shape, text, and 3D objects to bring virtual events into the physical world. By interacting with virtual geo-annotations, real users get haptic feedback when they interact with virtual avatars, and readers receive multimedia when they interact with the eBook texts.

1.5 Scholarly Output

The following papers have been published as a result of this research:

1.5.1 Papers in Referred Journals

- **Abu Saleh Md Mahfujur Rahman**, and Abdulmotaleb El Saddik. *Mobile PointMe-based spatial haptic interaction with annotated media for learning purposes*. Multimedia Systems, vol.19, no.2, pp.131–49. March 2013.
- K M Alam, **Abu Saleh Md Mahfujur Rahman**, and Abdulmotaleb El Saddik. *Mobile haptic e-book system to support 3D immersive reading in ubiquitous environments*, ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP), vol.9, no.4, pp.27, 2013.
- Ali Karime, M. Anwar Hossain, **Abu Saleh Md Mahfujur Rahman** et. al., *RFID-based interactive multimedia system for the children*, Multimedia Tools and Applications, vol.59, no.3, pp.749–774, 2012.
- S K Alamgir Hossain, **Abu Saleh Md Mahfujur Rahman**, and A El-Saddik, *Measurements of Multimodal Approach to Haptic Interaction in Second Life In-*

terpersonal Communication System, IEEE Transactions on Instrumentation and Measurement, vol.60, no.11, pp.3547–3558, Nov 2011.

- **Abu Saleh Md Mahfujur Rahman**, M. Anwar Hossain, and Abdulmotaleb El Saddik. *Spatial-geometric approach to physical mobile interaction based on accelerometer and IR sensory data fusion*, ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP), vol.6, no.4, pp.28, Nov 2010.

1.5.2 Papers in Referred Conferences

- Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *Promoting Active Participation of the Learners in an Authoring based Learning Movie System*, AICCSA 2016, Nov 29 Dec 02, Agadir, Morocco. (Best paper award).
- Abu Saleh Md Mahfujur Rahman, Kazi Masudul Alam, Abdulmotaleb El Saddik, *A Prototype Haptic EBook System to Support Immersive Remote Reading in a Smart Space*, HAVE 2011, 14-17 October, Jiangxi, China.
- Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *Mobile based Multimodal Retrieval and Navigation of Learning Objects using a 3D Car Metaphor*, ACM International Conference on Internet Multimedia Computing and Service (ICIMCS 2011), 5-7 August 2011, Chengdu, China.
- Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *Remote Rendering based Second Life Mobile Client System to Control Smart Home Appliances*, IEEE Virtual Environments, Human-Computer Interfaces, and Measurement Systems, 19-21 September 2011, Ottawa, Canada.
- SK Alamgir Hossain, Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *Bringing Virtual Events into Real Life by Bridging the Gap Between Virtual and Real in Second Life Home Automation*, IEEE Virtual Environments, Human-Computer Interfaces, and Measurement Systems, 19-21 September 2011, Ottawa, Canada.
- Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *Virtual Rendering based Second Life Mobile Application to Control Ambient Media Services*, 4th International Workshop on Semantic Ambient Media Experience (NAMU Series), SAME 2011, 29th June-2nd July 2011, Brisbane, Australia.

- Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *Mobile PointMe based Pervasive Gaming Interaction with Learning Objects Annotated Physical Atlas*, IEEE International Conference on Multimedia and Expo (ICME WS 2011), AIME2011, July 11-15, 2011, Barcelona, Spain.
- Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *hKiss: Real World based Haptic Interaction with Virtual 3D Avatars*, IEEE International Conference on Multimedia and Expo (ICME WS 2011), HFM2011, July 11-15, 2011, Barcelona, Spain.
- SK Alamgir Hossain, Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *Fusion of Face Networks through the Surveillance of Public Spaces to Address Sociological Security Recommendations*, IEEE International Conference on Multimedia and Expo (ICME WS 2011), July 11-15, 2011, Barcelona, Spain.
- Abu Saleh Md Mahfujur Rahman, K M Alam, Abdulmotaleb El Saddik, *Augmented HE-Book: A Multimedia Based Extension to Support Immersive Reading Experience*, International Conference on Autonomous and Intelligent Systems (AIS 2011), June 22-24, 2011, Burnaby, BC, Canada.
- K M Alam, Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *HE-Book: A Prototype Haptic Interface for Immersive E-Book Reading Experience*, IEEE World Haptics Conference 2011, June 22-24, 2011, Istanbul, Turkey.
- Abu Saleh Md Mahfujur Rahman, SK Alamgir Hossain, Abdulmotaleb El Saddik, *Bridging the Gap between Virtual and Real World by Bringing an Interpersonal Haptic Communication System in Second Life*, IEEE International Symposium on Multimedia (ISM2010), Nov 27, Taiwan, 2010.
- A S M Mahfujur Rahman, A El Saddik, *Touch me interaction paradigm for physically browsing personal learning spaces*, IEEE International Conference on Multimedia and Expo, 19-23 July, Singapore, 2010.
- SK Alamgir Hossain, Abu Saleh Md Mahfujur Rahman, Abdulmotaleb El Saddik, *Interpersonal Haptic Communication in Second Life*, International Symposium on Haptic Audio-Visual Environments and Games, Oct 16-17, Arizona USA, 2010.
- A S M Mahfujur Rahman, M. Anwar Hossain, Jorge Parra and Abdulmotaleb El Saddik, *Motion-Path based Gesture Interaction with Smart Home Services*, ACM

International Conference on Multimedia, HCM track short paper, Beijing, China, October 2009.

- Jongeun Cha, Mohamad Eid, Ahmad Barghout, A S M Mahfujur Rahman, and Abdulmotaleb El Saddik, *HugMe: Synchronous Haptic Teleconferencing*, ACM International Conference on Multimedia, The Multimedia Grand Challenge, pp. 1135-1136, Beijing, China, Oct. 19-24, 2009.
- Abu Saleh Md. Mahfujur Rahman, Jongeun Cha, and Abdulmotaleb El Saddik, *Authoring Edutainment Content through Video Annotations and 3D Model Augmentation*, IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems (VECIMS), pp. 370-374, Hong Kong, China, May 11-13, 2009.
- A S M Mahfujur Rahman, M. Eid, A. El Saddik, *KissMe: Bringing virtual events to the real world*, Virtual Environments, Human-Computer Interfaces and Measurement Systems, VECIMS 2008, Istanbul, Turkey, pp. 102-105, 14-16 July 2008.
- Juan M. Silva, A S M Mahfujur Rahman, A. El Saddik, *Web 3.0: a vision for bridging the gap between real and virtual*, In Proceeding of the 1st ACM international workshop on Communicability design and evaluation in cultural and ecological multimedia, Vancouver, British Columbia, Canada, pp. 9-14, 27 Oct - 1 Nov 2008.

1.5.3 Chapter in Book

- Abu Saleh Md Mahfujur Rahman, SK Alamgir Hossain, Abdulmotaleb El Saddik, *Haptic Interaction and Avatar Animation Rendering Centric Telepresence in Second Life*, Multimedia Image and Video Processing (2nd edition), 2011.

1.6 Thesis Outline

This thesis has been organized into six chapters.

Chapter 1 introduces the motivation of the research, which is followed by the research objectives, and contributions. Further, a list of publications that have been produced as an outcome of this research are also presented. The chapter is concluded with a detailed description of the organization of the thesis.

Chapter 2 presents background information of the thesis and briefly describes the state of the art related to smart mobile interactions.

Chapter 3 discusses the design and model of the proposed framework to implement smart mobile interaction applications. We propose a spatial-geometric approach for smart mobile interaction. We also detail the novel features of the proposed approach that facilitates interaction using commonly used paradigm such as TouchMe, PointMe and ScanMe to interact with virtual and real multimedia things in a smart environment setting.

Chapter 4 illustrates the realization of pointMe, touchMe, and context-aware interaction with IoT in a smart environment. We have incorporated the proposed framework to realize three prototypes. For each prototype, we provide short description of the prototype, describe the implementation challenges, relate the development of the prototype with the proposed framework, and illustrate the achieved results.

Chapter 5 describes the realization of touchMe based interactions with IoT in hand-held device based GUI setting. The proposed framework is incorporated to realize two prototypes, where we document the approaches of geo-spatial annotation mapping of the virtual things. For each prototype, we provide short description of the prototype, describe the implementation challenges, relate the development of the prototype with the proposed framework, and illustrate the achieved results.

Chapter 6 documents the realization of pointMe, touchMe, and scanMe based interactions with IoT. Here we present the details of three prototypes that realize pointMe, touchMe, and location-aware interactions with geo-spatial annotations by incorporating the proposed framework. For each of the prototypes, we provide short description of the prototype, describe the implementation challenges, relate the development of the prototype with the proposed framework, and illustrate the achieved results.

Chapter 7 draws conclusions and offers suggestions for future work.

Chapter 2

Background and Related Work

In this chapter, we firstly introduce some general background knowledge of smart mobile interactions. Then, we present the state of the art research work. We discuss traditional physical object annotation works and compare them with the spatial geometric annotation approaches that attempt to address interactions with both virtual and real world multimedia objects. The related work is categorized into several groups to address the context and rationale of the application prototypes that highlights some unique interaction cases and to show the suitability of the proposed framework.

2.1 Background

2.1.1 Pervasive and Ubiquitous Computing

Weiser [144] compares the way we interact with our environment and the way people interact with computers. His vision of disappearing technology will allow people to use technology, invisible computers woven into the environment, without thinking about it. Mobile devices are seen by Weiser as just an intermediate step to his vision of pervasive computing. Satyanarayanan [119] supports this by saying that distributed systems are predecessors of mobile computing which itself is a predecessor of pervasive computing.

Ubiquitous computing [95][16][56] is a model of human-computer interaction in which information processing has been thoroughly integrated into everyday objects and activities. In the course of ordinary activities, someone "using" ubiquitous computing engages many computational devices and systems simultaneously, and may not necessarily even be aware that they are doing so. More formally, Ubiquitous computing is defined as "machines that fit the human environment instead of forcing humans to enter theirs."

When primarily concerning the objects involved, it is also physical computing, the Internet of Things [2], haptic computing, and things that think [95]. Tangible user interfaces are important step towards Ubiquitous computing. The vision of tangible user interfaces and tangible bits were introduced by Hiroshi Ishii and Brygg Ullmer [66]. In this pervasive computing research, physical objects are linked with bits and users can access the bits via interacting with these objects. Tangible user interfaces attempt to augment real world objects with digital information that can be used by a user through interactions of things.

2.1.2 Mobile Spatial Interaction

In the last decade IoT research progressed on how to combine virtual information with the users physical location and its surroundings [2][43]. Users have the ability to access digital information that has been attached to physical places called points of interest (POI). PointMe based natural interaction can be used to point and direct people to places of interest or act as virtual geographic pointers for the selection of surrounding objects [36]. The Geo-Wand [51] is a research concept that enables tourists to point their mobile phone at a historic building and learn about it. In this way, users can access spatial information wirelessly [120]. The RELATE system presented in [52] shows a list of available target devices in users display with respect to their spatial arrangement. Their approach use magnetometer/accelerometer and GPS data and is suitable for outdoor-application.

To understand how such spatial mobile interaction work, at first we discuss the point-to-discover (p2d) project [50]. The project turns a mobile phone into an interactive pointing device. In the inner city of Vienna, p2d can guide a user to nearby restaurants, cafes, and bars. In order to obtain recommended listings of restaurants, cafes, and other point of interest the user points her mobile device to the desired locations. A smart, orientation-aware map on the mobile device guides the user to the point of interest location. The p2d system offers the main technological components needed for mobile spatial interaction and information access: (a) mobile user interface techniques, (b) platform for geo-referenced services, (c) 3D-GIS data aggregation algorithms, and (d) hardware sensor board prototypes. These components are shown in Figure 2.1. The p2d project relies on the GPS data in order to calculate point of interest searches by using its proprietary geo-spatial selection algorithms. The short description of the main components of this project are detailed below:

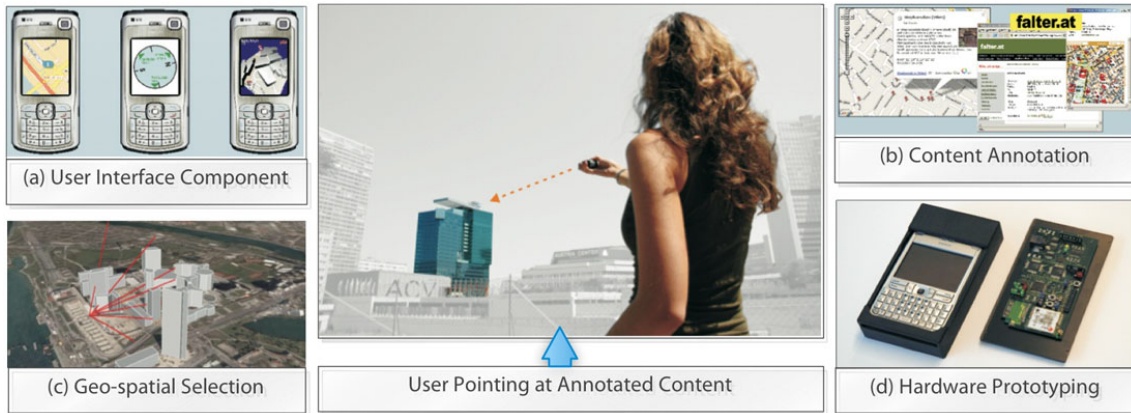


Figure 2.1: Point to discover system components. [50]

- (a) *User interface component*: A mobile application has been installed in the user's handheld device that provides visual cues and directions to the pointed locations. The application enhances the user's understanding of the geo-spatial data obtained from the GIS system.
- (b) *Content annotation*: The p2d platform provides interfaces to different sources of geo-referenced content from various sources on the Web. Through the interface it is possible to annotate various point-of-interest location data into the GIS system.
- (c) *Geospatial selection algorithms*: A p2d search result indicates which points of interest are visible from the user's location, and which are hidden.
- (d) *Hardware prototyping*: To enable experimentation, custom device prototypes have been developed combining GPS, compass and tilt-sensors (see Figure 2.1).

However, the GPS system does not provide any geo-spatial location data in an indoor environment. Hence, in order to implement the said pointMe interaction in an indoor environment, new approach is required that addresses not only the development of the selection algorithm but also addresses geo-spatial annotation, user position mapping, and mobile interaction application. In this thesis, we attempt to provide such a framework that provides guidelines for the implementation of the above components in an indoor setting and supports novel interaction paradigms.

2.1.3 Bridge between Virtual and Real

Collaboration between physical and virtual world is not new. Fleck et. al [45] presented the first notable work for controlling the physical space monitoring by using a virtual user interface. The authors presented an integrated 3D world model, where the physical objects were collected through inter-camera tracking system and embedded as live texture in the 3D world. Presumably the interface minimized the overhead of the observing person who was serving the system to understand the events that happened in the real environment. One of the supporting studies related to object-based smart home control using a 3D virtual environment is described in [17]. As the prices of smart home automation devices are decreasing, the need to control those devices with an efficient user interface is increasing. L. Borodulkin et. al. [17] show that a 3D user interface can improve the intuitiveness and adaptivity of smart home control, as compared to an ordinary 2D user interface. The authors also show that with the 3D user interface a user requires less training and learning over 2D user interface. With a virtual reality approach, the user fully benefits from the advantages of the 3D view design: instead of abstract objects labelled as "room 1", "light left" or "heater 2", etc., a realistic view of the house with its structural components such as rooms, stairs, and windows is given to the user. The work presented by Zhang et. al. [153] show how those factors affect the design of user interfaces, specially for the smart environment. Those factors are very important and have a high impact on users' cognitive tasks inside a home.

University of Arkansas installed a prototype system called Virtual Razorback [133] in Second Life on "University of Arkansas Island"¹ which is a virtual hospital for modelling certain aspects of health care and a hospital's supply chain, from receiving goods to patient care. The intention of this project was to use a virtual world as a platform for modelling certain aspects of health care and a hospital's supply chain-from receiving goods to patient care. They built a virtual hospital called Razorback Hospital in the Second Life world. Just like a real-world hospital, the virtual hospital has virtual furniture and equipment, as well as toilets, sinks, showers, chairs, classrooms, patient rooms, beds, etc. It also has virtual patients, doctors and nurses with their correct uniforms. Moreover, the hospital has functioning units including virtual pill bottles, medicine supplies, rubber gloves, X-ray units, virtual RFID readers, blood pressure monitors, and CAT scan machines.

¹University of Arkansas SL project, <http://vw.ddns.uark.edu>; last accessed on Jan 2011

Second Life (SL) ¹ is a popular online 3D virtual world, and is developed by Linden Lab². Second Life provides a free client program called a viewer, allowing for interaction with one another through 3D avatars called resident [74] [74]. A subscribed user can enter into a virtual 3D sub-region and move around within the region by using mouse based navigation. Like other virtual environments, each Second Life resident has a limited visibility area, called an Area of Interest (AoI). The avatars automatically receive all the messages and events within an AoI. Linden Lab provides both a commercial and an open source version of the SL viewer. The open source viewer is called Snowglobe [125]. The viewer allows its users to create virtual objects through a 3D modelling tool. It also allows them to add more functionality to these virtual objects by incorporating rules based on its own scripting language, called Linden Scripting Language (LSL). One thing that makes LSL unique is its emphasis on States and Events. A door can be "open" or "closed" and a light can be "on" or "off". This state-based idea came from real world objects, since many real-world object behaviours can be modelled in the same way. We have developed a preliminary prototype, where we described the basic virtual real communication approach by using a Second Life framework [100].

2.1.4 Context-aware Smart Interactions

Context also can be defined as an applications ability to detect and react to environment variables [14]. Researchers [25] have defined three levels of interactivity for a context-aware application, they are, personalization, passive context-awareness, and active context-awareness. In a personalization based context-aware application, the user specifies his/her own settings for how the application would respond to a given situation [92]. Similarly, a passive context-awareness based application provides the users with sensory updates and allows them to change its behaviour. Lastly, an active context-awareness based application autonomously changes the behaviour in response to sensory updates [70]. An example of an active context-aware application is a mobile phone system that changes its time automatically as the phone enters a new time zone. The corresponding passive context-aware application is where the mobile phone system prompts the user with information about the time zone change and lets the user choose whether the time should be updated or not. In context-aware applications, the researchers have to choose whether information should be pushed towards the user or the user should be

¹Second Life, <http://secondlife.com>; last accessed on Nov 2011

²Linden Lab, <http://lindenlab.com>; last accessed on Nov 2011

left to pull the information on his own [27].

Tao et al [53] have proposed an ontology based approach to modelling context. They attempt to describe contexts semantically in a way which is independent of programming language, underlying operating system or middleware. Based on the location and posture status of the users, the proposed system can determine whether a person is 'Sleeping', 'Showering', and 'Watching TV'. For example, if a user is located in living room and the TV is ON then it is considered that the user is watching TV. Similar constructs can be used to conserve energy by keeping the lights ON at locations where there are user 'activity.' However, very few users have understand the proposed ontology and use them to construct their own context events. The system also lacks personalization options and requires advanced technical knowledge to use the system.

2.2 Types of smart mobile interaction

We define smart interactions as the interactions between a user and a smart multimedia object that has been augmented with computation services [43]. The generic term smart objects or Internet of Things is used within the context of this thesis to summarize things, people and places with which the user can interact with [71]. The IoT is thereby often augmented to store and provide information used for the interaction.

Mobile user interactions with places target location based mobile services in which one can interact with places. In such mobile interaction scenarios, the location of the user and the object is used to start or to control an application [70]. The mobile interaction with things like advertisement posters, public displays, objects within smart environments or other electronic devices is relatively new [128]. The built-in camera of the mobile phone can be used to scan the visual markers on advertisement posters that may represent URL to the product. RFID reader enabled handheld devices can touch RFID-tag attached to a printer in order to print files stored on the mobile phone, etc.

The following subsections present a classification of the interaction techniques touching, pointing, and scanning. The description of each interaction technique uses a key name, a compact description of the usage and principles of the interaction technique. This is followed by a discussion of the advantages and disadvantages of this particular interaction technique. We present the categories of smart mobile interactions in Table 2.1.

Table 2.1: Common mobile interaction techniques

No	Category	Reference
1	PointMe	[107][102][50]
2	TouchMe	[143] [38] [96] [104][97]
4	ScanMe (location based)	[1] [98]
5	Context-aware	[53][121]
6	RFID/NFC	[113] [40][69]
7	Visual code	[47] [81]

2.2.1 TouchMe Interaction

We define TouchMe as an interaction technique, where the user can select a real world things by touching it or use a handheld device to touch a real or virtual representation of things. TouchMe is a direct interaction technique. TouchMe interaction becomes more appealing when things are spatially close to the users. In order to use touchMe interaction, the user is responsible to ensure that s/he makes the accurate selection of the objects to be touched through careful maneuver of her/his handheld device.

TouchMe interaction technique is natural and familiar. The interaction conforms to one of the everyday physical natural interactions. Want et al. [143] are one of the first who have presented a prototype for the touchMe interaction technique, which incorporates Radio Frequency Identification (RFID) tags and an RFID reader connected to a handheld device. In order to implement the proposed touchMe interaction techniques, short range RFID and Near Field Communication (NFC) have been used. NFC chip can read information to and from a tag [123]. This functionality is also provided by conventional RFID chips in handheld devices [152]. NFC chip can also emulate an NFC tag. NFC supports bidirectional communication between two devices, however, it does not allow communication with more than one devices at a time. We compare RFID and NFC tags in Table 2.2.

One disadvantage when using this physical mobile interaction technique is that the user must be aware of the augmentation of the object and the provided services. However, sometimes it is not possible to physically touch an object. Enns and MacKenzie [38] are one of the first who have used a remote control equipped with a touchpad to control a remote display. With their solution the user is able to control a television screen by entering commands via the touchpad. Similarly, a handheld device with stylus and a touch screen can be used as an input device for the interaction with a Personal

Table 2.2: Comparison of RFID and NFC interaction schemes

	RFID	NFC
Pros	Unobtrusive tagging, simple, inexpensive, disposable, can be easily sensed. NFC devices can emulate tags	
Cons	No visual awareness, additional visual marker like the NFC logo is needed. RFID devices can not emulate tags	
Unit Cost	Each RFID and NFC tags used in the application have cost associated with it. The cost depends on the standard of the tag. It is relatively cheap.	
Data Transfer and Accuracy	Bi-directional read and write. 75 – 85% [145]	
Capacity	Multiple tags	One tag at a time
Range	0 – 5 cm	0 – 3 cm
Interference	Reader-to-tag interference, tag collision, tag retransmission, tag jamming	

Computer(PC) or a whiteboard [89]. In the RemoteCommander prototype application the handheld device acts as the mouse and the keyboard. The prototype maps predefined commands that are made on the handheld device and sends them as inputs to a PC. By incorporating similar adaptation techniques, application may provide a shared screen with which the users can interact by using their handheld devices.

2.2.2 PointMe Interaction

By means of the pointMe interaction technique, the user can select or control IoT by pointing at it with her/his handheld device. PointMe interaction technique is natural to the user because it conforms to one of our everyday physical interactions. Pointing can be realized by several technologies and interaction concepts which can be based on visual markers, image recognition, or infrared technology [103]. Fitzmaurice [44] is one of the first who has described the concept of using mobile devices for pointing based interactions with IoT. He has described a map at which the user can point to get information about a specific area in this interaction technology.

Tagging real world objects with visual markers and their interpretation has its origins in the research fields of computer vision and augmented reality [101]. Some of the popular visual markers are listed in Table 2.3. One important factor for the usage of marker based

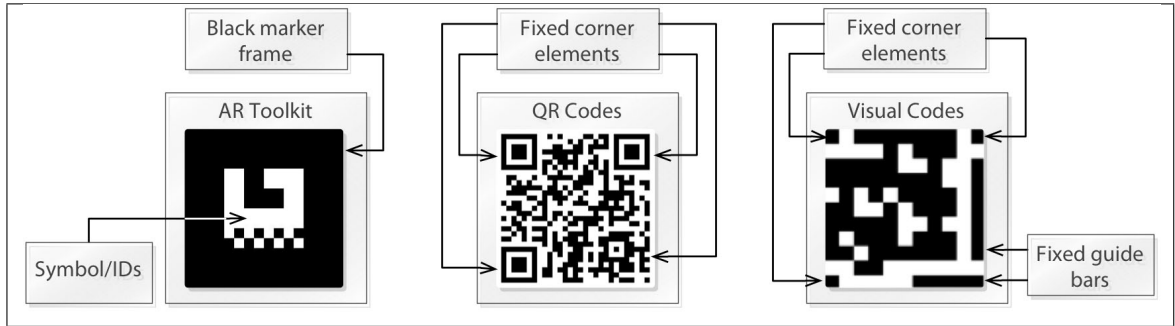


Table 2.3: Comparison of various visual code based mobile interaction approaches.

approaches is the minimal and maximal distance considerations between the handheld device and the visual markers. During the interactions, for a successful operation, the handheld device should be able to capture the visual marker in a sufficient quality. When using a small marker and a low-resolution camera, the handheld device has to have a distance of not more than just a few centimetres to be able to take a picture of the marker.

One disadvantage of those types of implementations is that IoT need to be tagged with often visually notable markers. Costanza et al [31] presented a concept which tries to solve this problem by proposing visually acceptable markers such as QR Codes [127].

A different approach also uses camera-equipped handheld devices for the implementation of the pointMe interaction techniques but employs an object recognition based approach. Object recognition not only places high demands on the processing power of the mobile handheld devices, but the application on it must also be aware of the features of the target IoT. Furthermore, there is no visual indication whether an IoT can be interactable. Fritz et al. [49] have presented a system for outdoor IoT recognition with camera equipped handheld devices whereby, the analysis of the focused object is done by a server. The advantage of the usage of visual markers or image recognition based approaches is that no power supply for the augmentation of the smart IoT is needed and it can be used with current handheld devices without additional hardware. Table 2.4 illustrates many of the pointMe interaction techniques and compares their features.

One disadvantage of the pointMe interaction technique is the dexterity demanded from the user. When using a marker based approach the user must correctly focus to detect the whole marker or, when using a laser pointer based approach, the user has to point directly to the corresponding sensor on the smart object.




Visual	Ref	Pros	Cons
	[117]	Makers are simple, inexpensive, disposable, and require no power supply.	Visual obtrusiveness of the marker, limited information storage capabilities.
	[46]	Smart objects do not need to be augmented, and require no power supply.	Great demands on image recognition capabilities and data model representing the smart object.
	[137]	Very intuitive interaction, Sensors can be small and unobtrusively embedded.	Smart object must be enhanced by a communication channel (e.g., RF, Bluetooth technology).

Table 2.4: Related work comparison in pointing interaction techniques.

2.2.3 ScanMe Interaction

The scanMe interaction technique is based on the proximity of handheld device and IoT which can be a real world object as well as a location in general. The mobile device scans the environment for nearby smart IoT. This action can be triggered by the user or the system that incorporates the sensors augmented in the smart environment. The result in both cases is a list of nearby IoT. The system automatically monitors the location of a user in relation to several nearby IoT. In this case, when a predefined distance between user and IoT is sensed, a corresponding service is started. For example, the system provides the user with detailed information of the IoT. The location aware interaction is intuitive because it corresponds to our everyday behaviour. For example, we approach a person when we intend to talk with her or we approach an object to see it in detail.

The advantage of this interaction technique is the possibility to discover nearby smart IoT and the services they offer. The user does not need to be aware of the augmentation of a real world IoT. Also, the IoT does not need to be visually changed to draw the attraction of the user. The disadvantage of this interaction technique on the other hand is that there is no direct link for the user between an item on the list of nearby smart

IoT and a concrete augmented object in the environment. This might be simple when thinking about an augmented microwave but might be complicated when a room has for instance several lamps and just one is augmented or controllable. Tuomisto et al. [136] used SoapBoxes which communicate with each other via radio communication to implement this interaction technique. In this approach, all nearby beacons that are attached to real world object are sensed.

In general, we currently see a huge interest regarding the provision of location services which provide information about location of the user or any other object. Typical approaches are based on cell identification, the Global Positioning System (GPS), distance to near field networks such a Bluetooth or WLAN, RFID, infrared or ultrasound [5] [77][109] [142]. A very comprehensive overview of indoor and outdoor positioning techniques that can be used for user and system triggered scanMe interaction technique can be found in [5].

2.2.4 Location Aware Interaction

The proposed smart mobile interaction framework can adopt many different location calculation methods based on the application. Many existing location calculation schemes can readily be adopted into the framework. Location calculation is central to context-aware and scanMe type application development. For example, active badges [142, 55] or tags [103] are worn about the person and communicate a unique identifier approximately every 10 seconds using an infra-red transmission. Infra-red does not penetrate solid surfaces such as walls. Electromagnetic based tracking ¹ are capable of resolving the location and orientation of a tracked object with very high spatial and temporal resolution. Optoelectronic tracking [6] optical sensors built into a head mounted unit. These sensors are sensitive to infra-red beacons built into a room's ceiling. With knowledge of the positions of these beacons and which beacons are currently in view, the location and orientation of the user's head can be determined.

As an example, we present a Radio frequency (RF) based system [7] for locating and tracking users inside buildings. The system is dependent on specialized hardware. It automatically create a radio map of the building by using propagation modelling and record information about the radio signal as a function of user's location. However, the radio signal strength is affected by many factors including physical walls of the building.

¹Real-World 3D Tracking Technology, "http://www.ascension-tech.com/products/trakstar-drivebay/"

The system further requires a physical layout of the building floor including coordinates for each room. In each operation, users locations must be entered into the system by manually clicking on a map of a floor. Afterwards, the system guess the user's location to be the same as the location of the base station that records the strongest signal (70 base stations used).

Active Badge [142] system is an IR based location identification system. It brings significant contribution to the field of location aware systems. In this system, a badge is worn by a person that emits a unique IR signal every 10 seconds. Next, sensors placed at known positions within a building pick up the unique identifiers and relay these to the location manager. While this system provide accurate location information, it suffers from several drawbacks (a) it scales poorly due to limited range of IR (b) it performs poorly in the presence of direct sunlight, (c) it has installation and maintenance costs. In another such system [6], IR transmitters are attached to the ceiling at known positions of a building. User are equipped with an optical sensor on a head mounted unit that senses the IR beacons. The periodic sensing of these beacons enables the system to determine a user's location at a given time.

2.2.5 Summary

The aim of this subsection is to provide a compact overview of the previously discussed physical mobile interaction techniques. Every interaction technique in the following Table 2.6 is described by its name, the technology used for the interaction between the mobile device and the smart object, the real world aspects of the interaction, the advantages and disadvantages and finally typical application areas.

We now present you a comparison between various annotation schemes in Table 2.5. The proposed geometric annotation scheme may not provide the optimal replacements for the convention RFID, NFC or Visual Code based applications, for example in Enterprise or business to track inventory; however, it brings a new approach to smart pervasive computing applications. In many scenarios, such as in virtual things annotation, multimedia object annotation, etc the proposed geometric annotation scheme presents many advantages. Unlike visual code based annotation, the geometric based annotation does not hide or add image/sticker on top of the IoT. The geometric based annotation map of IoT can further be changed, be transformed, or can be reused in another application or to develop a new type of interactions very easily. The geometric annotation map can easily be changed by using a software interface. Geometric annotations are not hardware

Table 2.5: Comparison between different annotation schemes

	RFID	Visual Code	Geometric	NFC
Side by Side Annotation	No	Yes	Yes	No
Maximum Annotation Size	Limited	Limited	As big as Object	Limited
Change of Annotation	Physical	Physical	Virtual	Physical
Hides Annotated Things	No	Yes	No	No
Whole Object as an Annotation	No	No	Yes	No
Annotation Type	Physical	Virtual and Physical	Virtual and Physical	Physical
Reuse Annotations	No	Yes	Yes	No
Interaction Distance	1-5cm	1m	Flexible	1-3cm
Interaction Types	ScanMe, TouchMe	PointMe	PointMe, ScanMe, TouchMe, Context-aware	ScanMe, TouchMe, PointMe
Interaction Devices	Mobile or Static Reader	Camera	Human, Mobile, Camera	Mobile or Static Reader
Change of Interactions	Not Possible	Not Possible	Possible	Not Possible
Annotation as Context	No	No	Yes	No
Hardware Dependent	Yes	Yes	No	Yes
Nearby Annotation Search	Not Possible	Not Possible	Possible	Not Possible
Passive Interaction	No	No	Yes	No
Read Capacity	Multiple	Single	Multiple	Single
Programmable Interactions	No	No	Yes	No

Table 2.6: Overview of smart mobile interaction methodologies

	Pointing	Touching	Scanning
Device Smart Object Interaction	Visual marker, light beam, IrDA	RFID, NFC, Bluetooth, WiFi	Bluetooth, WLAN, GPS
Real World Aspect	Distance object - Mobile device: $10cm - 10m$ line of sight	Object - Mobile device: $0 - 10cm$ line of sight	Distance object - mobile device: $10m$ (Bluetooth), $45m$ indoors, $90m+$ outdoors (GPS)
Pros	Natural way of interaction. Distance to the smart object.	Unambiguous selection of an object.	Possible to discover nearby objects and services
Cons	Dexterity is needed.	Proximity of user and object.	Problematic localisation
Application Area	Getting information about a remote object.	Picking up a hyper-link represented by an advertisement	Searching for nearby smart object to interact with them

dependent, for example, an IoT annotated with respect to an IR camera can easily be used in a 3D camera based interaction development.

Unlike RFID, visual code based solutions, in geometric annotation it is possible to scan neighbouring IoT annotation data and develop interactions that take their proximity into consideration. For example, a set of geometric annotations can be used to locate a user's position with respect to those annotated IoT and to define context-aware interactions. Passive automated interactions can also be defined when a user is within a set of geometric annotations, or a user is pointing near a set of geometric annotations etc. When we attach RFID or visual tags to IoT then we have to bring a camera or a reader near those tags to begin an interaction. On the other hand, by using geometric based annotations, an user can begin interactions with a very big object, for example a large dinosaur at a Museum, by pointing to any part of that object.

Chapter 3

Smart Mobile Interaction Framework

We propose a Smart Mobile Interaction Framework that supports the annotation of things within a hardware independent geometric entity in order to support the rapid development of user interaction models for IoT systems. The framework supports the development of different interaction models such as touchMe, pointMe, scanMe, context-aware object selection, etc. Since the proposed annotation approach is not dependent on a particular hardware setup, it is possible to integrate new interaction models without changing the base design and development of an IoT system. The philosophy behind the geometric annotation based development of user interaction models is explained in Figure 3.1.

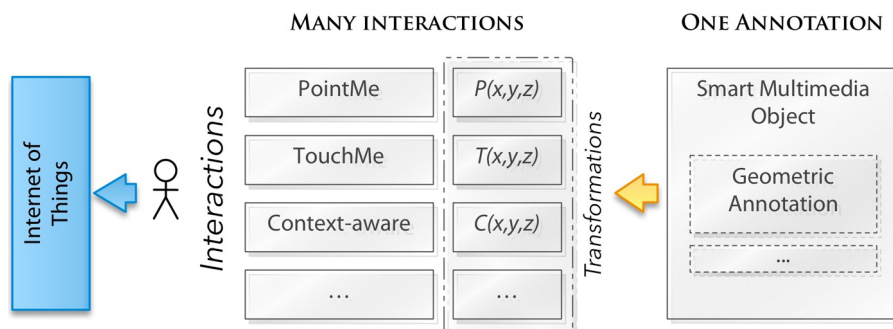


Figure 3.1: Geometric annotation based interaction model development in an IoT system.

In the proposed framework, IoT are termed as Smart Multimedia Objects (SMOs). SMOs are geometric entities that encapsulate the annotation data of things. SMOs can

encapsulate data of virtual and real things. In order to annotate virtual and real SMOs separate annotation editors can be used. When a new interaction model is introduced in the system the SMOs are transformed from local annotation space into smart interaction space. To clarify, each interaction model can use its own set of transformations to bring the SMOs into its own interaction space. We term the processing unit of the system as the Interaction Controller. The Interaction Controller incorporates the respective sensory data of the interaction models to match the data of a certain interaction space.

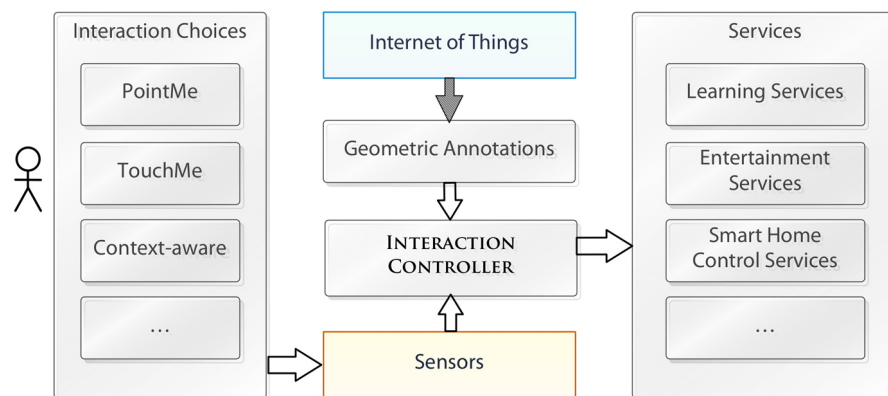


Figure 3.2: Overview of the framework.

Figure 3.2 depicts an overview of the proposed framework. The framework supports different interaction models such as pointMe, touchMe, scanMe, etc. In the proposed framework the things – virtual and real objects are tagged with geo-spatial annotation that can be read by the smart interaction controller. Different applications and service end points are supported such as smart home entertainment, learning, and haptic feedback services.

The proposed framework supports the implementation of user, or mobile device mediated interaction developments. For example, inside a smart room, we geo-spatially annotate smart devices with various services. Based on the position of the user with respect to the annotated smart devices, various services can be automatically activated. In order to foster virtual to real communication, the proposed framework supports annotation based interaction with virtual objects. In the physical home environment, different devices and sensors are connected in order to ensure a safe and automated home. Any event that occurs in the physical space is then synchronized with the virtual environment. More importantly, the virtual interface provides the option to control the physical smart devices. Overall, the proposed framework supports various interaction models that can

incorporate haptic, visual (image, video), and auditory (music, ambient sound) channels in their implementations. The interaction channels are shown in Figure 3.3.

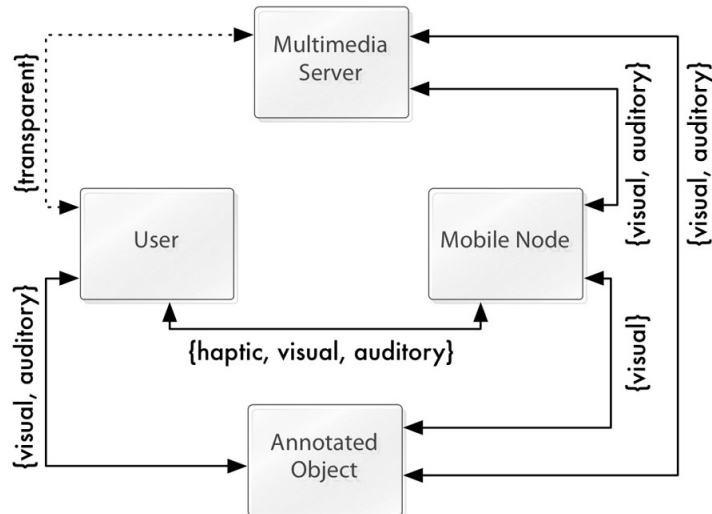


Figure 3.3: Interaction channels of the system.

3.1 Overall Architecture

Within a physical mobile interaction, the mobile device and the mobile user act as mediators between the physical and the digital worlds. The server represents the digital world, which offers information and services related to the smart object. The latter represents the physical world and provides entry points into the digital world. Generally, the smart object provides a link to the corresponding services that are made available by a corresponding server. The components of the server side are referred to as the Smart Environment Server as seen in Figure 3.4.

3.1.1 Smart Environment Server

The Smart Environment Server side consists of several components. The main task of the server is to provide contents or services when an interaction with IoT takes place, as specified in the framework implementation. The Service supports several types of objects such as application/message, application/uri, application/html, application/image, application/text, application/video etc. However, the implementation of the interactions

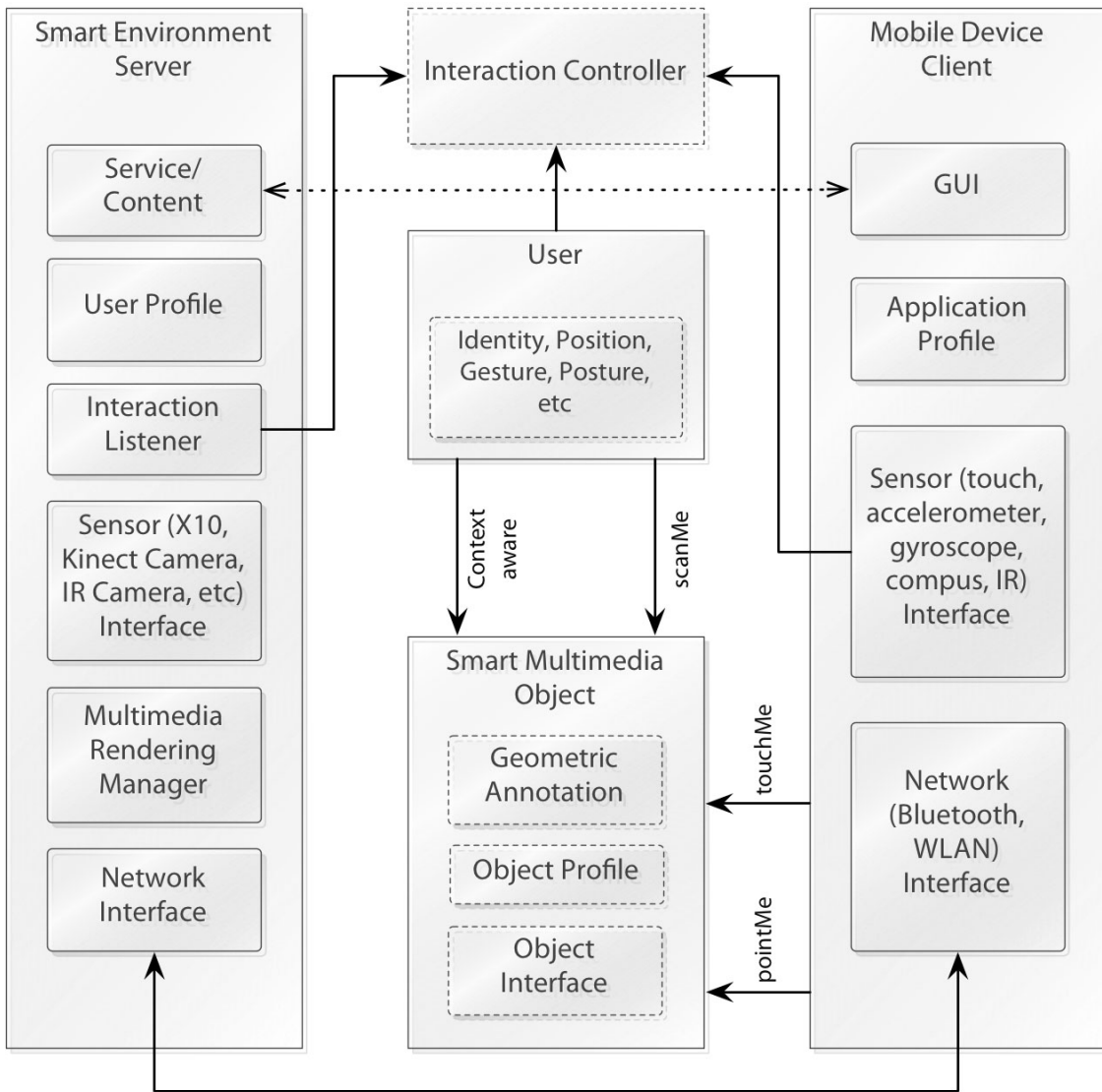


Figure 3.4: Smart Mobile Interaction framework.

are independent on the Service or Content definitions. Hence, new service endpoints and application contents can easily be incorporated into the framework.

The User Profile class processes the data that are obtained from the User Client. The profile XML definitions may contain a user or mobile identity, or position, user gestures, mobile events, etc. The profile class can be extended to store new data types as needed by the application. The Interaction Listener component contains the runtime stub and interface headers for the Interaction Controller. Through the stubs the Interaction Controller transmits messages and data to the Server.

Similarly, the Sensor class stores data that are obtained from sensory objects such as X10, IR Camera, Accelerometer, etc. The Sensor class extends the interface of the Interaction Controller and implement a set of initialization, rendering, input, and output modules. Any sensory element that adopts the interface presented by the Interaction Controller is supported by the server. For example, the server side supports sensors, such as a Kinect Camera, IR Camera, X10 Controller, Light sensor interface etc. Also, any sensory element can instantiate and access User Client and Mobile Device Client events and states in order to dynamically process user and device data. The Sensor interfaces of the Interaction Controller can be implemented to produce custom algorithms by using the sensory data that are available from other sensors on the Server.

3.1.2 Interaction Controller

The Interaction Controller contains stub interfaces and allows the client side to implement them in order to ensure interoperability. By implementing the interface procedures, it is possible to design and develop pointMe, scanMe, and context-aware interactions. We have presented guidelines and prototypes to demonstrate the use of each of those interactions. The controller may interact with the other components of the framework by using the Simple Object Access Protocol (SOAP). Other communication protocols can be incorporated as well, if needed. The Interaction Controller receives and/or transmits messages to/from the Mobile Device Client and the User Client through the SOAP communication channel and captures the events that are generated from the Mobile Device Client message transmission module. The Interaction Controller works as a core service and takes action according to user inputs. In Figure 3.5, the interaction packages are organized according to their respective implementation targets. For example, the scanMe interaction package is a collection of Gesture API, Kinect Camera API, and User Selection API. However, in the framework definition, the physical rendering of

a multimedia device is always contextual and requires the addition of context-aware interaction packages.

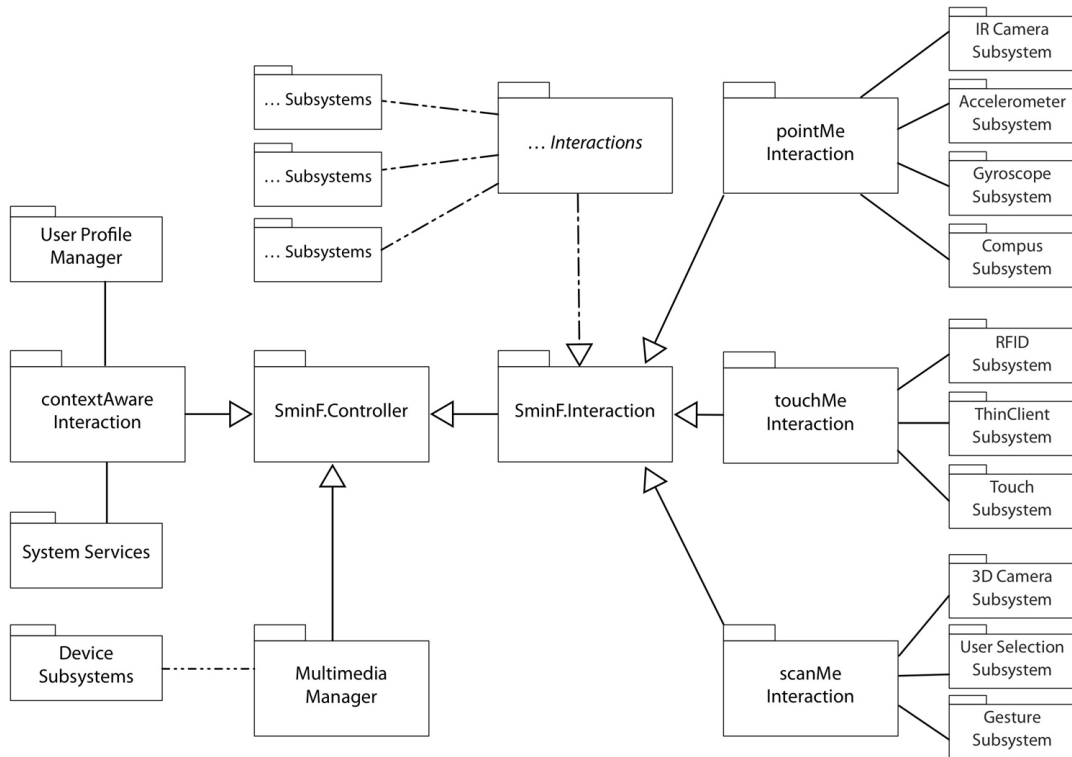


Figure 3.5: Interaction packages of the system.

3.1.3 Mobile Device Client

The purpose of the Mobile Device Client in the framework is multi-fold. A mobile device can be used to provide Service interaction points or contents through a mobile app with a graphical user interface. In addition, the mobile clients act as a collection of sensors that are used to track the user or the user inputs through accelerometer sensor, touch interface, etc. For example, by using the mobile sensor interface, IR emitter positions that are attached to a mobile device can be tracked in order to locate a mobile user. Additionally, data from the accelerometer, the compass, and the gyroscope of the mobile device are aggregated into the Server through the sensor interface for further processing. The IR position of the mobile device, as well as accelerometer and gyroscope sensory data are further fused to obtain the pointMe interaction of the mobile handheld device. An implementation of such a process is discussed in Section 3.5.1.

3.1.4 User Client

The User Client refers to a set of services that capture and process a user’s face, 3D position, and 3D skeleton data in order to determine position, posture, gait, gesture, and other profile data of the user. The Interaction Controller usually mediates the transfer, processing, and event based handling of the user data to produce meaningful interaction scenarios with a target application.

Depending on the application profile, the Interaction Controller prepares the sensory services to capture user client data. For example, in a 2D application or in a touch based application, the user’s position, and facial and skeletal data are not required. At the application package level, the user client is represented by a set of classes that stores user profile data such as age, user preference, accessibility, gaming score etc.

3.1.5 Smart Multimedia Objects

An important part of IoT applications is the augmentation of multimedia things in order to enable natural interaction possibilities with them. The augmentation of things is usually performed by using visual or sensory tags. In this thesis, we have introduced a logical geometric annotation based approach to accomplish this task. In the proposed annotation approach, we support the augmentation of both real and virtual multimedia things. We term such multimedia things as smart multimedia objects. A colour-coded sticker or overlay can be used to distinguish a smart multimedia object from the other things inside an interaction space. By using the colour coded sticker or overlay the user is able to determine the type of interaction techniques that are supported by the objects and the required steps needed to interact with them. Augmentation and the annotation of things are described in Section 3.4.

Examples of physical smart multimedia objects include but are not limited to indoor digital devices, indoor furniture, indoor pictures or maps, and indoor places. Furthermore, examples of virtual smart multimedia objects include 3D virtual avatars, 3D virtual furniture, 3D virtual places, 2D video content (specific to timelines), and eBook content (paragraphs of pages). In Section 3.4, we present examples of geo-spatial annotation schemes for virtual and real multimedia things.

3.2 Multimedia Renderer

The Multimedia Renderer is comprised of a set of services that control the playback of various multimedia contents for predefined multimedia devices. A new multimedia device can implement the multimedia interface module, extend a multimedia rendering type that it supports, and subscribe to the Multimedia Renderer. As all of the multimedia devices use a parent multimedia interface module, a multimedia device added to the Multimedia Renderer has a common device profile that the Interaction Controller can use. We have shown the use of a common multimedia interface in Figure 3.6. This scheme significantly reduces the compatibility issues of the application. As an example of multimedia rendering devices, in the following section we present the home entertainment controller, the haptic renderer, and the physical device controller.

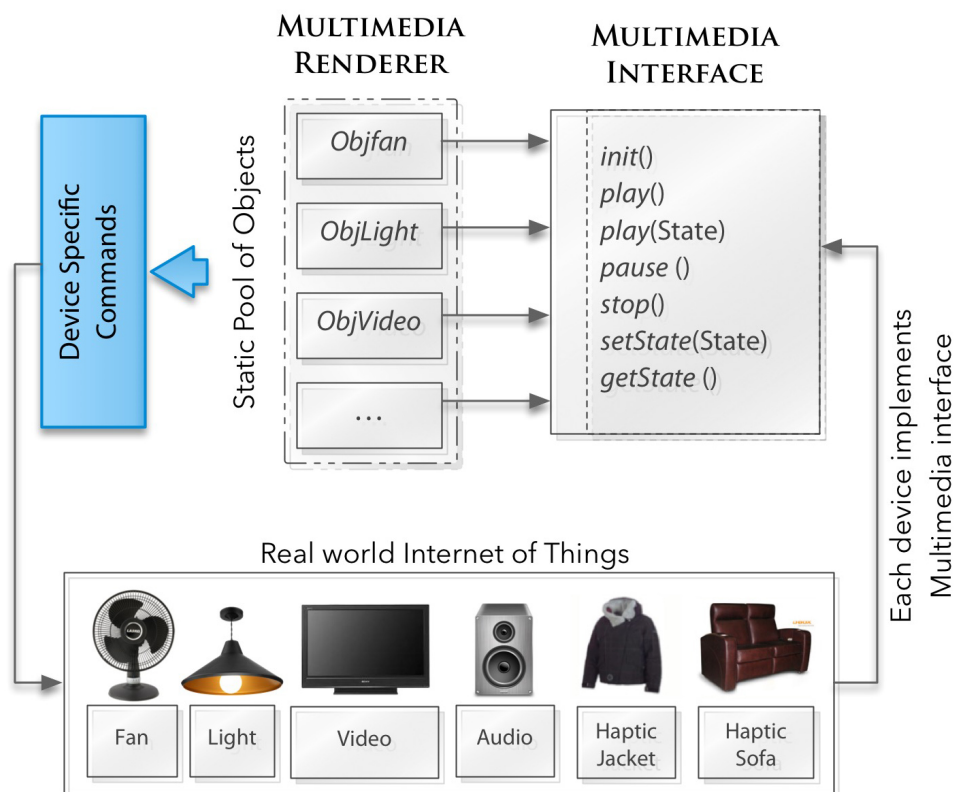


Figure 3.6: Heterogeneous device types are added to the Multimedia Renderer as they each implement a specific Multimedia Interface.

3.2.1 Physical Device Renderer

In order to interact with all smart home devices such as the smart door, smart light, and smart audio device we extend the multimedia interface module and add the object to the Multimedia Renderer. At the physical level, we incorporate X10, Wi-Fi, and Bluetooth interfaces to communicate messages to the physical devices. The Interaction Controller directly references the objects included in the Multimedia Renderer to transmit command according to user interactions. The smart home service receives the message and sends the associated command (written in the play module) by using X10 or Wi-Fi modules. When the physical object tries to turn off the device or resets its states, the Multimedia Renderer receives the updated states and synchronizes those states with the Interaction Controller. For example, in order to render synchronized animation in the virtual environment as per the changes in the state of the real devices, a state-based event communication mechanism can be incorporated. When a user touches a 3D object in the virtual environment, the system receives the event message, which is then pushed into an event queue. Furthermore, the system processes the event queue sequentially and transmits the message respectively to the physical devices. In the next step, the system sends the actual X10, Bluetooth, or Wi-Fi commands to the smart device in order to process "turn light ON", "open door", "close door", "increase light intensity", "turn fan ON", or other interaction requests.

3.2.2 Home Audio Video Renderer

The Home Audio Video Renderer is extended from the Physical Device Renderer module. This module can adapt to the ubiquitous home multimedia entertainment scenario and provide synchronized media playback options. For example, when a user/teacher reads annotated eBook content by using his/her mobile device (e.g. iPad, Kindle), the surrounding listeners/learners may enjoy the related learning contents through a sound system, HDMI based TV, haptic devices etc. While reading the book, an active user (teacher) can touch annotated parts of the content and consecutively a list of images may be queued to be displayed on a multimedia screen. At the same time related audio content can be played on a sound system. The image list and audio content are previously geo-spatially annotated in the eBook.

3.2.3 Haptic Renderer

In order to send /receive haptic feedback, in the framework design, haptic description block is sent to the Multimedia-Haptic Renderer module, where the haptic signal generator component determines device specific haptic signals. As the target haptic device can be heterogeneous, we have defined specific type of signals for each haptic device. The Haptic Renderer incorporates Bluetooth and Wi-Fi based communication methods to exchange commands with haptic devices. In our work, we have shown the use of a Bluetooth enabled haptic jacket, a D-Box haptic sofa, and a Bluetooth enabled haptic arm-band [100][99][4] in order to implement various haptic interactions. We have selected three basic emotions, love, joy, and fear based on the haptic devices available for our example prototype systems. We have annotated real life scenarios such as riding a bike, driving a car, travelling by an airplane, travelling by a boat, sea storm tide effects, tickles, a hug, a poke, a touch, etc.

- Touch: In order to mimic touch haptic interaction we have incorporated funnelling illusion based actuator pattern generation. The patterned vibrations of the set of actuators on the human skin produce a sensation of touch.
- Poke: A haptic poke incorporates similar patterned vibrations as the touch. However, the funnelling parameters for the motors are reduced and the vibration strength is increased in this type of haptic simulation.
- Tickle: Through the haptic jacket interface, we empirically have positioned the actuator motors around the belly, armpit and neck areas and leveraged a set of patterned touch sensations in order to generate tickle haptic feedback.
- Holding Hands: By using the haptic armband we have achieved the holding hand haptic feedback. In this type of haptic rendering, the haptic touch is focused on the armband area.
- Sea Storm, Travelling in a Boat/Airplane/Car, Bike Riding: We have customized the vibration patterns for the actuators of the haptic sofa to generate these feedbacks. The sea storm haptic feedback is a special type of boat riding haptic feedback. In order to generate a boat riding haptic sensation, we send customized vibrations parameters to the D-Box interface to modulate the actuators and generate a wave sensation. In order to generate a car travelling haptic feedback, we have

modified the boat travelling feedback and have enhanced it with frequent vibrations. In order to simulate a bike-riding experience, we customize the parameters of the haptic sofa to reduce the motor vibrations and increase the force of the actuators in a timely manner.

- Fear, Joy, Love: We again capitalize the haptic touch and haptic poke feedbacks in order to generate these emotional responses. In the case of fear haptic rendering, we enable the periodic patterned actuator touch sensation along the backbone. The joy effect is produced by incorporating a series of mild-poke and mild touch sensation around the stomach area, which simulates the butterfly effect. Whereas the love effect is produced by enhancing the haptic touch around the left-chest area.

3.3 User’s Mobile Position Determination

In this section, we present user 3D position determination approaches that are incorporated in the framework. In Section 3.3.1 we describe the IR Camera based user tracking and position estimation approach. Further, in Section 3.3.2, we describe a 3D camera based user tracking solution. In order to implement the scanMe and context-aware interactions we have incorporated a Kinect camera based user tracking solution in our prototypes. However, because of higher accuracy and better distance coverage, in the realization of pointMe interaction scenarios, an IR camera based solution has been incorporated.

3.3.1 IR Camera Based Approach

A single Infra-red (IR) camera can be placed inside an indoor environment in order to track a user’s mobile position in a particular scene. The IR motion camera consists of an IR motion receiver and an IR capture interface. The IR motion receiver is responsible for capturing the IR signals emitted from IR emitters. IR emitter node is a small LED that emits an IR signal that is then captured by an IR motion tracker camera. The IR emitter is connected to a small battery unit to receive power for the LED. As depicted in Figure 3.7, the IR capture interface component maps the captured IR signals into motion points. The proposed system incorporates multiple motion tracker cameras in order to calculate 2D and 3D motion points. A mobile phone is one of the target devices, where IR emitters can be attached; other choices include, pens, hand gloves, etc.

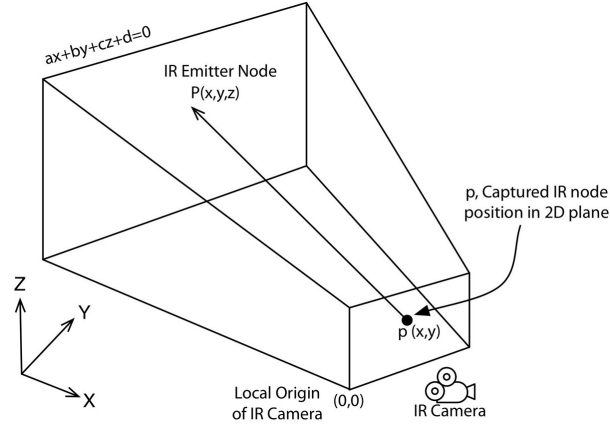


Figure 3.7: The 2D IR position capturing system of the IR camera. Here the IR emitter node $P(x, y, z)$ is activated in the virtual $ax + by + cz + d = 0$ plane facing the camera.

The IR capture interface delivers a clear and noise-free image of the IR emitter node even when the physical objects and the environment are complex. In our previous work [98] we have deduced that capturing the motion points of an IR emitter node in a 2D space are fairly accurate and the solution is robust enough to be used in other similar applications. For the proposed physical mobile interaction mechanism, we need to estimate the 3D position of the mobile devices and hence, we extend our previous work in order to calculate 3D positions of motion points in an indoor environment by using two IR cameras. A smart environment usually embeds many sensors and devices and the placement of multiple IR cameras in a room can therefore be considered as practical, especially since the IR cameras perform accurately in complex environment. The IR cameras can be purchased with minimal cost, at less than that of traditional cameras.

Camera Arrangements

Let us consider a logical point p that is visible in the space and is part of two images that are obtained from two IR cameras. The depth information of p can be calculated by corresponding the pixels in the left and right camera images. For simplicity reasons, in our approach the cameras are positioned perpendicular to each other and projecting the same 3D scene. Their location and orientation information is registered into the system beforehand. The following steps are followed to realize such an arrangement:

- A reference point λ in the physical environment is marked with an IR emitter that is viewable from both IR cameras. The reference point should be inside the region

from where a high volume of user interaction is expected. This point is registered to the system.

- We then position the first IR camera in the physical environment. To do so, we make sure that the projected 2D position of the reference λ is $(\alpha_{IR1}^\lambda, \beta_{IR1}^\lambda)$. We position the camera and maneuver its location until the said projected IR location of the reference point is achieved. As we change the position of the camera, we read the current projected coordinate of the reference node $(\acute{\alpha}_{IR1}^\lambda, \acute{\beta}_{IR1}^\lambda)$ from the computer screen. Furthermore, the system provides sound based feedback as we move the camera and confirms when the parameters match (i.e. $(\acute{\alpha}_{IR1}^\lambda, \acute{\beta}_{IR1}^\lambda)$ becomes $(\alpha_{IR1}^\lambda, \beta_{IR1}^\lambda)$). The whole process usually requires 5 ~ 10 minutes to complete.
- As with the first IR camera, the second IR camera also undergoes the aforementioned process. However, in this case for the reference point λ the target projected 2D position is $(\gamma_{IR2}^\lambda, \beta_{IR1}^\lambda)$ to make sure that the vertical location parameter for both the cameras are equal.
- Afterwards, we calculate the 3D position of the reference IR node λ . If we calculate with respect to the first IR camera then the resultant 3D location becomes $(x, y, z) = (\alpha_{IR1}^\lambda, \beta_{IR1}^\lambda, \gamma_{IR2}^\lambda)$, where $W = (0, 0, 0)$ is the world reference point in the environment.

Using this simplified scheme and employing a computer vision correspondence technique, meaningful depth information for a given point in the scene is produced from the two IR images [54]. An example scenario of the arrangement is depicted in Figure 3.8. In this arrangement, the IR emitter node that is attached to the mobile device is the point p . The point p is tracked by the IR cameras placed in an indoor environment. The IR camera positions and offsets from a known reference point (world coordinate) in the environment are known (or approximated) in advance. Based on the location and orientation information of the user, only two cameras of a scene are needed for a 3D location information estimation. The parameters used to select the two cameras are 1) The clear visibility of the IR emitter node from the camera for which the 2D orthogonal position scaling process results in suitable coordinate information for further processing and 2) The distance of the camera from the IR emitter node position. Hence, in our system the smaller the distance between the mobile user and the IR camera, the greater

the probability that the nearest IR camera is selected for the geometrical information fusion process.

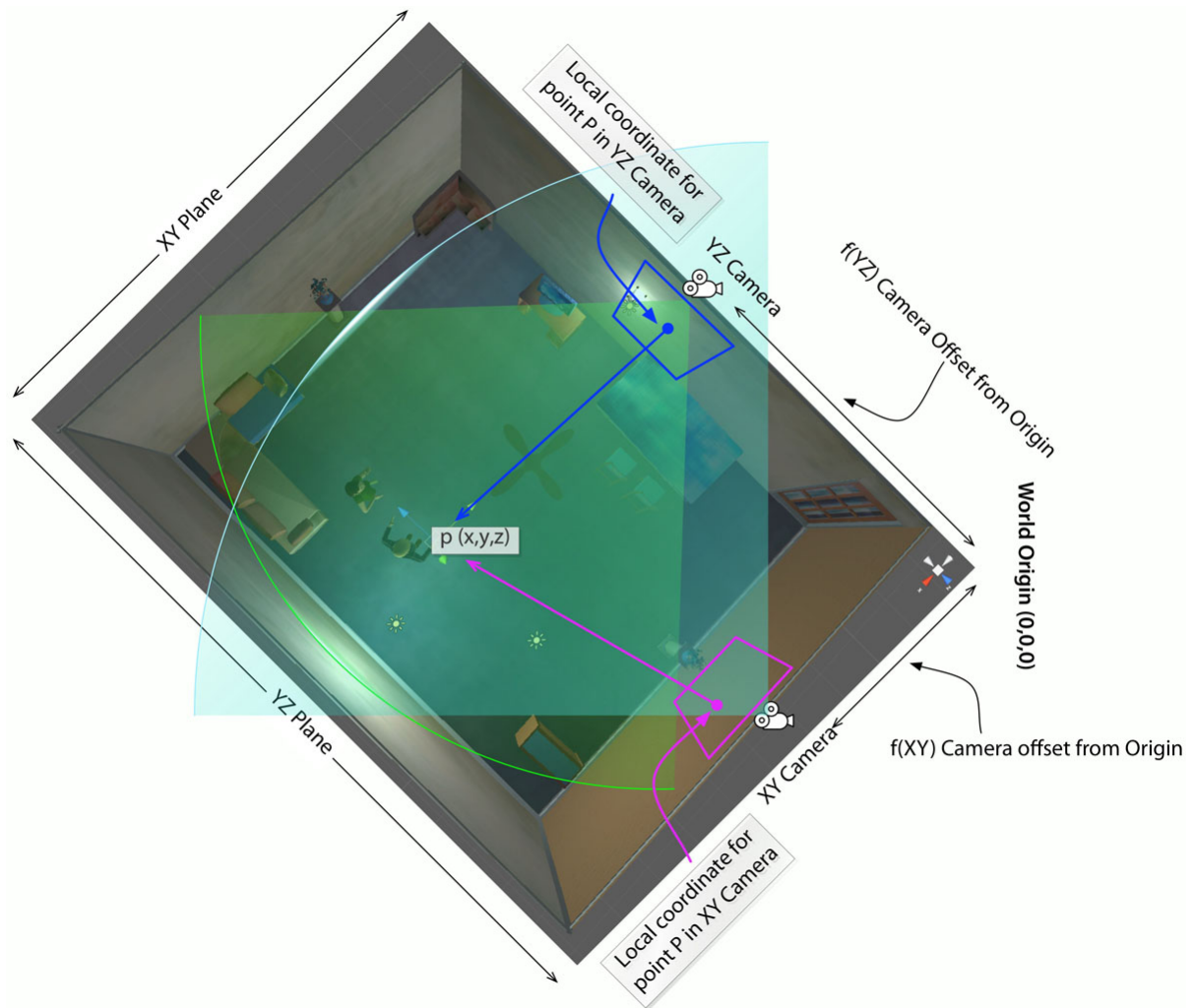


Figure 3.8: IR Camera positions shown in the indoor environment that are used to calculate the 3D location and orientation information of the user using two 2D IR positions.

Based on the IR camera arrangement explained above, we highlight the following points to improve the IR image capturing process and subsequently fuse the observations of individual IR cameras:

- The geometric conversion always provides the coordinate information of the IR emitter node with respect to a reference point inside the indoor location (world coordinate, $W(x, y, z)$).

- The entire object position mapping also happens with respect to the world coordinate system.
- Keeping one single coordinate system for detection and analysis reduces the complexity of the overall system. The reference or world point could also be calibrated in order to simplify the annotation/ pointing process. For example, it is advantageous to consider a closed corner points of the densely located target physical object, in an indoor environment, which will in turn reduce the transformation error of the pointing and mapping process.
- For simplicity reasons, object mapping are always considered to be rectangular prisms (voxels). This assumption quickens the annotation and registration process of the objects and simplifies the realization of the object selection algorithms.

IR Camera Image Fusion

The fusion process begins by recording sample IR emitter points at fixed interval and calculating the fundamental matrix, which updates when a set of successive IR node points are found for any new organization in the scene. Afterwards, for all IR emitter points that are found in the first IR camera image, we calculate the corresponding set of points with respect to the second IR camera image. All of these IR emitter points, along with their camera information, are further sent to the aggregation module for real-time processing. For each sampling, these point sets are analyzed to find the rectification homography matrices for the first and then second images. By using the homography matrices, the distance of the corresponding points are calculated in order to obtain the rectified points for each image. Therefore, with the rectified points, the distance is inversely proportional to the difference between the two shared coordinates of the images. The steps to calculate the 3D IR position can be illustrated as follows:

- For each camera, transform the local points into world points with respect to the reference point (see Figure 3.9). The visible IR emitter node is part of the images produced by the two IR cameras. Their correspondence is achieved by choosing the cameras from which the IR node is clearly visible.
- Afterwards, using the fundamental matrix, the correspondence between the two images are triangulated into a 3D point. This 3D point and the actual euclidean 3D point [42] in space will differ by a projective transformation. The rectification

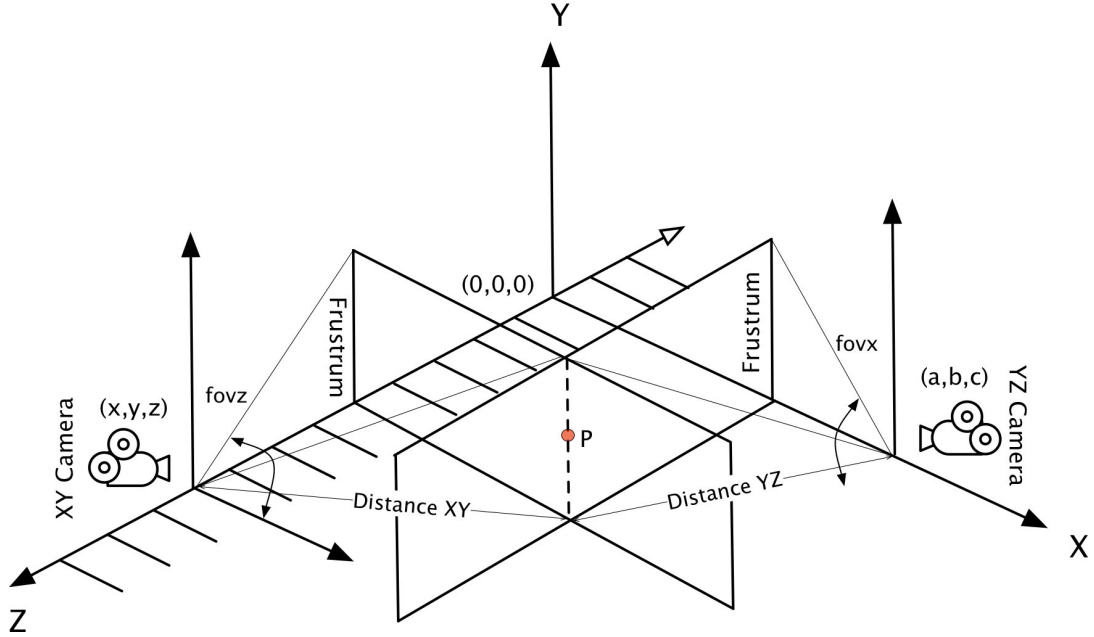


Figure 3.9: Geometric orientation, IR camera frustum, FOV angle and distance based relation of the IR emitter point P, with respect to the two IR cameras.

homography matrix [39] counteracts that projective transformation [86] in order to provide triangulation options.

- By using the fundamental matrix and the camera matrices from the two cameras with the direct linear transformation algorithm, we calculate the 3D position.

A sample pseudo code of the calculation is depicted in the following algorithm. Here, we have demonstrated the process used to calculate the fundamental matrix and to reconstruct 3D points from the two IR images. If any two 2D vertices, belonging to different views, have the same coordinate value for the shared coordinate axis, then the third view is searched for the 2D vertex, which has the same values as the remaining two coordinates from the original two 2D vertices under consideration. If the search is successful then a 3D p-vertex containing the common (x, y, z) coordinates from three 2D vertices is found. This procedure is carried out for all the 2D vertices. In the following, the Calibration algorithm uses the Zsense algorithm in order to calculate the 3D user's mobile position.

The time complexity of the fusion algorithm is $O(n \log n)$ and its running time is dominated by the factor of the number of updated points in a sample frame. On average, in our application the number of updates for a 200 Hz sampling rate frame is less than

Algorithm 1 Update Structure List

Require: Input *List* to be non empty. {The algorithm creates the point sets from the sensors, calculates the fundamental matrix and calibrates the updated data sets for processing in the ZSENSE algorithm based on the List structure.}

Ensure: Calculated fundamental matrix $M_F \neq null$ { N is the number of elements for which updated values are requested in this sampling. P_1 and P_2 are two array lists designed to hold normalized N number of *key* elements.}

- 1: **while** ($key = List.nextItem(void) \neq null$) **do**
 - 2: Append P_1 with $Normalize_{IR^1}(key, N)$
 - 3: Append P_2 with $Normalize_{IR^2}(key, N)$
 - 4: **end while**
 - 5: Create pointer handler IR_{p^1} for P_1 . Initialize matrix M_{IR^1} for IR_{p^1}
 - 6: Create pointer handler IR_{p^2} for P_2 . Initialize matrix M_{IR^2} for IR_{p^2}
 - 7: Create fundamental matrix, M_F from M_{IR^1}, M_{IR^2} . Store the state bits in M_{State} .
 - 8: **for** $i = 0$ to $length(M_{State})$ by 1 **do**
 - 9: $P_{v^1}[i] \leftarrow P_1[i]; P_{v^2}[i] \leftarrow P_2[i]$ {Store the matching points.}
 - 10: **end for**
 - 11: Create pointer handler $h_{leftView}$ and $h_{rightView}$ for P_{v^1} and P_{v^2} respectively
 - 12: Load identity matrix M_{LPMat} and M_{RPMat} from $h_{leftView}$ and $h_{rightView}$ respectively
 - 13: Call method $StereoRectifyConstruct(M_{LPMat}, M_{RPMat}, M_F, AR)$ {Here AR is the aspect ratio for both of the IR cameras}
 - 14: Return $ZSense(P_{v^1}, P_{v^2})$
-

Algorithm 2 ZSense (P_{v1}, P_{v2})

Require: Two structure lists of points P_{v1} and P_{v2} from the respective IR cameras {The algorithm inputs two sensory data packaged in a structure list and applies a set of equations and conversions to output the Z distance of the IR emitter nodes}

Ensure: Calculated X offset $\neq 0$

- 1: **for** All points P_{LIR} in $P_{v1}.next()$ AND P_{RIR} in $P_{v2}.next()$ **do**
 - 2: Create and initialize matrix $M_{L3,1}$ and $M_{R3,1}$
 - 3: Set $M_{L3,1} = \begin{bmatrix} P_{LIR}.getPosX() \\ P_{LIR}.getPosY() \\ 1 \end{bmatrix}$ and $M_{R3,1} = \begin{bmatrix} P_{RIR}.getPosX() \\ P_{RIR}.getPosY() \\ 1 \end{bmatrix}$
 - 4: Multiply $M_{L3,1}$ with M_{LPMat} to obtain rectified matrix M_{LPane}
 - 5: Multiply $M_{R3,1}$ with M_{RPMat} to obtain rectified matrix M_{RPane}
 - 6: Set ∂x to $M_{LPane_{0,0}}/M_{LPane_{2,0}}$ {Division by Z to produce X displacement}
 - 7: Set ℓx to $M_{RPane_{0,0}}/M_{RPane_{2,0}}$
 - 8: $X \leftarrow abs(\partial x - \ell x)$
 - 9: Set ∂y to $M_{LPane_{1,0}}/M_{LPane_{2,0}}$ {Division by Z to produce Y displacement}
 - 10: Set ℓy to $M_{RPane_{1,0}}/M_{RPane_{2,0}}$
 - 11: $Y \leftarrow abs(\partial y - \ell y)$
 - 12: Set $Z = f/X$ {Divide the displacements with the camera param, f . Ensure, $X \neq 0$ }
 - 13: Append new $Vector_i(X, Y, Z)$
 - 14: **end for**
 - 15: Return $Vector_{list}(X, Y, Z)$ {Let N be the number of elements in each point structure}
-

256. Hence, time delay is almost negligible in the estimation of a user's 3D position and for interaction tasks.

3.3.2 3D Camera Based Approach

3D camera such as Microsoft Kinect camera [141][151] introduces affordable motion capture technology that can be incorporated to track users in a pervasive computing applications. Kinect¹, a skeletal tracking sensor introduced in November 2010 by Microsoft Research, is a new game controller technology. The device features an RGB camera, depth sensor and multi-array microphone running proprietary software. With the advancement of infrared sensors and computing powers, Kinect camera provides relatively accurate human tracking and movement acquisition interfaces. The depth sensor consists of an infrared laser projector combined with a monochrome CMOS sensor, which allows the Kinect sensor to process 3D scenes in any ambient light condition.

The widely adopted Kinect application program interface allows human full body tracking and joint calculation options. Many other 3D cameras such as Xtion by Asus and RealSense by Intel has mimicked the technology of the Kinect camera. These 3D cameras have been adopted in various applications that utilize the realtime 3D user position tracking and skeletal joint calculation techniques. Recently, in 2014 Kinect camera has seen a major revision of its hardware capabilities that include wider camera angle, higher resolution of the depth and color images, and improved skeletal tracking algorithms. The sensing range of the depth sensor is adjustable, and the Kinect software is capable of automatically calibrating the sensor based on gameplay and the player's physical environment. The monochrome depth sensing video stream of the Kinect camera is in VGA resolution ($640 \times 480 \text{ pixels}$) with 11-bit depth, which provides 2,048 levels of sensitivity. The Kinect sensor has a practical ranging limit of $1.23.5m$, however, the sensor can maintain tracking through an extended range of approximately $6m$. The sensor has an angular field of view of 57° horizontally and 43° vertically. The depth accuracy of Kinect camera is relatively constant within a specific capture volume. The Kinect SDK v2.0 features skeletal tracking of up to 6 users with 3D locations of 25 joints for each skeleton.

Kinect is capable of simultaneously tracking up to six people for motion analysis with a feature extraction of 25 joints per user [141]. The tracking technique brings a robust and proprietary user position and user skeletal tracking technology in which user

¹<http://kinectforwindows.org/>

position, gait, and gesture development can be leveraged. We have incorporated the real-time skeletal and user tracking interfaces in the proposed framework by using the depth data of the Microsoft Kinect camera.

3.4 Annotation Scheme

The annotation mechanism is the core of the proposed smart mobile interaction framework. We have both designed and developed XML based automata in order to describe the geo-spatial metadata of the IoT and further specify rules that we logically assign with the things. The IoT will behave based on these specified rules. Our proposed mobile interaction approach requires the annotation of IoT or sub-regions of interest of the IoT with their geometric coordinates. Unlike RFID chip, NFC chip, or visual marker based annotation approaches, which operate at the physical or device level, the geometric annotation approach incorporates logical annotation that is leveraged at the application level. The application level annotation implementation option provides deep and easy integration with the developed pointMe, touchMe, scanMe type interactions. In Section 3.4.2, firstly we will present the Annotation Language (container) that we have leveraged in order to store the geo-spatial metadata. In Section 3.4.3, we describe 3D annotation schemes for both real and virtual things. Further, in Section 3.4.4, we illustrate the 2D picture based annotation scheme.

3.4.1 Annotation Requirements

We propose a novel spatial geometric annotation based mobile interaction framework that mediates many of the limitations that are present in the known RFID and visual tag based solutions. The framework allows the annotation of both virtual and real world multimedia IoT in the development of various intuitive, multimedia applications. In addition, it aims to accomplish the following:

- Find a way to facilitate communication between virtual and real world.
- Design and develop an intuitive, user-friendly and flexible 3D annotation mechanism to actuate the communication between virtual IoTs and the associated real world things.
- Incorporate haptic interaction between remote users through the virtual world.

- Design and implement intuitive touchMe, pointMe, scanMe interaction scenarios with various IoT by using the spatial geometric annotation.

As we shall present in the remaining sections, the geometric spatial annotation approach provides better flexibility and allows for deeper integration with certain multimedia applications. The integration is not restricted to the modes of interaction or certain hardware gadgets and setups. The framework allows the customized implementation of interaction modalities. Furthermore, once the geometric annotations are realized, the application can choose to implement any of the mentioned touchMe, pointMe, scanMe, or context-aware smart mobile interactions at a later time. Based on the requirement and ease of use of the application, it is possible to extend the built in interactions or build an entirely new set of interactions to be used with the geometric spatial annotations.

3.4.2 Annotation Language

The geo-annotation of smart IoT use geometric and textual data. We have incorporated a Learning Object Metadata (LOM) based annotation specification language in order to store the geo-annotations information. All annotated data is encoded into a customized LOM[65][35] container. For example, the container provides an access point for available information about a specific country on the map. In our geo-spatial annotation scheme, we author the smart real/virtual IoTs in a LOM container by specifying sub-regions of the screen along with their geometric coordinates, which is specified a priori based on the IR camera, 2D camera, or 3D camera placed in the environment.

In order to promote the sharing and reusing of Learning Objects (LOs), LOM is defined as a hierarchical structure consisting of nine broad optional categories. We use four of the categories of the structure as listed in Table 3.1. In this structure, we fit the LOM description with annotation geometric data and store the content in XML format. We use the General field in order to store the name and description of the geo-annotation data. Additionally, geometric data are stored in the Technical field where we added format descriptor to support polygonal, and complex geometric type records. Furthermore, the content type of the geo-annotation is specified here. The Educational field is used to store interaction data, such as supported and preferred interaction types, supported physical/virtual distance, etc. Lastly, the Annotation field has been incorporated to store game specific rules for the geo-annotation data.

The LOMs are stored in the server, which allows for optimal searching and retrieving of the metadata through the point based spatial geometric query process. We use the

NO	Fields Used	Explanations
1	General	General metadata, such as identifier, title, language, keyword, structure, or description of a LO.
4	Technical	Covers all technical information about a LO, such as format, size, location, duration, browser requirements, etc.
5	Educational	Describes the educational and pedagogic characteristics of the LO, such as interactivity, difficulty, end-user type, etc.
8	Annotation	Provides additional information about a LO including assessments and suggestions by other learners.

Table 3.1: LOM categories used to package geo-spatial data.

geometric information of the annotation tag to respond to a user’s spatial query and later pass the associated LOM to the Interaction Controller interface so that the content is rendered with the help of Multimedia Renderer.

3.4.3 3D Annotation Scheme

Microsoft’s Kinect¹ camera features an RGB camera, and a depth sensor. The depth sensor consists of an infrared laser projector combined with a monochrome CMOS sensor, which allows the Kinect sensor to process 3D scenes in any ambient light condition. The sensing range of the depth sensor is adjustable, and the Kinect software is capable of automatically calibrating the sensor based on the physical environment. The monochrome depth sensing video stream of the Kinect camera is in VGA resolution ($640 \times 480 \text{ pixels}$) with 11 – *bit* depth, which provides 2,048 levels of sensitivity. The depth sensor image, VGA image, and the processed 3D skeletons are displayed in Figure 3.10.

The proposed 3D camera based annotation scheme is suitable for indoor spaces, where interaction with a number of physical multimedia IoT e.g. television, audio player, movie player, air conditioner etc. are intended. By using the 3D depth sensor data, we geometrically annotate the physical devices in the smart environment as shown in Figure 3.11.

Based on the 3D position estimation methodology, we introduce a spatial-geometric approach to annotate the physical IoT on the specification of geometric coordinates of

¹<http://kinectforwindows.org/>

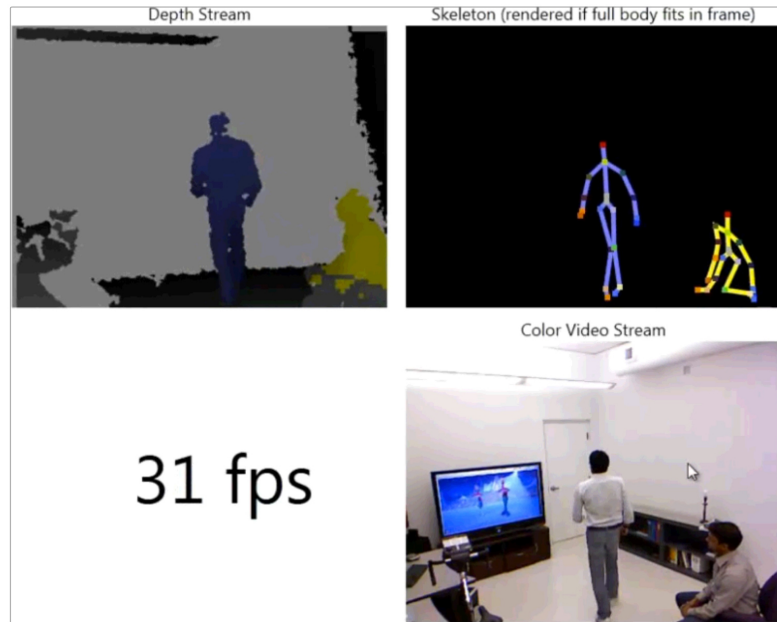


Figure 3.10: Streaming 3D depth images of a 3D camera.

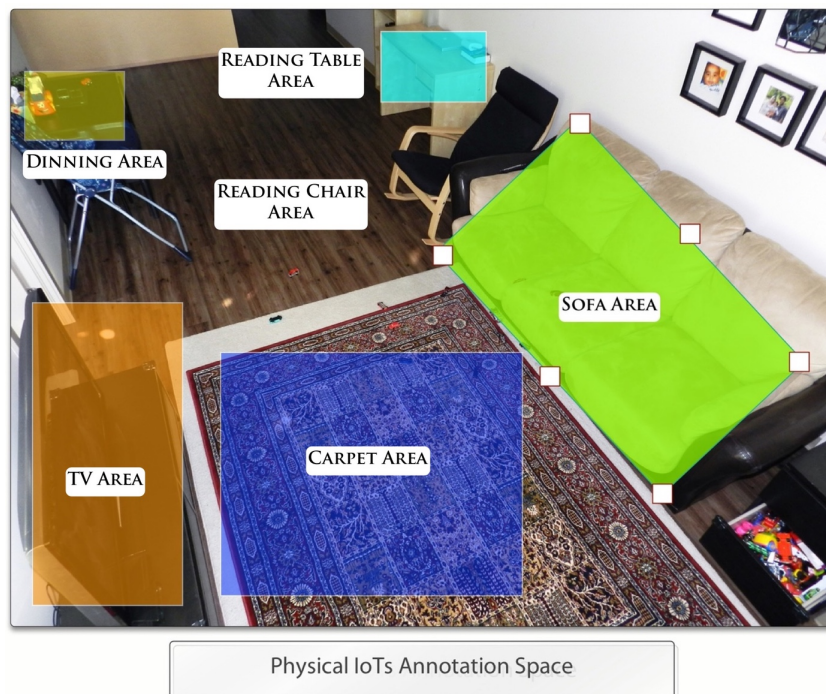


Figure 3.11: 2D layout based physical IoTs annotation approach by using 3D image data.

the approximate boundaries (voxels) of an object, which is facilitated by using IR or 3D cameras. In order to capture the motion path and formulate shape vector for the target objects, we have used our previous motion point to symbol mapping scheme [98]. Using that approach, we have annotated an object in both XY and YZ plane. The geometric annotations are voxel based and uses the IR camera reference point. Each of the geometric annotations are at least 2 – 3 feet in each 3D directions so that pointing and other form of mobile interaction becomes easy. Based on these voxel based geometric annotation, a physical environment device map is prepared. Afterwards, all annotation data is stored in the database and is associated with information/services, which are accessed when interacting with that annotated object.

We have considered a 3D virtual environment, where users can be represented by their virtual avatars. The users are able to control the movement of their avatars through WIMP based interactions or touch based [97] interactions. We have assumed the 3D virtual environments to be populated with virtual IoT such as multimedia devices, lights, carpets, sofas, etc. The avatars also interacts with virtual 3D IoT in the virtual environment, mimicking real world based interactions such as touchMe, pointMe etc. The 3D virtual environment recognizes these interactions with the virtual IoTs and provides APIs where the interaction events can be customized to provide real world based services.

We have created a virtual home that resembles a real world physical home. The virtual home contains certain 3D IoT, corresponding to their physical home IoT. We use the 3D virtual environment APIs to annotate these virtual IoT. In our system, we have annotated 3D IoT and have specified the corresponding physical device addresses, which we have assigned to each devices. Every real life object has a unique identifier by which smart home services can interact with it. We have provided a detailed description in the next chapter of such a prototype [61], where we show how the real world IoT are controlled by interacting with their respective virtual IoT in the 3D virtual environment. For example, we annotate a 3D lamp inside a virtual living room and when the virtual avatar incorporates touchMe interaction with the 3D lamp, we toggle the switch of a real-world physical lamp in a real living room. By using the same mechanism, virtual doors can be interacted with to control real world doors in a physical space. We have used X10, Bluetooth, and Wi-Fi communication channels in order to realize such scenarios. Figure 3.12 depicts the geometric based avatar annotation scheme of a 3D virtual environment.

Furthermore, we have extended the virtual interactions to a virtual social space, where avatars can interact with each other by touching, or hugging in the virtual space and feeling the touch, hug etc in the real world [64]. The real world users wear haptic devices

such as a haptic jacket [100] to feel those social interactions. In order to realize such a system, we have annotated the visible body parts of the 3D virtual avatar and have specified the corresponding physical haptic actuators to render the haptic feedback. For each haptic signal, we have also annotated the avatar animation. For example, we have annotated the 3D male avatar’s left arm and specified a particular vibrotactile actuator stimulation for it. The description of such a prototype is provided in the subsequent chapters.

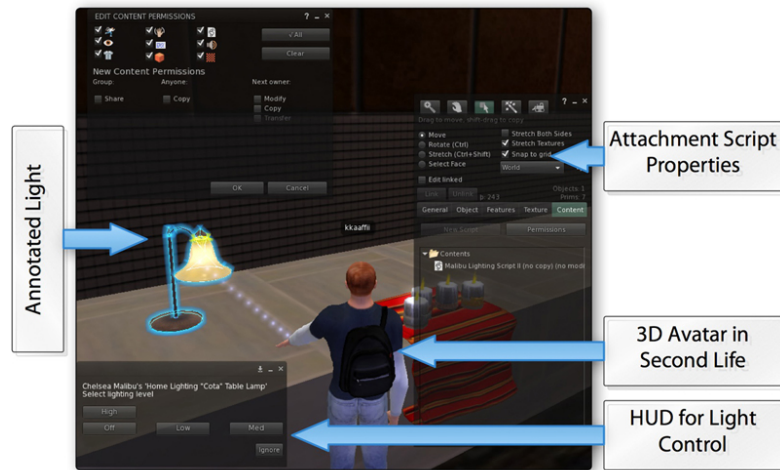


Figure 3.12: 3D virtual IoT annotation.

3.4.4 2D Annotation Scheme

In the proposed 2D annotation we have provided template based interfaces with the framework. The interface and GUI containers can be extended to realize the annotation of any 2D IoT or streaming video with respect to a camera or a sensor reference points. In the annotation of streaming video or dynamic interfaces such as eBooks, we store the annotation keyframes based on certain properties of the IoT. For example, for streaming video, annotations can be specified for the keyframes with respect to timelines. In the case of eBooks, each page of the book can be treated as a separate canvas for the 2D geometric annotations. An example movie annotation editor is depicted in Figure 3.13.

Physical IoT can be annotated with the help of a 2D camera image or a sensor. To illustrate, in order to annotate a physical atlas, the annotations can be performed over a 2D image of the map with respect to a reference point. In the interaction controller, the targeted point in the map is transformed and is scaled to ensure that the point can

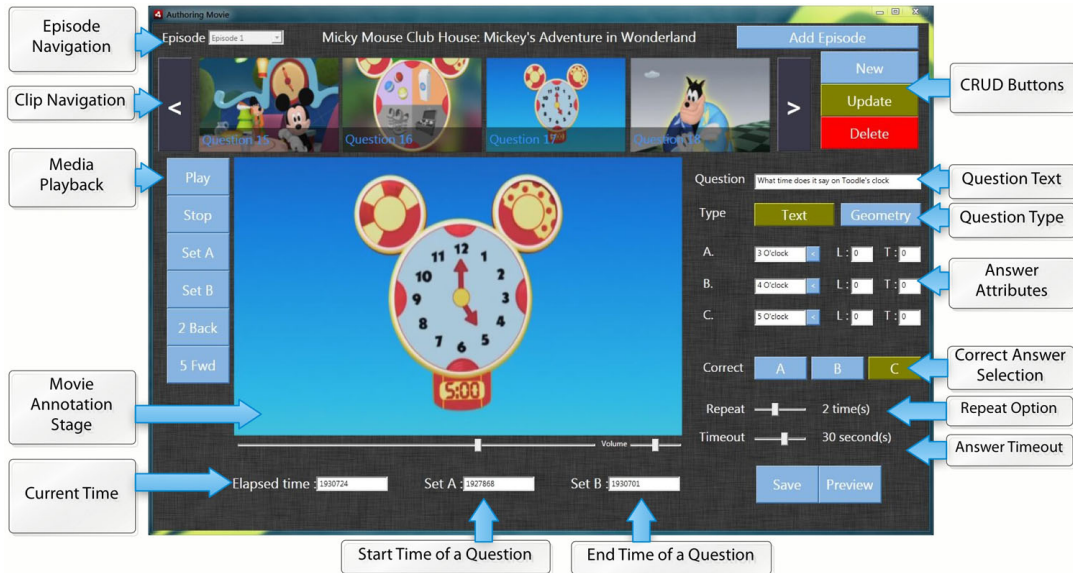


Figure 3.13: Movie objects annotation with respect to timeline.

be used to spatially search the map for a particular country. This type of 2D annotation scheme is feasible when the target's physical size is large (more than $3sqm$) and static. In such physical IoT, the number of annotation regions and the density of the annotations on the canvas can be complex.

It is also possible to annotate the physical IoT directly by using an IR emitter pen. The components of the IR emitter based physical map annotations are depicted in Figure 3.14. This type of annotation mechanism is feasible when the physical object's size is moderate (no more than $3sqm$) and the number of annotation regions, as well as the density of the annotations on the canvas are manageable. One of the advantages of the IR tracking based 2D annotation scheme is that the annotation process itself is rewarding and can assist in the learning process when used as a learning prototype design [104].

3.5 Interaction with Geo-Annotation

3.5.1 Annotation to pointMe Mapping

For certain interaction scenarios of our smart mobile interaction approach, we may need to estimate the 3D position of a user with respect to the annotated content when the user is interacting with physical IoT in an indoor environment. In some interaction scenarios, we may want to calculate a user's gestures, body posture, or hand movement with respect

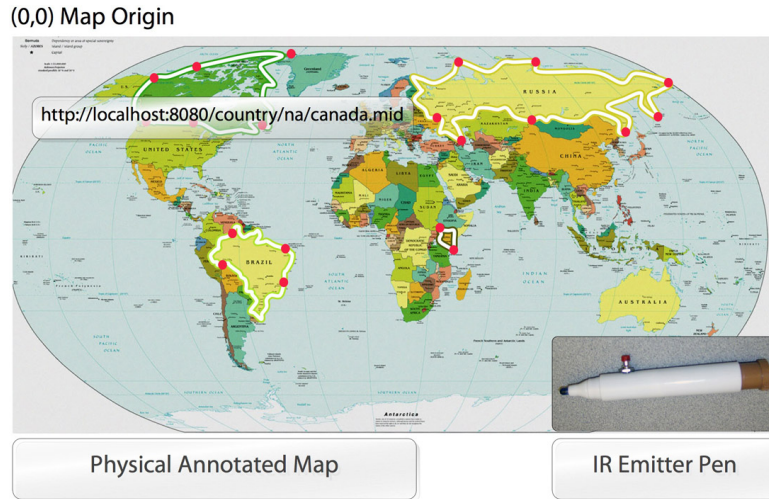


Figure 3.14: Manual annotation process of a map.

to the annotation. In mobile based interaction scenarios, we may also need to determine the annotated content at which the user is pointing with their mobile devices or their hands. In order to describe the approach of estimating a 3D position, in Section 3.3.1 we present the arrangements and settings of the IR cameras. Further, in Section 3.5.1, we present an IR position and accelerometer data fusion algorithm to calculate pointing direction and approximate pointing location of the user in the smart environment. In Section 3.5.1, we further discuss, how the calculated data is utilized to interact with geo-spatially annotated objects.

Position and Accelerometer Data Fusion

The accelerometer reading obtained from the mobile device is aggregated with the IR emitter based 3D location estimates and the resulting 3D location and tilt data is used for target object selection. The algorithm accepts the mobile object as a parameter through which it can access the raw accelerometer data. In this simplified calculation process, we have created a virtual screen that touches the Field-Of-View (FOV) boundary of the 3D mobile position. The process estimates the 2D position $G(x, y)$ in the virtual screen and connects the 3D IR emitter position to determine the slope by which the mobile is tilting. The complete description of Algorithm 3 and its stepwise illustration can be seen in the following section. The output of this algorithm is the targeted 2D virtual screen coordinate G .

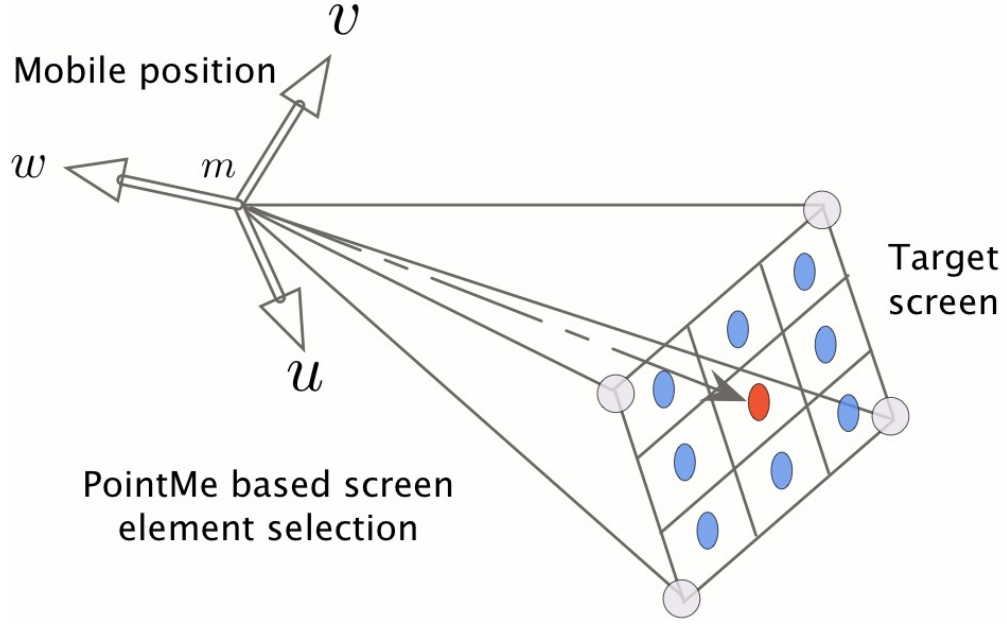


Figure 3.15: Mobile pointMe based object selection approach.

Algorithm 3 Fusion of Accelerometer and IR Sensory Data

Require: Object A_M that holds updated accelerometer parameters. Object A_C holds the updated compass parameters. {We suppose that the mobile is facing a virtual screen at a distance d and we determine the point G on this screen so that if a line is drawn from the mobile FOV it will intersect G . Let the width of this screen be W and height H }

Ensure: The IR node is visible, otherwise the previous screen position is returned

- 1: Set $R_x \leftarrow (1 - (\text{round}(P.\text{getPosX}()) / W)) * A_M.\text{getRawX}()$
 - 2: Set $R_y \leftarrow ((\text{round}(P.\text{getPosY}()) / H) * A_M.\text{getRawY}())$ {Get the IR Point Coordinates (let these two points are in the same Plane with common Z. Here, P is an element of the $Vector_{list}(X, Y, Z)$ }
 - 3: Set $\partial x \leftarrow (((R_x / (A_M.\text{getRawX}() / 2)) - 1) * \varphi_{SR}) + x_{Offset}$ { φ_{SR} is primary sensitivity variance of the accelerometer}
 - 4: Set $\partial y \leftarrow (((R_y / (A_M.\text{getRawY}() / 2)) - 1) * \varphi_{SR}) + y_{Offset}$ { x_{Offset} and y_{Offset} are IR emitter smoothing offsets, used for calibration purposes}
 - 5:
 - 6: Update $G \leftarrow \partial x, \partial y, P.\text{getPosZ}()$ with respect to A_C parameters.
 - 7: Calibrate G with accelerometer parameters A_M to adjust for speed and direction.
 - 8: Return vector G { G holds target object position on the virtual screen}
-

Target Physical Object Selection

The target physical object selection is based on the fused sensing data, namely the 3D location information of the user’s mobile and the tilt angle to the virtual screen. In order to determine the targeted physical objects, we have constructed a geometric spatial query. In this query, let the position of the user be $P(x, y)$. For different FOV angle, we want to determine the spatial triangle $\triangle PLR$ (see Figure 3.16). Let us assume that only the objects that are d distance from the user’s mobile will be considered for selection. The position of the point F , which is opposite to P along the centre line of the FOV angle is $(x + d * \sin R, y - d * \cos R)$. Similarly, the position of the left point L of the FOV triangle is $(x + d * \sin R - \theta/2, y - d * \cos R - \theta/2)$ and the position of the right point R is $(x + d * \sin R + \theta/2, y - d * \cos R + \theta/2)$. After calculating these three points, the process works by finding all the candidate objects that appear in the viewing triangle and then providing them weight so that one possible target object can be selected. The weight is determined based the proximity of the candidate objects with respect to the FOV centre line. It is also possible to show all the candidate objects inside the FOV instead of just selecting a single object. In this case, the user would have to determine which particular objects she/he intended to select.

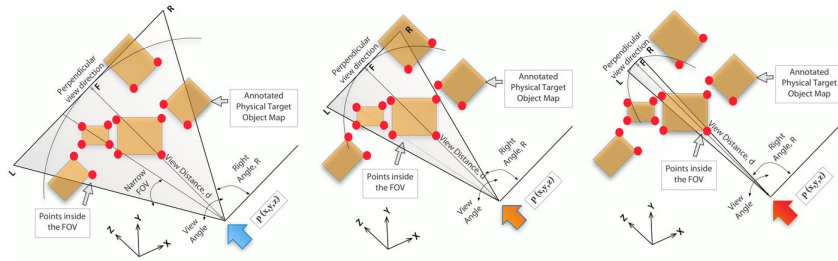


Figure 3.16: FOV angle variation and physical target object selection. Narrow FOV angle selects only the target objects but user maneuvering becomes difficult and results in poor accuracy.

We now provide an explanation of the selection process in detail, by considering the FOV angle and the tilting angle. Our target object selection process is based on a horizontal spatial query and a vertical spatial query. In the horizontal spatial query, the FOV triangle is used to apply a divide and conquer strategy to locate the physical object coordinates and a temporary stack is created to store the references of those objects. The objects in the stack are sorted in ascending distance order from the viewer’s position $P(x, y, z)$. Afterwards, another search to the sorted list is performed to determine the

weight of each objects based on their distances from the centre line PF of the FOV. This process is followed by a vertical spatial query, where we verify if the tilting line from the mobile device passes through the weighted objects. If the vertical spatial query does not return any result (i.e. the user is pointing slightly higher or lower than the target object or the view angle is narrow) then the first element sorted from the horizontal scan could be returned as the target physical object.

For example, in Figure 3.17 (left), a horizontal geometric scan is performed to obtain the list of objects: Fan object, Video object and Light object. Afterwards, in the vertical scan (right in Figure 3.17), only the Light object is found to intersect with the accelerometer tilting line. Hence, the light device is returned as the selected object to the feedback generation module.

As mentioned previously, another condition for object selection is the distance from the mobile device to the target object. In order to simplify the discussion, geo-spatial objects are modeled as rectangular prisms that enclose real object polygons in a map; since a concave polygon is much more complex in computing visibility than a convex one. Our triangular geo-spatial search is therefore, dependent on five parameters namely the 2D position of the user P , the distance d of the objects from P , the right angle R , the camera angle θ and the vertical tilt line PG. Using these parameters, the geo-spatial query is constructed. However, among these parameters, the selection of the view angle θ and of the distance d is crucial. As depicted in Figure 3.17, the number of objects inside the FOV triangle is selected according to the view angle θ .

The more narrow the angle, the fewer objects there will be in the frustum and hence, the more the user will need to correctly point to the target physical object. Figure 3.18 (a) explains such a case, where the narrow view angle would only cull portion of the targeted object and as a result would not register as a selection. Also as evident in Figure 3.18, the view distance needs to be chosen accordingly in order to ensure that the sub-regions at the end of the objects are included while culling the target physical objects. In cases where the view distance is kept at a minimum, instead of a sub-region based culling, an object based selection method can be employed. To ensure accurate physical mobile selection, these two parameters need to be adjusted based on the placement of the physical objects and the space inside the indoor physical environment.

Also, the geometric query execution time depends on those parameters. Typically, the horizontal geometric query execution requires more time than the vertical geometric query execution. However, in both cases, the amount of CPU and physical system memory resources that have been used is negligible. For the above interaction scenario,

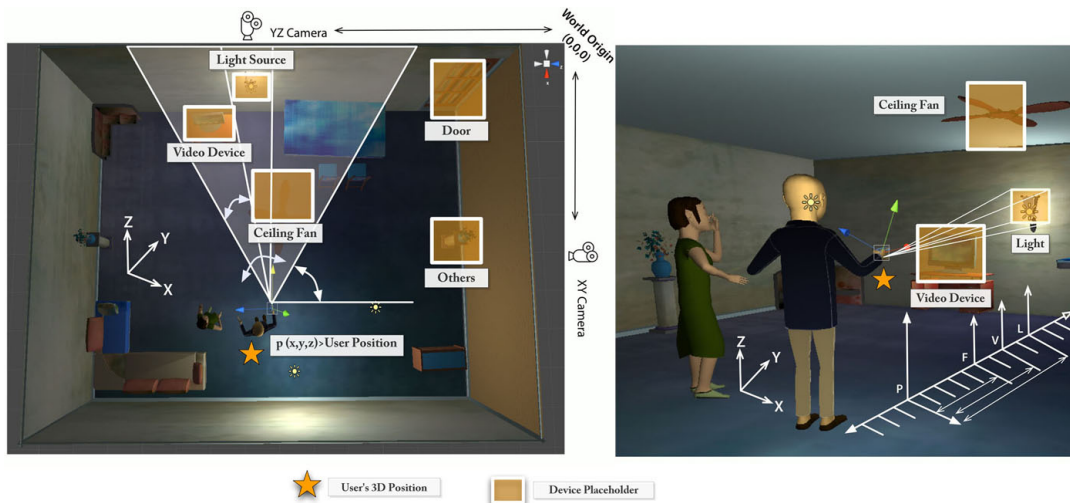


Figure 3.17: Closed FOV based physical object selection approach. On the left we graphically show the horizontal spatial query process in overhead view. On the right we graphically show the 3D vertical spatial query process. Here, point p is the mobile phone position, F is the fan distance, V is the video device distance and L is the distance of the Lamp source.

the horizontal geometric scan requires 30 ms and the vertical query scan requires 20 ms. The overall time that the system requires to respond to a user's interaction is 1280 ms, where three geometrically annotated objects have been scanned in the process.

3.5.2 Annotation to touchMe Mapping

We implement two different touchMe interactions of the proposed smart mobile interaction framework. First, the implementation of THIN mobile clients to incorporate touchMe interactions with geo-spatially annotated objects in graphics intensive applications and second, dedicated touchMe applications. By implementing the mobile touchMe interaction applications, we present guidelines that can be used in the development of similar touch oriented smart mobile applications.

In the THINC [10] and the MobiDesk [11] architectures, virtual display buffer management and screen resizing are used for devices with different display resolutions. One of the closely related works is presented in ReDi [132]. The work proposes an interactive virtual display system for ubiquitous devices using thin clients, through which users can efficiently leverage the local display capabilities and remote computing resources. For

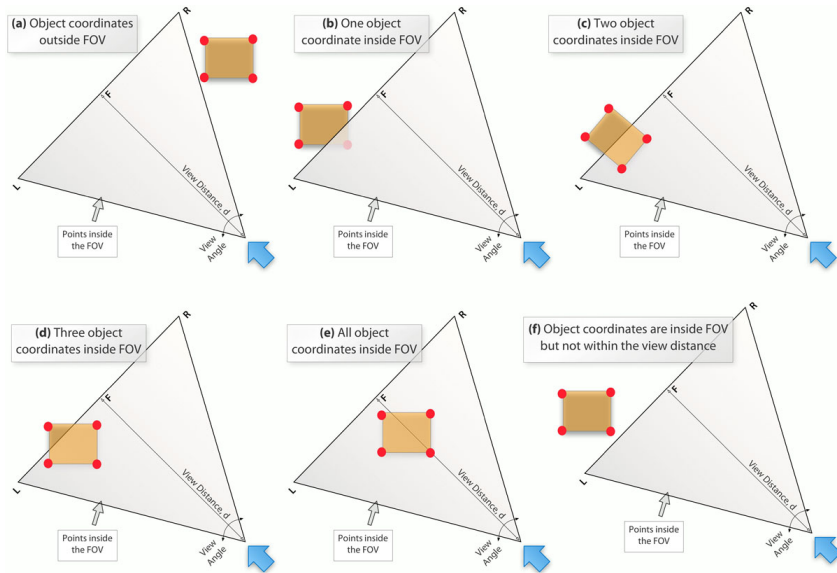


Figure 3.18: PointMe scenarios with nearby geo-annotated objects

example, currently mobile devices may lack the graphics horsepower to render a 3D environment that uses gigabytes of textures, sounds and animation data that requires a huge amount of network download time and CPU processing time. These processing overheads can adversely affect frame rate, draw distance, render detail, and the battery life. Therefore, in order to render to such graphics intensive user interfaces in mobile devices, we have adopted a display virtualization technique that incorporates device collaboration [132][110]. In device collaboration heavy computations are carried out remotely by leveraging remote computing resources, and screen updates are compressed and transmitted together with low-level display commands from servers to clients [126].

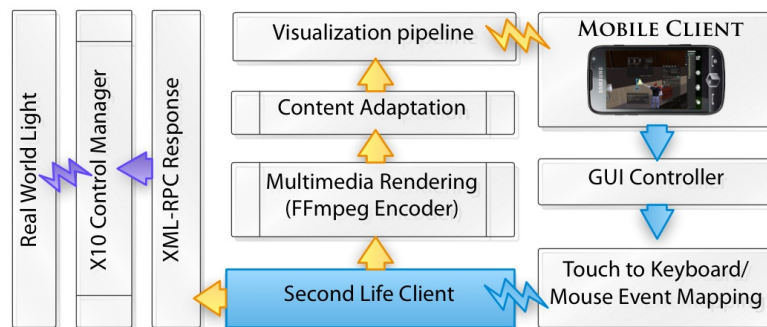


Figure 3.19: Mobile thin client based touchMe interaction.

As an example of such a realization, we present the THIN client based Second Life (SL) GUI virtualization on a mobile screen. As shown in Figure 3.19, the Region of Interest (ROI) of the SL desktop window is encoded in real-time. Furthermore, the compressed content is adapted for the mobile screen. We have established a UDP pipeline with the mobile client to transmit these adapted frames in real-time. In the mobile THIN client, we have adapted the touchMe GUI interactions and have mapped those to mouse events. The mouse events are further virtualized in the desktop client by using mouse event scripts. Other type of event virtualization and transformation can easily be implemented by following the recommendations presented in the examples of the framework design. By using the guidelines in the design we realized a THIN client based prototype in order to interact with a geo-spatially annotated movie game. The movie has been annotated at predefined timelines. We present the detailed illustration of the prototype in a later chapter of this thesis.



Figure 3.20: Mobile touchMe app examples.

The interaction scenario is depicted in Figure 3.20 (1). During the gameplay, the movie pauses at each of the annotated timelines and shows interactive multiple choice questions related to that specific scene of the movie. It also provides interaction options to that s/he can easily select an answer option that is geo-spatially annotated. In order to interact with the movie, we have realized a touchMe interaction based mobile application. Instead of rendering the content of the television frames, we have associated and have rendered GUI buttons with the presented quiz answers in the mobile app. Figure 3.20 (2) depicts the mobile app. As shown in the figure, the answers presented on the multimedia screen are coloured in Red, Green and Magenta. In order to provide a touchMe based answer selection option, we have rendered Red, Green, and Magenta

coloured GUI buttons in the mobile app. The captions and colours of these buttons correspond to the geo-spatially annotated answer options. For example, touching the Red GUI button will select the answer option Red. In addition to colour mapping, we also have displayed the Quiz question and have labeled the GUI buttons with the answer options in the mobile app. The colour association with the geo-spatially annotated tags significantly enriches the usefulness of the touchMe interaction of the mobile app.



Figure 3.21: Mobile touchMe app for Haptic Book.

Lastly, in another touchMe interaction prototype, we have presented the Haptic Book mobile app (see Figure 3.21). We geo-spatially have annotated certain paragraphs of an eBook with haptics, audio, and visual data. For example, when the reader touches the annotated eBook sections (highlighted in colours), the prototype automatically sends haptic commands to the haptic devices in order to mimic train, boat, or car riding experiences. Additionally, the prototype renders audio and video materials relevant to the reading content on the home entertainment system. The standard touch APIs are used both to determine touch screen coordinates and mapping of the identification of the paragraph of the ebook that have been touched. We have incorporated Bluetooth, and WiFi communication channels to communicate with the haptic and audio-visual devices.

We have also used thin client based mobile TouchMe interactions to interact with geometrically annotated IoT. A touchMe interaction with an annotated IoT is easy to realize in real world applications. Geometrically annotated IoTs permit the realization of multiple interactions. We have provided a detailed description of such a prototype

in the next chapter, where we show the realization of pointMe, touchMe and other interactions with the same set of geometrically annotated IoT.

3.5.3 Annotation to scanMe Mapping

In a scanMe type interaction in the proposed framework, we devise service or content delivery based on particular user gestures, or user-activities (e.g. walk three steps, or do a certain activity a certain number times to activate an answer of a learning question). In a pointMe based map learning game, in addition to providing information about the pointed country on the map, we scan the nearest countries on the map and present them on the mobile screen based on their proximity to the pointed country. A screenshot from the implemented scanMe mobile application is depicted in Figure 3.22.



Figure 3.22: Mobile scanMe client for map selection.

The scanMe parameters for the annotated IoT that we incorporate in the proposed framework are

- User position with respect to geo-spatially annotated IoT: for example, when answering a quiz question depicted on the multimedia screen with three possible answers, we define three coloured area in front of the multimedia screen that corresponds to the colour of the answer choices. By standing on a particular colour and

by performing a gesture (e.g. step three times) the learner will be able to confirm her answer choices. We present the description of the prototype in the next chapter (see Figure 3.23).

- User posture: when the user is near a geo-spatially annotated IoT, certain postures would activate predefined services specific to that nearby annotated IoT. For example, if a user is near a reading table and then sits on the reading chair, the reading light will go "ON" (see Figure 3.25).
- User gesture: this scanMe parameter works in a similar way to the user posture parameter, however, the service activation or content delivery occurs when the user performs certain gestures. A single and easy to perform gesture can be used for delivery services, where the service selection depends on the proximity of different annotated IoT.

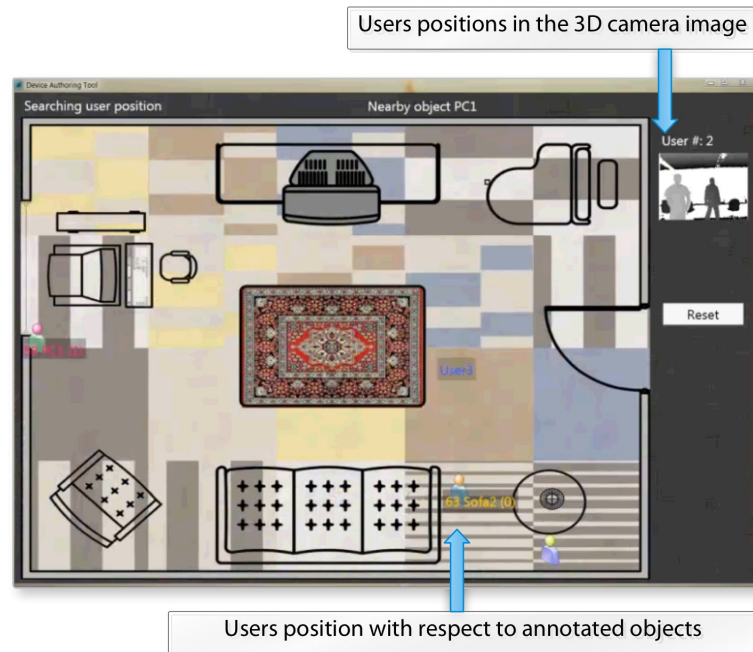


Figure 3.23: User position with respect to geo-spatially annotated objects.

In order to measure user position, user posture, and user gestures we have incorporated a Kinect sensor's skeletal tracking technique. We have used the skeletal data from the Kinect sensor in order to deduce the user's arm joint and leg joint data in real-time. The gesture recognition module tracks the detected joints and continuously monitors the

joint angles. To describe the working process of the Algorithm 4 we want to present the joint detection and joint angle determination data. The joint angles are calculated in real-time when the user is moving their hands, their wrists, their elbows, or their shoulders. An analysis of these two angular results is further mapped with the interaction commands. In order to calculate the angle between the three joints that represents essentially three vectors, we have used Equation 3.1, which uses only one $acos()$ method call.

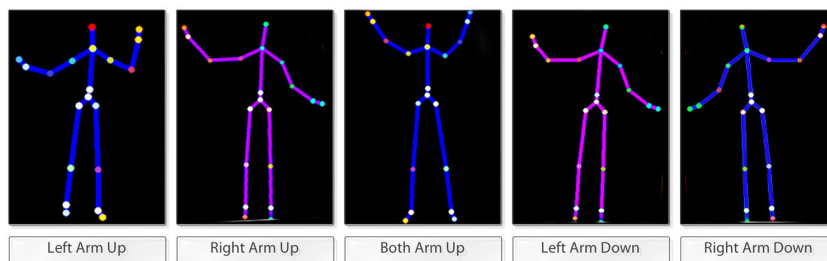


Figure 3.24: 3D skeletal gestures.

$$\begin{aligned}
 uv &= P_{j1} - P_{j2} \\
 wv &= P_{j2} - P_{j3} \\
 \theta &= \text{acos}f(2 * ((\vec{u}\vec{v}.\vec{w}\vec{v})^2 / \vec{u}\vec{v}^2 / \vec{w}\vec{v}^2) - 1)
 \end{aligned}
 \tag{3.1}$$



Figure 3.25: Skeletal 3D standing postures of two users.

The process used to recognize gestures from the hand drawn joint motion points is multifaceted. Whenever a user moves/rotates his/her arm and performs a gesture in the

Algorithm 4 3D Kinect Gesture

Require: Object K_M that holds updated skeletal parameters from the Kinect camera.

The update is available after a valid geometrical skeleton construction. {The gesture calculation starts by using the current body joint positions. The algorithm updates variable $gStates_i$, where $0 \leq i \leq n$ and n is the number of gestures.}

- 1: Calibrate the user with the predefined poses;
 - 2: **if** $gUserGenerator \rightarrow GetSkeletonCap.IsTracking(K_M)$ is *true* **then**
 - 3: Calculate $gUserGenerator \rightarrow SkelJointPosition(K_M, \Gamma_i, L)$;
 - 4: For all J in Γ_i , calculate $\alpha = \sum_i^L \{\Gamma_i.getConfidence()\}$ { L is the number of joint positions to be tracked}
 - 5: **if** $\alpha \geq L/2$ is *true* **then**
 - 6: Convert 3D world to projective plane points (Γ_i, L) ;
 - 7: Update depth, Z for the projective plane points;
 - 8: Look for right hand 3D joints, e.g. *Wrist (Head of Radius) W_{HR}* , *Elbow (Distal Radio-Ulna) E_{RU}* , and *Shoulder(Cartilage) S_C* changes and calculate the angle φ_{Right} that is created by the vectors;
 - 9: Look for left hand 3D joints W_{HR} , E_{RU} , and S_C changes and calculate the angle φ_{left} that is created by the vectors. {The LOM capture gesture uses timer to look for Z disparity, ϑ . Here, S_ϑ is the hand stretch disparity.}
 - 10: **if** $W_{HR}\{Z\}(t) - W_{HR}\{Z\}(t-1) \geq \vartheta$ AND $\varphi_{right}(t)/2 * 180/PI \geq S_\vartheta$ **then**
 - 11: Compute left/right hand up and down gestures using $\varphi(t)$ AND $\varphi(t+1)$ by comparing their elevations with empirically calculated S_ϑ ;
 - 12: Set the gesture variable flags $gStates_i$, where $0 \leq i \leq L$;
 - 13: **end if**
 - 14: **end if**
 - 15: **end if**
 - 16: Return the gesture variable, $gStates_i$
-

air by using free hand movements, the motions of the hand joints produce a sequence of motion points and hence, generate joint angles. The gesture recognizer module inputs these motion points, calculates the joint angles and sends them to the grammar processing unit of the architecture, as primitives for syntactic analysis, to ensure that the whole gesture can be recognized according to the training production rules. The process uses string matching and regular expression parsing techniques to accurately distinguish a gesture drawing. By tracking the arm motion, we continuously determine the angle of the arms and the rotation angles of the hands in real time. We monitor the changes in the raw depth data of the assigned hand. We further extend the gesture recognition algorithm and use machine learning technique to teach our system to recognize user postures. By extending the user posture recognition approach, we define 4 user gaits, a) Standing, b) Sitting relax, c) Sitting normal, and d) Walking. We incorporate these gaits in the context-aware interaction implementation, which is discussed in the following section.

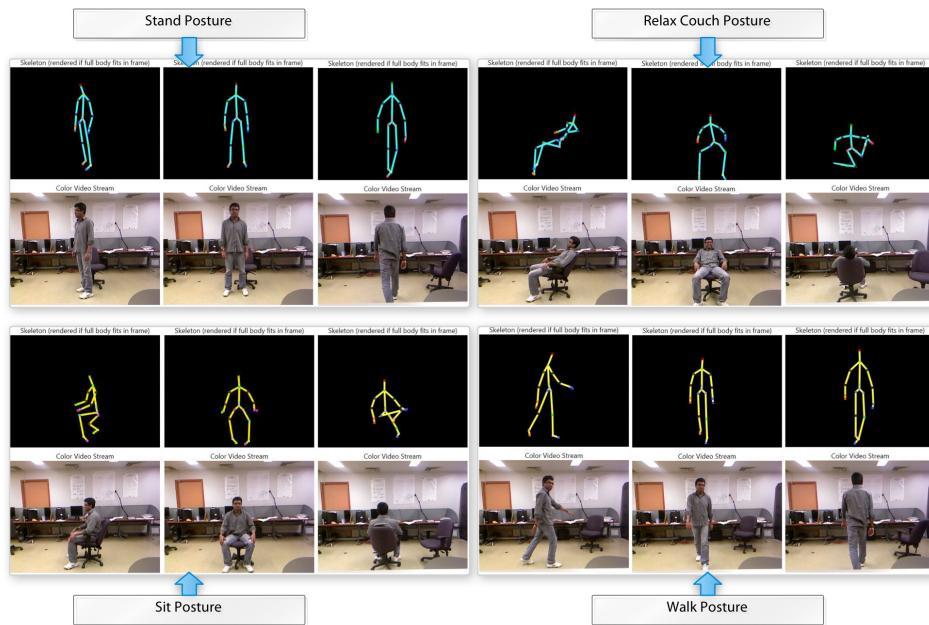


Figure 3.26: List of user gaits used in the implementation of context-aware interaction.

3.5.4 Annotation to Context Mapping

Schilit et al. [79] (1994) [121] have categorized context into three categories using a conceptual categorization technique based on three common questions. 1) Where you

No	Module Used	Parameters
1	System Local	Time, Date, Calendar appointments, Previous activity record
2	Tracking Services	User 3D position, User position with respect to annotated smart objects, Number of users
3	Skeleton Processing Service	User posture, gait
5	Audio Recognition Services	User voice commands
6	Camera and Gait Recognition Services	User authentication

Table 3.2: Context parameters used in the framework

are: This includes all location related information such as GPS coordinates, common names (e.g. coffee shop, university, police), specific names (e.g. Canberra city police), specific addresses, user preferences (e.g. users favourite coffee shop). 2) Who you are with: The information about the people present around the user. 3) What resources are nearby: This includes information about resources available in the area where the user is located, such as machinery, smart objects, and utilities.

In our context-aware interaction realization, we suppose the context to be a boolean combination of a set of sensory data. We term these combinations as context rules. In our work, we argue that given the sensory data, it is possible to define useful context scenarios with respect to annotated IoT. We relate the sensory context parameters with the geo-annotated IoT to obtain more realistic contextual rules. A context rule also acts as geo-spatial annotation to the smart environment. Next, we show some examples of such context rules and highlight their realizations for interaction with geo-annotated IoT. The context parameters that are used in the realization are listed in Table 3.2.

Overview of the Context Framework

We present the overview of the context design framework in Figure 3.27. We have designed the context framework with respect to geometrically annotated IoT. We describe the components of the framework in the following section.

- **Event Processing Nodes:** The modules of this component work in real time to deliver sensory data to the Intelligent Decision Services component. It consists of a 3D position calculation module that identifies a user's position in the 3D

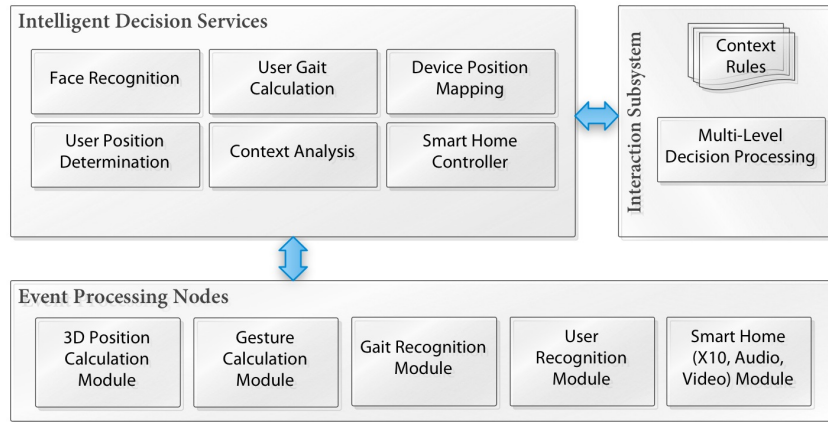


Figure 3.27: Context analysis system.

environment by utilizing a 3D camera sensor. The gesture calculation module produces valid gestures that a user performs. We have implemented six hand gestures that uses the joint angle, as explained in the previous section. The gait recognition module is responsible for calculating a user's posture and mapping it further to obtain Sitting, Lying down, Standing gaits. In the user recognition module, we incorporate a camera based facial recognition approach and a skeletal joint ratio based gait comparison approach. The smart home module provides a wrapper to call the Server side sensory APIs such as X10 API to control home devices.

- **Intelligent Decision Services:** The decision services utilizes the data obtained from the event processing nodes. Here, the actual processing of the obtained data is performed.
- **Interaction Subsystem:** This component is a sub-component of the Interaction Controller of the proposed framework. It defines a set of context rules and leverages the processed sensory data stored on the server to produce context-aware interaction options. Based on the available sensory data , a complex multi-level context parameter analysis could be performed to define complex context scenarios as shown in Figure 3.29.

In addition to user face recognition, we have also incorporated a user's skeletal joint ratio based continuous gait recognition approach. As shown in Figure 3.28, the skeletal joint length and ratio of two users are distinguishably different from each other. We

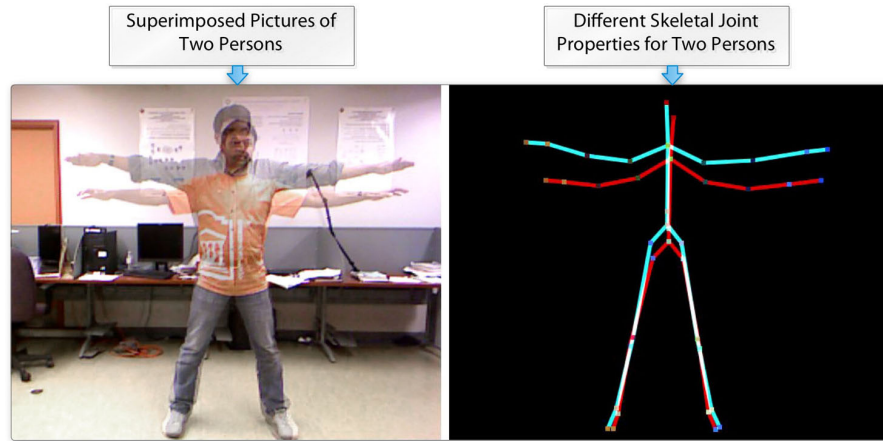


Figure 3.28: Difference between user postures based on 3D skeletal joint ratios.

have streaming depth images from the 3D camera that we utilize to calculate major joint lengths (normalized by the distance of the user from the camera) and their ratios. For each user, we train the system with a number of such joint lengths and joint ratio binary data as we capture their depth images from varying distances and angles. For a small set of users (for example, members of a family), this approach proves to be a quick user recognition technique.

Programmable Context Rules

We want to emphasize that the design of the context rules are possible as we leverage the geo-annotation data of the IoTs in the smart environment. We have programmed the following context rules by utilizing many of the above context parameters in order to activate specific services in the smart environment.

- A. Turn on reading lights near a bookcase, a magazine stand, or a reading table: The context parameters that are used in this interaction are, 1) Bookcase geometric annotation data , magazine stand geometric annotation data, or reading table geometric annotation data, 2) User position with respect to those geometrically annotated IoT, 3) User gesture, and 4) User gait. We have programmed the context rule as, *Readinglightstate* = User position is within the boundary of (1) *AND* User gait is Sitting position (when considering a reading table) *AND* User gesture is Right hand up. User may use a gesture to confirm the activation of such services (we terms such gestures as active). By utilizing this context rule, the interaction

subsystem only turns the reading light *ON* when the user is within the proximity of the reading table or bookcase etc.

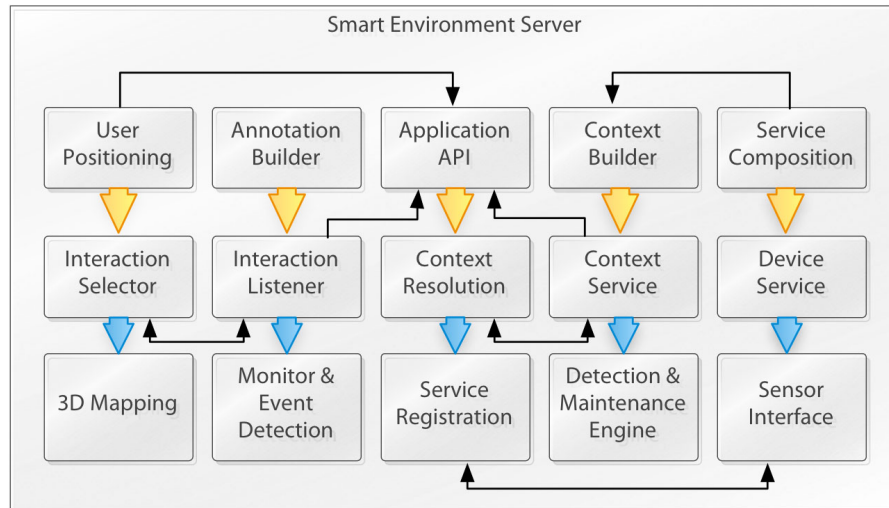


Figure 3.29: Context rules incorporated into the framework.

- B. Play cartoon when the child is $5m$ away from the television and is sitting on the sofa: The context parameters that are used in this interaction are, 1) User recognition, 2) User gait, 3) User position with respect to the Television and Sofa geometric annotation data, and 4) Playback media type is cartoon. We have programmed the context rule as, $Playmediastate = Playback\ media\ is\ cartoon\ AND\ User\ is\ a\ child\ AND\ User\ position\ is\ 5m\ away\ from\ the\ Television\ geometric\ annotation\ data\ AND\ User\ position\ is\ within\ Sofa\ geometric\ annotation\ data\ AND\ User\ gait\ in\ Sitting\ position$. The interaction subsystem utilizes the rule to make sure the child is at least $5m$ away from the television (thereby ensuring a safe distance for watching movie without discomforting or harming the eyes, etc.) and sitting on the sofa when a cartoon is playing on the television. If the conditions of the programmed rule are not met, the interaction subsystem automatically pauses the movie.
- C. Device activation based on proximity: Based on a user's position with respect to the annotated IoT in the smart environment, the interaction subsystem may automatically activates the nearest IoT when the user performs an active gesture. This context-aware scenario is suitable for elderly home care. As an example, an elderly person may use an active gesture to turn ON the playback

of the television, when s/he is sitting on the living room sofa. In this manner, only one or two gestures are used to access many services in the smart environment. The context rule for device activation based on proximity can be written as, $StateDevice(X) = \text{Position of User is within } Y \text{ AND User gesture is ACTIVATE AND User gait is } Z$. In order to implement an example scenario, we set $X = Televisiongeometricannotationdata, Y = Sofageometricannotationdata$, Activate gesture = Right hand up, User gait = Sitting position. The rule implies that, when a user is sitting on a sofa and lifts her right hand up then the television is turned ON. As implied in this example rule, a user can choose an ACTIVATE gesture from a set of predefined gestures that are realized in the framework.

- D. Energy savings: We implement many automatic device turn OFF/ON interaction rules in order to save energy. In this context scenario, when a user is not sitting near the reading table, the interaction controller may turn the reading light OFF (if programmed in such a manner). We use the following context rule: $StateDevice(X) = \text{Position of User is away from } Y \text{ by } Pm \text{ AND User gait is } Z$. In order to explain the rule, let us assume that $X = readinglightgeometricannotationdata, Y = readingtablegeometricannotationdata, P = 5m, Gait = any$. Hence, when the user is further than 5m away from the reading table, the reading light automatically turns OFF. Similarly, based on the predefined user posture and the time of the day, the lights or other electrical smart home appliances of the smart home can be programmed to turn OFF/ON to conserve energy. As another example, the smart home appliances can be turned OFF when the user is not inside the smart home at certain time of the day.
- E. Media playback customization : When a child enters the living room area and is in front of the television and playback media type is horror (R type), then the interaction controller automatically switches the media playback to a G type media (e.g. Animation). We write the context rule as follows: $Mediaplaybackstate = \text{Playback media type is Horror AND New user is a child (based on Gait) AND New user position is in front of television geometric annotation data}$.
- F. Elderly assistive living: emergency calls or alarm signals to care person is configured in cases, where the elderly person is lying down in areas defined with the geospatially annotated smart objects. In order to realize the context we have the following rule, $Emergency, Contact(X) = \text{User gait is lying position AND User}$

position is Z . Here, X is the contact care person and Z is assigned multiple OR values, such as Floor area of the living room, Washroom floor area, Kitchen floor area, Entrance door floor area, etc. These are the positions, where the person is not supposed to be lying down for a long time.

- G. Other programmable scenarios: In fact, with the 3D smart IoT editor and built-in Interaction editor, the user can program the system to realize many such useful scenarios. For example, the bed room light can be turned OFF/ON when an elderly person or a mother is at the bed, using a hand-up gesture or when a simple voice command is performed. The benefits of many programmable interactions, which are supported in our framework, are presented in the next chapter.

3.6 Summary and Conclusion

In this chapter, we have presented the smart mobile interaction framework. This framework provides implementations for the interaction techniques of touching, pointing, and scanning, as well as context-aware object selection. We illustrate a generic spatial geometric annotation based mobile interaction approach that mediates many of the limitations that are present in the known RFID, NFC, and visual tag based solutions and allows tagging of both virtual and real world multimedia objects to develop different intuitive applications. Since, the geo-spatial annotation mechanism operates at the application level, it is not dependent on the type of interaction devices or on a particular hardware setup. Additionally, the framework provides interfaces for an easy integration of additional interaction implementations with the annotated IoT.

We have presented an algorithm to fuse the IR based motion tracking data and the handheld device's accelerometer data to both approximate the user's handheld device position and calculate the target direction in which a handheld device is pointing. User customizable context-preferences are presented for the scanMe based interaction techniques, to easily incorporate context-aware interactions with the geo-spatially annotated IoT. The smart mobile interactions framework provides many benefits and advantages in areas such as learning, haptic communication, smart environment control, context-aware smart home, elderly assistive home, etc. In the next chapter we discuss the application areas of many of these smart mobile interactions by using the proposed framework. The realizations of the user centric interactions show the importance of geometrically annotated IoT in industry and academia.

Chapter 4

Annotated IoT in Smart Home

4.1 Introduction

We have used the proposed framework to realize pointMe, touchMe, and context-aware interaction with IoT in a smart home. In this chapter, we show three prototypes, where we document the approaches of geo-spatial annotation mapping of the physical things in smart homes. The annotation maps are further used both in physical and virtual environments to realize natural interaction with the annotated IoT. These intuitive interactions are used to gain access to many of the available smart home services. For each prototype, we provide short description of the prototype, describe the implementation challenges, relate the development of the prototype with the proposed framework, and illustrate the achieved results.

4.2 PointMe Interactions with Home Things

In this prototype, a mobile handheld device points from a distance to annotated smart home things for the purpose of interaction. The pointing direction and location is determined based on the fusion of IR camera and accelerometer data, where the IR cameras are used to calculate the 3D position of the mobile handheld device and the accelerometer in the device provides its tilting and orientation information. We have discussed the fusion algorithm, the annotation technique, and the pointMe data measurement details earlier. Here, at first in Section 4.2.1 we discuss the details of the prototype with respect to the proposed smart mobile interaction framework. The annotation of the home things are explained in Section 4.2.2. Lastly, in Section 4.2.3 the realization steps and

key results have been illustrated in detail.

4.2.1 Prototype Description

In this prototype, we extend the proposed framework in order to realize pointMe based interactions with annotated smart home things. The system is suitable in a living room like space, where a number of physical devices e.g. television, audio player, movie player, and air conditioner are located. In order to interact with things in the smart home the mobile users points his/her handheld device towards the things. In Figure 4.1 we detail the components of the prototype and show how it extends the smart mobile interaction framework. In our physical mobile interaction approach, we first estimate the 3D position of the user who is interacting with physical smart home things by using his/her mobile handheld device. Next, we calculate the direction and location in which the user is pointing at by using his/her mobile handheld device. As explained earlier in the framework description, we have placed two IR cameras in the target environment to track the mobile IR emitters. IR emitter node is a small LED that emits IR signal. The signal can be captured by an IR motion tracker camera. IR emitter is connected to a small battery unit to receive power for the LED. The IR emitters are attached with the user's handheld device.

The objective of PointMe type interaction is to point to an object from a distant position and obtain relevant information or services corresponding to that object. We have realized the proposed mechanism for physical mobile interaction in a lab environment. We have integrated the different objects in this environment using a web-service based framework. The various entities that we integrated are lights, fan, television, media player, door lock control, etc. We use X10 to send interaction commands to the connected smart home things and control them. In the scanMe interaction paradigm, a list of object reference that is in close proximity to the mobile handheld device is obtained. In our approach, we prepare such list of the surrounding objects by using spatial geometric query with the current position of the mobile user. In this interaction process, the list of objects is prepared based on the horizontal spatial query by considering 360 degree Filed Of View (FOV) angle. In our approach, we also realize the TouchMe type interaction paradigm by bringing the mobile handheld device very close to a target object. This is possible first by constructing a horizontal spatial query based object search with narrow FOV angle and minimal query distance to select only the nearby objects. A vertical spatial query is used to sort the nearest smart home objects and create an

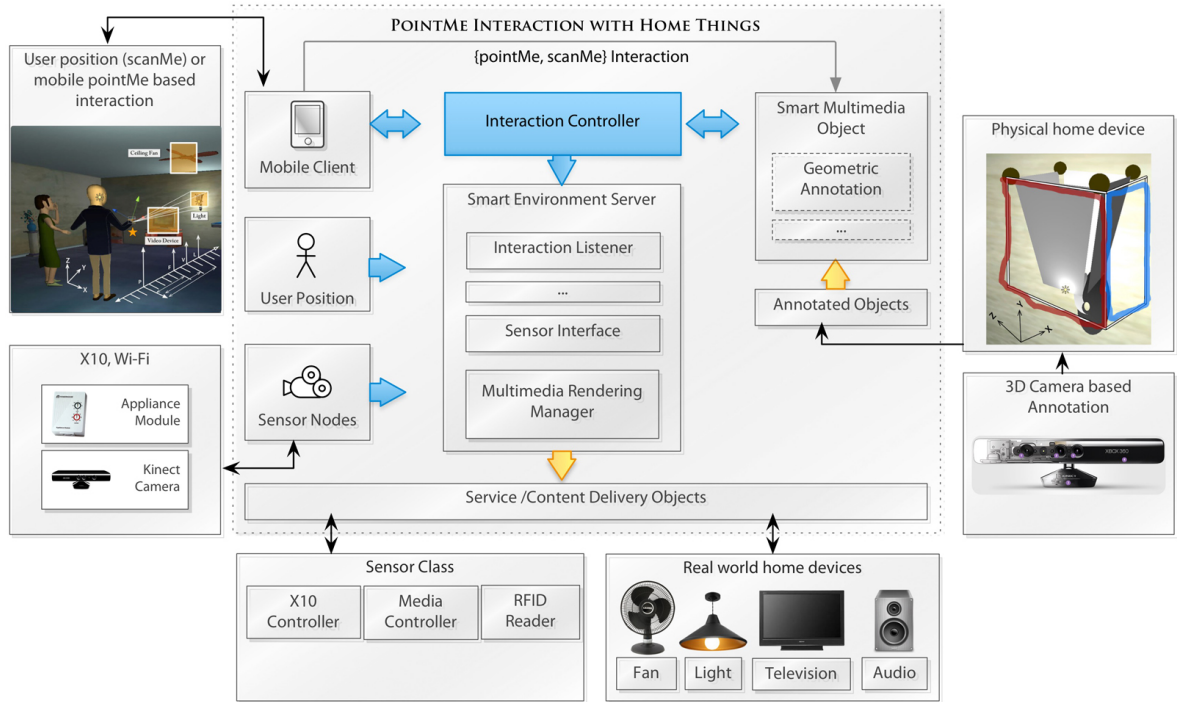


Figure 4.1: PointMe interaction with annotated smart home objects.

interaction event for the Interaction Controller.

We construct the smart multimedia object repository by annotating the smart space with a 3D camera. IR camera based annotation can also be incorporated in this prototype. We extend the Mobile client class, User Position class to realize the handheld device client and IR based user mobile position detection algorithms. We extend the class of the Multimedia Rendering Manager to realize X10 controller based interface for the Fan and Light devices. We extend Multimedia Rendering Manager class to realize Media controller for the Television and Audio devices.

4.2.2 Annotation of IoT

Based on the 3D position estimation methodology, we introduce a spatial-geometric approach to annotate the physical things on the specification of geometric coordinates of approximate boundary (voxels) of an object, which is facilitated by using two IR cameras. While annotating, in order to capture the motion path and formulate shape vector for the target objects, we have used our previous motion point to symbol mapping scheme [98]. Using that approach, we have annotated an object in both XY and YZ plane as

shown in Figure 4.3. The geometric annotations are voxel based and uses the IR camera reference points. Each of the geometric annotations are at least 1 – 3 feet in each 3D directions so that pointing and other form of mobile interaction becomes easy. Based on these voxel based geometric annotation, a physical environment device map is produced. Afterwards, each annotation data is stored in the database and are associated with information/services, which are accessed when interacting with that annotated thing.

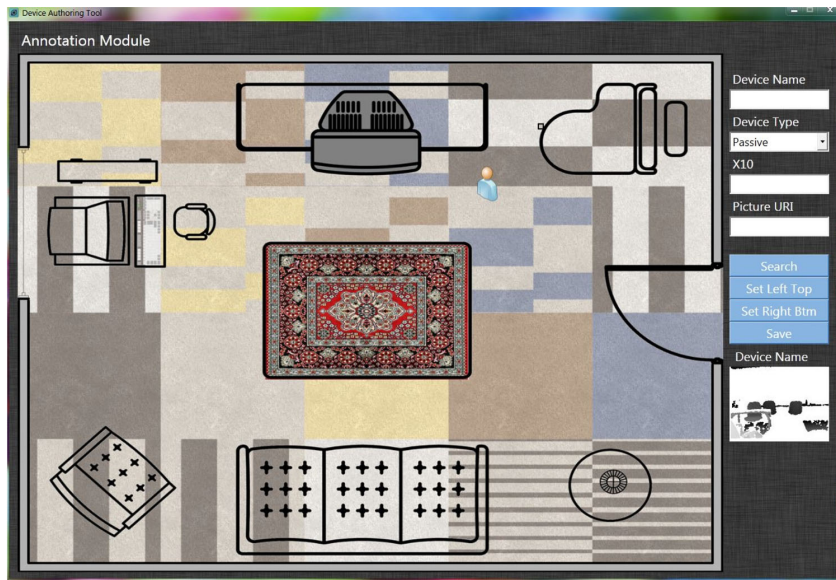


Figure 4.2: 3D physical objects annotation approach.

We have also developed a 3D physical object annotation editor as shown in Figure 4.2. Here, the objects are annotated in the XY plane. The 3D image captured from the camera is used for the purpose of annotation. The elevation data of the annotated objects are obtained from the 3D camera. While annotating the 3D things, the user only maps the things in 2D space by using a colour image. The 3D elevation data are computed in the background by the system to construct the voxels of the things.

4.2.3 Realization and Measurements

In our realization of the system, we have placed two IR cameras that are aligned in perpendicular to each other. The cameras are attached to the room wall at fixed positions. We have used the HTC touch diamond mobile phone with .net compact framework support to realize the Mobile client. The mobile handheld device is also equipped with an accelerometer. An IR emitter node is attached with the mobile phone and a switch to

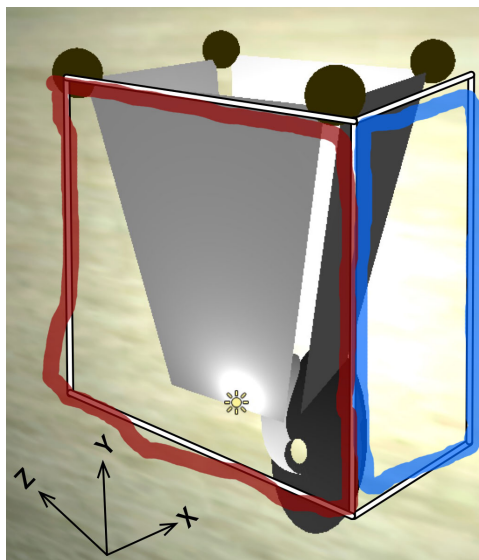


Figure 4.3: Annotation process of an object (Lamp).

turn it on/off is attached on the back of the device. Figure 4.1 shows the phone and the IR emitter equipment. The development platform is visual studio 2008 .net and C# is the chosen programming language. We have illustrated in Figure 4.4 the overall workflow of the system by labeling different steps. First, the IR cameras installed in the environment capture the location of the mobile device when pointing to a target physical object (step 1). The IR capture interface in an IR motion receiver obtains the 2D IR motion points relative to the camera's orthogonal origin space. Similarly, for different IR receiver, multiple motion points respective to their origin space are calculated and are sent to the fusion process (step 2). The individual 2D location points of different IR cameras are fused and a 3D position is estimated, which is transferred for further aggregation (step 3). The accelerometer reading obtained from the mobile device is aggregated with the IR based 3D location estimates and the resultant 3D location and tilt data is transferred to the object selection component (step 4 and step 5). Based on the 3D position and tilt data a FOV is calculated with respect to the current mobile position and pointing direction. The calculate FOV is used to generate the spatial query to identify the physical target objects and their corresponding information/service specifications are sent to the feedback generation component (step 6). At last, a suitable GUI interface representing the generated feedback is presented to the user's mobile screen (step 7).

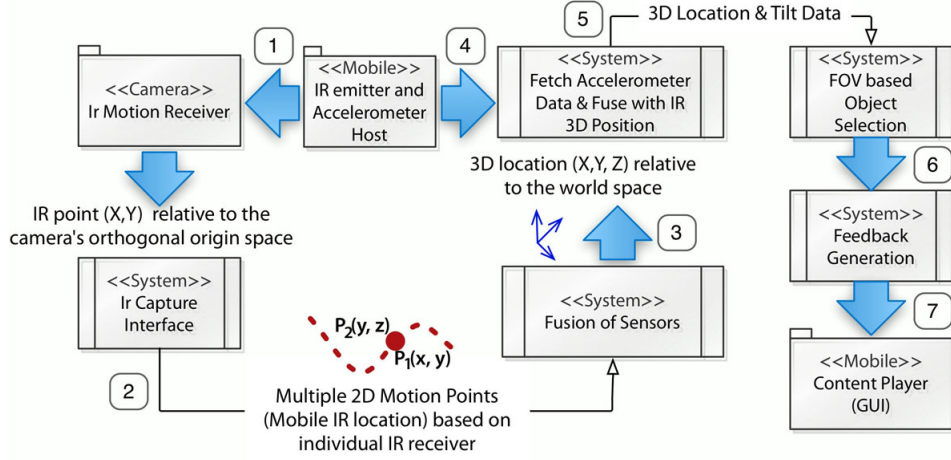


Figure 4.4: The overall system components that works by fusing the 2D IR emitter locations into a 3D point and then applying accelerometer parameters in order to determine the user's point of interest.

Correctness of User's Mobile Position and Orientation

In order to estimate the correctness of the user's mobile position and orientation calculation methodologies, we devised the following experiment. Inside the lab environment, we defined three physical locations namely A , B , and C . We marked those locations with tape and paper and labeled them for easy identification. Afterwards, we calculated their approximated 3D locations (in cm) with respect to a physical environment reference node $K(x, y, z)$. The reference node in our lab environment was the west-north corner of the room. In order to transfer the approximated 3D coordinates from the room coordinate space to the IR camera coordinate space, we performed firstly translation of the origin (reference node). The translation vector $T_x = (tx, ty, tz)$ is found by computing the difference between the physical environment reference node, $K(x, y, z)$ and the IR camera reference node $L = (\alpha_{IR1}^\lambda, \beta_{IR1}^\lambda, \gamma_{IR2}^\lambda)$. Hence, we performed translation operation, $(\hat{x}, \hat{y}, \hat{z}) = (x + X, y + Y, z + Z)$ and determined the 3D coordinates of A , B , and C with respect to L . We renamed these three coordinates as A_L, B_L , and C_L respectively. The coordinates are scaled with respect to the physical environment and not the IR camera environment. Therefore, appropriate scaling parameters are needed in order to determine the latter. The scaling variable, $M(mx, my, mz)$ was measured based on the parameters of the room space and the camera space. This is to be noted that the camera coordinate space is smaller than that of a room.

We determined the Cartesian distances of the measured A_L, B_L , and C_L . Three Cartesian distances are calculated, namely d_{AB}, d_{BC}, d_{CA} that represent Cartesian distances between A_L and B_L , B_L and C_L , A_L and C_L respectively. The 3D distances between A_L and B_L and B_L and A_L will result the same values. The Cartesian distance, d between two 3D coordinates $P(x_1, y_1, z_1)$ and $Q(x_2, y_2, z_2)$ is given by, $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$ [42].

We positioned the three IR emitter nodes at the physical coordinate places of A, B , and C respectively and determined their 3D coordinates IR_A, IR_B , and IR_C respectively by using the above approach. Further, we measured the Cartesian distances of those coordinates. Three Cartesian distances are calculated, namely d_{ab}, d_{bc}, d_{ca} that represent Cartesian distances between IR_A and IR_B , IR_B and IR_C , IR_A and IR_C respectively. Afterwards, the ratio of the Cartesian distances $d_{ab} : d_{AB}$, $d_{bc} : d_{BC}$, and $d_{ca} : d_{CA}$ are compared. The average of the ratios are used to obtain the scaling parameter, M . Each ratio of the Cartesian distances resulted seemingly identical values that signifies the successful camera space conversion of the physical nodes A, B , and C . We performed scaling operations $(\hat{x}, \hat{y}, \hat{z}) = (mx, my, mz)$ on the 3D coordinates of A_L, B_L , and C_L . Additionally, we calculated the 3D coordinates of the triangular path between these three points and store the path information in the database. The 3D triangular path is depicted in the following Figure 4.5.

By using a mobile IR node, further we recorded the 3D path coordinates. A single IR node was moved between the physical points C and A , then A and B and lastly between B and C . We performed 4 sampling of the IR triangular path traversal. In each sampling, we measured the 3D coordinates of the moving IR node by using the proposed approach at a constant time interval and recorded those information in the database. Each of these samples were also plotted in Figure 4.5. The calculated 3D sampled paths resembled that of the original physical 3D triangular path and proved that the IR camera based 3D coordinate calculation could be used to obtain the original physical coordinates of the sampled IR node. However, we found that the average of the calculated 3D sampled paths differs slightly. As evident from the depiction, the difference is because the physical environment space and the IR camera space needed a rotation transformation. These rotation variables are depicted in the Figure 4.5.

The orientation calculation algorithm is based on the 3D position of the IR node and the accelerometer data of the mobile device. We considered a gyro-meter assisted accelerometer of the mobile phone to obtain the orientation of the mobile user with respect to the IR cameras that will be used to calculate the projected screen coordinates

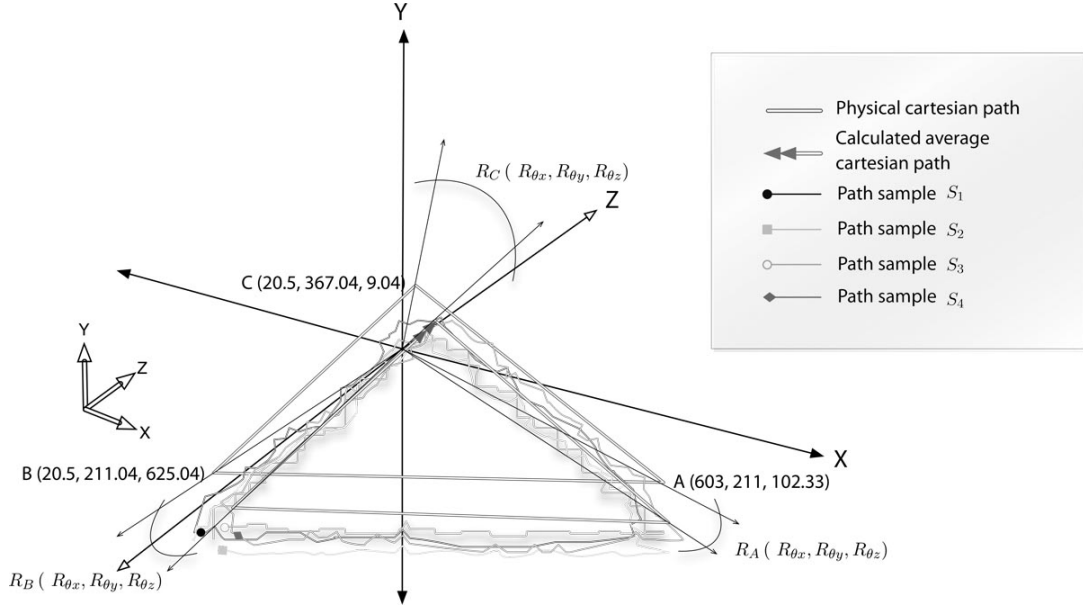


Figure 4.5: Triangular IR path sampling and rotation correction using the cartesian distances.

at the virtual screen. As recorded during our experiment, when the mobile device is facing a complete horizontal left while facing eastward then the accelerometer data obtained was $(X_{raw}, Y_{raw}, Z_{raw}) = (0, 0, 1.04)$. Similarly, for the complete horizontal right, the accelerometer reading was $(X_{raw}, Y_{raw}, Z_{raw}) = (0, 0, -1.04)$. Moreover, when the mobile phone was oriented to a complete vertical up then the accelerometer data was $(X_{raw}, Y_{raw}, Z_{raw}) = (0, -1.04, -0.04)$ and for the complete vertical down the reading was, $(X_{raw}, Y_{raw}, Z_{raw}) = (0.04, 1, 0)$. The other accelerometer data and the calculated virtual screen coordinates by the proposed algorithm are given in the following Table 4.1.

Therefore, we were able to determine the 3D coordinates of the mobile IR node and its orientation by using the proposed algorithms. The 3D coordinates were used to calculate the user's mobile position in the physical environment and the orientation information is used to create a FOV based selection of the geometric target objects. Different metrics that influence the selections of the target objects are illustrated in detail in the following.

Selection Accuracy Parameters

We set up experiments with ten participants to determine the ease of selection, selection accuracy and selection performance of target objects inside the indoor environment.

Table 4.1: Accelerometer raw data based target analysis

Elevation	Orientation	X_{raw}	Y_{raw}	Z_{raw}	$G(x, y)$
0 ~ 10	V <i>up</i>	-0.04	-0.2	1.04	305.8, 191.33
0 ~ 10	V <i>down</i>	-0.08	0.28	0.96	294.03, 303.05
10 ~ 30	V <i>up</i>	0.0	-0.28	1	318.96, 169.54
10 ~ 30	V <i>down</i>	-0.04	0.4	0.92	305.58, 330.0
0 ~ 10	H <i>left</i>	-0.28	0.0	0.96	228.76, 239.1
0 ~ 10	H <i>right</i>	0.24	0.0	1	393.35, 239.11
10 ~ 30	H <i>left</i>	-0.4	0.0	0.92	186.02, 238.7
10 ~ 30	H <i>right</i>	0.4	0.0	0.96	445.89, 239.2

Each experiment targets to analyze whether certain parameters such as distance of the user from the target object affects user’s selection performance. Each user were properly introduced to the system and briefed about the purpose of the experiment at the beginning. They were shown video clips of some usage scenarios of the system and a dry run was conducted before they start with their assigned tasks. Selections of the target objects were achieved by obtaining the associated mapped information of the object. The user was given the option to choose suitable places inside the room boundary before pointing and pressing the activate button (IR ON/OFF switch) in his/her mobile device. Afterwards, the user is asked to point to another object, which continues until all the objects for the intended experiments were selected.

In each attempt of the user, one facilitator took notes and recorded whether the user received correct feedback from the system regarding the target physical object and if it was a successful endeavour. In the following section, we provide details of our experiments. In Section 4.2.3, Section 4.2.3 and Section 4.2.3, we evaluated whether the object arrangement affects the user’s selection performance based on random, row and columnar object arrangement respectively.

Random Target Object Position

In this experiment, the participating users were asked to point and select physical target objects, which were arranged without necessarily following any order. The distances among the target objects are kept 4 – 7 feet. As discussed earlier, each of the geometric target objects are at least 1 – 3 feet 3D boxes (voxel) that sometime exceeds the actual

physical object's size but the chosen geometric volume boundary provides flexibility in the selection process. Empirically, 7 distances from the physical objects and 7 viewing FOV angles were selected for the experiments. These distance and FOV angle choices are marked in the figures. For each distances and FOV angle settings, the user has performed the selection tasks. From their task analysis that is depicted in Figure 4.6, we found that with the increase in distances, the selection tasks of the user became more challenging. For example, when the users are at 6 feet distance, it is very easy for them to select the objects, as inside the FOV angle there are only a few physical objects along with the target objects. Hence, selection becomes easy and more accurate. As the distance increases, number of target objects increases inside the FOV angle and accurate selection requires more effort. On an average, as depicted by the plot line, users were more precise within the 6 – 20 feet boundary and in this boundary their selection accuracy is more than 81%. Also, as the physical mobile interaction occurred in indoor space settings, the accuracy inside this 20 feet boundary may be considered satisfactory.

For randomly organized target object selection, FOV angle of the mobile viewpoint plays a very important role. As depicted in Figure 4.6, we found that between 40 and 90 degree FOV angle the physical object selection accuracy is comparatively higher. Inside this angle boundary, the selection accuracy of the users is 88%, which we found acceptable during the routine interactions and exploration of our system.

Target Object Arranged in Row Fashion

In this series of experiments, the arrangement of the target physical objects was changed compared to the previous experiment. The target objects were placed in row fashion so that at a particular viewpoint inside the FOV angle there will be fewer targets. The distances among the target objects are kept 4 – 7 feet. As discussed earlier, each of the geometric target objects are at least 1 – 3 feet 3D boxes (voxel). This arrangement of the physical objects also mimics the practical arrangement inside our homes; hence, this arrangement has more significance. The result analysis of this study is depicted in Figure 4.7. As can be seen, in this arrangement when the user is closer to the target physical objects, the accuracy of the number of object selection is comparatively higher than what we found in earlier experiments. Close distance object selection often reached maximum possible accuracy; this is due to the fact that in row fashion arrangement in close distance there will be no more than two objects in the FOV of the user. This also depends on the dimension of the target objects. In between the distance range 6 – 32 feet if we aggregate the accurate number of object selections, we can see that the accuracy rate is just above

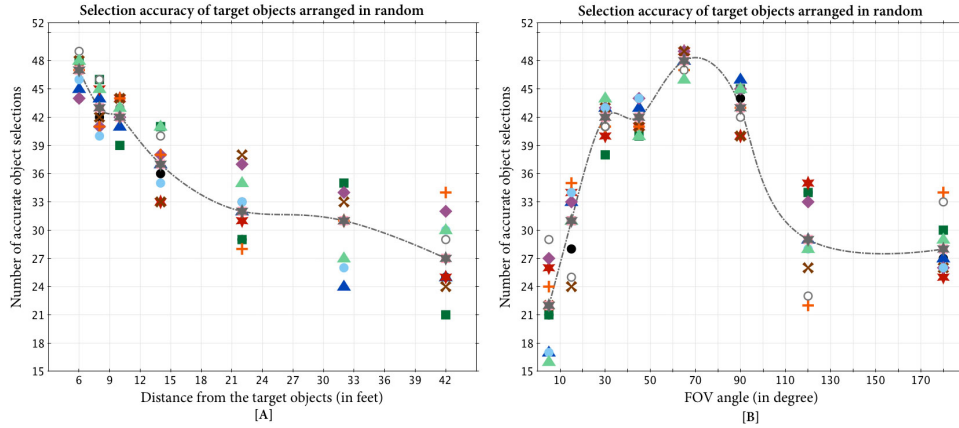


Figure 4.6: Number of accurate object selections of the user at varying distance (6 – 42) feet as depicted in [A] and at different FOV angle ranging from (5 – 180), shown in [B]. The 50 voxel based target objects were placed in random order in the environment. Each of the sampling points represents a trial by the respective users at the said arrangements. E.g., the blue triangles in [A] shows the selection accuracy achieved by the subject 4 who was standing 6, 8, 10, 14, 22, 32, and 42 feet away from the randomly arranged targets in the said evaluation.

90%. In the same arrangement, correct object selection is also dominated by the choice of FOV angle for the mobile viewpoint. However, less FOV angle is still inappropriate to achieve better accuracy. As depicted in Figure 4.7, the adequate choice for the FOV angle that results in more number of accurate object selections is between 45 and 100 degree. In that FOV angle range, the accuracy for physical object selection is 94%. This finding is similar to our previous experiment but the improvement found is largely due to the arrangement of the target objects in row fashion.

Target Object Arranged in Columnar Fashion

In order to examine whether columnar target object arrangement affect the accuracy of object selection, we set up another series of experiments. Here, the object map is used for the physical objects where they are placed in a columnar fashion and asked the user to select them. We observed noticeable change in the number of accurate object selection. As shown in Figure 4.8, we have found that between 6 – 12 feet distance from the physical objects, the average accuracy of selection is 79%, which is comparatively less than that of arrangement in row fashion. This reduced performance arises due to

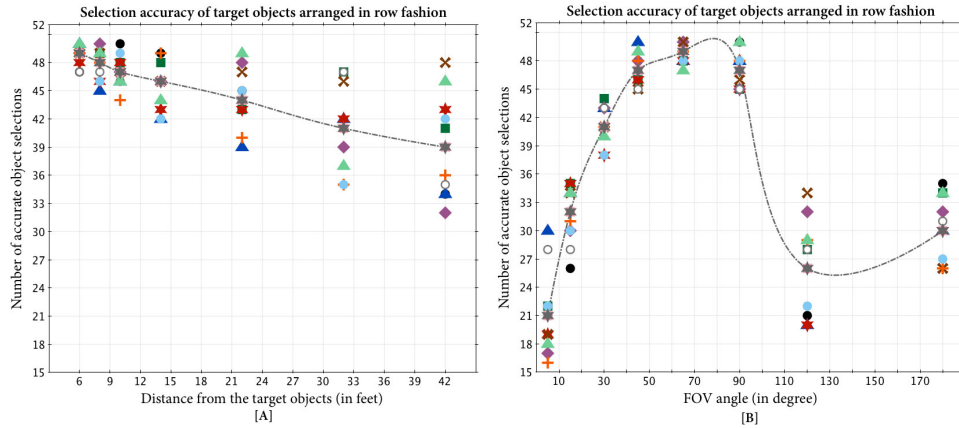


Figure 4.7: Number of accurate object selections of the user at varying distance (6 – 42) feet as depicted in [A] and at different FOV angle ranging from (5 – 180), shown in [B]. The 50 voxel based target objects were arranged in Row Fashion. Each of the sampling points represents a trial by the respective users at the said arrangements and the curve represents their averages. e.g., the blue triangles in [A] shows the selection accuracy achieved by the subject 4 who was standing 6, 8, 10, 14, 22, 32, and 42 feet away from the targets in the said evaluation.

the fact that at any moment inside the FOV angle multiple objects are considered for selection.

Similarly, in the most accurate FOV angles in columnar object arrangement, we see less accurate selections. We have observed that between 40 and 90 degree FOV angle the physical object selection accuracy is comparatively less accurate. Within this angle boundary, the selection accuracy of the users is 80%. Hence, even though the users have found it easy to search targets with wide FOV angle, there is a point where clear selection becomes difficult as the number of objects are arranged in columnar fashion.

Usability Evaluation

The participating users in previous experiments were told to fill out a questionnaire and to rate some given assertions according to their usage experience of the system in Likert scale [79]. The average responses from the users are depicted in percentage form. In Figure 4.9, assertions where the users responded positively are shown. For example, 34% of the participating users acknowledged that in order to familiarize with the system probably it is a better idea to provide training to the users. They mentioned that

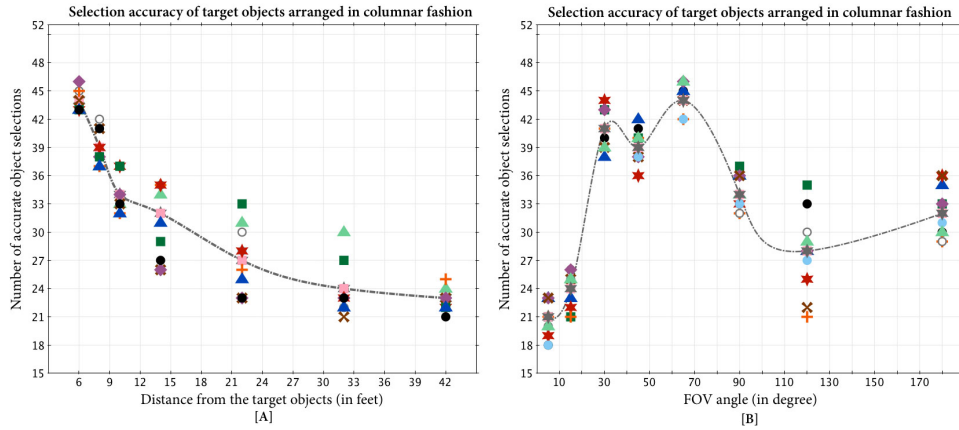


Figure 4.8: Number of accurate object selections of the user at varying distance (6 – 42) feet as depicted in [A] and at different FOV angle ranging from (5 – 180), shown in [B]. The 50 voxel based target objects were arranged in columnar fashion. Each of the sampling points represents a trial by the respective users at the said arrangements and the curve represents their averages. e.g., the blue triangles in [A] shows the selection accuracy achieved by the subject 4 who was standing 6, 8, 10, 14, 22, 32, and 42 feet away from the targets in the said evaluation.

without the introductory dry runs of the system at the beginning of their experiments, the selection performance could have been more inaccurate. Some users also explained that video demonstration of the usage were helpful to them and provided insight to the use of mobile device in physical object selection.

In our experiment, we found the proposed interaction style to be user friendly and easy to get familiar with. Regarding age, we observed that the elderly people had more difficulty in using the system than the young people, as the young were very familiar in using mobile phone and many of the existing mobile phone based services. However, one particular observation we had is that the young, once trained in using the system, were frequently trying to use the developed interaction mechanism to control household appliances, such as controlling a lamp. Finally, based on their test experiences, the users filled out a questionnaire where they were requested to provide ratings of their likeliness, familiarity, ease of usage etc of the system.

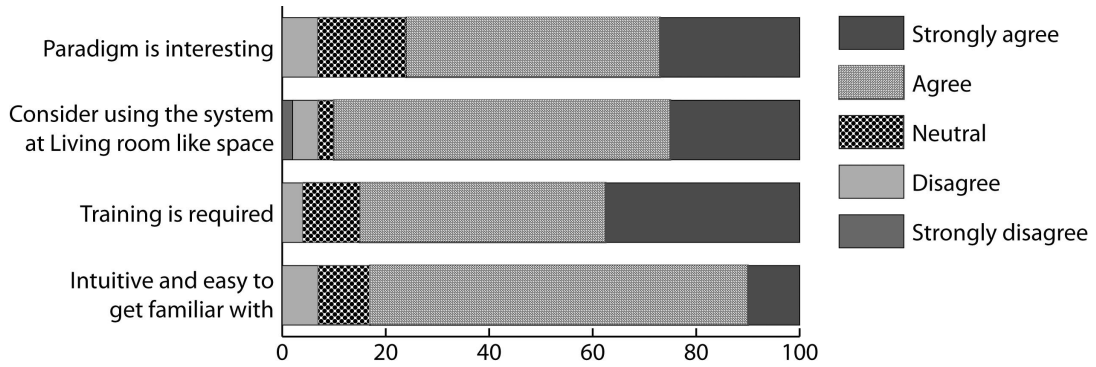


Figure 4.9: Usability study with the mobile interaction system. The participating users responded to the listed assertions and rated those in Likert scale. The average of their ratings are shown in percentage.

	Mean	Std. dev.	Mean Percentage
Acceptability	4	0.7303	80%
Device Feedback	2.6875	1.07	53.75 %
Likeness	3.875	0.806	77.5%
Delay	3.78	1.31	77.35%
Ease of Use	4.25	1	85%

Table 4.2: User satisfaction on the overall evaluation in Likert scale.

4.3 TouchMe Interactions with Home Things

Researchers have successfully developed realistic model for real world things/phenomena and then have simulated that in the virtual world. In this prototype, we attempt to instantiate virtual world events into the real world. The interactive 3D virtual environment provides a useful and realistic 3D world that resembles objects/phenomena of a real world, but it has limited capability in communicating with the physical environment. Now a days, in smart spaces, a range of wired and wireless sensory devices such as temperature, humidity, and pressure sensors are available. By using these sensory data, it is possible to automate and control various physical devices to create a soothing environment. The prices of these sensors and automation devices (such as X10[150] and *Wi-Fi*) are decreasing. At the same time, their utility in providing control and various entertainment facilities in smart spaces are making them increasingly popular. A graphical user interface (GUI), which is a medium with which to interact with programs, plays an important role in controlling physical space elements like smart homes.

We explore the possibilities of integrating real world object interactions with Linden Lab's multiuser online 3D virtual world, Second Life. We enhance the open source Second Life viewer client in order to facilitate communications between the real and virtual world. Moreover, we analyze the suitability of such an approach in terms of user perception, intuition and other common parameters. Here, at first in Section 4.3.1 we discuss the details of the prototype with respect to the proposed smart mobile interaction framework. The annotation of the home things are explained in Section 4.3.2. Lastly, in Section 4.3.3 the realization steps and key results have been illustrated in detail.

4.3.1 Prototype Description

Controlling a physical space by using the traditional icon-based GUI and limited mobile screen space might not be sufficient for effective management of a range of smart physical devices spread randomly across the physical space. In order to efficiently control different intelligent physical devices, researchers have focused their attentions on the intuitive GUI design and explored its usability issues [17] [112]. In our research, we leverage the advantages of 3D virtual interfaces, propose a framework to control and visualize the real sensory data from the smart physical environment. In the physical home environment, different devices and sensors are connected in order to ensure a safe and automated home. Any event that occurs in the physical space of the smart home can be synchronized with the virtual environment. We annotate 3D virtual devices in Second Life and define events

that would trigger a real device ON/OFF.

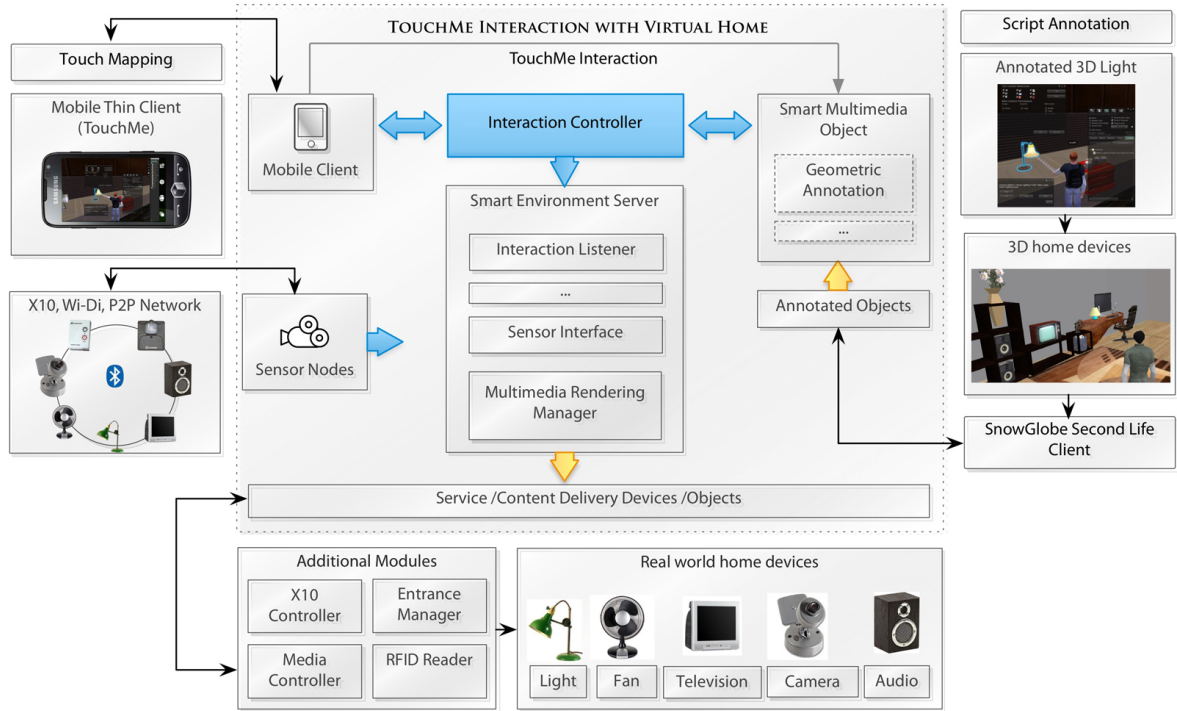


Figure 4.10: TouchMe interaction with annotated smart home things.

We extend the proposed framework to realize touchMe based interactions with annotated virtual things and bridge communications with physical objects. We extend the Mobile Client class to implement a THIN mobile client that renders a 3D virtual environment by using remote computer resources. Currently the mobile devices lack the graphics horsepower to render SL's 3D environment in a satisfactory way. SL uses gigabytes of textures, sounds and animation data that entail a huge amount of network download time and CPU processing time that can adversely affect frame rate, draw distance, render detail, and mostly the battery life. Therefore, in order to render SL in mobile devices we adopt virtual display technique that incorporates device collaboration [132][110], where most computations are carried out remotely by leveraging remote computing resources, and screen updates are compressed and transmitted together with low-level display commands from servers to clients [126]. We realize X10, Media Controller, RFID Reader, and Entrance Manager class from Multimedia stub class. Later, we have registered those classes to the Multimedia Rendering Manager. The components of the framework are shown in Figure 4.10. We have created 3D virtual home in Second Life that resembles a physical smart home and have annotated virtual 3D objects inside

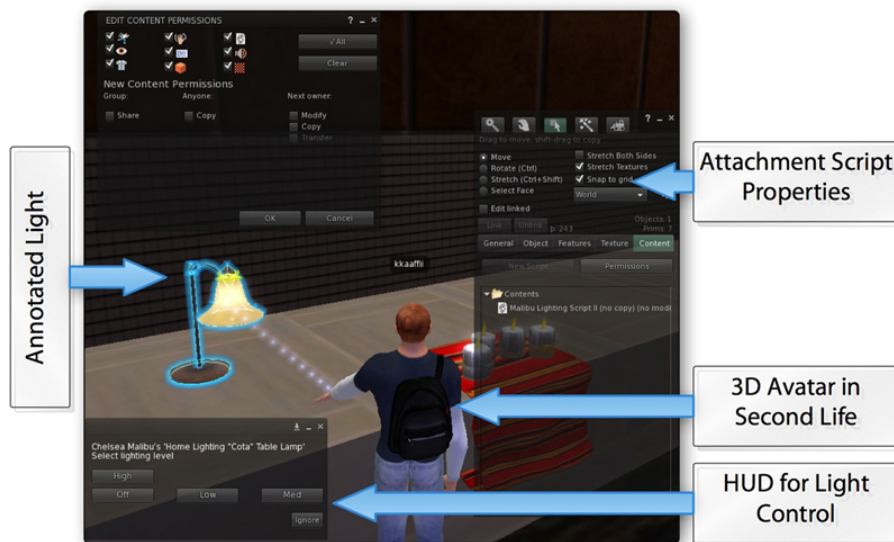


Figure 4.11: The flexible object annotation scheme allows the user to annotate objects and to set rules that create events to render real world services.

the virtual home. By using the Second Life virtual interface, a home owner can have a better overview to monitor and control the home appliances as the virtual home interface provides the option to control the physical smart devices.

4.3.2 Annotation of IoT

Annotation mechanism is one of the primary parts of our proposed system. We can create *XML* based rule and can attach the rules to the virtual 3D objects at a sub-region level. As the users interact with the annotated objects, the system will create events based on the attached rules. The 3D objects are annotated using the script annotation mechanism that is demonstrated in Figure 4.11. In this figure, we geometrically annotated a 3D virtual lamp by storing a script file with the annotated objects that is recognized by the system.

The annotation file is an *XML* file that specifies the animation unique identifier (*UUID*), the animation file (*Biovision Hierarchy - BVH*), the speed of animation, duration of animation etc. Each annotated object contains its respective animation file that we attach during the authoring process. For example, the 3D door object is provided with “doorOpen” and “doorClose” animation properties. Similarly, the 3D virtual fan object contains rotation of its blades as part of its animation. These interaction ani-

```

<?xml version="1.0"encoding="utf-8"?>
<AnnotationRules>
  <animationModules >
    <animationModule id="doorOpen1">
      <UUID>6b61c8r8-6878-0h87-12hg-j1hj3h4us8g9</UUID>
      <animationBVH>doorOpen.bvh</animationBVH>
      <animationLooped>NO</animationLooped>
      <animationSpeed>30</animationSpeed>
      <animationSpeed>LOW</animationSpeed>
      <animationScaleTo>1</animationScaleTo>
    </animationModule>
    <animationModule id="doorClose1">
      <UUID>6f51c8r8-4555-0h87-89dw-a02j4b49sk2b</UUID>
      <animationBVH>doorClose.bvh</animationBVH>
      <animationLooped>NO</animationLooped>
      <animationSpeed>30</animationSpeed>
      <animationSpeed>LOW</animationSpeed>
      <animationScaleTo>1</animationScaleTo>
    </animationModule></animationModules>
  <Appliances>
    <Appliance name="Main Door 1">
      <ApplianceType>Door</ApplianceType>
      <DeviceType>X10</DeviceType>
      <PhysicalAddress>A1</PhysicalAddress>
    </Appliance>
  </Appliances>
</AnnotationRules>

```

Figure 4.12: An overview of the target object specific interaction rules stored (and could be share) in an *XML* file.

mations of the 3D virtual objects make the interaction more fluid and attractive to the users. This file also specifies the physical device specific data like device address, name, type etc. In Figure 4.12 depicting a sample *XML* specification that demonstrates a door open close scenario where two animations “doorOpen1” and “doorClose1” are specified for open/close the real door that is connected with an *X10* appliance module. Here the device address is also specified which is “A1” in this case. In the annotation phase we specify the LSL scripts [94] using the Second Life viewer obtains control of the objects and performs read/write operation into its communication channel.

In order to get refined response and to provide high flexibility to control the physical objects, our annotation scheme introduces a generic interface. This is especially important for those devices in which different parts of a virtual device need different annotation rules. For example, a table fan have different switch for speed control whereas a lamp has different illumination intensity levels or a temperature sensor have different temperature levels. In order to provide this type of facilities we developed a suitable annotation mechanism that allows annotation into different parts of the 3D virtual object. In this manner, we specify rules in the *XML* data that we need to attach with specific region of the object. In our prototype system all the events are controlled by predefined names

such as: light, door, fan, and music player.

4.3.3 Realization and Measurements

Our system can be used in virtual object-based home automation systems, where multiple devices are seamlessly connected through an event-driven architecture. In order to implement this scenario a virtual home is created using Second Life, which mimics the layout of a physical home. The system is comprised of three primary components that are Second Life Viewer plug-in, home automation web services and the point-to-point physical connection between services to the wireless and/or wired devices. In order to communicate with the core part of the second life client program called *viewer* we have developed a Second Life interaction event handler as an add-on to the viewer. This coupling architecture provides the option to listen to the communication channel of the Second Life without affecting the functionality of the core system. Furthermore, the event controller interacts with the other parts of the home automation system by using Simple Object Access Protocol (*SOAP*). The event handler performs actions by using text based messaging protocol. A message contains event trigger data, animation data or simple communication data. The Message Transmission Module captures all the messages that are generated in the core section of the Second Life. When through the avatar the user issues events in the 3D environment, e.g., *a touchMe event to open a door*, the message transmission module captures those events and transfers the event messages to the nearby interaction event handler for processing.

We have incorporated Microsoft Visual Studio 2005 IDE to develop our system, and the primary language used was Visual C++. We adopted Microsoft Foundation Class (MFC) library and an asynchronous socket-programming scheme to create a socket-based secure communication channel. We have used CMake for building and listening to the communication protocols of the open source version of the Second Life viewer. We have implemented various other components of the framework on top of this viewer system.

Usability Evaluations

We have incorporated the usability evaluation guidelines [28] [131] to qualitatively measure our proposed system. We have designed our tests accordingly with the sensory analysis [73] of the system involving both the user and the targeted sensory communication modules. Before performing the usability tests, we have designed a test plan, where we define our evaluation objectives, develop questions for the participants, identify the

measurement criterion and decide upon the target users of the system. The tests take place at our university laboratory with sixteen (16) participants comprising of different age groups. Five (5) of the participants are in age group 13-18, eight (8) of them are in age group 18-36 and the rest three (3) are in age group 36+.

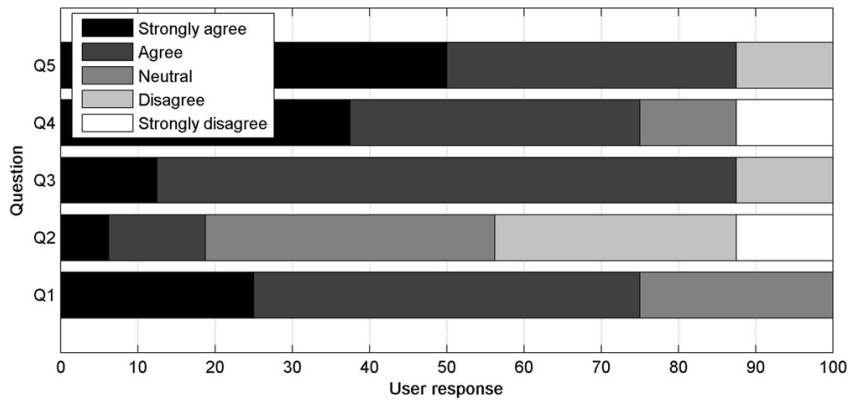


Figure 4.13: User response for the usability test questions that are listed in Table 4.3.

Table 4.3: Usability test questions to the user for smart home interaction.

#	Question
Q1	Easy to get familiar with
Q2	The system responded promptly to the virtual interactions
Q3	The device state animations (ON/OFF) were helpful to the overall management of the smart home
Q4	Users were immersed into the system while using it

In a user's test machine the enhanced Second Life viewer with the add-on is installed to provide animation and GUI based interactions. We have annotated two fans, three lights, and one door object in the Second Life virtual home. Those objects are installed into two separate virtual rooms. At a time, the selected volunteers are told to interact different physical devices through the annotated virtual objects. Their activities are monitored throughout the experiment and are recorded for analysis. Finally, based on their test experiences the users filled out a questionnaire, where they are requested to provide ratings of their likeliness, familiarity, and ease of usage of the system.

Table 4.4: User satisfaction on the overall evaluation in Likert scale.

	Mean	Std. Dev.	Mean Percentage
Easy to Learn	3.66	1.09	73.12%
System Response	3.72	0.77	74.37%
Device States	3.97	0.862	79.38%
Immersion	3.94	1.16	78.75%

The user responses are shown in Likert Scale [79] in Figure 4.13. The ratings of the questionnaire are in the range of 1-5 (the higher the rating, the greater is the satisfaction). The average of the responses of the users are calculated in percentage form and are measured after the usability tests. Figure 4.13 shows the user’s responses for each given assertions. We have found that more than 80% of the users would like to use the enhanced system if they were available in Second Life. Overall, 75% of the users are also satisfied with the synchronized animation and real world responses of the physical objects. Table 4.4 summarizes the overall performance score of the users. The higher mean values of system response, device feedbacks, and Easy to familiar represent a very satisfactory user response, while the moderate mean values of system in Second Life and the perceived delay show relatively good user satisfaction.

In our study, we also have attempted to evaluate the acceptability of the system by the users from different genders, age groups and technical backgrounds. The results of these studies are depicted in Figure 4.14. In case of different age groups, we have divided the users into three age groups, namely *group – 1*: ages 13 – 18, *group – 2*: ages 18 – 36, and *group – 3*: ages 36+ and have recorded their responses in Figure 4.14 (a). We conclude that compared to the older group of users, the users from the younger group seem to be more comfortable in using the system. Also from Figure 4.14 (b) we have received favourable responses and recommendations from users with non-technical background than that of the technical people. The users also have affirmed that should the system become available in Second Life they might consider using it in their management of Smart Homes.

4.4 Context-aware Interactions with Home Things

Natural interaction with devices frees people from working in a desktop-like setting and provides intuitiveness in accessing various services of interest [67]. For example, when

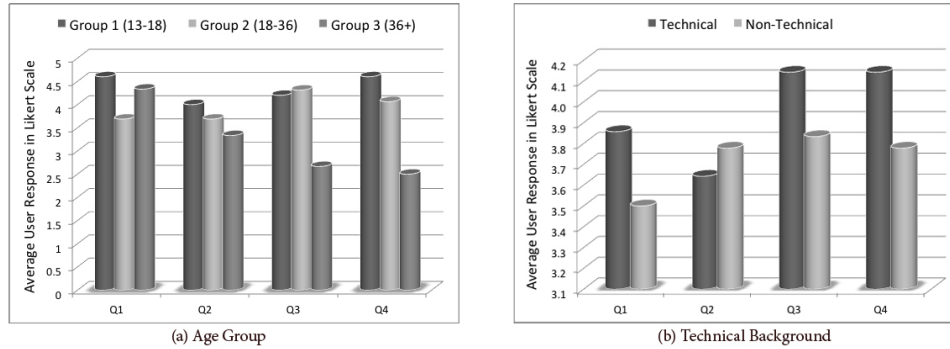


Figure 4.14: Comparison between the responses of users from different (a) age groups and (b) technical backgrounds.

reading a newspaper, a person might want to turn on the light or the media player while sitting on a couch. Users need support to get access to smart home anywhere and anytime. This is of special interest in a smart home environment [26], where various sensors and services are available in the environment to support human activities. Intelligent systems promise to do more in terms of automating tasks and increasing awareness by inferring the states of users and environments (things). However, knowledge of the location of the users and the smart home things are prerequisite for the realization of context-aware applications.

We argue that given the sensory data, it is possible to define useful context scenarios with respect to annotated IoT. We relate the sensory context parameters with the geo-annotated IoT to obtain realistic contextual rules. A context rule also acts as geo-spatial annotation to the smart environment. We have incorporated the proposed framework to realize a prototype that implements many of the context rules that we have presented.

Context-aware systems are in nature ad-hoc based. They are developed by incorporating a particular hardware based platform that supports a particular way of interactions with things. Similarly the location aware systems are ad-hoc based and do not follow a particular framework or guidelines. The proposed geometric annotation based thesis presents a framework that allows the use of annotated things to program context-aware applications. To illustrate the system, first in Section 4.4.1 we discuss the details of the prototype with respect to the proposed smart mobile interaction framework. The annotation of the home things are explained in Section 4.4.2. Lastly, in Section 4.4.3 the realization steps and key results have been illustrated in detail.

4.4.1 Prototype Description

Knowledge of the users location in the smart environment is prerequisite to any context-aware applications. In our prototype we have ensured that at the point of interests users location is available to the system. Unlike outdoor applications where GPS location data is present, we have developed our own location determination system. We have extended the proposed framework to realize the context-aware interactions in a smart environment. We have used 3D Kinect camera to calculate users 3D position in real-time inside the smart environment. However, the location data is available only around the IoT where the interactions take place. We term these collections of IoT annotation data for a particular location as a Scene. Hence, the location data of the users are always determined with respect to a Scene. Hence, in our context-interactions based prototype, we have created an annotation map of all the IoT that we want to control.

Smart homes are technologically augmented spaces where several interconnected devices; artifacts and other services are available to support people. The lighting control service, media service, and security service are few examples of services a smart home application incorporates. Interaction with such an application occurs either implicitly, where the environment automatically invokes context-aware services, or explicitly, where people select services of their interest through several interaction devices such as remote control and cell phone. However, unlike existing techniques, gesture based interaction, as a mode of explicit interaction is more natural and appealing to people while accessing various services. In automated context-aware interactions, the users do not require to confirm his/her interaction commands. However, in a smart home environment, the users may want to use gestures and associate a gesture with an interaction commands. In such cases, for a large number of IoT interaction commands, we want to avoid gesture overload by using our proposed context-aware interactions approach. In our usability experiment we have found that associating individual services/commands with individual gestures increases users cognitive load [98]. In such usability experiments, we have found that users will forget the gestures that s/he has associated with certain actions.

We want to note that Mobile app based IoT interaction is limited by the UI. In most cases, simple IoT interaction becomes tedious and repetitive resulting time waste. In this prototype, we suppose context to be a Boolean combination of a set of sensory data. We term these combinations as context rules. When a context rule becomes true, then the user have to confirm his/her interaction commands by using a simple gesture. Hence, in our context-aware interaction prototype, the user only needs to remember one gesture

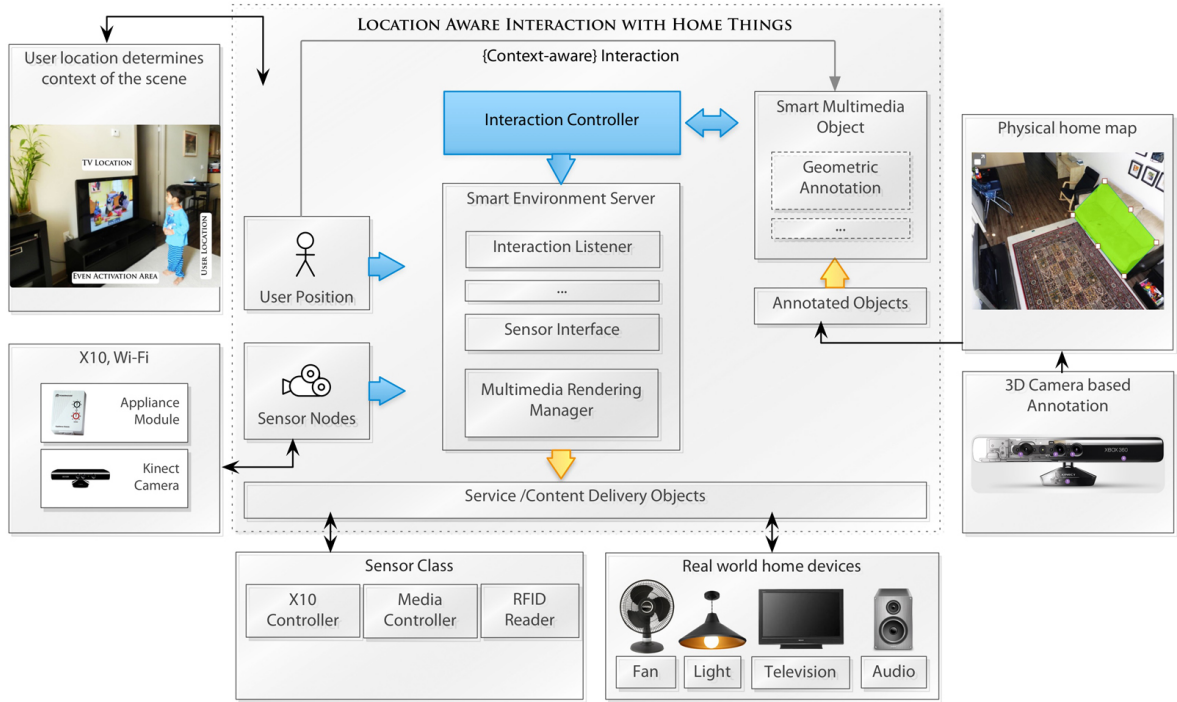


Figure 4.15: Context aware interaction with annotated smart home things.

command. In addition, we have realized Audio Recognition Service to make it possible for the users to assign a voice command instead of a gesture.

In addition to user location and IoT Scenes, we have combined our gesture rule calculations with the System Local. It contains time of the day, current date, calendar appointments, and previous activity records of the users. Number of users present in a Scene is also used as a parameter for a context-rule calculation. In order to calculate context in a Scene, in this prototype we have realized Skeleton Processing Service that determines User posture. We have considered standing, sitting, lying down postures in this prototype. We have implemented Audio processing service, User location tracking service as classes of Sensor Nodes. The Interaction Controller uses Event Processing Node modules in real time to deliver sensory data to the Intelligent Context calculation service component. This component is a sub-component of the Interaction Controller of the proposed framework. We show the components of the prototype in Figure 4.15. Next, we show some scenarios that we have developed with the help of specific context rules. We highlight their implementations for the interaction with geo-annotated IoT.

Child Aware Device Programming

During Television playback the kids have the tendency to come near the television to watch their favourite characters up close or get immersed in the action of the medium, which is not good for their eyes [124]. In order to assist the caregiver to instill good practices in the kids television watching experience we have considered the following geo-annotation programming scenario in our system. Figure 4.16 explains the scenario.

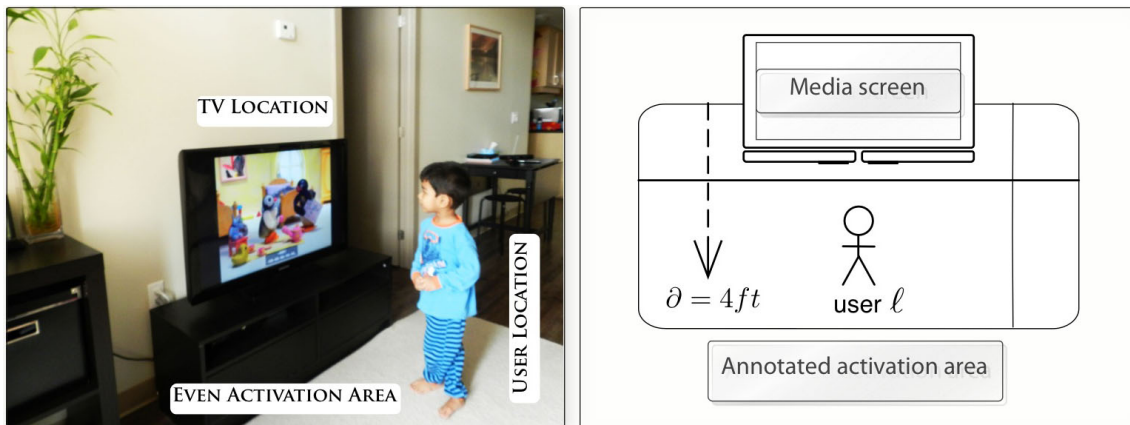


Figure 4.16: Programming the television to pause when kids are very close to it.

When the kids physical location is within the boundary of the annotated carpet and the location of the kids is within the threshold defined by the caregiver the television screen automatically dims the brightness (by using IR controller) or the playback of the television medium is paused/stopped. The caregiver can program the distance, the height of the user who should be considered as kids and many other parameters to effectively train the kids. It is also possible to automate a voice control instruction that in addition to pausing the television media content will ask the kids to go back to their sits repeatedly.

Location Aware Device Programming

Based on user's proximity with respect to the geo-annotated IoTs in the smart environment, the interaction subsystem may automatically active the nearest device (as programmed) when the user performs an activate gesture. This context-aware scenario is suitable for elderly home care. An elderly may raise his/her right hand or provide a voice command to turn ON the reading light when s/he is sitting on the reading chair. Figure 4.17 explains the use case in more detail. Similarly, playback of the television

can be controlled when the user is sitting on the living room sofa etc. As evident the geo-annotations of the IoT play an important role in the design and development of these context aware interaction scenarios.

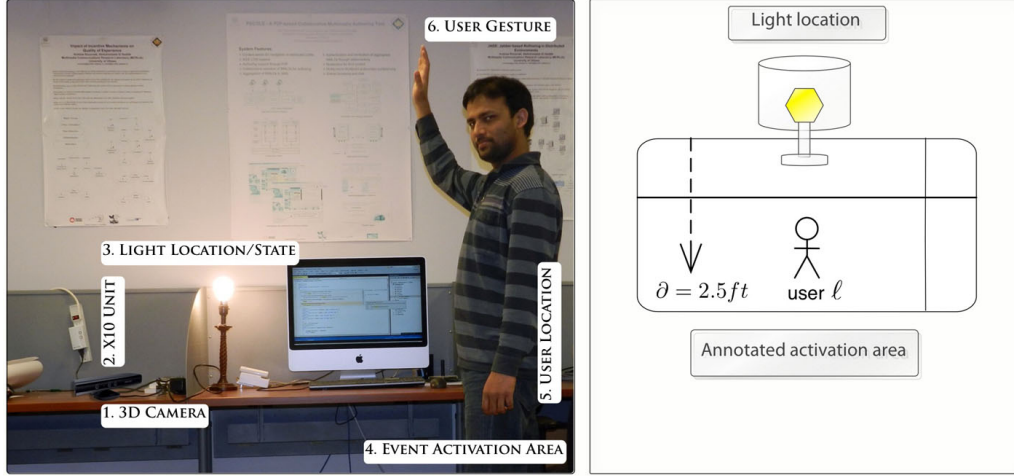


Figure 4.17: Programming lights that responds to a user gesture when s/he is nearby.

Encouraging Healthy Posture

In the geo-annotated environment, where we not only track the location of the user but also can determine the posture through 3D analysis of the skeletons (as discussed earlier) we can program scenarios so that user can develop healthy habits. Based on the duration of time the user is watching the television we can provide warning augmented on the playback screen. The caregiver can also program to completely stop playback of the television after the user has watched it for a predefined time. Figure 4.18 explains such a scenario.

$$Event(E_i) = \sum_1^T |E_i + Score(P_i)|$$

Here, P_{T_i} is the posture score over time T_i and $TVState = f(E_i)$ is the Boolean function that controls the media playback state. Through intelligent programming whenever the user's posture changes and accumulates bad score the television content can be paused or a non-intrusive message appears on the screen. When such message appears the user can correct his/her posture and continue enjoying the media playback.

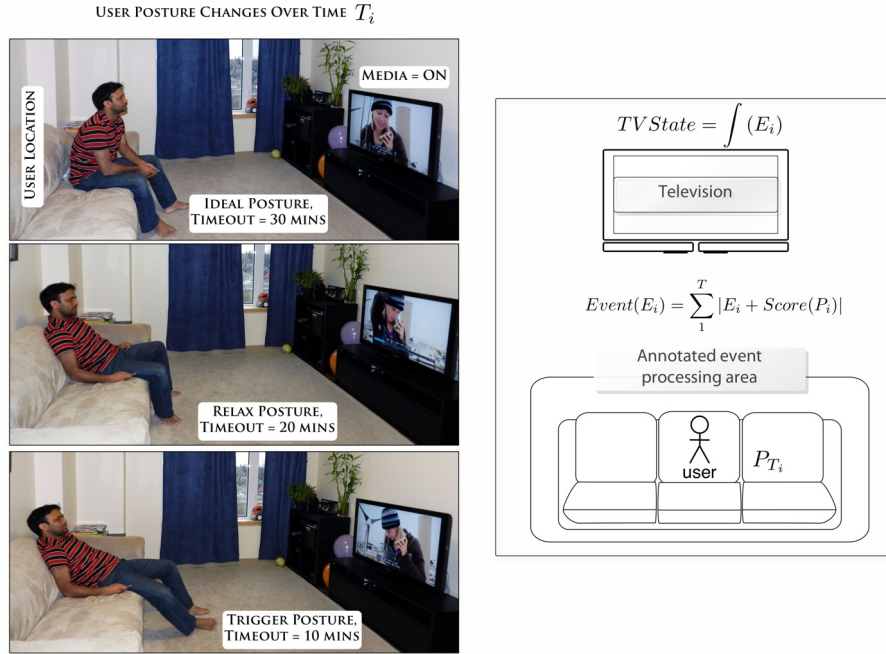


Figure 4.18: Programming to encourage breaks in a long television watching session.

4.4.2 Annotation of IoT and Scene Programming Editor

The proposed 3D camera based annotation scheme is suitable for indoor spaces, where interaction with a number of physical multimedia IoT e.g. television, audio player, movie player, air conditioner etc. are intended. By using the 3D depth sensor data, we geometrically annotate the physical devices in the smart environment as shown in Figure 4.19. Based on the 3D position estimation methodology, we introduce a spatial-geometric approach to annotate the physical IoT. Using that approach, we have annotated an object in XY plane. The geometric annotations are voxel based and uses 3D camera reference point. By using the 3D annotation editor the user can draw a rectangular shape to identify an IoT inside the 3D image. With the editor the use can assign name, colour, properties and associate that IoT with a particular Scene. Based on a collection of geometric IoT annotations a physical scene map is created for a specific location of a smart environment.

We now describe the context-rule editor that is shown in Figure 4.20. Context-rule editor incorporates the Annotation database and also uses Service list from the Multimedia Rendering Manager. Context-rule editor has been created by following the design of an email query editor. Here, the user can learn from examples and tutorials

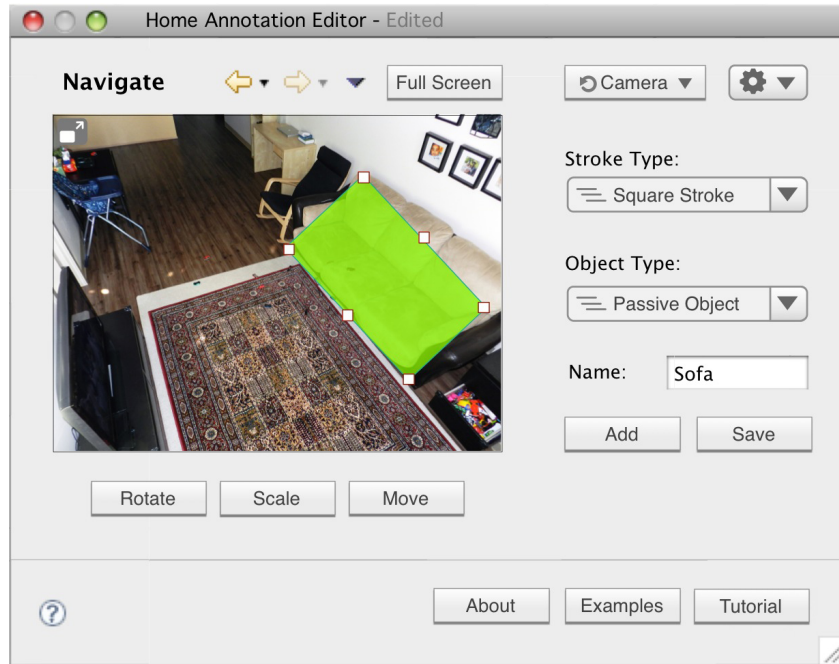


Figure 4.19: Annotating smart space elements to allow programmable physical interaction.

to familiarize himself/herself with the editor. For example, in order to create a context rule that will turn ON a reading light when the user is sitting on a reading chair, a user does the following. First, the user selects the IoT annotation s/he wants to interact with, the name of the target IoT, and select Activation options. The list of annotated IoT is available in the Object menu. In the next menu we need to select for which user groups this interaction applies to. For all the users we select the value ANY (default value). Next, we have to assign a gesture command that will confirm the gesture command to the Interaction Controller. We assign a gesture command to the context rule from the Trigger menu. We select a posture trigger for the context-rule from the Posture menu in the editor. Other System Locale commands are available from the settings menu located at the top right corner in the editor. Afterwards, each rule data is stored in an Scene database and are associated with information/services. These rules are triggered by the Interaction Controller when a user is interacting with an annotated IoT.

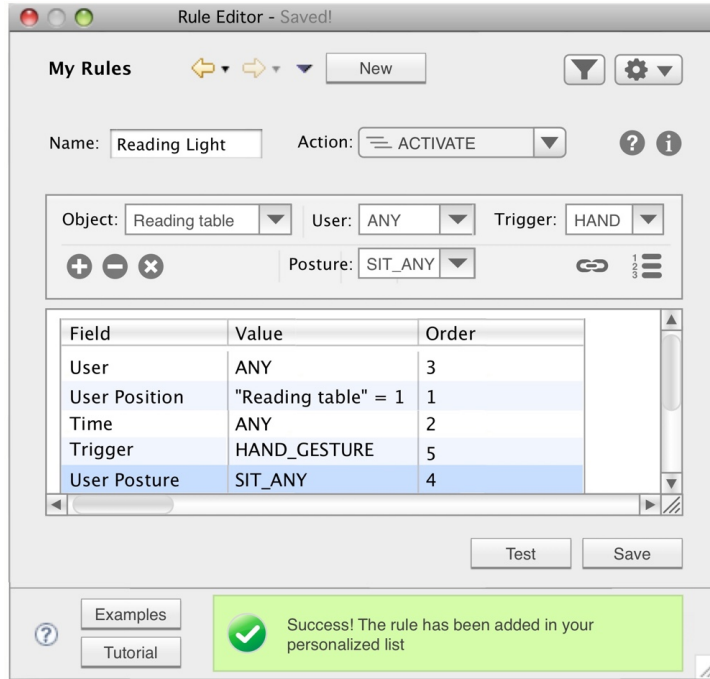


Figure 4.20: Programming smart space by using the Interaction Rule Editor.

4.4.3 Realization and Measurements

We have chosen the development platform of the prototype to be visual studio 2012 .net and C# is the chosen programming language. In order to communication messages between the processing units of the prototype, we have used Simple Object Access Protocol (*SOAP*). The event handler performs actions by using text based messaging protocol. We have used XML based simple database to store IoT annotation data and Scene annotation data.

Location accuracy

Location always calculated in terms of distances. Event triggered from the user was paired with the id of the nearest physical object in order generate query to the Service database. In our experiments we used 21 physical object locations. We annotated the physical environment in a manner that we avoided nested location calculations. As evident from our use cases, nested physical location mapping is unnecessary to build a valid and workable location aware service activation. As we can activate a depending on the position of the user with respect to any objects that are mapped in the system.

In order to avoid false positives we subdivided the physical annotation map of larger objects in order to work with correct radius that did not intersect with smaller physical objects. Hence, for larger objects we considered 2 or 3 points of centres (POC). For example, we considered 2 points of centres for big sofa, television, child carpet etc objects. Similarly, for the queen bed and living room carpet we considered 4 points of centres for the annotated object maps. These additional points made the distance based location calculation with respect to the annotation maps more accurate. We listed the annotated objects, their POCs, actual triggered events, and obtained calculated events in Table 4.5.

Table 4.5: Location calculation accuracy with respect to the geometric annotation maps.

Object IDs	ID	POC	Actual Events	Obtained Events	False +tives	Accuracy in %
1	Light Living Room	1	37	37	0	100
2	Light Reading Room	1	47	43	1	90
3	Light Washroom	1	45	44	0	98
4	Light Bedroom	1	47	42	0	89
5	Sofa Big	3	42	40	2	91
6	Sofa South Wall	2	41	38	1	90
7	Sofa Reading Table	2	45	44	1	96
8	Chair Reading Table	1	47	46	0	98
9	Couch Bed Room	1	34	32	0	94
10	Chair Child Reading	1	32	27	0	84
11	Television	2	46	41	0	89
12	Carpet Living Room	4	50	47	3	89
13	Queen Bed	4	49	49	2	96
14	Corner Table South	1	43	41	0	95
15	Side Table East Wall	1	43	41	0	95
16	Drawer with Knife	1	43	43	0	100
17	Oven/Heater	1	47	43	0	91
18	Child Play Carpet	2	47	42	0	89
19	Kid Sound System	1	31	27	0	87
20	Fan Living Room	1	49	48	0	98
21	Fan Bed Room	1	40	35	0	88

Comparisons of Interactions with Large Annotated Objects

In this series of usability experiments, we set up the study with 9 and 22 participants on day 1 and day 2 respectively to compare the programmable geometric annotation interactions. We devised two simple experiments to compare context-aware interactions with other conventional interactions. In experiment one, by using geometric annotations we could annotate a large object such as a television at a living room. In this experiment setup, we asked the users to turn a television ON by using a remote control interaction, a RFID based interaction, a visual QR Code based interaction, a pointMe based interaction, and a context-aware interactions. We explained the use of these interaction to the users in detail. In order to turn the television ON by using a remote control, the user was needed to press the power switch of the remote and to complete his task. Similarly, we asked the users to scan a RFID chip to power ON the television. We placed a visual code at the corner of the television and as soon as the users scanned to code we turned the television ON. In addition, the user can point a mobile handheld device at the television screen and press a switch to turn the television ON. Lastly, in the context-aware interactions, we automatically turned the television ON as soon as the users sat on the sofa in front of the television.

The participating users in the experiments were told to fill out a questionnaire and to answer some given questions according to their usage experience of the system. During the experiments we also recorded the interaction actions of the users for further analysis. From the recorded data, we compared the usability data of the selected interaction schemes. In Figure 4.21, we presented the average rating of the users in Likert scale. Here, users rated whether they considered remote control, RFID, QR Code, pointMe, context-aware interactions to be appealing and easy to use. As evident, RFID and visual code based interaction approach were not appealing to the users even though users agreed that RFID based approach was easy to use. Moreover, location-aware based automated television start was appealing to the users.

Study of Interactions with a Light

In this experiment setup, we asked the users to turn a light ON by using a light switch, a clap, a mobile app, and with a context-aware based programmable interactions. We explained the use of these interaction to the users in detail. In order to turn the light On by using a switch, the users were requested to press the power-switch of the light ON to complete their tasks. Similarly, we asked the users to use a mobile app and to turn the

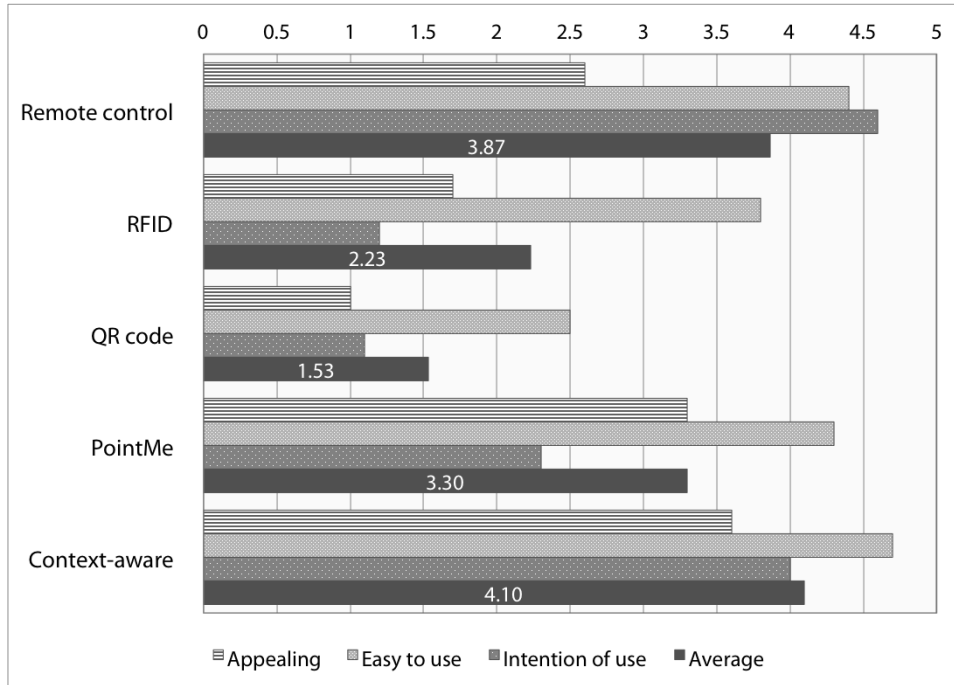


Figure 4.21: Geometric annotation of large objects and its advantages.

light ON. We used a smart light and the accompanied app to accomplish this task. In addition, the user was told to clap with his/her two hands to turn the light ON. Lastly, in the context-aware interactions, we automatically turned the light ON when the users sat on a reading chair and raised their right hands.

The participating users in the experiments were told to fill out a questionnaire and to answer some given questions according to their usage experience of the system. During the experiments, we also recorded the interaction actions of the users for further analysis. From the recorded data, we depicted the usability data of the selected interaction schemes. In Figure 4.22, we presented the average rating of the users in Likert scale. Here, users rated whether they considered a light switch, a mobile app, a clap, and a context-aware based interaction approaches to turn a light ON to be appealing and easy to use. As evident, a light switch was considered by all users to be very easy to use compared to other interaction approaches. Users found both mobile app and context-aware based interactions to turn a light ON to be appealing.

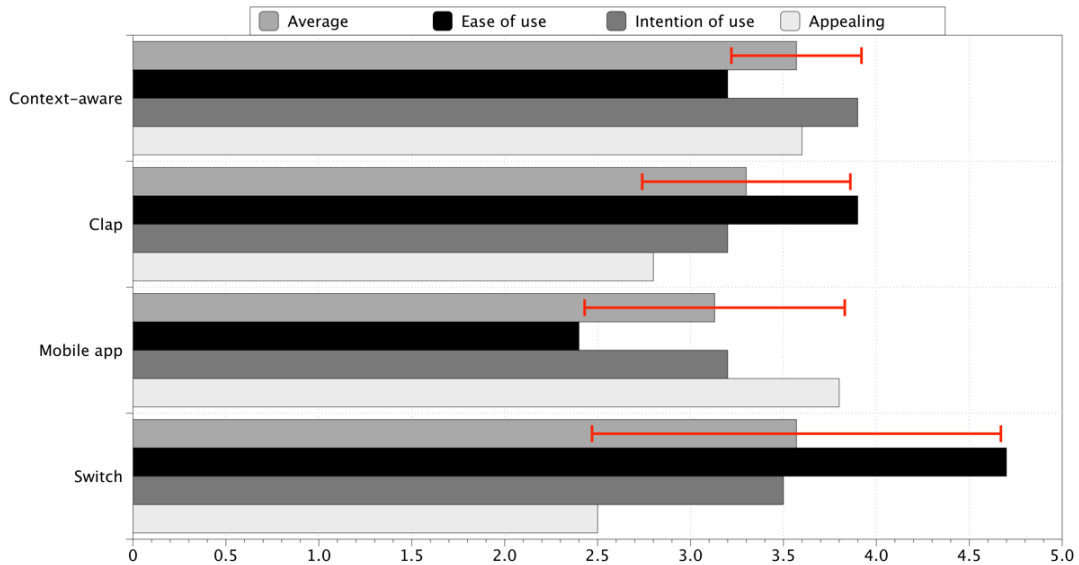


Figure 4.22: Usability comparisons of various interactions with an annotated light.

Usability Study of Healthy Posture Awareness

In this experiment setup, we asked 9 users to participate in the study of a context-aware based healthy posture awareness application. We considered a scenario, where users would watch television for long durations and would remain sit without straightening their backs. In such cases, the context-aware app would wait for a given time (20 minutes) and would provide warning to the users that they needed to fix their sitting postures. The app would also be able to notify the users to take a break and to pause the playback of the movie to enforce a break automatically. To participate in the experiment, users were requested to watch three movie clips, each of 5 minutes length. The system would calculate users posture after every 3 minutes by using a 3D camera and would notify the users in a non-obtrusive manner to correct their postures or to take a break if needed.

The participating users in the experiments were told to fill out a questionnaire and to answer some given questions according to their usage experience of the system. From the recorded data, we depicted the usability data of the selected interaction schemes. In Figure 4.23, we presented the average rating of the users in Likert scale. Here, users rated whether they found the proposed system to be appealing and easy to use. As evident, an overwhelming majority of the users appreciated the appeal, ease of use of the context-aware application. In addition, they showed their interests to use the system in real world if it was made available to them.

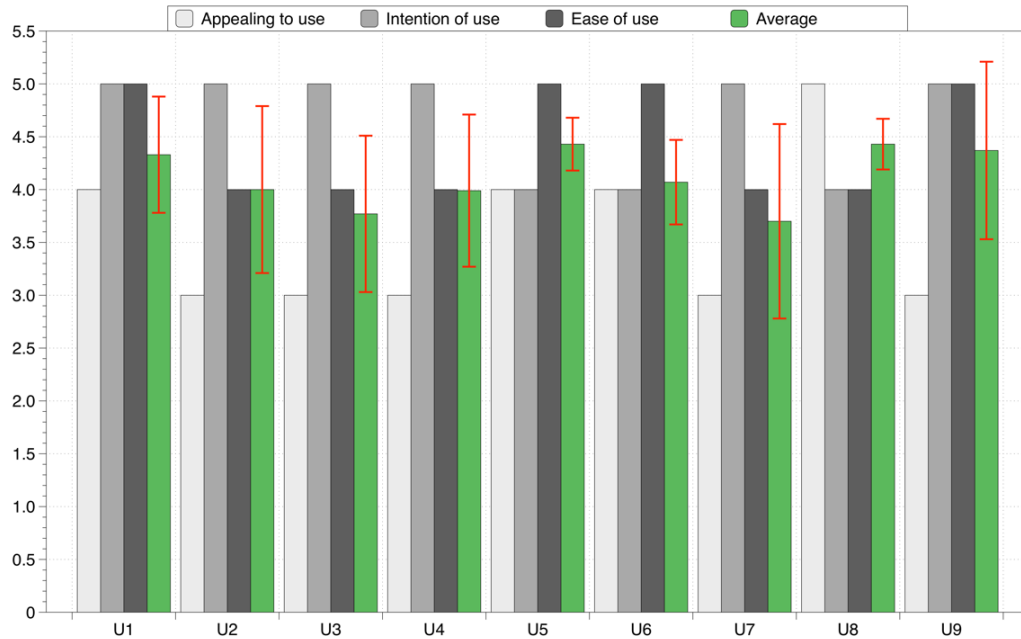


Figure 4.23: Usability study of a context-aware healthy posture awareness application.

Usability Study of Location Aware Television Watching Habits

We observed that the kids often would watch the television by standing very close to the screen. In order to discourage young learners to watch television by standing at a close proximity of the television, we devised a context-aware application. We asked 22 users to participate in the usability evaluation of such an application. In this application, we annotated the carpet and television objects and included them in the annotation map. Further, we created a context rule, where the television playback was automatically paused as soon as a user was at a certain distance threshold of the television screen. During content playback, the app would also be able to make a loud noise to warn the users, in case they were standing too close to the television screen. To participate in the experiment, users were requested to watch three movie clips, each of 5 minutes duration, however, by standing at various distances with respect to the television screen. The system would calculate users position after every 20 seconds in the smart environment and would notify the users immediately when they would stand close to the screen.

The participating users in the experiments were told to fill out a questionnaire and to answer some given questions according to their usage experiences of the system. In Figure 4.24, we presented the average rating of the users in Likert scale. Here, users rated whether they found the proposed system to be appealing and easy to use. As evident,

an overwhelming majority of the users found the system to be appealing and ease to use.

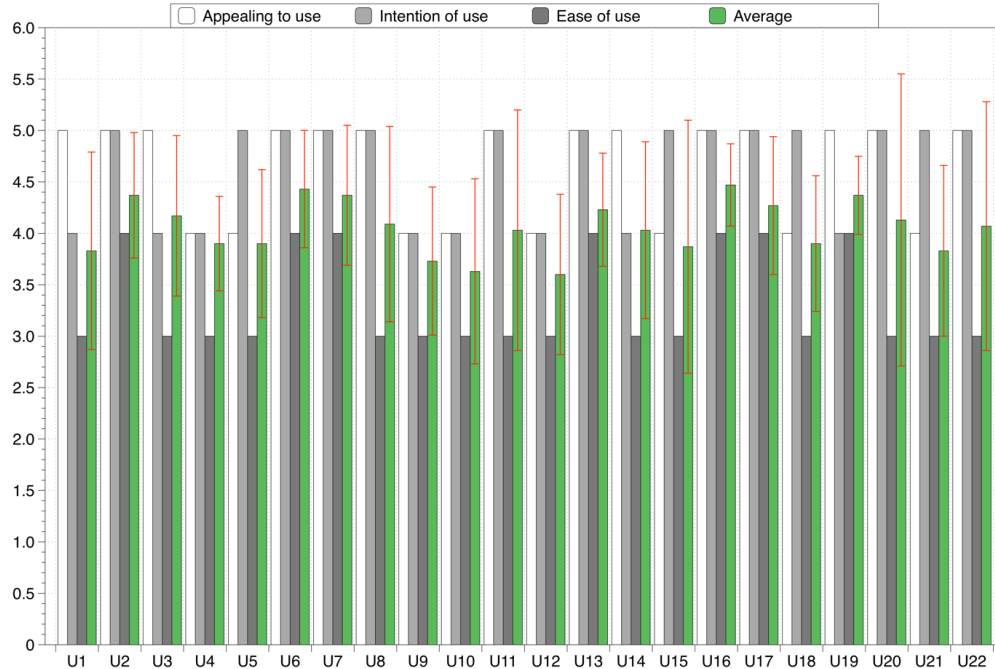


Figure 4.24: Usability study of a context-aware applications to monitor baby locations in front of a multimedia screen.

Usability Study of the Context Editor

In this experiment setup, we asked the users to create context-rules by using our developed context editor. We separated the users into three groups. Each group contains 3, 2, 2 users respectively. We explained the tutorial of the editor to the users. In the tutorial we presented demo and video to explain programmable context rules with examples. We provided five easy to understand scenario implementations by following four context rules. We explained the users the scenarios, where 1) Light would turn On depending on the location of the users with respect to a geometric annotation, 2) Television would automatically turn ON when users sit on the annotated sofa, 3) Television movie playback would pause as soon as a user stands at a close distance from the multimedia screen, and 4) Turn a light OFF when user would leave a geometrically annotation location.

The participating users in the experiments were told to fill out a questionnaire and to answer some given questions according to their learning experience of the system. Afterwards, we asked the users whether they were able 1) To follow the context-rule

editor tutorials, 2) To create at least 1 context rule for a scenario that was presented in the tutorial, 3) To create at least 2 context rules for two scenarios that were discussed in the tutorial, and 4) To create a scenario by using their own context rules. We showed their score in Figure 4.25. Overall, the users in each group were able to develop two scenarios by using context rules in the proposed context editor.

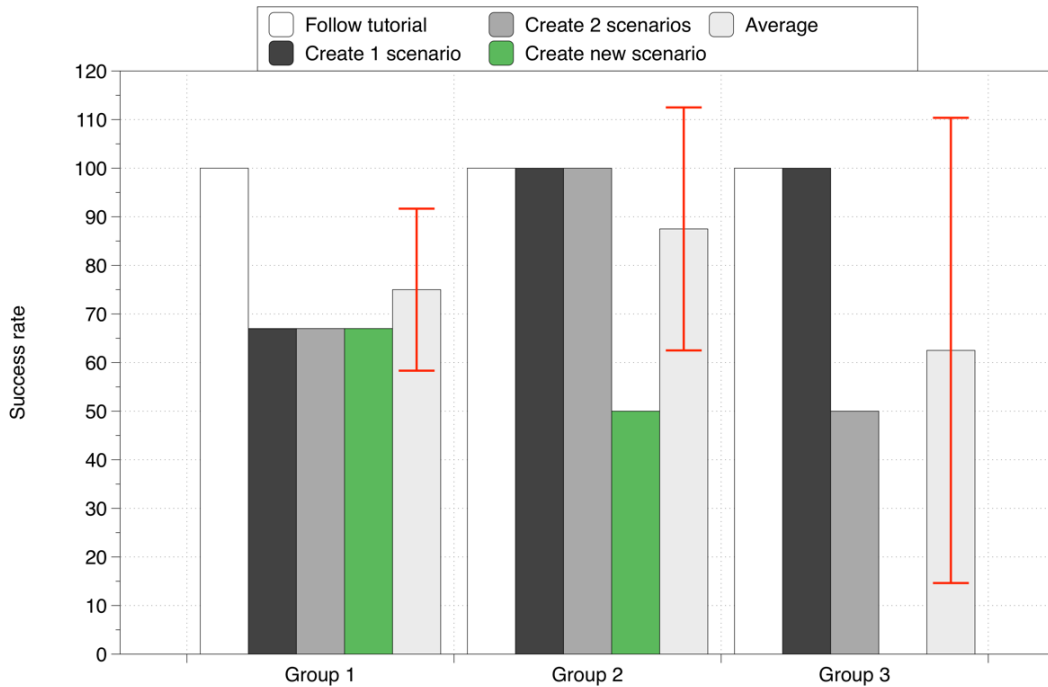


Figure 4.25: Ease of use of the context editor application.

From the questionnaire feedback data we found that more than 50% of the users found the context-rule editor to be easy to use. However, most users provided suggestions to improve the editor in future. In addition, most users agreed that users who were able to create an advanced email search would be able to create context-rules easily.

User Posture Accuracy

To assess the accuracy of posture classification, we have created a confusion matrix. In the confusion matrix, we compare the classification results to additional ground truth information. We want to use the confusion matrix to identify the nature of the classification errors, as well as their quantities.

In Table 4.6 we depict the confusion matrix. We have Stand, Sit, Recline, Lie-Down classifications. In the table REL denotes the reliability of the classifications and ACC

column shows the accuracy of the classifications. In the matrix, the rows correspond to classes in the ground truth map and columns correspond to classes in the classification result. The diagonal elements in the matrix represent the number of correctly classified images of each class. In the table, 467 pixels of 'Stand' in the test set were correctly classified as 'Stand' in the classified image. The off-diagonal elements represent misclassified pixels or the classification errors. In the matrix for example, 7 samples of 'Stand' in the test set were classified as 'Sit' in the classified image and 87 ground truth images of 'Lie Down' were included in the 'Recline' class by the classification.

Table 4.6: User posture accuracy.

	Stand	Sit	Recline	Lie Down	ACC
Stand	467	7	0	0	0.985232068
Sit	4	245	92	11	0.696022727
Recline	0	50	150	67	0.561797753
Lie Down	0	21	87	130	0.546218487
REL	0.991507431	0.758513932	0.455927052	0.625	

In the confusion matrix, the ACC column presents the accuracy of the classification. Accuracy is defined as the fraction of correctly classified images with regard to all images of that ground truth class. For example, for the 'Stand' class, the accuracy is $467/474 = 0.985$, meaning that approximately 98.5% of the 'Stand' ground truth data also appear as 'Stand' data in a classified image. The figures in row REL presents the reliability of classes in the classified image. Reliability is the fraction of correctly classified data with regard to all data classified as this class in the classified image. For example, for the 'Stand' class, the reliability is $467/471 = 0.99$, meaning that approximately 99% of the 'Stand' data in the classified image actually represent 'Stand' on the ground.

The average accuracy is calculated as the sum of the accuracy figures in column ACC divided by the number of classes in the test set. The average accuracy of the proposed classification is 0.697317759. The average reliability is calculated as the sum of the reliability figures in row REL divided by the number of classes in the test set. The average reliability for the proposed posture classification is, 0.707737104. Lastly, we calculate the overall accuracy by the total number of correctly classified data (diagonal elements) divided by the total number of test data. In this case overall accuracy of the classification is, $(467 + 245 + 150 + 130)/1331$, which is approximately 75%. From the confusion matrix, we can further conclude that the test set classes 'Recline' and 'Lie-

Down' were difficult to classify as many of such test set data were excluded from the 'Recline' and the 'Lie-Down' classes. On the other hand, class 'Recline' in the image is not very reliable as many test set images of other classes were included in the 'Recline' class in the classified image.

Table 4.7: User stand posture classification accuracy.

	Side	Front	Back	Unclass	ACC
Side	292	4	171	4	0.619957537
Front	11	360	9	2	0.942408377
Back	155	12	280	7	0.616740088
REL	0.637554585	0.957446809	0.608695652		

In Table 4.7, we now present classification for the stand posture considering the Front, Side, and Back view. The confusion matrix in column Unclassified represents the ground truth data that were found not classified in the classified image. The average accuracy of the classification is 0.726368667. Average reliability and overall accuracy of the classification are 0.734565682 and 0.713083397 respectively. From the confusion matrix, we can conclude that the test set classes for 'Side' and 'Back' stand postures were difficult to classify as many of such test set data were excluded from the 'Side' and the 'Back' classes. On the other hand, class 'Front' in the image is very reliable as no test set images of other classes were included in the 'Front' class in the classified image.

Table 4.8: User sit posture classification accuracy.

	Front	Side	Back	Unclass	ACC
Front	124	0	1	40	0.751515152
Side	7	230	1	7	0.93877551
Back	5	2	12	110	0.093023256
REL	0.911764706	0.99137931	0.857142857		

In Table 4.8, we now present classification for the sit posture considering the Front, Side, and Back view. The confusion matrix in column Unclassified represents the ground truth data that were found not to be classified in the classified image. The average accuracy of the classification is 0.594437973. Average reliability and overall accuracy of the classification are 0.920095624 and 0.67903525 respectively. We can further conclude from the confusion matrix that the test set class 'Back' was difficult to classify as many

of such test set data was excluded from the 'Back' class. When user is in sitting posture then from the back it is difficult to accurately detect that classification from the image.

Table 4.9: User recline posture classification accuracy.

	Front	Side	Back	Unclass	ACC
Front	120	0	4	20	0.8333333333
Side	0	184	1	4	0.973544974
Back	5	2	7	105	0.058823529
REL	0.96	0.989247312	0.5833333333		

In Table 4.9, we show classification for the recline posture considering the Front, Side, and Back view. The average accuracy of the classification is 0.621900612. Average reliability and overall accuracy of the classification are 0.844193548 and 0.688053097 respectively. We can further conclude from the confusion matrix that the test set class 'Back' was difficult to classify as many of such test set data was excluded from the 'Back' class. When user is in recline posture then from the back it is difficult to accurately detect that classification from the image.

In Table 4.10, we now present classification for the lie-down posture considering the Front, Side, and Back view. The confusion matrix in column Unclassified represents the ground truth data that were found not classified in the classified image. The average accuracy of the classification is 0.664236569. Average reliability and overall accuracy of the classification are 0.817651369 and 0.673015873 respectively. From the confusion matrix, we can further conclude that the test set classes 'Back' and 'Front' were difficult to classify as many of such test set data were excluded from the 'Back' and the 'Front' classes. On the other hand, class 'Side' in the image is reliable as not many test set images of other classes were included in the 'Side' class in the classified image.

Table 4.10: User lie-down posture classification accuracy.

	Front	Side	Back	Unclass	ACC
Front	50	12	6	22	0.555555556
Side	5	90	10	10	0.782608696
Back	8	5	72	25	0.654545455
REL	0.793650794	0.841121495	0.818181818		

4.5 Summary and Conclusion

In this section, we have presented the details of three prototypes that have been realized by using the proposed smart mobile interaction framework. The prototypes first demonstrate with examples how smart home things are annotated geometrically. Smart home things such as fan, light, television are used as examples to demonstrate the annotation process in a simple manner. Second, by using the proposed framework, pointMe, touchMe, and context-aware interactions have been realized. Next, in context-aware interactions realization, we see how user's position, and geometric annotation of places and things can be intelligently processed in order to calculate contexts. We show programmable context rules that are used to create natural interactions event with things. Lastly, we present the detailed measurements data that we have obtained from the implementations of the pointMe, touchMe, and context-aware interactions.

Chapter 5

Haptic Interactions with Annotated IoT

5.1 Introduction

We have incorporated the proposed framework and have realized touchMe based interactions with IoT in handheld device based GUI setting. In this chapter, we show two prototypes, where we document the approaches of geo-spatial annotation mapping of the virtual things. The annotation maps are further used both in physical and virtual environments to realize natural touchMe based interactions with IoT to bring virtual events into real life. In Section 5.2, we describe how we can use our interactions virtual 3D avatars to touch real people. In Section 5.3, we illustrate a haptic eBook system, where reading a scary parts of the story will bring horror haptic sensations to the readers, or reading about a moon adventure will present moon expedition images on the media screen, etc. For each prototype, we provide short description of the prototype, describe the implementation challenges, relate the development of the prototype with the proposed framework, and illustrate the achieved results.

5.2 TouchMe Based Social Haptic Interaction

The sense of touch has much importance in technology-mediated human emotion communication and interaction. Many researchers around the world are aiming to leverage the sense of touch in the communication between multiuser 3D virtual world and real environment. Driven by the motivation, we explore the possibilities of integrating haptic

interactions with Linden Lab’s multiuser online virtual world, Second Life. We have enhanced the open source Second Life viewer client in order to facilitate the communications of emotional feedbacks such as human touch, encouraging pat and comforting hug to the participating users through real-world haptic stimulation. The emotional feedback that is fundamental to physical and emotional development in turn can enhance the users interactive and immersive experiences with the virtual social communities in the Second Life. Here, we describe the development details of a prototype that realizes the aforementioned virtual-real communication through a haptic-jacket system by using the framework proposed in this thesis.

In this prototype system we aim to achieve the following. First, in order to bridge the gap between virtual and real, we present a Second Life viewer add-on, where we provide haptic interaction opportunity between the real users and their respective virtual avatars through a 3D graphical user interface (GUI). Second, we introduce touch, hug and tickle haptic features for the Second Life users through touchMe interactions. Third, we incorporate 3D annotation mechanism for the Second Life avatar so that user dependent interpersonal haptic and animation interactions become possible. To illustrate the system, first in Section 5.2.1 we discuss the details of the prototype with respect to the proposed smart mobile interaction framework. The annotation of the home things are explained in Section 5.2.2. Lastly, in Section 5.2.3 the realization steps and key results have been illustrated in detail.

5.2.1 Prototype Description

We extend the proposed framework to realize touchMe-based interactions with annotated virtual things and bridge communications with physical objects. We extend the Mobile Client class to implement a THIN mobile client that renders a 3D virtual environment by using remote computer resources. Currently the mobile devices lack the graphics horsepower to render SL’s 3D environment in a satisfactory way. In order to render SL in mobile devices we adopt virtual display technique that incorporates device collaboration [132][110], where most computations are carried out remotely by leveraging remote computing resources, and screen updates are compressed and transmitted together with low-level display commands from servers to clients [126]. We explore the possibilities of integrating haptic interactions in Second Life [63][62]. We have enhanced the open source Second Life viewer client and have introduced a communication channel that provides physical and emotional intimacy to the remote users. In the prototype system a user can

take advantage of touch, tickle, and hug type haptic commands in order to interact with the participating users. A haptic stimulation of touch and other touch based interactions are rendered to the remote user on the contacted skin through our previously developed haptic jacket system [23] that is composed of an array of vibrotactile actuators. This thesis illustrates a preliminary prototype exploring the aforesaid haptic interactions between virtual and real environment actors. We show the components of the system in Figure 5.1.

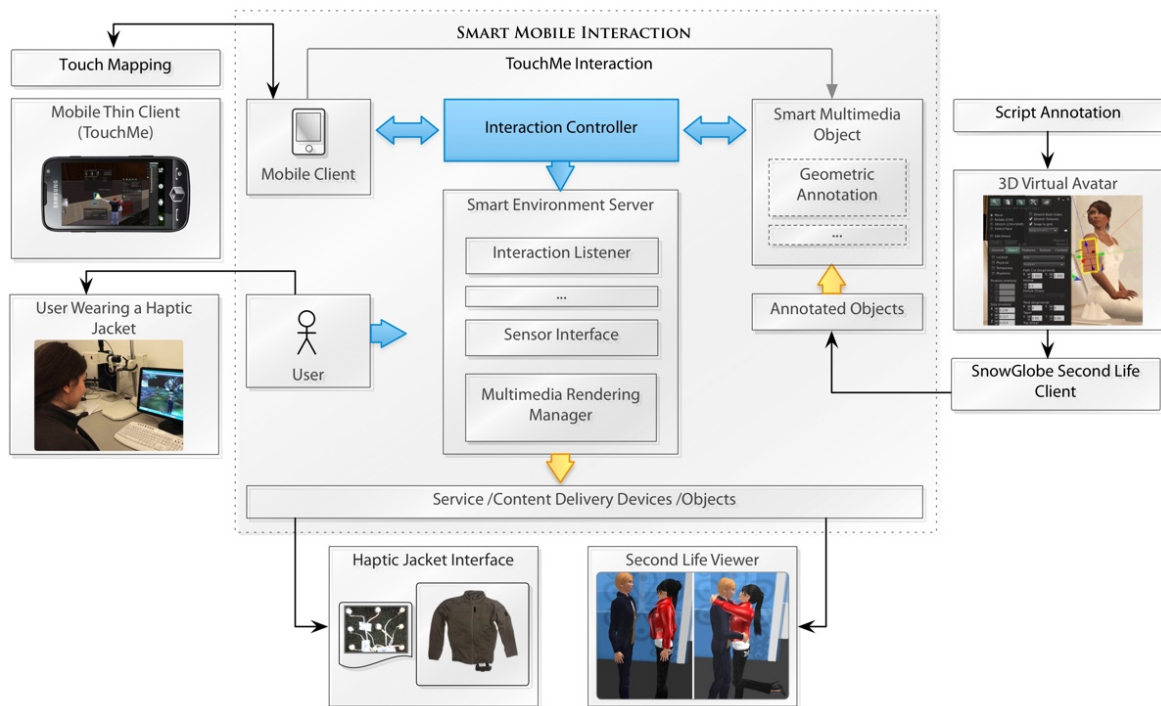


Figure 5.1: TouchMe based inter-personal social haptic interactions, such as hug, kiss, and tickle in Second Life.

We attach linden script on avatars in the Second Life virtual environment to make them interactive. When a user touches a scripted part of the avatar, annotated events are sent to the haptic jackets of the real users. Below we illustrate Linden script and haptic jacket in detail.

Linden Script

Linden Script (LSL)[75] is a scripting language developed by Linden Lab, which imparts the capability to create custom objects in Second Life. LSL script ¹ can be attached to any primitive object in the virtual world, but not to an avatar. Avatars, however, can wear scripted objects. The syntax of LSL script is similar to the syntax of popular programming language Java. Like other scripting languages, LSL script has variables, functions, and events. One thing that makes LSL unique is its emphasis on States and Events. In SecondLife, a door can be "open" or "closed" and a light can be in the "on" or "off" state. This state-based idea came from real world objects, since many real-world object behaviours can be modelled in the same way.

Each LSL script has at minimum one state, which is the scripts default state. An event can be thought of as a "Trigger". Events are predefined, and are triggered when two objects or avatars are colliding. For example, when an avatar touches an object, a "touch start" message is sent to the object, which causes the touch start() event handler to begin executing. LSL script has over 500 functions available. Users can also define additional functions.

Haptic Jacket Interface

Vibrotactile actuators communicate sound waves and create funnelling illusion when it comes into the physical contacts with skin. The haptic jacket consists of an array of vibrotactile actuators that are placed in particular portions of the jacket and their patterned vibration can stimulate touch in the user's skin [13]. A series of small actuator motors are placed in a 2D plane in the jacket in a certain manner. An AVR Micro-controller controls the vibration of these actuators. We have configured the Micro-controller so that it processes input commands that are sent from the haptic interaction controller. In order to achieve the input command transmission the haptic interaction controller uses the Bluetooth communication channel. Figure 5.2 depicts the components of the jacket into more detail. In order to create hug type haptic feedbacks for the participating users, we systematically increased the jacket's *leftChest*, *rightChest*, *neck*, *leftBackShoulder*, and *rightBackShoulder* motors intensity levels to produce the funnelling illusion. The systematic control of the actuator intensity levels creates the touch effect in those areas and offers a hug type haptic stimulation. The lover type hug is different to that of the formal hug. In addition to the areas defined above, we decided to add haptic touch stimulation

¹Linden Script, [http://wiki.secondlife.com/wiki/LSL Portal](http://wiki.secondlife.com/wiki/LSL_Portal)

in the stomach area to emulate the joy emotion [34] [135]. Hence, by following the laws of funneling illusion, we activated the arrays of vibration motors attached to the abdomen area of a person.

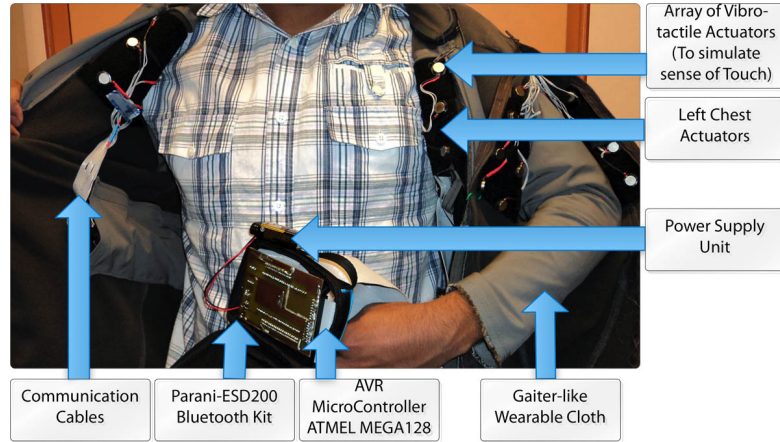


Figure 5.2: The Haptic jacket controller and its hardware components. Array of vibrotactile motors are placed in the gaiter like wearable cloth in order to wirelessly stimulate haptic interaction.

5.2.2 Annotation of IoT

In our system, we annotate visible body parts of the avatar in Second Life and specify the corresponding physical haptic actuators to render the haptic feedback. For each haptic signal, we also annotate the avatar animation. Figure 5.3 depicts the geometric based avatar annotation scheme. We attach LSL scripts [76] in each of the annotated parts of the avatar that contain the haptic commands as well as the identification number of the animation sequences. For example, we have annotated the 3D male avatar’s left arm and have specified particular vibrotactile actuator stimulation for it. Further, we have specified the interacting animations for both of the participating male and female virtual avatars. Afterwards, when the user representing the female avatar issues a touchMe interaction command to the male avatar arm then the annotated haptic stimulation is rendered at the real male user’s arm through the haptic jacket.

For intimate interactions such as a hug, we employ group based annotation scheme. As evident, hugging with parents is different to that of a friend. Hence, we use separate animation and haptic rendering for each type of hugs, touch etc interactions. We have created groups and have incorporated group based annotation scheme of the 3D avatars.

For each group, we have created different avatar animations and haptic rendering options. By using the script based dialog interface any interacting contacts are then assigned to a group (default is formal). We provide four different groups namely family, friend, romantic, and formal. This group based haptic interaction in Second Life further assists the user to personalize his/her experience.

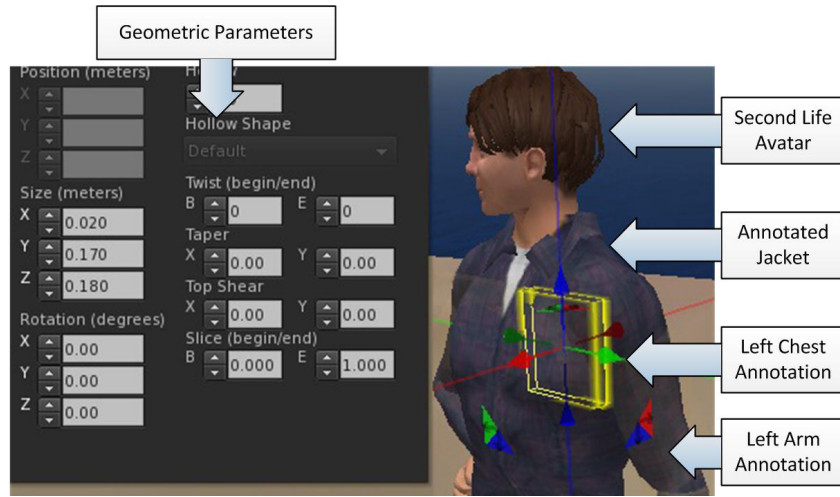


Figure 5.3: The flexible avatar annotation scheme allows the user to annotate any part of the virtual avatar body with haptic and animation properties. When interacted by the other party, the user receives those haptic rendering on his/her haptic jacket and views the animation rendering on the screen.

5.2.3 Realization and Measurements

In order to develop the Second Life add-on, we locally build the Second Life open source viewer Snowglobe [125] version 1.3.2 by using the latest version of CMake (version 2.8.1) [29]. Second Life message transmission module is responsible for dispatching all the messages to handle virtual environment. All the messages are in XML (Extensible Markup Language) format with detail avatar and virtual environment related data. All the event type messages are filtered by nearby interaction event handler. Interaction Event Decoder is a component in the interaction listener. It receives all the output messages from the Nearby Interaction Event Handler in an encrypted XML [147] format. The primary responsibilities of the module are to decrypt those received messages and transmit them further to the communication channel towards the Permission Manager.

The Haptic Renderer (HR) operates the haptic jacket and notifies the Animation Parcel Manager for synchronized animation feedback. In order to control the jacket motors, it parses an XML file containing haptic patterns and sends a message to the micro controller unit of the jacket accordingly. In our implementation the actuator motors have a total of 16 intensity levels from 0 to 15. Where, 0 means no vibration and 15 indicates the maximum vibration level. To repeat the vibration patterns we set the value for the *numberOfRepetition* attribute.

Usability Study

We have incorporated the usability evaluation guidelines [28] and have designed our tests accordingly with the sensory analysis [73] of the system involving both the user and the targeted sensory communication modules. Before performing the usability test, we design a test plan where we define our evaluation objectives, develop questions for the participants, identify the measurement criterion and decide upon the target users of the system. The test took place at a university laboratory with sixteen (16) participants comprising of different age groups. Five (5) of the participants are in age group 13-18, eight (8) of them are in age group 18-36 and the rest three (3) are in age group 36+. Furthermore, in a multi-user interpersonal communication setting, the users are divided into two groups namely Group A and Group B.

Table 5.1: Usability test questions to the user.

#	Question
Q1	Perceived system response was acceptable
Q2	The haptic feedbacks are realistic and/or acceptable
Q3	Consider using the system in Second Life
Q4	Perceived delay between haptic response and avatar rendering was tolerable
Q5	Easy to get familiar with

For the traditional experiments, two users are chosen at a time. In order to ensure that each communicating participant can converse with different age groups their selection is made randomly. Moreover, to ensure the distributed communication behaviour the physical location of the users are separated. At a time, the selected volunteers are told to put on the haptic jackets and are requested to use the prototype system by participating in certain haptic interaction based tasks. Their activity is monitored throughout

the experiment and is recorded for analysis. Afterwards, based on their interaction experiences the users are requested to fill out a questionnaire where they provide ratings of their likeliness, familiarity, and ease of usage of the system.

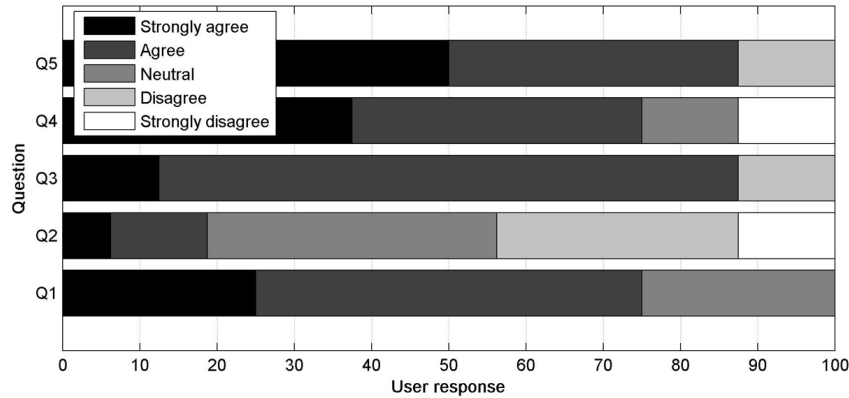


Figure 5.4: Usability study of the Second Life haptic interaction system.

The user responses are shown in Likert Scale [79] in Figure 5.4. The ratings of the questionnaire are in the range of 1-5 (the higher the rating, the greater is the satisfaction). The average of the responses of the users are calculated in percentage form and are measured after the usability tests. Figure 5.4 shows the user’s responses for each given assertions. It is worth mentioning that more than 80% of the users would like to communicate using the enhanced system through haptic and animation interaction if they are made available in Second Life. Overall, 75% of the users are also satisfied with the synchronized animation and haptic rendering responses of the system.

We have conducted usability tests to evaluate the user’s quality of experience with our proposed system and to measure the suitability of the approach. Table 5.2 summarizes

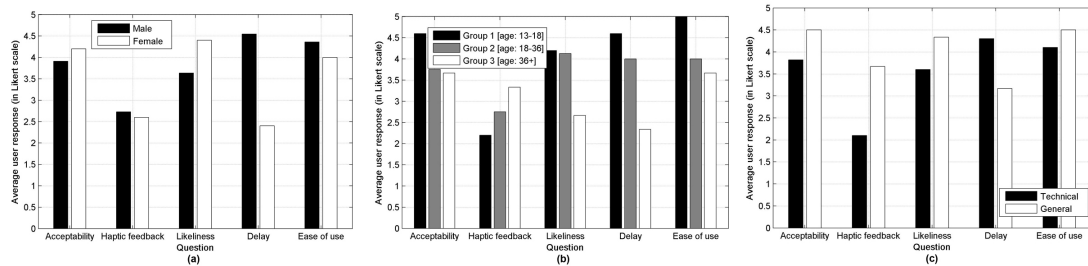


Figure 5.5: Comparison between the responses of users from different (a) gender, (b) age groups, and (c) technical background.

Table 5.2: User satisfaction on the overall evaluation in Likert scale.

	Mean	Std. Dev.	Mean Percentage
Acceptability	4	0.7303	80%
Haptic feedback	2.6875	1.0782	53.75%
Likeliness	3.8750	0.8062	77.5%
Delay	3.875	1.3102	77.5%
Ease of use	4.25	1	85%

the overall performance score of the users. The higher mean values of System response, haptic feedbacks, and Easy to familiar represent a very satisfactory user response, while the moderate mean values of System in Second Life and Perceived delay show relatively good user satisfaction.

In our study, we also have attempted to evaluate the acceptability of the system by the users from different genders, age groups and technical backgrounds. The result of these studies are depicted in Figure 5.5. We infer from the Figure 5.5 that (a) the female users give more positive feedback in acceptability and likeness than the male users. However, the male users have confirmed that it is easier for them to use the system after a couple of dry runs. Moreover, all the users have agreed that a refined haptic rendering is needed to make the interaction experience natural and more realistic. In case of different age groups, we have divided the users into three age groups, namely *group* – 1: ages 13 – 18, *group* – 2: ages 18 – 36, and *group* – 3: ages 36+ and recorder their responses in Figure 5.5 (b). Our analysis shows that compared to the older group of users, the users from the younger group seem to be more interested in using the system and want to participate in remote touch, hug, and tickle interactions. Also from Figure 5.5 (c) we have received favourable responses and recommendations from users with non-technical background than that of the technical users.

5.3 TouchMe Based Haptic Interaction with EBooks

In order to leverage the use of audio-visual materials in instilling effective learning behaviour, we present an intuitive approach of annotation based haptic-audio-visual interaction with the traditional digital learning materials such as eBooks. By integrating the home entertainment system in the user’s reading experience combined with haptic interfaces, we want to examine whether such augmentation of modalities influence the

user’s learning behaviours. The proposed Haptic E-Book (HE-Book) prototype leverages the haptic jacket, haptic armband as well as haptic sofa interfaces to receive haptic emotive signals in the form of patterned vibrations of the actuators. By extending the proposed smart mobile interaction framework, we implement a Haptic eBook system. In this system, we have proposed an annotation based hapto-audio-visual feedback integration in traditional digital learning materials. We present a generalized framework that describe diverse IoT annotation scheme and illustrate touchMe interaction mechanisms with the annotated virtual IoT. In this prototype, we have conducted user studies with the proposed HE-Book system to investigate its suitability on instilling effective learning behaviour. To illustrate the system, first in Section 5.3.1 we discuss the details of the prototype with respect to the proposed smart mobile interaction framework. The annotation of the home things are explained in Section 5.3.2. Lastly, in Section 5.3.3 the realization steps and key results have been illustrated in detail.

5.3.1 Prototype Description

According to [84], *“Haptic perception is of vital importance to reading, and should be duly acknowledged. The reading process and experience of a digital text are greatly affected by the fact that we click and scroll, in contrast to tactilely richer experience when flipping through the pages of a print book”*. As today’s user community is widely accepting haptic interfaces, the effect of tactile feedbacks from the mobile reading materials has started to form a new genre of research interest. Our proposed Haptic E-Book (HE-Book) system is an e-book reader platform capable of delivering content related traditional media (audio/video) as well as vibrotactile feedbacks to the user’s reading experience. We have extended the proposed smart mobile interaction framework to implement the touchMe interactions with annotated eBook content. We use a eBook authoring editor to annotate certain paragraphs of an eBook and attach different context specific haptic commands in those. We have extended the Mobile Client class in order to develop eBook reader app for handheld devices. Haptic Jacket, Multimedia Image Player, Haptic Sofa, Audio Player are all developed by implementing Multimedia stub class and later added to the Multimedia Rendering Manager. We depict the component of the haptic book prototype in Figure 5.6.

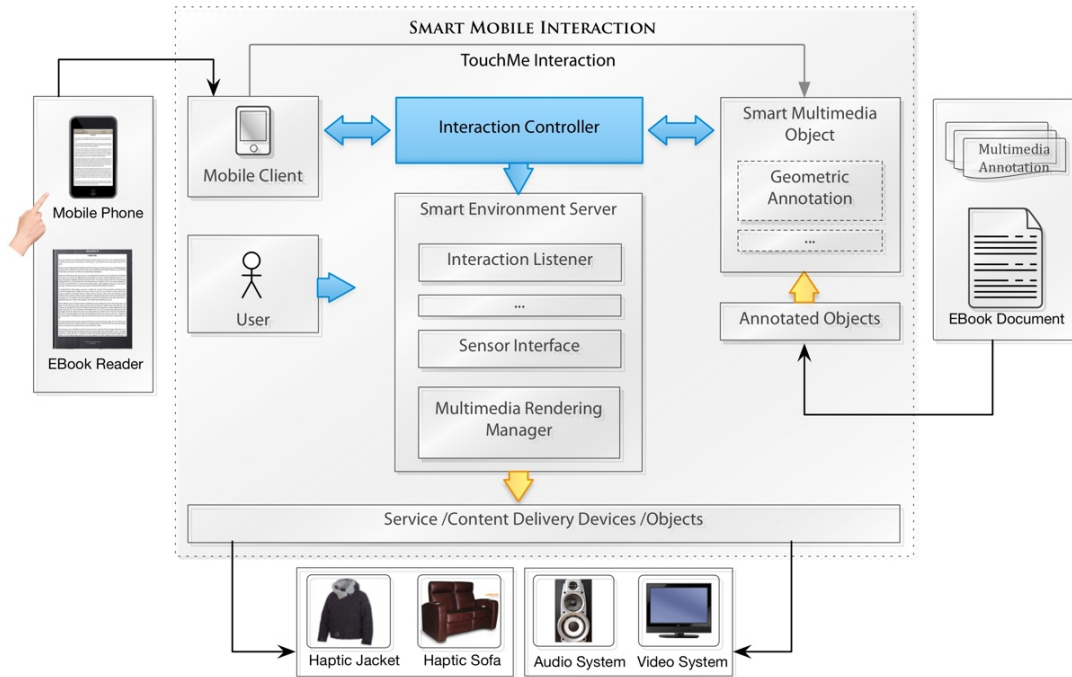


Figure 5.6: TouchMe based haptic interactions with an annotated eBook.

Reading Events

The haptic e-book system is deployed in a touch based mobile/tablet device by extending the Mobile Client class. Here, the user flips through the pages and scrolls the paragraphs of the pages using the touchMe based interactions. When the HE-Book reader system is used in a touch mobile device, the standard touch SDK available for that device is used to obtain the screen coordinates, using the *Reading Event Listener* component. The Listener uses the *Paragraph Locator* component in order to match the touched screen coordinate with the displayed e-book page area to deduce the paragraph that the user is currently pointing at. As soon as the user touches a paragraph of the page that has been previously annotated, the user is sent a motor vibration to confirm that and a valid haptic annotation for the selected paragraph is deduced. Further control is handed over to the Interaction Controller in order to render Haptic commands with the help of Multimedia Rendering Manager.

Reading Event Listener

When a reading event is generated and corresponding page and paragraph of the eBook content is detected, control is handed over to the *Interaction Controller* (IC) module which initiates reading event processing step. In this phase, *Annotation Manager Interface* of Interaction Controller uses the *Annotation Mapping Service* component of the *Annotation Manager* module to retrieve multimedia and haptic augmentation information from the annotated XML file. This component uses the previously calculated page ID and the corresponding paragraph ID information. The *Annotation Mapping Service* component returns the corresponding multimedia-haptic annotation details to the *Interaction Controller*, which continues to coordinate with rest of the system modules.

5.3.2 Annotation of IoT

Using the developed annotation scheme, an author selects a paragraph of a certain page of an eBook and tags those by using various haptic and multimedia properties. There are four supported haptic devices: haptic jacket, haptic armband, haptic sofa, and haptic mobile. For each of these devices, there is a list of supported vibration patterns. Those patterns are configured by using key control parameters for each haptic device. For *multimedia:image* annotation, the annotator is provided with a list of possible images from a keyword based web search. Later, an author selects a relevant image from the list and thus image metadata is attached to the annotation file. The annotation editor of the haptic book prototype is shown in Figure 5.7. All the multimedia annotation data for an eBook is stored as an XML file and later is processed by the Haptic-Book reader.

Advantage of separate XML based annotation file is that, a HE-Book user can decide to either use the multimedia and haptic extension or avoid it. For the annotated paragraphs of an e-book, we have created an XML element *Page*, which has attribute *ID* that denotes the unique number of the page. Every *Page* element is divided into *Para* that also has *ID* that denotes the paragraph number. Granularity level of our annotation ends at *Para* i.e. vibrotactile feedback and multimedia playback start and end in a paragraph level, which we term as scene or paragraph based annotation. Under the *Para* we have elements *Haptic* and *Image* for our prototype purposes. These elements are later extended to support internal, external *audio*, *video* urls too. *Haptic* is described by using *Device Type*, *Pattern*, *Repeat*, and *Delay*. Whereas *Image* element has attributes *Src*, and *URL*. Influenced by our past work [100], we annotate the e-book content with *Touch*, *Fun*, *Hug*, *Tickle*, *Kiss*, *Poke* type haptic feedbacks into our HE-Book system.

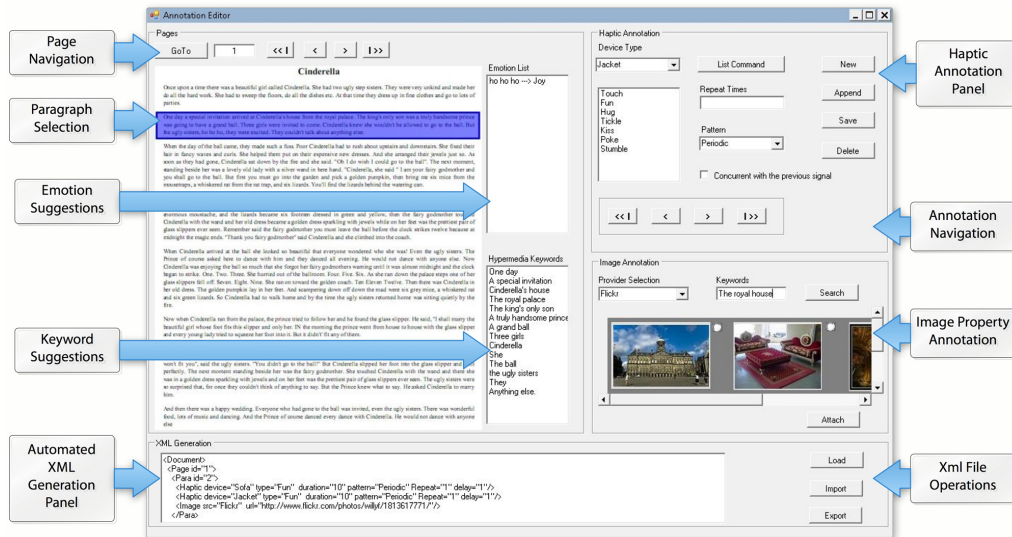


Figure 5.7: Annotation editor for the e-book document (i.e. PDF)

5.3.3 Realization and Measurements

Based on the page and paragraph information of *Touch Based GUI, Interaction Controller* gets haptic description from the *Annotation Retriever* sub-module and generates the described haptic signal for the targeted haptic device. *Haptic Signal Generator* plays an important role for device specific haptic signal generation. As the target haptic device can be heterogeneous, we considered device specific configuration. In our system architecture, we have considered Bluetooth as a method of communication between our system and the corresponding haptic device. Such communication can also be extended to other possible personal network communication methods based on the device support. In our prototype system, we have used a Bluetooth enabled haptic jacket, a D-Box haptic sofa, and a Bluetooth enabled haptic armband.

In our ubiquitous home multimedia setup, we have used Samsung LCD 46" 1080p TV for image/video display which is configured for HDMI usage. We also have a separate stereo sound system for the audio playback. Both of the TV and sound systems are connected with a mediator computer controller. Our mobile based HE-Book reader system is deployed in a low profile Samsung Omnia II mobile device, which is further wirelessly connected with a desktop computer, playing the role of a mediator. HE-Book annotation tool was developed by extending an existing open source PDF reader named

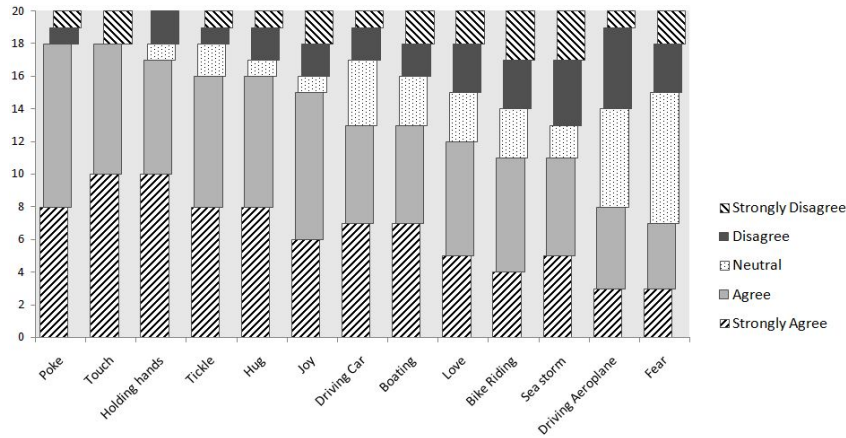


Figure 5.8: User response to various haptic feedbacks in Likert Scale

ICEPdf¹ developed using JAVA technology. We have added annotation facilities to the open source tool, where one annotator can annotate emotional/activity related haptics, scene related pics/video/audio etc. An annotation author can navigate between the PDF pages and select any paragraph to annotate.

After defining the haptic sensations, we have conducted usability experiments to understand how the users perceived the defined haptic feedbacks. Our target is to evaluate the user's quality of experience with our HE-Book system and to justify the suitability of the proposed approach. The usability tests take place at the university laboratory in a controlled environment with 20 volunteers of 12-18 ages and from different academic backgrounds. Among them 90% are from low to without haptic experience. Here, haptic experience means, utilization experience of test Haptic devices such as emotive jacket, Haptic arm band or Haptic sofa. Each user is requested to use the prototype setup and is given custom stories (PDF) to read. Each paragraph of the PDF documents is annotated with target haptic and related visual data. No haptic feedbacks are without visual support. According to our experience, haptics without visual support is difficult to follow. In addition, no haptic effect is repeated in a complete reading session for any user. Afterwards, they are asked to rate the following assertions, [*I felt myself "drawn in", I enjoyed myself, My experience was intense, I had a sense of being in the stories, I responded emotionally, I felt that all my senses were stimulated at the same time, Reading activity was not distracted, The system is easy to get familiar with, I would consider using this system*]. The ratings of the feedbacks are in the range of 1-5 (the higher the

¹Open Source Java PDF, <http://www.icesoft.org/java/projects/ICEpdf/overview.jsf>

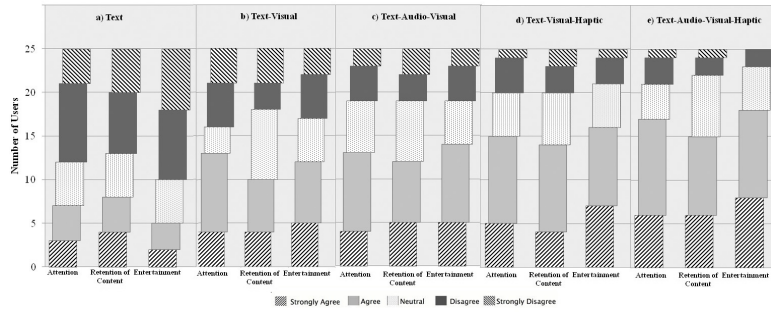


Figure 5.9: Usability study of the HE-Book system. a) Likert scale presentation of the attention, retention of content and entertainment metrics in a traditional text reading experience, b) The same study with text-visual rendering added in the reading process, c) Incorporating audio-visual simultaneously to determine the changes in reading process d) Visual-haptic feedbacks added in the reading process, e) Combining the haptic-audio-visual feedbacks in the paragraph reading experience to report the user scores.

rating, the greater is the satisfaction). Later, average rating to each haptic feedback for each user is projected in the final Likert scale of Figure 5.8.

From the empirical study, we have found that most of the users are neutral/disagreed about the haptic effect of *travelling aeroplane*, *fear*. Sea storm, boating, driving car, bike riding effects simulated by the haptic sofa are more acceptable. Overall, haptic jacket based effects are mostly appreciated above the haptic sofa. Our observation is that proper immersion scenarios and emotions depend upon the design/capability of the haptic devices and related visual annotation.

In our test (Figure 5.9), we have considered *Strongly Agree*, *Agree* as positive score and *Disagree*, *Strongly Disagree* as negative score. Overall, positive score of text based learning was 27% in our experiments which shows an improvement up to 47% when any visual media is augmented with the text content. We find that the effect of audio augmentation to any media mostly has increased the score overall 6%, whereas addition of haptic media has increased the overall score to 13%. Part of the reasons, for scoring higher while receiving haptic feedback can be due to the excitement that the users feel in receiving haptic feedbacks that relate with the learning materials. As visual feedbacks by itself improves memorization [21], hence our analysis after the comparison with *text-visual* and *text-visual-haptic* is that haptic also has an effect on retention. And, when the combination of *text-audio-visual-haptic* media are used with previously mentioned order, then we observed an overall score upto 67% which is 15% improvement over

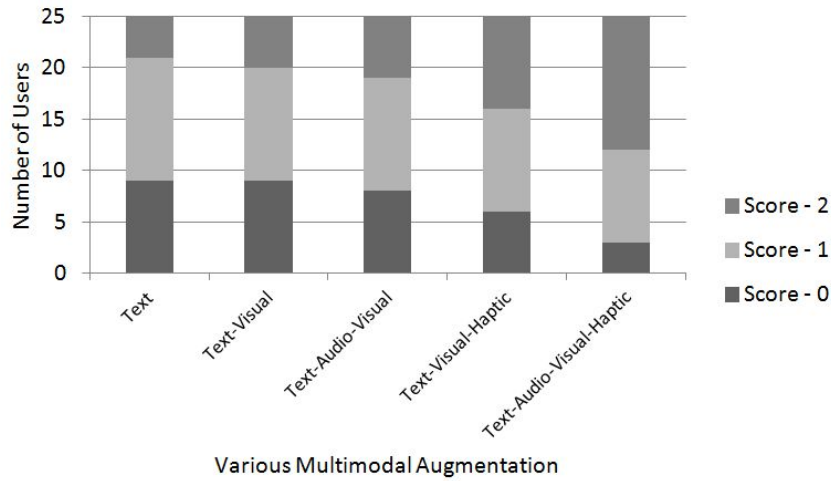


Figure 5.10: Knowledge acquisition scores of the users in different multimodal augmentation, where every user had to answer two questions related to each multimodality.

the *text-visual* or *text-audio-visual* based media augmentation. In figure 5.10, we have projected the knowledge acquisition score of all the 25 volunteers. Here for every type of multimodality, each user has answered two multiple-choice questions. For every right answer, we have recorded 1 point. From the figure we find that, overall knowledge acquisition score is the highest for text-audio-visual-haptic multimodality.

If figure 5.9 and 5.10 are compared, then a correlation can be established that is when Likert scale attention, retention score is high (i.e. agree, strongly agree) then knowledge acquisition score is also high (score 1, 2). However, there is always some users who has failed to make any scores for any type of multimodal augmentation. As in the study design, volunteers are given random fictions to read, so effect of story in attention, retention is not considered in the experiments. In order to get overall experience of the user, they are asked to make recommendations about the HE-Book system. Their immediate reactions were to improve the quality of haptic sofa vibrations to make it more enjoyable. In addition, they have suggested that some modifications to Haptic jacket design are needed. Nevertheless, overall they have enjoyed the reading sessions. When, they are asked the question, *Whether they would like to use a haptic book system in everyday life*, they have suggested that schools, colleges can have one such ubiquitous multimodal setup which is easily accessible for the students.

5.4 Summary and Conclusion

In this section, we have presented the details of two prototypes that have been realized by using the proposed smart mobile interaction framework. The prototypes first demonstrate with examples how virtual 3D avatars are annotated geometrically. By using THIN client based Mobile interface it is possible to touch the annotated virtual 3D avatars to send real haptic interaction commands to the real users. We demonstrate the 3D annotation process of a virtual avatar and present the scripting language that is necessary to accomplish the touchMe interactions. Next, in haptic book prototype, we illustrate an intuitive eBook system, where we annotate haptic data inside the eBook content. Based on the context of the eBook storm, bus riding, and boat riding haptic, audio, and visual commands can be annotated. When users touch the annotated parts of an eBook, haptic, audio, and visual contents are rendered to the user to augment his/her learning experience. Lastly, we present the detailed measurements data that we have obtained from the implementations of the prototypes with touchMe based interactions and report our findings.

Chapter 6

Annotated IoT in Edutainment

6.1 Introduction

We have incorporated the proposed framework and have realized pointMe, touchMe, and scanMe based interactions with IoT. In this chapter, we describe three prototype systems. In Section 6.2, we present a prototype that enables pointMe interactions with a time frame based annotated content on a Smart Multimedia Screen. We show the annotation of static slide based contents and dynamic continuous contents of a movie and illustrate pointMe, and touchMe based interactions with them. Next, in Section 6.3, we present pointMe based mobile handheld interactions with annotated physical atlas. We annotate major countries on the map to create pointMe interactions based innovative learning games. Lastly, in Section 6.4, we illustrate pointMe, touchMe, and scanMe interactions with an authored movie system. We annotate learning movies to make it possible for the learners to respond to learning based questions in the movie and actively participate in the learning process. For each of the prototypes, we provide short description of the prototype, describe the implementation challenges, relate the development of the prototype with the proposed framework, and illustrate the achieved results.

6.2 PointMe Interactions with a Smart Multimedia Screen

In this prototype, we extend the proposed framework to facilitate handheld device based PointMe interaction with annotated media content. In this prototype, the user points

his/her handheld device to the media presentation screen in order to interact with the presented media. The annotation approach incorporates customized Learning Object Metadata (LOM) format and are performed by specifying the spatial location on the media content in a particular timeline and further mapping related learning information or services to them. Spatial query is conducted over the annotated media contents in order to deliver interactive learning content to the user's handheld device. We incorporate IR based motion tracking [98] technology and handheld device's accelerometer data to approximate user's mobile position with respect to the media presentation screen. Further, we calculate the target direction at which the handheld device points. We perform experiments in a technology-augmented learning space to show the suitability of the proposed approach. To illustrate the system, first in Section 6.2.1 we discuss the details of the prototype with respect to the proposed smart mobile interaction framework. The annotation of the home things are explained in Section 6.2.2. Lastly, in Section 6.2.3 the realization steps and key results have been illustrated in detail.

6.2.1 Prototype Description

Mobile based learning material delivery as proposed by our work enables the system to track the progress of the user, provides relevant learning material that is of interest to the learners. We extend the proposed smart mobile interaction framework to realize pointMe-based interactions with the annotated media screen contents. We have developed two Mobile Clients in this prototype. First Mobile Client is used in order to send the accelerometer and IR emitter activation data to the Interaction Controller. The second Mobile Client is used to manage and organize the learning contents that the learner gathers from the live presentations. We have extended the Sensor Nodes interface in order to realize the 3D camera and IR camera subsystems. The components of the prototype are shown in Figure 6.1.

An IR camera is placed on top of the Multimedia screen in order to determine user's mobile position. The IR capture interface component maps the captured IR signals into motion points in a pre-defined viewport boundary that resembles the Multimedia screen. The proposed system incorporates two IR emitter points in order to calculate 2D and 3D motion points. In this section, we present a general walkthrough of the system considering a scenario, where the learner uses pointMe interactions in order to select the annotated vector LOMs (vLOMs). Afterwards, the learner added his/her recommendation for it. The flow of data has been shown in Figure 6.2 for clarification purposes.

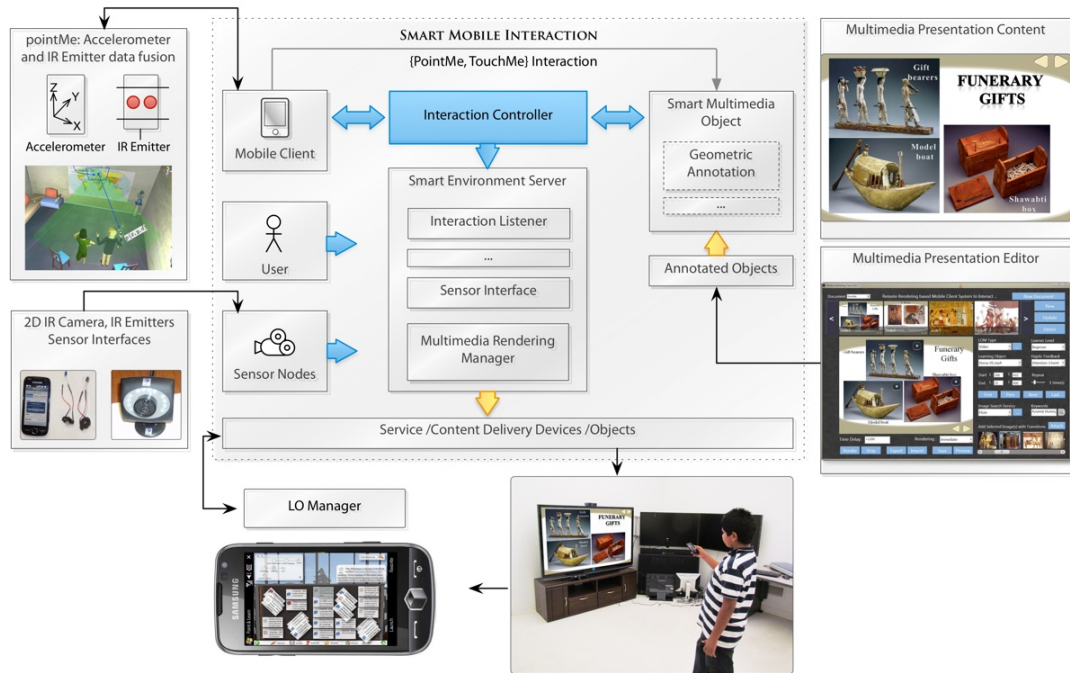


Figure 6.1: PointMe based interactions with annotated multimedia screen contents.

System Walkthrough

We begin our discussion when the learner activates his/her pointMe interaction by switching ON the IR emitters attached in his/her handheld device. In step 1, the system inputs the IR Camera message to determine the 2D IR emitter points with respect to the world coordinate system. Later in step 2, the coordinates are transformed into the Multimedia screen viewport. Further in step 3, the approximated z depth is determined. In this step, we determine the 3D alignment approximation and the system performs the scan conversion to obtain the targeted pointMe location in the Multimedia Screen S .

In step 4, the screen coordinate is taken into consideration to spatially determine the vLOM that has been targeted by the learner. Divide and conquer algorithm is used to efficiently scan the onscreen vLOMs to determine the targeted one. The system then inputs the learner profile information, looks upon his age, learning level, learning progress etc and updates the new interaction in step 5. Further, in step 6, the vLOM that the user targeted was calculated and suitable LO for the learner was sent to the interacting learner. The participating learner further updated the vLOM content, recommended it by using his/her handheld device and sent/shared that back to the MC . The MC attached the learners background information, created a new handle, and prepared new vLOM

package to be stored in the vLOM repository. User's ranking was also subsequently updated during this step.

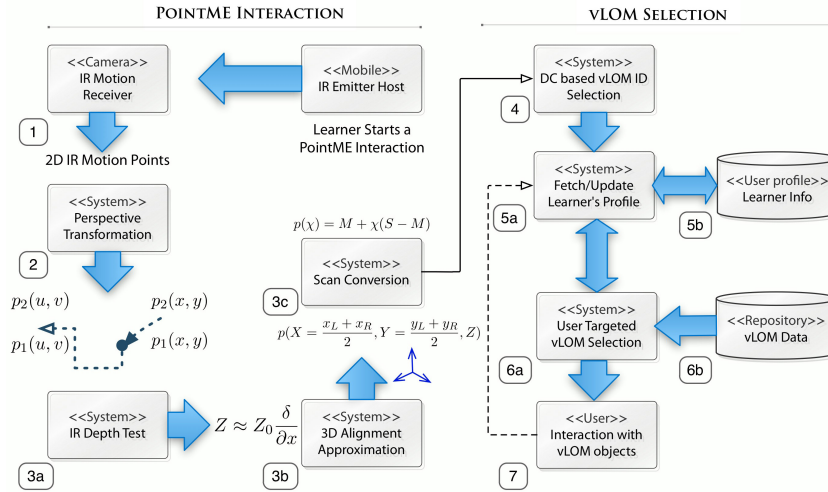


Figure 6.2: A description of the steps of the system after the learner activates the pointMe based searching and retrieval of the vLOM elements.

Mobile Client System

There are several modules in the overall system that performs different functionalities as depicted in Figure 6.3. The Search Interface invokes the Metadata Extractor module that retrieves LOM records from the distributed repositories. The search results are returned from the repositories, which are in XML format. The LOM-XML Parser module parses these results to extract the LOM elements. The parser is implemented by using the Java SAX ¹ parser. The parsed LOM elements are used by the LOM Clustering Module to cluster the search results into different groups. The 3D Graphics Engine is responsible for all the graphics generation and rendering mechanisms. It maps the different groups of search results to create the 2D interface by using templates of primitive graphic objects from the Metadata Template Repository. The User Manager authenticates the user's log in information and preserves his/her overall score, learning level etc information. It also maintains individual user profiles in the Profile Repository for providing a personalized learning environment.

The notice board metaphor that we have developed previously [105] have been re-implemented for the mobile delivery and organization interface of the vLOM contents.

¹Java SAX parser; <http://sax.sourceforge.net/>

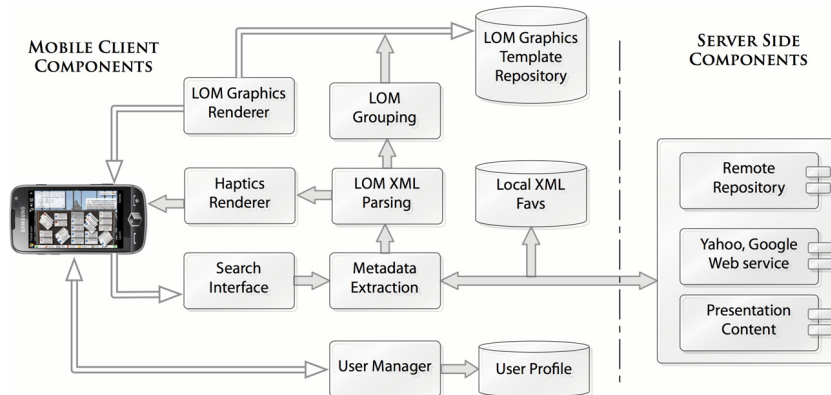


Figure 6.3: System architecture of the LOM viewing client.

The notice board based metaphor consists of a set of boards placed according to the users' preference. The metaphor has three wooden walls and a glass wall. The room, placed in an intuitive virtual environment, is enriched with a number of 3D rendering elements. Using touch interaction the user can easily navigate between different boards containing different groups of LOM information. The metaphor allows the user to tag a set of LOMs at any time, and store those to the directory list. Figure 6.4 shows a sample depiction of the implemented prototype.



Figure 6.4: Notice board based mobile metaphor application for viewing and organizing LOMs.

6.2.2 Annotation of IoT

In our proposed annotation scheme, the presentation content is pre-authored by the learning content provider or a teacher. We have developed a presentation authoring tool that is similar to YouTube ¹ based annotation scheme. In the authoring process, the teacher creates rectangular blocks on the scene and associates respective vLOM information with those annotations. For each of these vLOM annotated block, the teacher can further specify the length of time the vLOM will be available for interaction from the learners. The teacher also specifies vLOM type (*image, video etc*), learner's content selection level etc.

Unlike YouTube or video annotation scheme proposed in [12][22], where annotations are anchored to a point (or a range of time) in the timeline of a presentation, in our proposed approach, annotations are movable with the timeline. The generic presentation authoring tool is depicted in Figure 6.5. For example, when annotating a sprite in a scene, we define a path that resembles the sprite animation in the scene within a given timeline. The annotation is then attached with this path variable. Furthermore, as researchers have shown that traditional overlay based display of the annotation information is obtrusive [80][68], in our annotation displaying scheme, we display the visual cue to distinguish different types of annotations. In addition, during the interaction process, the user receives distinguishable haptic feedbacks when s/he selects different vLOMs on the screen. For example, in our annotation scheme, there are *quiz game, video, audio, html* type vLOMs that are distinguishable from their visual cues and haptic feedbacks. The learners by knowing the vLOM type will be able to point to it on the Multimedia screen and obtain the customize learning content in their handheld devices.

6.2.3 Realization and Measurements

We have realized the proposed mechanism for handheld device based Multimedia Screen interaction in a lab environment. We have integrated the different objects in this environment using a web-service based framework similar to [98]. We have placed one IR camera on top of the Multimedia screen that is attached with the room wall. We have used the Samsung Omnia 2 handheld device with .net compact framework support to implement the Mobile Clients. We have taken advantages of the mobile's touch based graphical user interface and have implemented the mobile interaction application. The integrated stream media player is used to play the LO content from the web-server.

¹Annotation, http://www.youtube.com/t/annotations_about

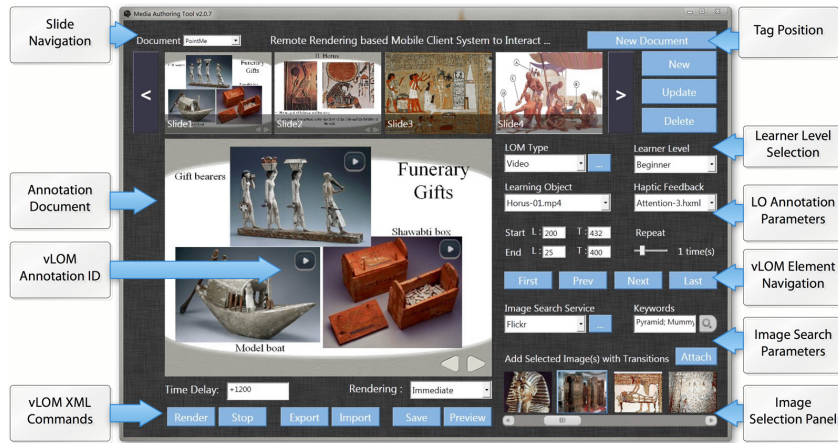


Figure 6.5: Presentation media authoring application. Any annotation of the media transforms along a path with the timeline of the presentation (if applicable).

As shown in Figure 6.6, two IR emitter nodes are attached with the handheld device and a switch to turn it on-off is attached on the back of the handheld device. On top right of the figure, handheld device and the IR emitter equipment are shown. The development platform used is visual studio 2010 .Net and CS was chosen as the primary programming language.

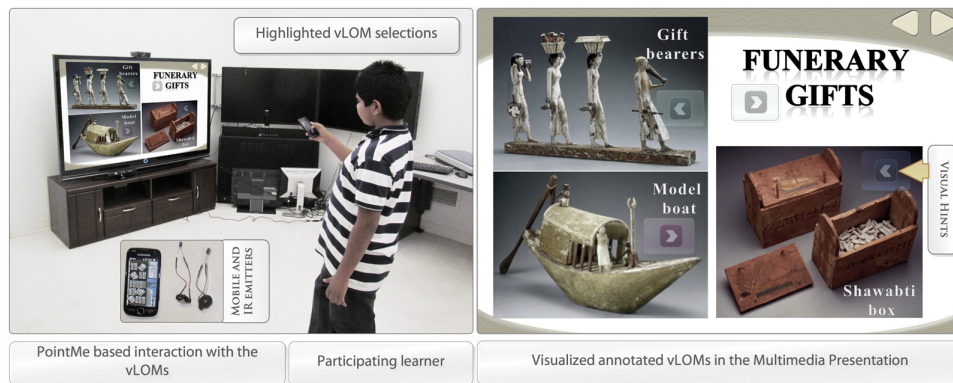


Figure 6.6: A learner pointing the targeted screen to select a vLOM using his handheld device. The IR camera, mobile and IR emitters are also shown.

The annotated objects are modelled as a rectangular prism that encloses a LOM object polygons on the screen. Therefore, our triangular geospatial search is dependent on five parameters namely the 2D position of the user P , distance d of the objects from P , right angle R , camera angle θ and the vertical tilt line PG . Typically, the horizontal ge-

ometric query execution requires more time than the vertical geometric query execution. However, in both cases the amount of used CPU power and physical system memory are negligible. For the above interaction scenario, the horizontal geometric scan required 30 ms and the vertical query scan required 20 ms. The overall time that the system required to respond to the user's interaction was 280 ms that scans three geometrically annotated LOM object polygon in the process.

The performance of the system is discussed in section 6.2.3, where the learners pointMe based selection accuracy is presented with different parameters. In section 6.2.3, the time complexity of searching and retrieving of the learning content is presented. Lastly, in section 6.2.3, the system's usability measurements are presented in terms of perceived ease of use, perceived usefulness and perceived control parameters.

Accuracy of selection

In our experiments, we have found that the learner's vLOM selection performance can be affected by inter distance of the vLOMs displayed on the screen, screen size to vLOM size ratio, distance of the user from the screen and lastly influence of the presence of point cursor in the screen in the selection process. The point cursor is an icon that denotes the current screen position where the user is pointing at using his/her handheld device. In each of the following experiments, we observe the user's selection performance by both hiding and showing the point cursor on the Multimedia screen and measuring the accuracy afterwards.

In Figure 6.7, we present the selection accuracy of the users in varying density of the vLOM elements. That is by modifying inter-distance of the vLOM elements, we observe that the selection accuracy changes. We have found that, it is challenging to the learners to select densely populated vLOM elements even when the screen pointer is present. However, as the inter distance of the vLOM elements increases after certain gap (around 6 cm) the selection accuracy remains the same throughout the experiment. Next, we observe whether the selection accuracy changes by modifying the user's handheld device location from the Multimedia screen. The result is depicted in Figure 6.8.

As could be expected the vLOM selection accuracy increases when the user is near the Multimedia screen and changes linearly with regression $R^2 = 0.98$ for visual aid based selection and regression $R^2 = 0.90$ for normal selection approach. This is because, when the learners are near the vicinity of the Multimedia screen, they have a clear view of the vLOM elements that they are trying to interact. The selection accuracy parameters obtained from these experiments are reflected back into the system to provide maximum

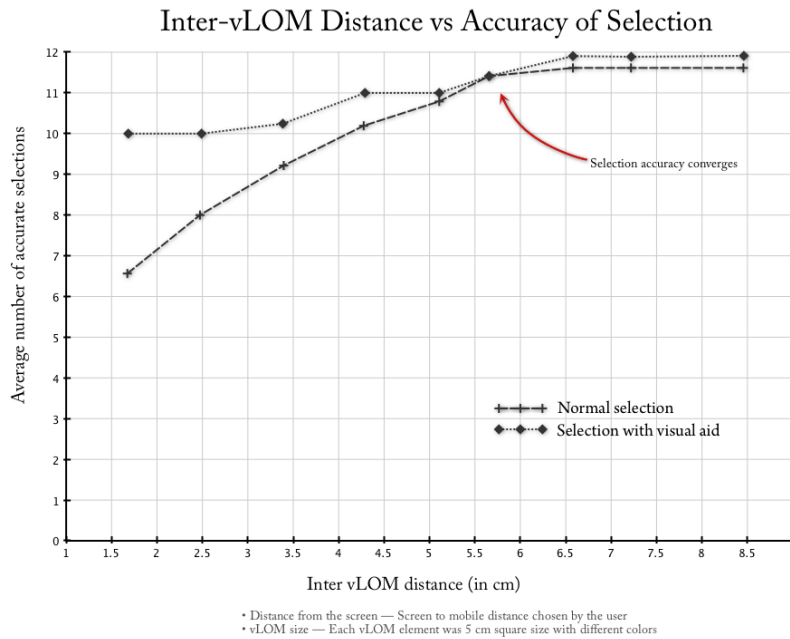


Figure 6.7: Comparison of the accuracy of the system with changing inter-distance of the vLOM elements.

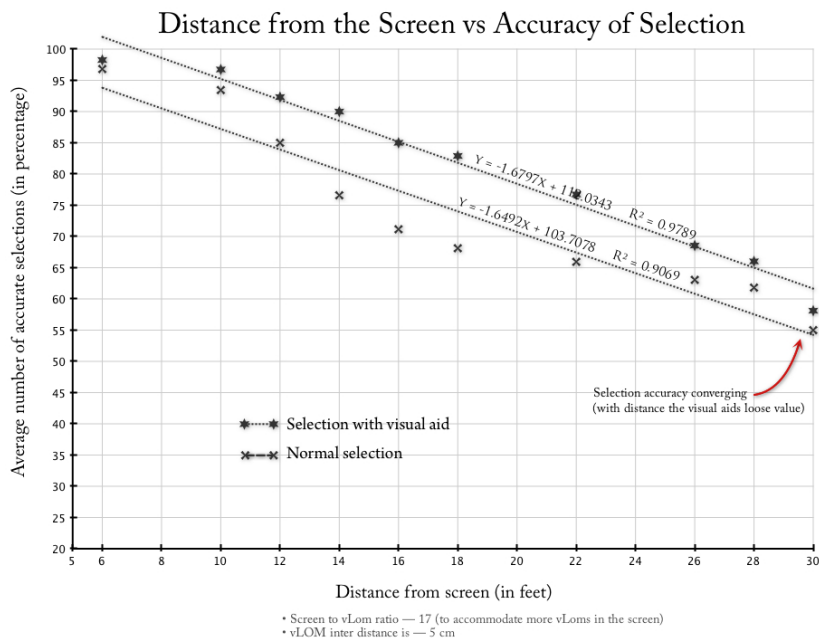


Figure 6.8: Accuracy of the system with varying position of the learner from the screen.

selection accuracy to the learners. For example, the Screen to vLOM ratio has been preserved to make the selection easier.

Time complexity

The time complexity of the selection and searching of vLOM elements are discussed in this section. One aspect that needs to be noted that the selection time or searching time was related to current number of vLOM that were present in the screen and the maximum number of vLOM elements for a given Multimedia presentation. However, it did not depend on the complexities of the presentation scene or length, type or content of vLOMs because those were stored separately as discussed earlier.

In selection of the vLOM we used divide and conquer algorithm. In the algorithm the base cases had constant-bounded size. The work of splitting the problem and combining the partial solutions was proportional to the problem's size n , and the number of three divided subproblems (namely to match the element or to look for the item in the left half of the screen or at the right half of the screen recursively) at each sub stages were bounded. Hence, the cost of this divide-and-conquer approach for finding n number of vLOM at a given time was $O(n \log n)$.

As the vLOM were stored in networked location, the vLOM retrieval time depended on network access and database access time. The time delays are depicted in Figure 6.9. As could be seen the retrieval time was linear with the regression $R^2 = 0.9474$. Nevertheless, in our Multimedia screen interaction application, time delay was almost not noticeable in the users selection tasks and mostly users were able to focus on pointing tasks as they were updated with the feedback promptly.

Usability Study

The usability tests took place at the university laboratory in a controlled environment with sixteen participants. We briefed each of the participants about the purpose of the study and how it is organized, and that we wanted them to think out loud as they were trying to carry out the tasks. We also showed the video on how to point and select the vLOM elements from the Multimedia screen and informed them the expected outputs. After that, we conducted the actual evaluation by consecutively handing out a sheet of paper describing the current task the participant had to carry out in the system. We identified personas of the participating users from the interview responses. Many of the user responses indicated strong tendencies towards intuitive interactions and

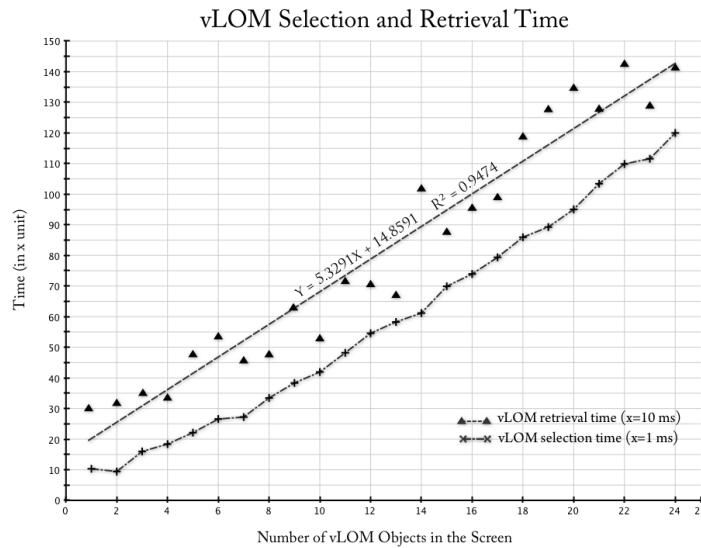


Figure 6.9: Searching and retrieval time of the different size vLOM elements.

incorporating new technologies (such as mobile pointMe interaction) in the traditional learning process.

During the study, an observer was taking notes on usability issues encountered by the participant. After all tasks had been completed, we conducted an interview with the users about their age, background, computer knowledge, their experience and learning behaviour while using the system. We also asked open-ended questions about their perception of the system’s usability. We also conducted an interview with respect to their learning behaviour. Before the study, we ran dry tests for each users to make sure that the assigned tasks and the interview questions were understandable. In the end of the experiments the participating users were told to fill out a questionnaire and to rate (in Likert scale [140]) some given assertions according to their usage experience of the system.

To have a better understanding of the user’s behavior in our learning system we divided the study into two categories. 1) The users obtained vLOM elements by pointing his/her handheld device. 2) The vLOM elements on a particular presentation screen were transferred to the users handheld devices as they activated IR emitter switches. The average responses based on the users answers were depicted in percentage form in Figure 6.10 for case 1 and in Figure 6.11 for case 2 .

When determining the Perceived Ease of Use of the system we focused on the aspects to find out whether the system was sufficiently easy to use and do not exclude any user

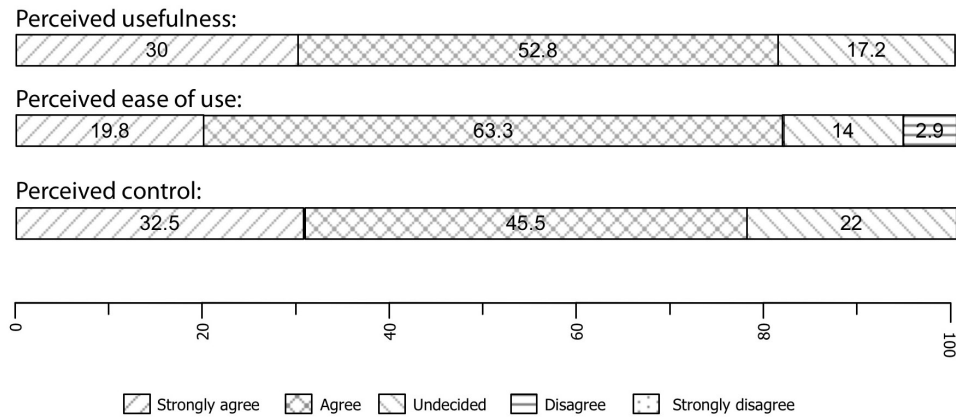


Figure 6.10: Usability study when the user is pointing in order to retrieve the vLOM elements.

in the process. In our study, we found that the user’s consented that PointMe interaction was clear and understandable, easy to use, and works more quickly. We asked the users to rate the following assertions to determine the ease of use of the learning system, a) Was easy to use b) Flexible to interact with c) It is easy for the learners to become skillful at using the system d) Predicted usage of the system if it is made available in the class room.

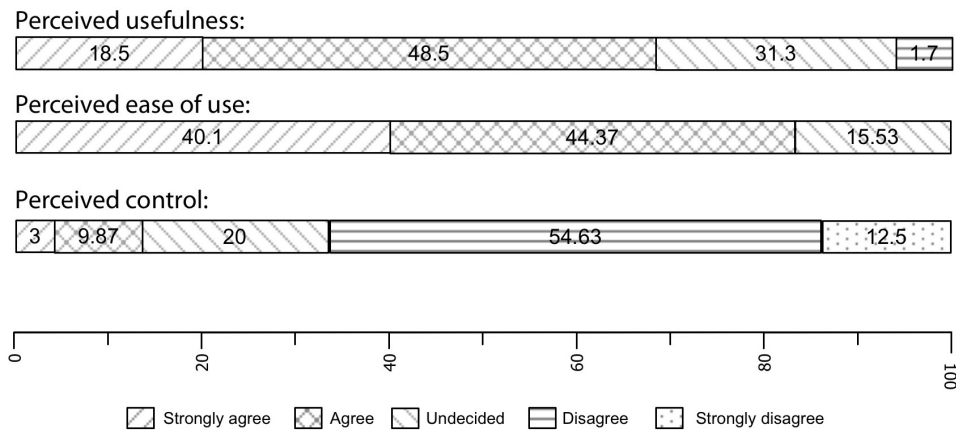


Figure 6.11: Usability study with automatic retrieval of the vLOM elements.

In order to find out the Perceived usefulness, which was a determinant of behavioural intentions to use technology [33], we focused on the topic whether the system is sufficiently useful in our experiments. In other words, whether the system provided enough

perceived learning value that the user would consider using the system on a regular basis. The participating users rated favourably that learning materials annotated in accessible and non-obtrusive manner and were useful in learning application. They also agreed that the interaction approach improves learning performance and recommended to be incorporated into a real world learning classes.

In order to rate the system usefulness we asked the user to rate the following assertions: a) The application increases learning productivity. b) The application enhances effectiveness of learning. c) The application makes it easier to perform a learning activity in a real world class.

6.3 PointMe based Interactions with Physical Atlas

The prevalent visions of ambient intelligence leverage natural interaction between user and available services in a learning space. In this pursuit, we develop a prototype by extending the proposed smart mobile interaction framework to facilitate handheld device based pointMe interaction with annotated physical atlas. In this prototype user points his/her handheld device to the world map in order to participate in learning based games. The proposed system performs geometrical annotations of the countries by specifying spatial locations on the atlas and further mapping related learning information to them. Each annotated data is encoded into customized Learning Object Metadata (LOM) format and it provides access point for available information about a specific country on the map. By incorporating a fusion of Infra-Red (IR) motion point and accelerometer data, the system determines the learner's pointMe location on the map. Based on the location, the system further calculates a particular country by performing spatial query over the geometrical annotations. In order to educate the learner about the world atlas, we design intuitive gaming scenarios by leveraging pointMe interaction scheme, where learners are challenged to select a country by listening to certain hints. This real world interaction technique with the physical map and seamless virtual learning information acquisition encourages the young learners to engage and concentrate in their learning activities. To illustrate the system, first in Section 6.3.1 we discuss the details of the prototype with respect to the proposed smart mobile interaction framework. The annotation of the home things are explained in Section 6.3.2. Lastly, in Section 6.3.3 the realization steps and key results have been illustrated in detail.

6.3.1 Prototype Description

Traditionally we know that the use of various modalities such as audio-visual materials can influence the effective learning behaviour [32]. According to Dale et al [32] effective learning styles are influenced by the use of modalities. PointMe is considered as a new mode of interactions that provides both independent and supplementary medium of communication channels. Fallahkhair et al. [41] argued that non-desktop technologies fit learning activities. In this prototype, we facilitate handheld device based PointMe interaction with annotated physical atlas, where the user points his/her handheld device to the world map in order to participate in learning based games. The learners use mobile pointMe [103] interaction approach to spatially search the annotated learning materials displayed on the annotated map. By incorporating a fusion of Infra-Red (IR) motion points and accelerometer data of the handheld device the system determines the learner's pointMe location on the physical map. By using the pointed location, a spatial geometrical query is performed over the annotated map objects. We extend the Mobile Client interface to implement Mobile app to assist the user in selecting the country of interest in an intuitive manner.

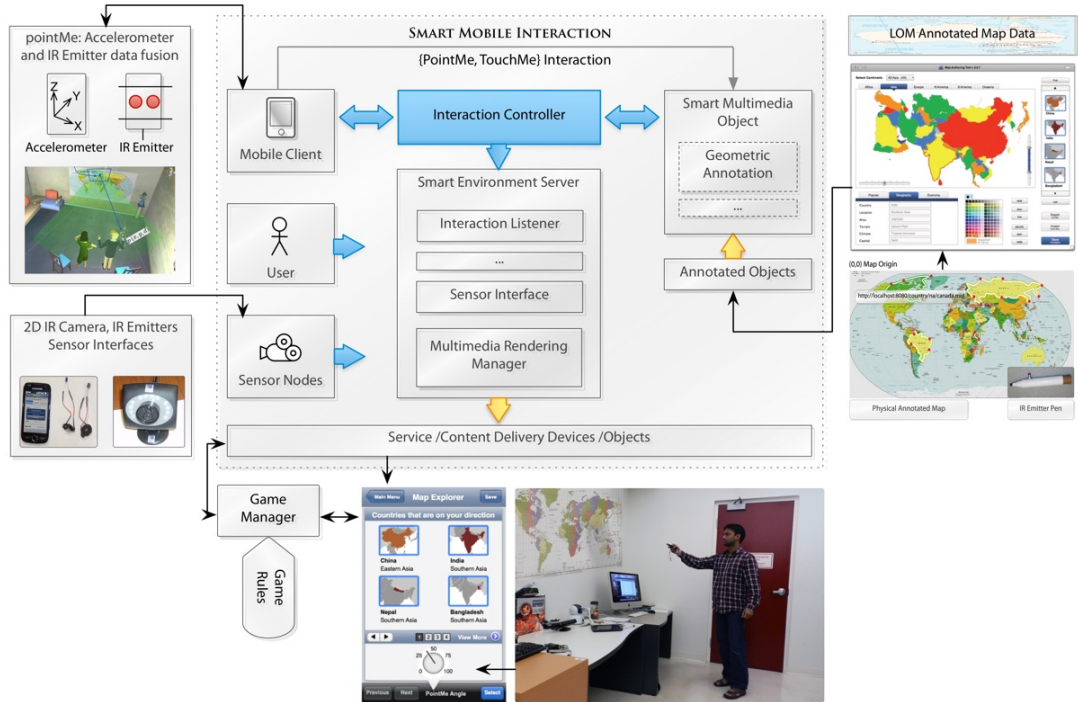


Figure 6.12: PointMe and TouchMe based interactions with annotated physical atlas.

Physical Map Selection

The target physical map selection is based on the fused sensing data, namely the 3D location information of the user's mobile and the tilt angle to the virtual screen. In order to find out the targeted physical objects, we have constructed geometric spatial query. In this query, the position of the user is $P(x, y)$. For different FOV angle we want to determine the spatial triangle $\triangle PLR$. Let us assume that only the objects, which are d distance away from the user's mobile, will be considered for selection. Now the position of the point F , which is opposite to P along the centre line of the FOV angle is $(x + d * \sin R, y - d * \cos R)$. Similarly, position of the left point L of the FOV triangle is $(x + d * \sin R - \theta/2, y - d * \cos R - \theta/2)$ and lastly position of the right point R is $(x + d * \sin R + \theta/2, y - d * \cos R + \theta/2)$. After calculating these three points, the process works by finding all the candidate vLOMS appearing in the viewing triangle and then providing them weight so that one possible target vLOM could be selected. The approach is depicted in Figure 6.13. The weight is determined depending on how close the candidate objects are with respected to the FOV centre line. It is also possible to show all the candidate objects inside the FOV instead of just selecting a single object. In this case the user would have to determine which particular objects she/he intended to select.

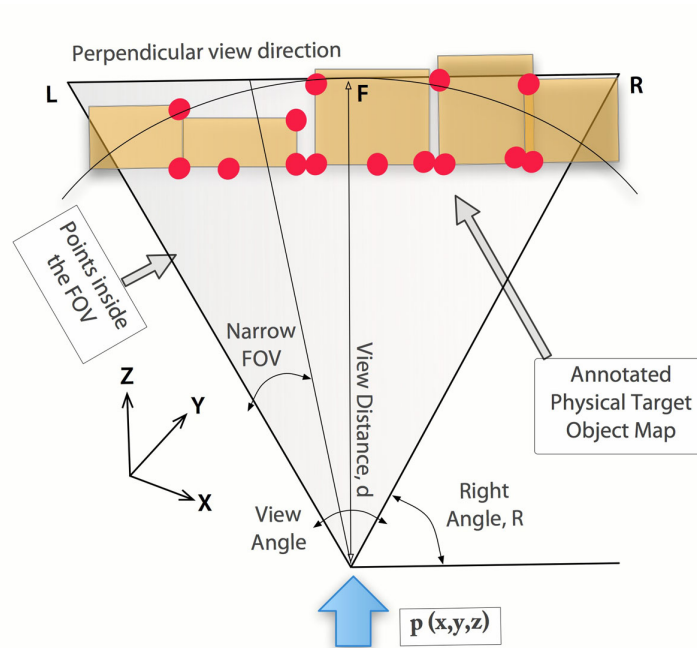


Figure 6.13: 2D IR point clouds and spatial selection query of the annotation map.

Our target object selection process is based on horizontal spatial query. In this approach, the FOV triangle is used to apply divide and conquer strategy to locate the physical object coordinates and a temporary stack is created to store the reference to those objects. These objects in the stack are sorted in ascending distance order from the viewer's position $P(x, y, z)$. Afterwards, another search to the sorted list is performed to find out the weights of the objects, which are more near to the centre line PF of the FOV. The sorted first element from the horizontal scan could be returned as a selected target physical vLOM. The annotated objects are modelled as a rectangular prism that encloses a country polygon on the map. Therefore, our triangular geospatial search is dependent on five parameters namely the 2D position of the user P , distance d of the objects from P , right angle R , camera angle θ and the vertical tilt line PG. Typically, the geometric query execution requires less time and the amount of used CPU power and physical system memory are negligible. The overall time that the system requires to respond to the user's interaction is less than 280 ms that scan thirty geometrically annotated vLOM polygons in the process.

Spatial Atlas Game

We describe the gaming scenario that we have designed by using the intuitive PointMe based selection approach. We define several gaming and scoring rules and develop a Game Interaction Controller to delegate the interaction and issue commands for the gaming interactions. The Interaction Controller transfer messages with the *Question Selection Module* and *Spatial Query based Map LOM Selection Module* to deliver appropriate vLOM data to the users mobile phones. Moreover, the progresses of the users are stored in their respective handheld devices as xml files. Hence, the PDA is used to store user profile and progress of the game. Below, we describe the developed gaming scenarios. In all of the following gaming scenarios, the users are given audio instructions by using the Text-to-speech (TTS) API¹ in the mobile phone speaker to perform certain actions relating to the game.

Scenario 1: In this scenario, the system speaks the name of the country and asks the user to locate that with their handheld device by using the PointMe interaction. For example, the system speaks *Locate* \langle *CountrName* \rangle . The user afterwards has to point and select the country in the physical map within a given time. The time duration is reasonable to allow the user to spend time learning about the world

¹Text to speech, <http://freetts.sourceforge.net>



Figure 6.14: Selection feedback from the game module in the mobile client.

map. The users are given three chances before a decision is made by the system on the correctness of their selections. In this case, users are either congratulated for their correct selection or given a brief explanation of the answer to the users to introduce them with more learning materials related to the country. The selection time and random country query information were stored as game rules.

Scenario 2: In this interaction scheme, instead of rendering the country names, hints are provided to the user about the country. Only name of the currency and capital are used as hints for a given country. The game rules are set so that the system could accept more than one answer for some queries. For example, the current version of the system for the following question, *Locate country that uses <currencyName>* accepts both USA, Canada as answers.

Scenario 3: The users are told to *Locate country that has < famous Landscape Name >*. For example, users interpret the command *Locate a country that has <Eiffeltower>* to be *France* and select accordingly. This scenario is targeted to increase the general knowledge of a user about certain countries. As discussed, in case of wrong answers, the mobile player reads out the complete answer along with details about that country.

The gaming experience is entertaining and engaging to the learners. Users report that after participating in the gaming interactions, they learn more about the atlas of the world. In our preliminary usability study, we have studied the users to evaluate their acceptance of these gaming scenarios and have tried analyzing the effectiveness of the individual scenarios in terms of learning parameters such as retention, ease of learning, and helpfulness in learning. After their gaming participation, we have presented these parameters to the 18 users and have inquired about their experience with each gaming scenarios. While the users have adapted with gaming *scenario – 1* very quickly they have argued that scenario 1 is simple enough and is not helpful in learning new things about the world map. However, the users have retained the country location information that they have learned from such experiences. Gaming *scenario – 2* is comparatively difficult to adapt as the hints are tricky. However, the hints and the description of the system have helped the users to acquaint with new things. Overall 78% learners have retained the hints and country information that they have learned in this gaming scheme.

6.3.2 Annotation of IoT

We define the annotation tag as a geometric identifier inside a physical environment with respect to the IR camera. In our experiment, we have deduced that the capturing process of the motion points of an IR emitter node in a 2D space is fairly accurate and mapping these points to lines is possible based on certain input patterns. In order to perform the annotation, we have used a custom built, inexpensive IR emitter pen to generate the successive motion points. A computer representation of the annotation map and the IR pen are depicted in Figure 6.15. When a user presses and holds the switch of the IR emitter pen it generates IR signals. The IR camera calculates the motion signals and converts them as motion polygons. Each polygon geometrically represents a country area on the physical atlas. A timeout of the IR emitter input distinguishes between two successive polygons. Further, The LOM information association has been conducted after the corrected visual shapes are stored in the computer. The annotation is a virtual placeholder to a physical object and does not interact with the object itself. It does not obstruct the object from the viewer. The geometric annotation tag is associated with a LOM and the user can access it by employing various interaction schemes such as PointMe. The geometric coordinate based annotation tag has many benefits if compared with traditional tagging methods.

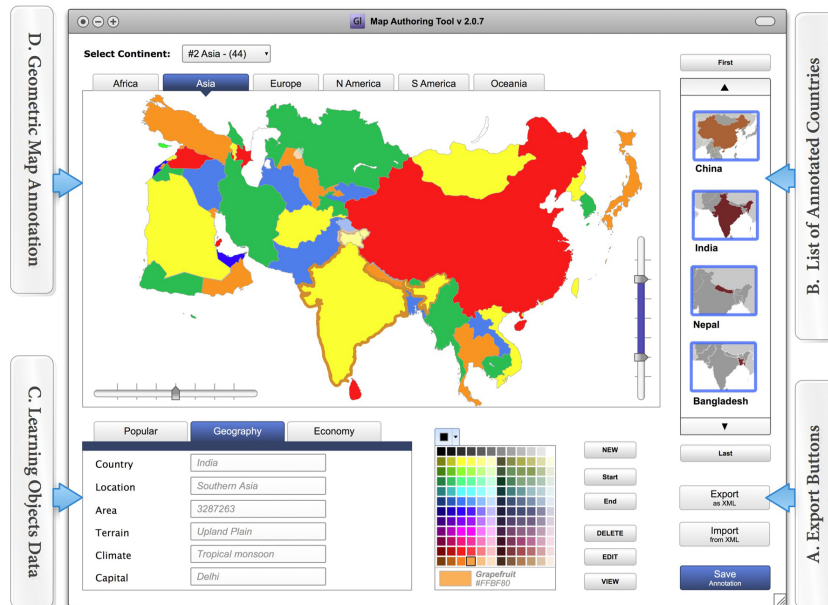


Figure 6.15: Map annotation and authoring scheme.

6.3.3 Realization and Measurements

This section presents the system component architecture and their communication relationships. As seen in the figure, the learner's handheld device communicates with the *InC* by using UPnP [98] protocol. Multiple handheld device can register with the *InC* by using this protocol. The LO access is a remote method invocation from the UPnP servers. In the user's PDA, *Mobile Interaction Application* communicates with the *Mobile Interaction Web Service* in the *InC*. The *Interaction Application* sends accelerometer data in this channel and obtains vLOM specific information from the *InC*. In this process the *Client Personalization Service* returns the learners profile information to the *Learner Specific vLOM Selection Service*. Also the *LO Navigator/Browser* returns the *InC* with IR emitter specific commands and obtains url type vLOM materials that are rendered in the mobile multimedia player. The *InC* uses the *IR Camera Unit* to determine the *IR Emitter Position* of the user's handheld device when required.

We have realized the proposed mechanism for physical mobile interaction in a lab environment. We integrated the different objects in this environment using a web-service based framework. We placed one IR camera in front of the physical object map that is attached with the room wall. We have used the Samsung Omnia 2 mobile phone with .NET compact framework support. We took advantage of the mobile's point based graphical user interface and implemented the mobile interaction application. The integrated stream media player is used to play the LO content from the web-server using the RMI interface. An IR emitter node is attached with the mobile phone and a switch to turn it on/off is taped on the back of the mobile phone. The development platform is Visual Studio 2010 Professional edition and C# was the chosen as programming language. The Interaction controller is configured in the web-server and an UPnP service is threaded in the same machine. The Interaction controller monitors the entire system message transfers and ensures consistency within the system's software elements.

Accuracy of Selections of vLOM

In this experiment setup, the distance of the mobile node is varied while attempting to select 20 different countries from the geometrically annotated map. The distances of the mobile node is within the range of 6 – 42 feet. The result analysis of this study is depicted in Figure 6.16. As can be seen, in this arrangement when the user is closer to the target physical objects the accuracy of the number of object selection is significantly higher. Close distance object selection often reached maximum possible accuracy; this

is due to the fact that in close distance there will be no less number of vLOM objects in the FOV of the user on which the spatial query will be performed. This is also due to the error in the distance calculation of the mobile node. At 6 – 32 feet if we aggregate the accurate number of object selection we can see that the accuracy rate is just above 90%.

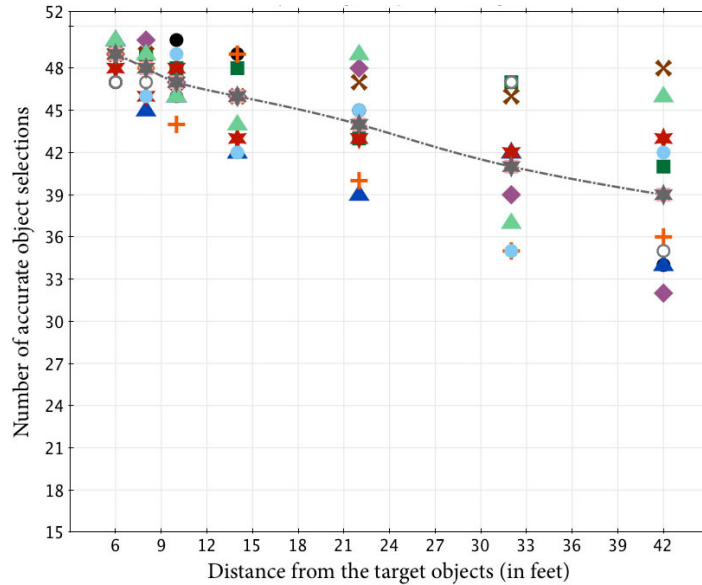


Figure 6.16: PointMe selection accuracy of the map LOMs.

The selected 20 countries are geometrically large in sizes (so as to avoid using the assistive selection scheme) and we consider $4cm$ of error boundary in users selections operations. Each symbol in the Figure 6.16 represent a user’s score on a particular distance from the physical map during the selection task. In the horizontal direction, we depict the distance of the users from the physical map during the experiments. For example at distance 6 on average most of the users are able to accurately point and select 18 countries on the map. However, as the distance increases we see the decrease of the selection performance of the same users. In order to evaluate the basic selection operation we turn off the assistive selection scheme during our experiments.

Time complexity

Now we present the time complexity of the selection and searching of vLOM geometric annotations. One aspect that needs to be noted that the selection time or searching time is related to current number of vLOM that are present on the screen and the maximum

number of vLOM elements for a given physical atlas map. In selection of the vLOM we use divide and conquer algorithm. In the algorithm the base cases has constant-boundary size. The work of splitting the problem and combining the partial solutions is proportional to the problem's size n , and the number of three divided subproblems (namely to match the element or to look for the item in the left half of the map or at the right half of the map recursively) at each sub stages are bounded. Hence, the cost of this divide-and-conquer approach for finding n number of vLOM at a given time is $O(n \log n)$.

As the vLOM are stored in networked location, the vLOM retrieval time depends on network access and database access time. The time delays are depicted in Figure 6.17. As could be seen the retrieval time is linear with the regression $R^2 = 0.9474$. Nevertheless, in our physical atlas interaction application, time delay is almost not noticeable in the users selection tasks and mostly users are able to focus on their pointing tasks as they are updated with the questions and question feedbacks promptly.

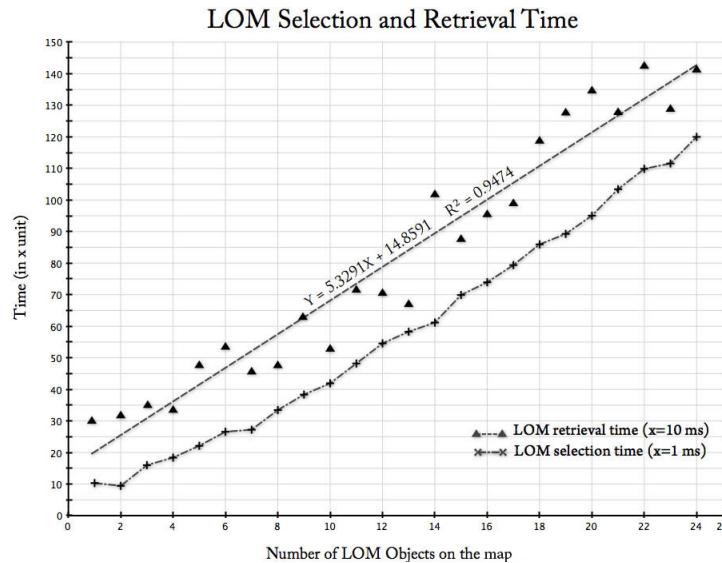


Figure 6.17: Selection and retrieval time of the map LOMs.

Usability Testings

In order to evaluate the proposed approach we set up experiments with 18 participants to determine some of the systems usability parameters. That is, whether 1) the interaction process is easy to learn and use in the learning environment, 2) the interaction paradigm is entertaining and interesting to the user, 3) the system is performing as an effective

learning tool and 4) the system’s learning process assists the users in remembering the subject matter that is taught. In our usability study, the users were categorized into two groups, namely adult and young groups based on their age ranges. Number of young users that participated in the study was 7 and their ages ranged between 14 and 17 (inclusive). The adult user group’s age ranged between 18 and 33. However, users from both of the two groups were assigned the same set of tasks in order to compare and better analyze their behaviours in the map selections.

Each user was given introduction to the system and briefed with the purpose of the experiment in the beginning. They were shown video clip of some usage scenario of the system and a dry run was conducted before they start with their assigned tasks. Selections of the target map objects were achieved by obtaining the associated mapped information of the object. Each user was told to point the mobile phone at the map and attempt to select a country before pressing the activate button (IR ON/OFF switch) in his/her mobile device. The user was then asked to view/listen the LO associated with the selected country and rate their experience. Afterwards the user was asked to point another country, which continues until all given objects for the intended experiments were selected.

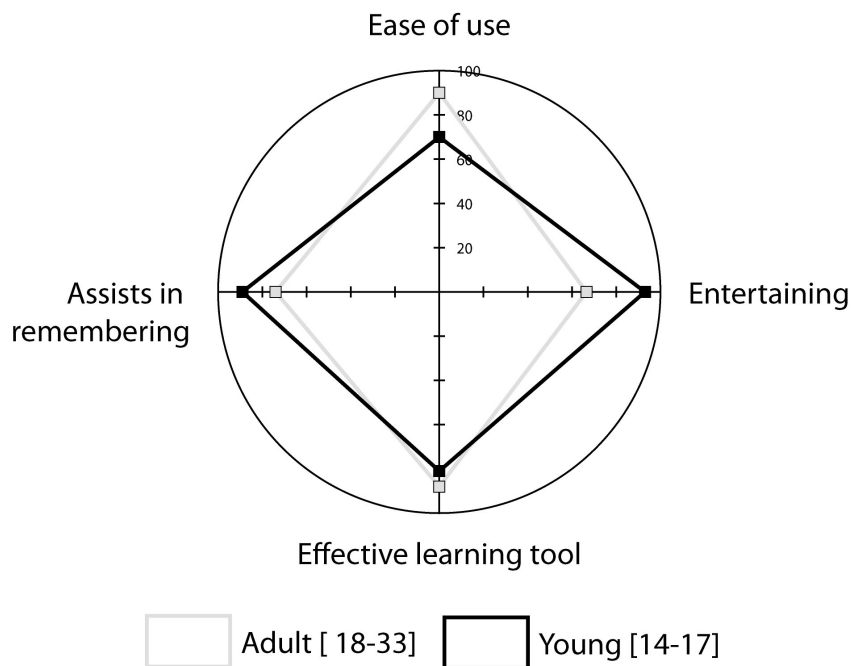


Figure 6.18: Average of the responses (in percentage) from Young and Adult users of the physical interaction system.

The participating users in the experiments were told to fill out a questionnaire and to answer some given questions according to their usage experience of the system. The average responses based on the users answer are depicted in percentage form in Figure 6.18. Overall, the young users responded favourably to the questionnaire compared to the adult users. For example 87% young users answered that the point based interaction approach to view learning information about a country from a map is entertaining and interesting while only 56.6% adult users responded the same. However, the learning curve for the young users was higher than that of adult users. 93% adult users found the interaction system to be easier to learn and be familiar with compared to 70.8% for the young users.

The users also rated some given assertions according to their usage experience of the system in Likert scale [79]. The average responses from the users are depicted in percentage form. In Figure 6.19, assertions where the users responded positively are shown. For example, 34% of the participating users acknowledged that in order to familiarize with the system probably it is a better idea to provide training to the users. They mentioned that without the introductory dry runs of the system at the beginning of their experiments, the selection performance could have been more inaccurate. Some users also explained that video demonstration of the usage were helpful to them and provided insight to the use of mobile device in physical object selection.

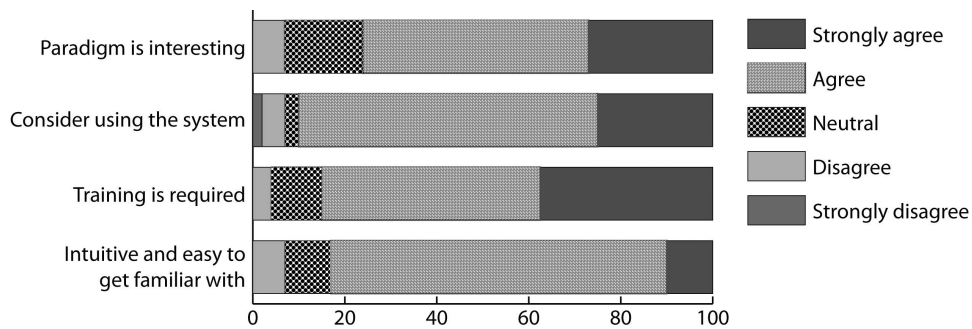


Figure 6.19: Likert scale response of the users of the given assertions.

Interactions with Neighbouring Annotations

In this usability experiment, we set up the study with 12 participants to determine some of the systems usability parameters. That was to compare the appeal and ease of use parameters for RFID and pointMe based approach to select 10 countries on a physical atlas. Selections of the target map objects were achieved by obtaining the associated

mapped information of the object. Each user was told to point the mobile phone at the map and attempt to select a country before pressing the activate button (IR ON/OFF switch) in his/her mobile device when using pointMe based interactions. In RFID based map interactions, the user scans the RFID chip by using a mobile RFID reader to select a given country. The participating users in the experiments were told to fill out a questionnaire and to answer some given questions according to their usage experience of the system. The average responses based on the users answer are depicted in Likert scale [79] in Figure 6.20. Overall, the users found pointMe interactions more appealing compared to RFID based interactions. However, on average the users have found RFID based interactions easier to use.

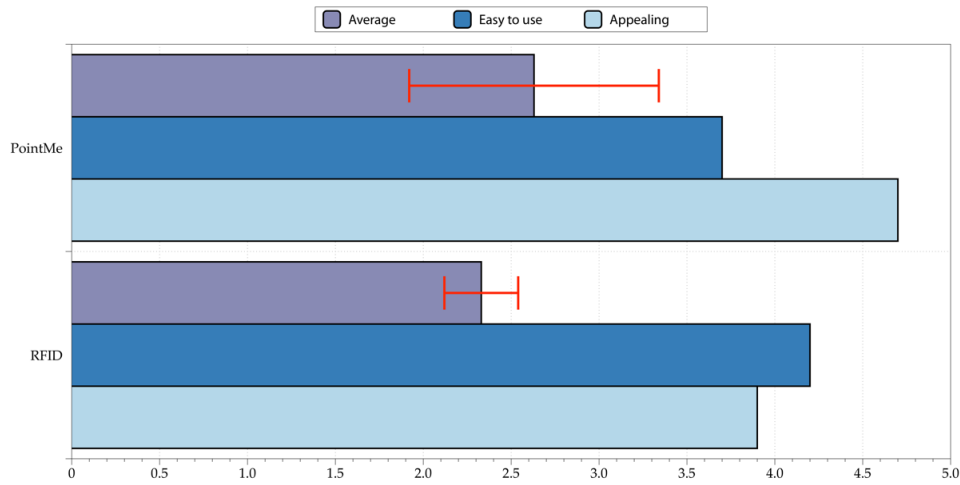


Figure 6.20: Usability study of pointMe and RFID based interactions with world map.

6.4 PointMe, TouchMe, and ScanMe interactions with an Annotated Movie System

Netflix, Hulu, etc are some of the most popular video content streaming services that are increasingly being accessed through many popular consumer devices such as Apple TV, Xbox, Wii, etc. It has now become possible to conveniently interact with the video contents by using the input hardware that these devices provide. We emulate the setups that many of these popular platforms provide in order to develop learning based video interaction games. The games leverage the user interaction feature with the video contents. In the award winning learning television series such as Mickey Mouse ClubHouse,

WordWorld, Super Why etc., the protagonists present learning based questions by using various scenarios and viewers learn the answers passively as they wait. In order to foster active participation of the viewers, we author the movie with learning questions at particular timelines of the video and provide interaction options. In those specified timelines, the learners interact with the presented questions by using Wii's pointMe or XBox Kinect's gesture based interactions and input answers. The interactions assist the learners to engage with the video contents and make it possible to actively participate in the learning process. In order to examine the suitability of the proposed approach, we perform usability experiments in a technology-augmented learning space and report our findings. To illustrate the system, first in Section 6.4.1 we discuss the details of the prototype with respect to the proposed smart mobile interaction framework. The annotation of the home things are explained in Section 6.4.5. Lastly, in Section 6.4.6 the realization steps and key results have been illustrated in detail.

6.4.1 Prototype Description

In order to bring active participation of the viewers, in our system, we author the movie with the learning questions at particular timelines of the video, where the viewers have interactions options. The authoring is performed by specifying the spatial locations of a video scene at particular timelines and further mapping learning questions to them. During the video playback, the system pauses the video at those specified timelines and produces intuitive game questions related to the played video. The answer choices are displayed as a form of geometrical markers/buttons over the video layer. When the movie pauses, the learners are able to actively participate and reflect on the content displayed on the screen. They are provided options to select the answers by using Kinect or Wii based simple and entertaining interactions. Edutainment systems, which combine entertainment with education are becoming popular form of e-learning. Studies [88, 108] have shown that entertainment improves learning skills by increasing learner's interests. We anticipate that the addition of gaming flavour in the movie watching experience will improve understanding of the viewed content and encourage active participation of the learners.

Interaction plays a vital role in promoting active learning and encouraging participations of the learners. According to Dale et al [32] effective learning styles are influenced by the use of interaction modalities. We contribute in this thesis by annotating the movie contents with learning object information to allow game like interaction with the movie

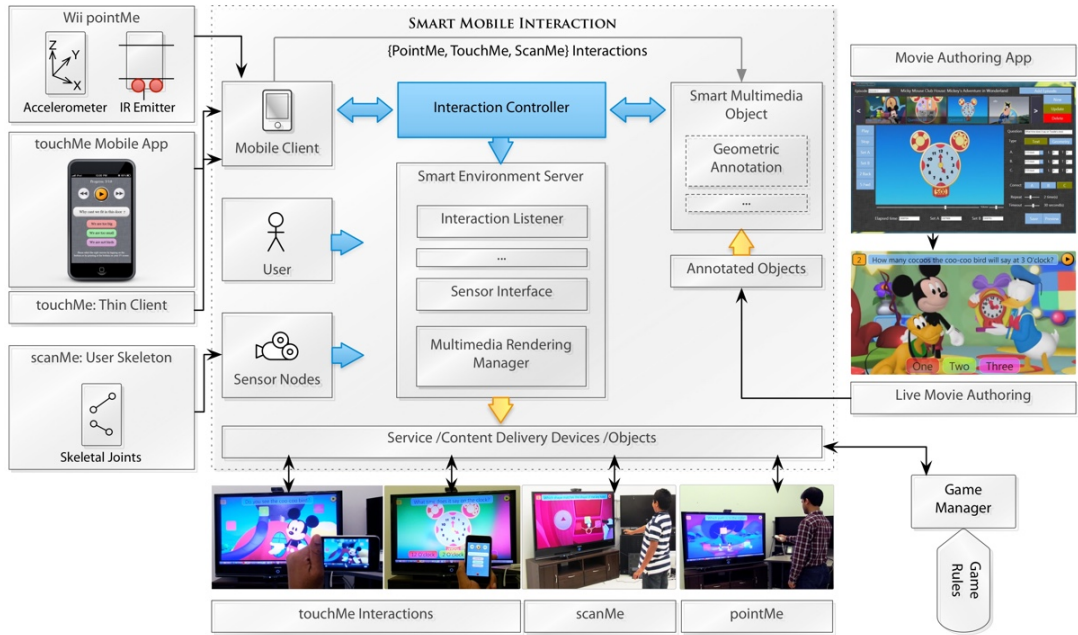


Figure 6.21: PointMe, TouchMe, and ScnaMe interactions with an Authored Movie System.

contents. By leveraging the intuitive interaction choices that are present in the gaming consoles and various set-top boxes, we promote active participation of the learners so that they can interact with the wealth of learning video resources in an intuitive and entertaining manner.

We analyze two different types of interactions in the learning video series. In one type of interaction, the protagonist asks the viewer about certain information that relates the previous discussion or the current visual elements on the scene. For example, in Figure 6.22, we see Mickey Mouse asking the viewer "How many coo-coo the clock bird says at 3 O'clock?". The answer of this question is textual as depicted in the figure. In this type of spatial game, we encourage the learner to contemplate, and listen the question carefully before answering the questions. In case the learner makes any error, the question is again repeated from "timeA" as defined in the LOM annotation file.

6.4.2 Spatial Television Game

We have analyzed two different types of interactions during the playback of the learning videos. In one type of interactions, the protagonist asks the viewer about certain information that relates to the previous discussion or the current visual elements on the

scene. For example, in Figure 6.22 (a), we see Mickey Mouse asking the viewer “How many coo-coo the clock bird says at 3 O’clock”. The answer of this question is textual as depicted in the figure. In this type of spatial game, we encourage the learners to contemplate, and listen the question carefully before answering the questions. In case the learner makes any mistakes, the question is again repeated from “timeA” as defined in the LOM annotation file.

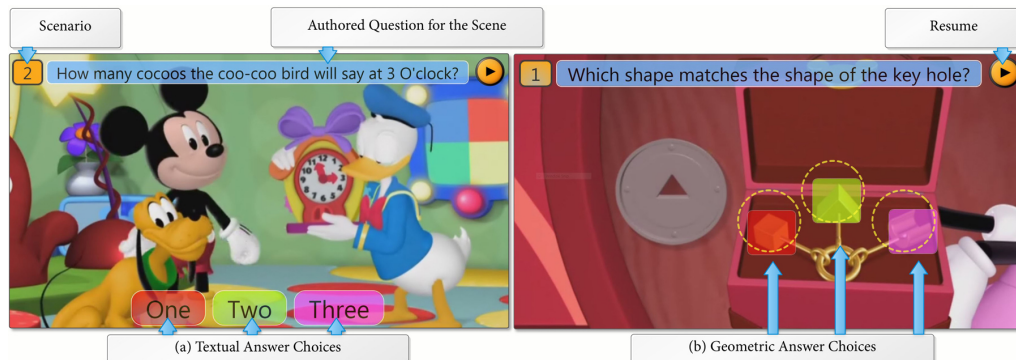


Figure 6.22: Textual and geometrical visual answer selection approaches.

In the second type of interactions, the protagonists ask the viewer to locate, identify, or search certain objects on the scene. These objects have certain shapes or visual hints by which the learners can identify them. For example, in the following Figure 6.22 (b), the protagonist asks the viewer “which shape matches the shape of the key hole”. The learners here must use the visual cue of the shape presented in the scene in order to determine the answer. In this type of gaming questions, we display the colored, transparent, and rectangular shapes over the visualized answer elements to provide visual cues to the learners for the possible answers. The learners take advantage of the interaction choices (pointMe or gesture) that we have developed to select the correct answer.

6.4.3 Wii based PointME Selection

We have used the hardware setup discussed in our previous work [98] and have incorporated an IR camera to track the users Wiimote. We have placed two IR emitter nodes on top of the media screen. An IR emitter is a small LED that emits IR signals that can be captured by an IR camera. Now, we want to determine the pointing location of the Wiimote on the media screen. We consider the IR emitter position m with respect to a known reference point, in our case the IR Camera origin. We know the screen parameters

of the media screen with respect to the IR camera parameters. The unit vectors $\vec{u}\vec{v}\vec{w}$ of the m are available from the Wiimote's accelerometer data. Using these available data we want to determine the pointing location on the media screen S . Using ray-tracing approach, the 3D parametric line from the IR target point M to the target screen point S on the screen is given by

$$p(\chi) = M + \chi(S - M) \quad (6.1)$$

Equation 6.1 represents that we advance from IR point M along the vector $(S - M)$ a fractional distance χ to find the screen projection point p . For simplicity reason we assume that $p(0) = M$ and $p(1) = S$, provided that $\chi \geq 0$ and $\chi \leq 1$.

In order to determine the pointing direction we need the Wiimote's position M and the Wiimote's axis unit vector (V_u, V_v, V_w) . First, we find the coordinate of S in the uvw coordinate system taking the IR point, M as the origin. Now, by considering the $S_w = \vartheta$ for the calculated screen point, the uv coordinates are found by the transformation that converts the coordinate space $[-0.5, \vartheta_x - 0.5] \times [-0.5, \vartheta_y - 0.5]$ to $[\alpha, \beta] \times [\pi, \tau]$. The transformation of the coordinate space is given by the following equation 6.2.

$$S_u = \alpha + (\beta - \alpha) \frac{x_{off} + 0.5}{\vartheta_x}, S_v = \pi + (\tau - \pi) \frac{y_{off} + 0.5}{\vartheta_y} \quad (6.2)$$

By using the user's approximated Wiimote location and accelerometer data, this pointing location at the screen is calculated. Furthermore, the screen location S is then spatially compared with the visual question elements that are present on the media screen in order to calculate the visual element that is pointed by the learner.

6.4.4 Kinect based Selection Approach

XBOX Kinect camera is a depth sensor that consists of an infrared laser projector combined with a monochrome CMOS sensor. The camera provides Application Program Interfaces (APIs) or to capture, process, and track a human skeleton in real-time. We devise a set of hand gestures that are recognized by capturing the motion-path of the hand skeleton. In order to capture the motion-path, we track the 3D hand movements (only one hand) [98]. As the camera tracks user's hand, it generates a sequence of motion-points of the hand joints, which are then analyzed syntactically to recognize the intended hand gestures.

The Kinect APIs can also be used to determine skeleton joint angles. When the user is moving the hands, the wrist, elbow and shoulder joints create angles that we calculate

in real time. We calculate the angle between the three joints that represents essentially three vectors. Whenever a user draws a gesture in the air using free hand movements, the motions of the hand joints produce a sequence of motion points. We transform these motion points into joint angles and later use them to calculate interaction gestures.

6.4.5 Annotation of IoT

In our proposed annotation scheme the presentation content is pre-authored by the learning content provider or a teacher. We have developed a presentation authoring tool that is similar to YouTube¹ based annotation scheme. In the media authoring process, depending on the learning question we annotate the screen at that timeline with geometrical or textual data. The authoring tool is depicted in Figure 6.23.

We target to provide interaction options at different timelines of the movie. We store the timeline information with the annotations. Next, we define the question that is relevant to the video learning content in that scene. For each question, we include at least three possible answers and also define the correct answer of the question. If the learning scenario involves locating some objects or things in the scene then the answer options are of the geometrical forms. i.e., the answers are displayed as an overlay with colorful rectangular blocks. In order to define the spatial coordinates of these colorful answer blocks in our authoring tool we click on the scene and store the coordinates. In this case, we also change the answer type to be of “Geometry” type. For the “Text” type questions, the answer elements are displayed on the bottom of the screen. The coordinate, size, and color parameters of these answer elements are selected by the media presentation module at runtime based on the screen resolution.

In our annotation step, we also define the maximum number of turns the user gets to answer a question. In case the user does not respond to a question, the movie automatically resumes after a timeout period. This timeout period is defined according to the difficulty level of a question. The varying timeout parameter allows the user sufficient time to contemplate on the questions. The annotation data of the movie is stored in a separate file, which is similar to a subtitle file. While watching the movie, parents both add this annotation file with the movie and create provision for the interaction for their children or they watch the original movie as they find suitable.

¹Annotation, http://www.youtube.com/t/annotations_about

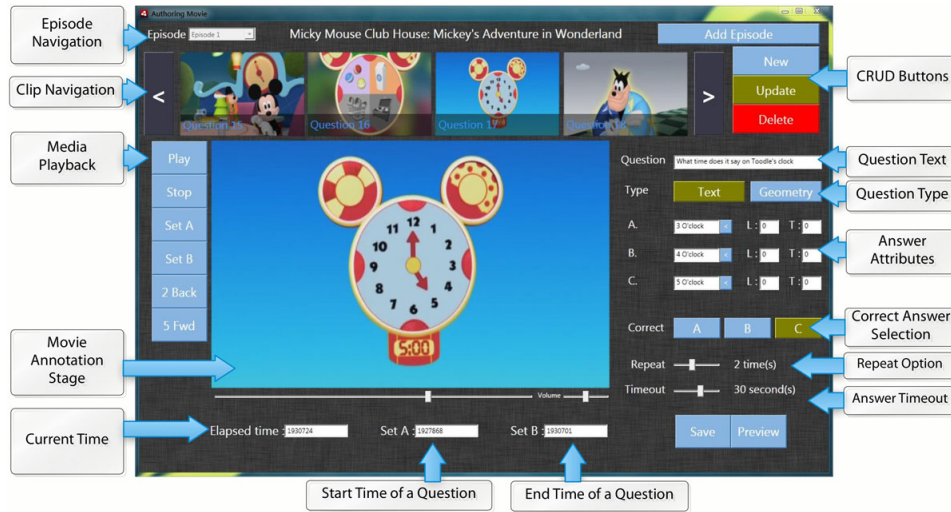


Figure 6.23: Presentation of media authoring application. Any annotation of the media transforms along a path with the timeline of the presentation.

6.4.6 Realization and Measurements

We incorporated the usability evaluation guidelines [28] to qualitatively measure our proposed system and designed our tests accordingly. Before performing the usability test we designed a test plan, where we defined our evaluation objectives, developed questions for the participants, identified the measurement criterion and decided upon the target users of the system. The usability tests took place at the university laboratory in a controlled environment with fourteen (14) participants comprising of different age groups. All of our participating users were in the age between 4–9 years and they were accompanied with their parents. Among the fourteen users three (03) of the participants were in age group 4–4.5, four (04) of them were in age group 4.5–6 and the rest seven (07) were in age group 6–9. In order to measure the acceptability of our system we also grouped the participating kids into two disjointed groups namely Group A and Group B.

Before we started the usability test, we installed our developed prototype system into a desktop computer, which was connected with a large 65” Samsung 8000 Series HDTV. The Kinect sensor and the Wii controller were connected with the computer through USB and Bluetooth connectivity. For better results we positioned the Kinect sensor 2 feet (0.6 m) from the floor and the participating user were stand back 6 feet (1.8 m) from the sensor. During the study, an observer was taking notes on usability issues

encountered by the participant. At the beginning, we briefed each of the participants about the purpose and the organization of the study. We explained the participants that we wanted them to think aloud as they were carrying out the tasks. As each kids had to participate individually in the experiment, the parents accompanied their kids to make them comfortable. We also demonstrated how to point and select the interactive movie elements by using the Wii controller and the Kinect sensor.

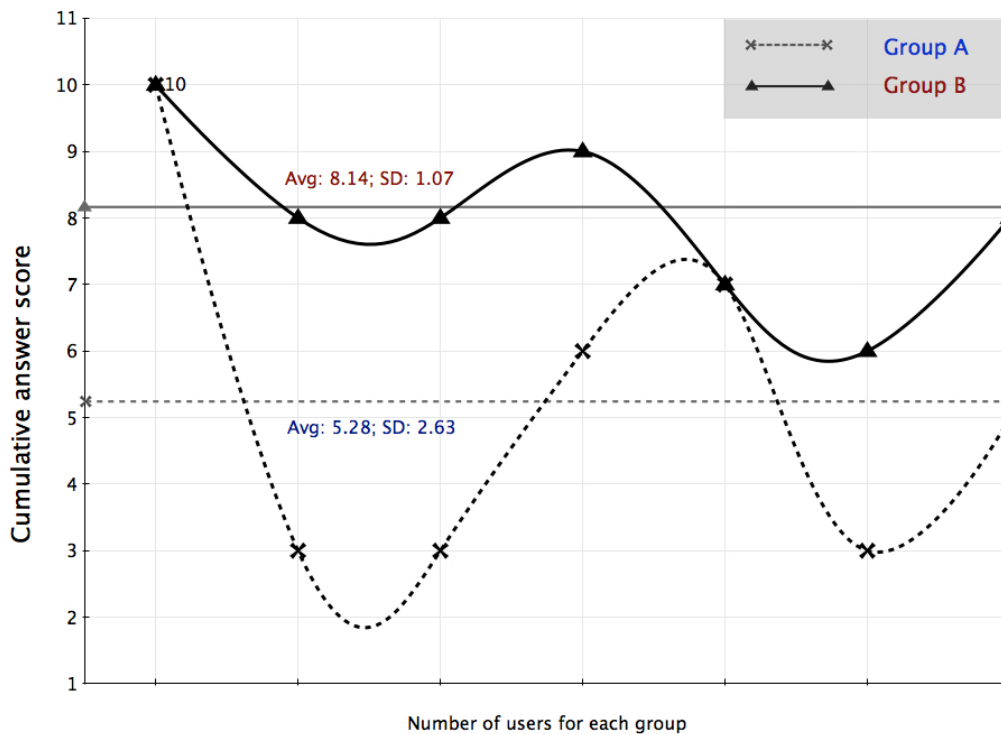


Figure 6.24: Comparison of the learning scores of two user groups with and without interaction options.

In Group A, each participant individually watched the movie in a traditional setting, where there was no option to interact with the movie. During the movie playback, the participants did not engage in any learning activities other than watching the movie. On the other hand, in Group B, each participant individually watched the same movie, where there was interaction options. While watching the movie, each participants individually answered the questions that were presented to them in the movie by using Wii or Kinect interaction options. In case the answer selections were wrong, the learning questions were repeated to help the learner understand the scenario. At the end of the experiments, the young viewers were asked to answer ten questions related to the viewed movie content.

In order to measure the acceptability of the system we noted their answers for analysis. The cumulative learning score for the viewers of Group A and Group B were illustrated in Figure 6.24. In this figure, the dotted and solid horizontal straight lines indicated the average scores for Group A and Group B respectively. It was clear from the figure that the learners of Group B achieved more learning score than that of Group A. The lower standard deviation of the learning score of the viewers of Group B indicated that the score of the users in Group B were close to the average value. The accurate number of selections by the users by using touchMe app, location-aware, pointMe, and scanMe based interactions were presented in Figure 6.25.

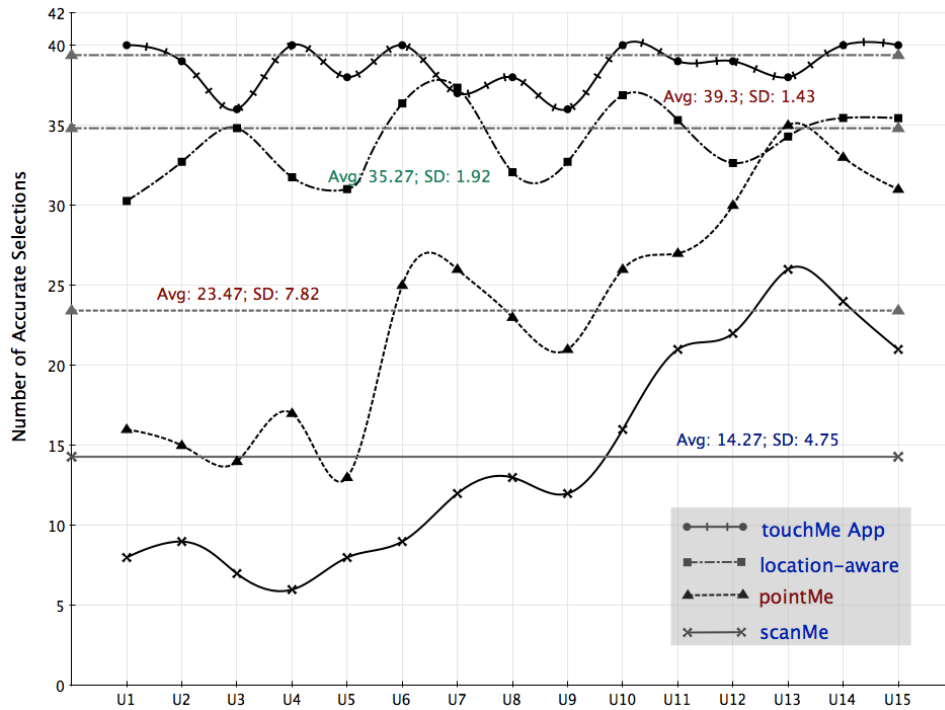


Figure 6.25: Comparison of the number of accurate selections by the users.

After all tasks had been completed, we conducted an interview with the parents about their kid’s age, background, experience, and learning behaviour while using the system. We also asked open-ended questions about their perception of the system’s usability. We also conducted an interview with respect to their children’s learning behavior. The parents were told to fill out a questionnaire and to rate (in Likert scale [140]) some given assertions according to their observation of the system. The ratings of the questionnaire were in the range of 1–5 (the higher the rating, the greater is the satisfaction). The usability test questions in the questionnaire to the parents are listed in Table 6.1.

Table 6.1: Usability test questions for the parents.

#	Question
Q ₁	If this system is made available in the movie series please rate whether you would be using the tool at home with your children.
Q ₂	During the interaction-based movie viewing, the children asked more questions compared to non-interactive viewing and raised more discussion on the topic.
Q ₃	Do you think this type of interaction options can positively influence the participation on the children and engage them in learning?

It was interesting that many of the parents indicated strong tendencies towards intuitive interactions and incorporating new technologies (such as Wii pointMe interaction) in the traditional video based learning process.

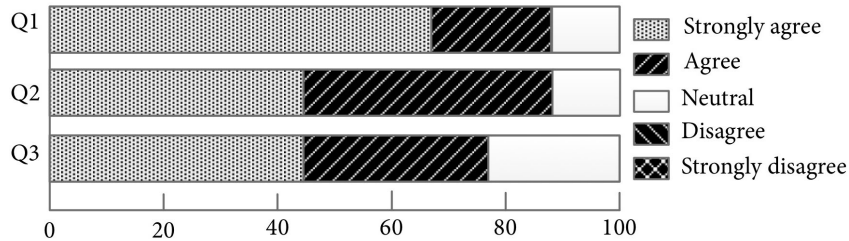


Figure 6.26: Parent's response on Likert scale for the given assertions [Q1~Q3] as listed in Table 6.1.

Their responses based on the questionnaire were shown in Likert scale in Figure 6.26. The average of the responses of the users were calculated in percentage form and measured after the usability tests. It was worth mentioning that more than 80% of the parents would like to recommend the interaction based movie for their kids at home. Also 45% parents strongly consented that during the interaction based movie viewing their children asked more questions. Table 6.2 summarized the overall performance score of the system based on the parents satisfaction. In this table low standard deviation indicated that the data points tend to be very close to the mean.

When determining the perceived ease of use of the system we focused on the aspects

Table 6.2: Parent’s satisfaction in Likert scale.

	Mean	Std. Dev.	Percentage of Mean
Q ₁	4.56	0.73	91%
Q ₂	4.33	0.71	86.67%
Q ₃	4.22	0.83	84.44%

to find out whether the system was sufficiently easy to use and did not exclude any user in the process. In our usability experiments, we observed that the participating learners liked the Wii based pointMe interaction scheme the most (72%). Accordingly, we found that kids who were 3.4 to 6 years old liked Wii based interaction scheme, where kids who were 6 to 9 years old liked Kinect based gesture interaction scheme.

6.4.7 Comparison between various Interactions

In this usability experiment, we set up the study with 11 participants to compare the movie authoring systems usability parameters when users incorporated RFID, Visual Code, pointMe, touchMe, scanMe, and Location-aware interactions. In each interaction experiment, the user selects and answers 10 interactive questions by using the movie authoring system. We explained the use of pointMe, touchMe, and scanMe interactions. In order to interact with the movie by using RFID, we prepared the following setup. We made a pad with three paper answer choices and hidden RFID chips behind the papers. The paper choices had "A", "B", "C" written as captions. In order to select the answer option "A", the users scan the paper with the "A" caption. For QR Code based interactions we prepared another answer pad with paper choices "A", "B", and "C" written as captions. Three separate QR Codes were printed on these paper buttons so that the user could select any of them to provide their answer choices by using a simple QR Code reader mobile app. In order to realize the location-aware based interactions with the authoring movie, we annotated three locations in front of the television and marked those places with "A", "B", and "C" respectively. In order to answer "A" the user stands on the annotation location that is marked "A".

The participating users in the experiments were told to fill out a questionnaire and to answer some given questions according to their usage experience of the system. During the experiments we also recorded the interaction actions of the users for further analysis. From the recorded data, we compared the accuracies of the selected interaction schemes.

In Figure 6.27, we present the interaction accuracies of the 11 users. On average selection accuracies for pointMe, touchMe, scanMe, visual code, and location aware interactions were 77.91%, 88.27%, 75.82%, 87.64%, 70.55%, and 84.18% respectively. As evident, RFID and touchMe based interactions provided better accuracy in the when selecting annotations in the learning movie system.

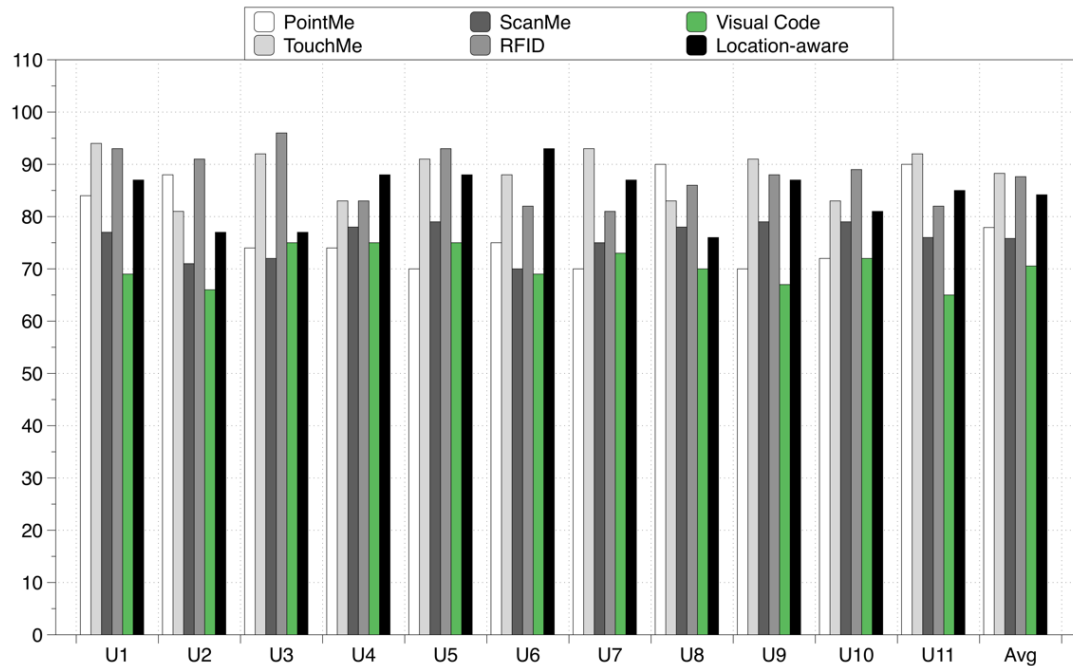


Figure 6.27: Comparison of the accuracies of various physical mobile interaction schemes.

In Figure 6.28, average interaction times to complete each of these tasks were listed. As evident, visual code interaction based answer selections required more time from the users. On the other hand, touchMe interaction based answer selections required the minimum amount of time from the users.

The average interaction times of pointMe, touchMe, scanMe, location-aware, RFID, and visual code based interactions were shown in Figure 6.29.

In the usability experiments users were requested to rate a number of given assertions in Likert scale. Based on their report, the ease of use of various physical mobile interaction schemes were listed in Figure 6.30. As evident from the figure, the results were mixed. The location-aware interactions, touchMe interactions, RFID based interactions all were rated favourably by the users as easy to use interaction systems.

We also asked the users to rate the physical mobile interactions based on their enter-

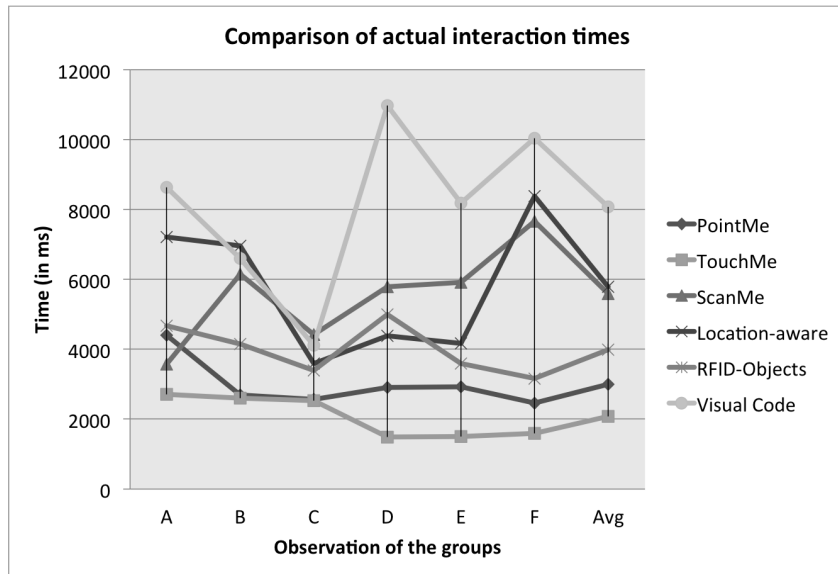


Figure 6.28: Comparison of the interaction times of various physical mobile interaction schemes.

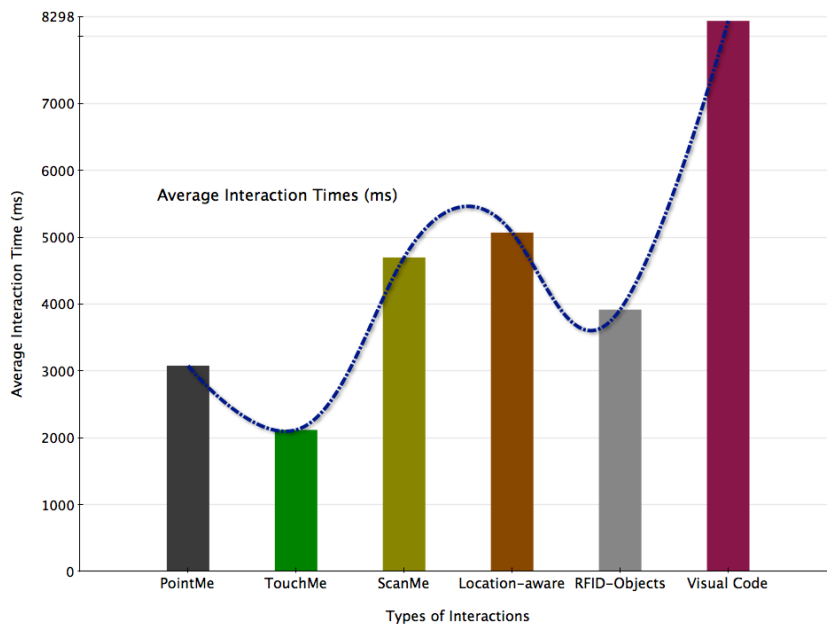


Figure 6.29: Comparison of the average interaction times of various physical mobile interaction schemes.

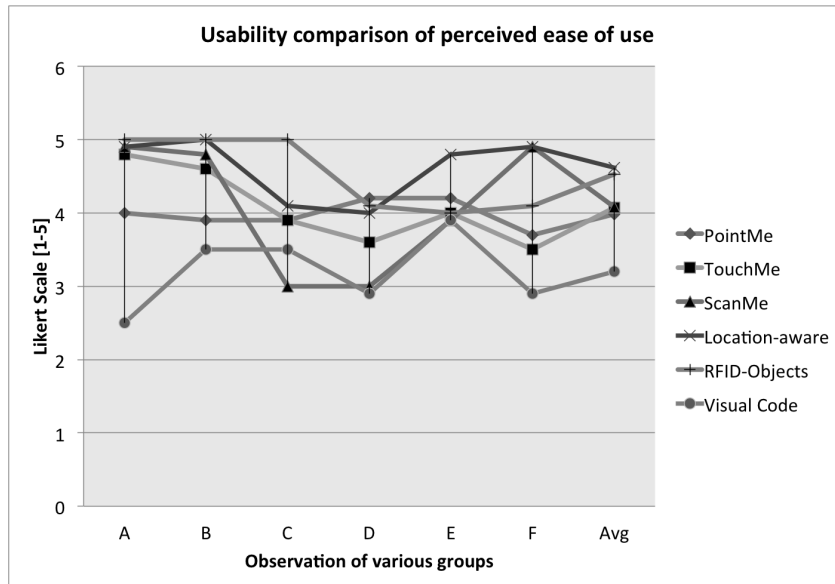


Figure 6.30: Usability comparison of perceived ease of use of various physical mobile interaction schemes.

tainment values in Likert scale. Based on their reports, the application and entertainment values were depicted in Figure 6.31. As evident from the figure, average entertainment values for pointMe, touchMe, scanMe, location-aware, RFID, and visual code based interactions were 4.53, 3.85, 4.23, 4.78, 2.63, and 3.03 respectively. The pointMe and touchMe based interactions with annotated movie system were appealing to the users and they perceived more application value in those two type of interactions in the system.

In addition users rated the physical mobile interactions based on their intention of use in real world in Likert scale. Based on their reports, the intention of use values were depicted in Figure 6.32. The pointMe and touchMe based interactions with annotated movie system were appealing to the users and they expressed their willingness to use these interactions in real world learning movie system.

We wanted to understand whether users were aware of intermediate computing units interpreting their interactions. We asked the users to rate the physical mobile interactions in Likert scale based on their understanding of intermediate computing units. We presented their reports in Figure 6.33. To conclude, users were aware that intermediate computing elements were processing their requests when they were using Location-aware and RFID based interactions.

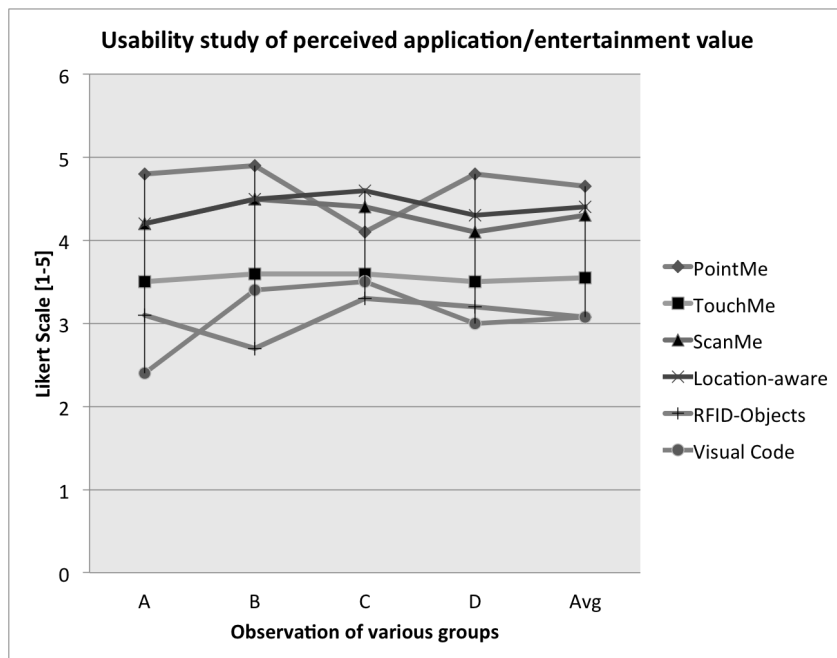


Figure 6.31: Usability comparison of perceived application and entertainment values of various physical mobile interaction schemes.

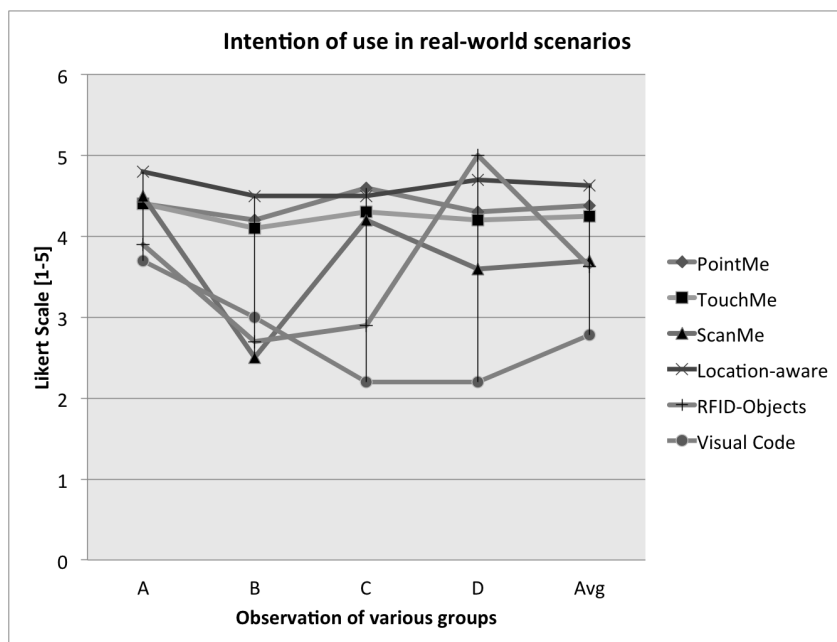


Figure 6.32: Usability comparison of intention of use of various physical mobile interaction schemes to solve real world problems.

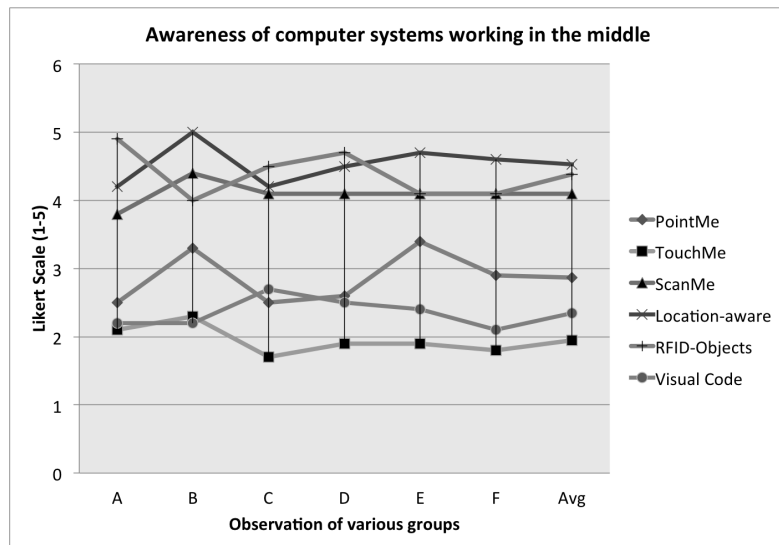


Figure 6.33: Usability comparison of perceived awareness of computer systems working in the middle (determining transparency of the computation in the user interaction)

6.5 Summary and Conclusion

In this section, we have presented the details of three prototypes that have been realized by using the proposed smart mobile interaction framework. We have presented a learning prototype that enables pointMe interactions with a time frame based annotated contents on a Smart Multimedia Screen. We show the annotation of static slide based contents and dynamic continuous contents of a presentation and illustrate pointMe, and touchMe based interactions with them. We highlight the mobile client that is used to gather learning objects from a presentation content. A similar prototype is also explained, where pointMe based mobile handheld interactions with annotated physical atlas are developed. In this prototype, we annotate major countries on the map to create pointMe interactions based innovative learning games. The learning games assist the users to remember country specific information and entertain the users at the same time. Lastly, we illustrate pointMe, touchMe, and scanMe interactions with an authored movie system. We annotate learning movies to make it possible for the learners to respond to learning based questions in the movie and help them actively participate in the learning tasks. We show two geometric annotation approaches that are used to create natural interactions event with multimedia things. Lastly, we present the detailed measurements data that we have obtained from the implementations of the pointMe, touchMe, and scanMe based interactions and compare their suitability.

Chapter 7

Conclusion and Future Works

7.1 Conclusion

IoT envisions a world, where billions of sensors are connected to the Internet. In the last decade, IoT research has progressed on how to combine virtual information with a users physical location and its surrounding things. IoT systems enable users to access digital information that has been attached to things. In this thesis, we propose a Smart Mobile Interaction Framework that supports the annotation of things within a hardware independent geometric entity in order to facilitate the rapid development of user interaction models for IoT systems. The framework supports the development of different interaction models such as touchMe, pointMe, scanMe, context-aware things selections, etc. The proposed geometric annotation approach is not dependent on a particular hardware setup. By using the annotation model, it is possible to integrate new interaction models without changing the core design and development of an IoT system.

Furthermore, in this work we have presented the structure, implementation, and evaluation of different interaction approaches conducted within the context of the proposed framework. In this work, we have described with example prototypes that the proposed smart mobile interaction framework can provide many benefits and advantages in areas like learning, haptic communication, smart environment control, context-aware smart homes, and elderly assistive homes. The summary of achievements and results of the proposed smart mobile framework are listed in the following :

- We study different IoT systems that combine natural interaction techniques and explore application areas in which these techniques are used.

- We illustrate a spatial geometric annotation based user interaction approach that mediates many of the limitations that are present in the known RFID, NFC, and visual tag based solutions. The geometric scheme allows easy and customizable annotation options for virtual and real world multimedia things.
- We present the design, development, and implementation of smart mobile interaction framework. By incorporating the framework we show the implementation of touching, pointing, scanning and context-aware things selection. Additionally, we illustrate how the framework interfaces can be used to integrate new interaction model with an existing IoT system without changing the core components of the application.
- We describe a comprehensive overview of the proposed geo-spatial annotation mechanism. Through a number of implementations of the smart mobile interaction framework we present guidelines to use the geometric annotation scheme in order to develop IoT interactions that are independent of a particular hardware setup.
- We illustrate user customizable and programmable context-preferences in detail in order to enable context-aware interactions with geo-spatially annotated multimedia things. We shown with developed prototypes, how geo-spatially annotated things can be used as building blocks in order to construct the context parameters.
- We present tools and algorithms to implement pointMe, touchMe, and context-aware interactions with geo-spatially annotated IoTs. We develop algorithms to fuse IR based motion tracking data and handheld device accelerometer data to approximate position and pointMe data. We present tools to geo-spatially annotate virtual and real things in order to realize applications with intuitive user interface and user interaction possibilities.
- We have used virtual interfaces and have annotated virtual things in order to develop touchMe based interactions. We propose haptic-audio-visual feedback annotation to the live presentation document and eBook document to enable pointMe and touchMe based learning interactions.
- We geo-spatially annotate Second Life virtual things in order to realize social haptic interactions and smart home control interactions. We extend the proposed smart mobile interaction framework to realize haptic social interactions by interacting with Second Life virtual avatars. Similarly, we use the framework to control real

world smart home appliances by interacting with Second Life virtual appliances that are geo-spatially annotated. By using the Second Life virtual interface, it becomes easier to monitor and control the real world home appliances.

7.2 Future Work

There are a number of open issues and challenges which were identified while conducting the research presented within this thesis and that deserve further investigation.

Mobile Apps and Ambient Interaction

We use the term ambient interaction to making technology invisible, where the user interface responds to a user's normal behaviour pattern, for instance his/her presence in a room or proximity with respect to certain home appliances. Previously, researchers have extensively explored the use of mobile handheld devices and mobile apps to interact with IoT. Mobile handheld technology has been designed for user-centered mobile communication. It is interesting to turn ON the living room light or unlock a door by using a mobile app. However, research has shown that the number of interactions needed to complete such trivial tasks increases when they are conducted through a mobile app. Hence, more research is needed to explore and improve user experiences for such interactions in order to make them appealing to mass users. We need to study how users are using the mobile apps to interact with IoTs in a physical environment. Currently, the user has to be aware that s/he uses a mobile device when interacting with IoT in the environment. An open question is whether mobile technology can be built to accomplish the same tasks without the user's direct awareness of a mobile computer use. This would fulfil the vision of the disappearing computer.

By using a mobile app, a connected home light can be controlled from hundreds of miles away. However, at home, mobile app based control of such smart appliances is positively retrograde. When controlling a smart light with a mobile app, for example, the increasing number of interactions needed makes it unappealing compared to a traditional wall switch. Interactions in such smart home environment should complement the existing interfaces of household things instead of changing the way we use them. Future research is needed to explore ambient interaction options with physical IoTs in a smart home. The mobile apps need to stay in the background and smart sensors should respond to the natural interactions of the users. More research is needed to provide a

single, inter-connected user experience in smart home.

Industry Standards for Ever Changing IoT Platforms

Web 1.0 has brought the opportunity to connect computers and their data, whereas Web 2.0 has brought forth the technology to connect people and their data. Web 3.0, on the other has brought the opportunity to connect things, places, people, computers etc. One of the main reasons for the huge interest in IoT services is that everybody could easily build new IoT platform to connect things, people, and services. In the last few years, we have seen a large increase in interest in context-aware IoT applications and novel mobile app interaction techniques that target IoTs. With hardware advancements, new companies and IoT products are emerging in increasing numbers.

The field has been evolving so rapidly in the last years that IoT app platforms are becoming outdated with the arrival of new ones. These changes sometimes make the IoT devices and their associate services obsolete. Many of the IoT platforms are based on mobile apps and oftentimes, an updated mobile app requires the purchase of a completely new set of IoTs. Moreover, some IoT services and devices are available only on certain mobile ecosystems. Hence, the adoption and growth of certain IoTs and services depend on the availability and evolution of these mobile ecosystems. In order to make the IoTs accessible and popular, new research is needed to address open industry standards for the enterprises. These open standards can be used to develop IoT that would be available to users irrespective of the mobile ecosystems they choose to use.

Privacy Issues for a Context Aware Home

Pervasive and ubiquitous computing affects many aspect of the way we work, live, and play. The ever increasing influence of pervasive computing will hugely impact our level of personal privacy. For example, a mobile network operator can potentially track the location and movements over time of a person who is carrying a handheld device. As with all applications related to IoT, security is a big concern. In pervasive environments such as smart homes, security and privacy are necessary for consumer acceptance of IoTs and services [15]. Looking at the number of smart sensors that may soon be utilized in IoT application in our homes, we recognize that more research is needed to address the privacy issues of the consumers.

In many digital domains we are noticing an alarming trend; people are providing personal information for their increased convenience. Our personal information such as

the location data of our car GPS, health data from a fitness tracker, eating pattern data from a smart refrigerator, etc can be harvested by businesses or governments to measure our creditworthiness, employability, or insurance premiums. For example, in exchange for a premium discount, an insurance provider might want its customers to wear a fitness tracker at home in order to measure active steps, to wear a heart rate monitor to determine stress, or to use a smart refrigerator in order to understand their eating habits, etc. We understand that in an environment infused by invisible sensors and gadgets, it is challenging to have absolute protection of personal information [78], however, steps must be taken to reduce privacy threats. Hence, it should be easy to understand if we are not careful, the pervasive collection of information from within our homes may create automated forms of profiling and exploitation opportunities for businesses. Therefore, if people prefer to live in a context-aware home to gain extra convenience, every measure should be taken to prevent the information about their private life from being accessed by others. One possible way to achieve this is to give users more awareness and control over the use of personal information. Future research should also focus on establishing industry standards for data gathering and use in order to minimize privacy risks.

Beacons Based Indoor Position Finding Tools

IoT applications can greatly benefit from a robust indoor position finding tool similar to GPS. WiFi and bluetooth signal triangulation techniques may find the indoor position of a mobile sensor, however, more research is needed to achieve a degree of accuracy that would make it usable in location aware IoT applications. Beacons that have limited advertising range can be attached to a thing, for example a table in a furniture store. The beacon periodically broadcasts a unique identifier which can be sensed by a mobile application on a smartphone/smart watch. When the mobile app receives the identifier of the physical location of the thing, a location based event can be triggered. Therefore, when a customer is near the table, the store can provide a sales offer to the customer in an intuitive manner. When placed in an indoor environment, these beacons may also be used to approximate location information of a mobile user. An initial configuration step is required to map the locations of the beacons. When a user is within this mapped environment, based on the signals received from the beacons and the mobile sensor, the proximity information of the user can be determined. More research is needed to find novel user interfaces and novel use cases of such location data in IoT systems.

Improving User Experiences

The home, business and work environments of today are poised to get smarter in the coming years. With the generation witnessing the proliferation of smart things, the focus now shifts to improving the interactions with everyday things and thereby improving the overall user experience. The technology advancements now focus on moving away from the smartphone tied experiences to more real-time interactions. The increasing transition of everyday things into smart things has sparked the growth of novel application areas for IoT.

In this era of IoT applications, the user does not need to scan a tag or open an app to control an appliance. Rather, the technology will take a backstage and the user and their interaction with things will come at the center stage. Therefore, context-awareness and intelligent mobile interactions [24] will play a critical role [92] in the user's interactions with IoT. The emphasis should be on making the user's interactions smart, convenient, and friendly. More research is needed to bring forth technology that facilitates seamless interaction with nearby connected things. These things may have attached sensors that work as transmitters and broadcast contextual data including location, sensor readings and time. IoT research represents the focal points of the upcoming technological advancements aimed at providing an improved digital experience. The new era of IoT research is here, where everyday things can be made smarter by being equipped with an array of sensors that are capable of broadcasting digital data to enable real-time contextual interactions.

Bibliography

- [1] G.D. Abowd, C.G. Atkeson, J. Hong, S. Long, R. Kooper, and M. Pinkerton. Cyberguide: A mobile context-aware tour guide. *Wireless networks*, 3(5):421–433, 1997.
- [2] Ejaz Ahmed, Ibrar Yaqoob, Abdullah Gani, Muhammad Imran, and Mohsen Guizani. Internet-of-things-based smart environments: state of the art, taxonomy, and open research challenges. *IEEE Wireless Communications*, 23(5):10–16, 2016.
- [3] Heikki Ailisto, Lauri Pohjanheimo, Pasi Valkkynen, Esko Strmmer, Timo Tuomisto, and Ilkka Korhonen. Bridging the physical and virtual worlds by local connectivity-based physical selection. *Personal Ubiquitous Comput.*, 10(6):333–344
physical and virtual worlds by local connectivity-based physical selection
- [4] Kazi Masudul Alam, Abu Saleh Md Mahfujur Rahman, and Abdulmotaleb El Saddik. Mobile haptic e-book system to support 3d immersive reading in ubiquitous environments. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, 9(4):27, 2013.
- [5] K. Axel. Location-based services: Fundamentals and operation. *John Wiley & Sons*, 2005.
- [6] Ronald Azuma. Tracking requirements for augmented reality. *Communications of the ACM*, 36(7):50–51, 1993.
- [7] Paramvir Bahl and Venkata N Padmanabhan. Radar: An in-building rf-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784. Ieee, 2000.

- [8] R. Ballagas, J. Borchers, M. Rohs, and J.G. Sheridan. The smart phone: a ubiquitous input device. *IEEE Pervasive Computing*, 5(1):70–77 ubiquitous input device
- [9] Rafael Ballagas, Michael Rohs, and Jennifer G. Sheridan. Sweep and point and shoot: phonecam-based interactions for large public displays. In *CHI '05 extended abstracts on Human factors in computing systems*, Portland, USA, 2005. ACM.
- [10] R. A. Baratto, L. N. Kim, and J. Nieh. Thinc: a virtual display architecture for thin-client computing. In *20th ACM Symposium on Operating Systems Principles*, pages 277–290, 2005.
- [11] R. A. Baratto, S. Potter, G. Su, and J. . Nieh. Mobidesk: mobile virtual desktop computing. In *MobiCom '04*, pages 1–15, 2004.
- [12] David Barger, Anoop Gupta, Jonathan Grudin, and Elizabeth Sanocki. Annotations for streaming video on the web. In *CHI '99: CHI '99 extended abstracts on Human factors in computing systems*, pages 278–279, New York, NY, USA, 1999. ACM.
- [13] A. Barghout, Jongeun Cha, A. El Saddik, J. Kammerl, and E. Steinbach. Spatial resolution of vibrotactile perception on the human forearm when exploiting funneling illusion. In *Haptic Audio visual Environments and Games, 2009. HAVE 2009. IEEE International Workshop on*, pages 19 –23, 7-8 2009.
- [14] Louise Barkhuus and Anind Dey. Is context-aware computing taking control away from the user? three levels of interactivity examined. In *UbiComp 2003: Ubiquitous Computing*, pages 149–156. Springer, 2003.
- [15] Victoria Bellotti and Abigail Sellen. Design for privacy in ubiquitous computing environments. In *Proceedings of the Third European Conference on Computer-Supported Cooperative Work 13–17 September 1993, Milan, Italy ECSCW'93*, pages 77–92. Springer, 1993.
- [16] Thierry Bodhuin, Gerardo Canfora, Rosa Preziosi, and Maria Tortorella. Hiding complexity and heterogeneity of the physical world in smart living environments. In *Proceedings of the 2006 ACM symposium on Applied computing*, pages 1921–1927, Dijon, France, 2006. ACM.

- [17] L. Borodulkin, H. Ruser, and H.-R. Trankler. 3d virtual "smart home" user interface. In *IEEE International Symposium on Virtual and Intelligent Measurement Systems VIMS '02*, pages 111 – 115, 2002.
- [18] Gregor Broll, Markus Haarländer, Massimo Paolucci, Matthias Wagner, Enrico Rukzio, and Albrecht Schmidt. *Collect and Drop: A Technique for Multi-Tag Interaction with Real World Objects and Information*, volume 5355/2008, pages 175–191. Springer Berlin / Heidelberg, 2008.
- [19] Gregor Broll, Enrico Rukzio, Massimo Paolucci, Matthias Wagner, Albrecht Schmidt, and Heinrich HuBmann. Perci: Pervasive service interaction with the internet of things. *IEEE Internet Computing*, 13(6):74–81 Service Interaction with the Internet of Things, 2009.
- [20] Gregor Broll, Sven Siorpaes, Enrico Rukzio, Massimo Paolucci, John Hamard, Matthias Wagner, and Albrecht Schmidt. Supporting mobile service usage through physical mobile interaction. In *Fifth Annual IEEE International Conference on Pervasive Computing and Communications*, White Plains, NY, 2007.
- [21] Russell Carney and Joel Levin. Pictorial illustrations still improve students' learning from text. *Educational Psychology Review*, 14:5–26, 2002. 10.1023/A:1013176309260.
- [22] Pablo Cesar, Dick C. A. Bulterman, Jack Jansen, David Geerts, Hendrik Knoche, and William Seager. Fragment, tag, enrich, and send: Enhancing social sharing of video. *ACM Trans. Multimedia Comput. Commun. Appl.*, 5(3):1–27 Fragment, tag, enrich, and send: Enhancing social sharing of video 1551–6857, 2009.
- [23] Jongeun Cha, Mohamad Eid, Ahmad Barghout, A S M Mahfujur Rahman, and Abdulmotaleb El Saddik. Hugme: synchronous haptic teleconferencing. In *MM '09: Proceedings of the seventeen ACM international conference on Multimedia*, pages 1135–1136, New York, NY, USA, 2009. ACM.
- [24] G. Chavira, S. W. Nava, R. Herv, V. Villarreal, J. Bravo, S. Mart, and M. Castro. Services through nfc technology in ami environment, 2008.
- [25] G. Chen, D. Kotz, et al. A survey of context-aware mobile computing research. Technical report, Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College, 2000.

- [26] Qing Chen, Marius D. Cordea, Emil M. Petriu, Annamaria R. Varkonyi Koczy, and Thomas E. Whalen. Human computer interaction for smart environment applications using hand gestures and facial expressions. *Int. J. Adv. Media Commun.*, 3(1/2):95–109, 2009.
- [27] Keith Cheverst, Keith Mitchell, and Nigel Davies. Investigating context-aware information push vs. information pull to tourists. In *proceedings of Mobile HCI*, volume 1, 2001.
- [28] Dana Chisnell. Usability testing: Taking the experience into account. *Instrumentation Measurement Magazine, IEEE*, 13(2):13–15, 2010.
- [29] CMake. Cross platform build system. Technical report, <http://www.cmake.org/cmake/resources/software.html>, 2010.
- [30] Michael H Coen et al. Design principles for intelligent environments. In *AAAI/IAAI*, pages 547–554, 1998.
- [31] E. Costanza and M. Leinss. Telling a story on a tag: The importance of markers’ visual design for real world applications. In *Workshop Mobile Interaction with the Real World (MIRW 2006) in conjunction with 8th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI 2006), Espoo, Finland*, 2006.
- [32] Edgar Dale. *Audio-Visual Methods in Teaching*. Rinehart and Winstor, 1969.
- [33] F. D. Davis. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3):319–340, 1989.
- [34] C. DiSalvo, F. Gemperle, J. Forlizzi, and E. Montgomery. The hug: an exploration of robotic form for intimate communication. In IEEE Press, editor, *RO-MAN 2003*, pages 403–408, 2003.
- [35] E. Duval and W. Hodgins. A lom research agenda. In *Proc. 12th Int. Conf. World Wide Web (WWW)*, pages 1–9, 2003.
- [36] M. Egenhofer. Spatial information appliances: A next generation of geographic information systems. In *First Brazilian Workshop on GeoInformatics*, Campinas, Brazil, 1999.

- [37] Josel Encarnacao. *HEI! –The Human Environment Interaction*. Springer Berlin / Heidelberg, 4557/2007, Aug 18, 2009 2007.
- [38] N.R.N. Enns and I.S. MacKenzie. Touchpad-based remote control devices. In *Conference on Human Factors in Computing Systems: CHI 98 conference summary on Human factors in computing systems*, volume 18, pages 229–230, 1998.
- [39] Muhammad Fahiem, Shaiq Haq, and Farhat Saleemi. A review of 3d reconstruction techniques from 2d orthographic line drawings. *Geometric Modeling and Imaging, 2007. GMAI '07*, pages 60–66 A Review of 3D Reconstruction Techniques from 2D Orthographic Line Drawings, 2007.
- [40] O. Falke, E. Rukzio, U. Dietz, P. Holleis, and A. Schmidt. Mobile services for near field communication. *University of Munich, Department of Computer Science, Media Informatics Group, Tech. Rep., LMU-MI-2007-1*, 2007.
- [41] S. Fallahkhair, L. Pemberton, and R. Griffiths. Dual device user interface design for ubiquitous language learning: Mobile phone and interactive television (itv). In *WMTE 2005: Proceedings of the IEEE International Workshop on Wireless and Mobile Technologies in Education*, pages 85–92, 2005.
- [42] O. Faugeras. *Three-Dimensional Computer Vision*. MIT Press
- [43] Shuo Feng, Peyman Setoodeh, and Simon Haykin. Smart home: Cognitive interactive people-centric internet of things. *IEEE Communications Magazine*, 55(2):34–39, 2017.
- [44] George W. Fitzmaurice. Situated information spaces and spatially aware palmtop computers. *Communications of the ACM*, 36(7):39–49, 1993.
- [45] S. Fleck, F. Busch, P. Biber, and W. Straber. 3d surveillance a distributed network of smart cameras for real-time tracking and its visualization in 3d. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on*, pages 118 – 118, 2006.
- [46] P. Föckler, T. Zeidler, B. Brombach, E. Bruns, and O. Bimber. Phoneguide: museum guidance supported by on-device object recognition on mobile phones. In *Proceedings of the 4th international conference on Mobile and ubiquitous multimedia*, pages 3–10. ACM, 2005.

- [47] G.A. Fowler. Qr codes: In japan, billboards take code-crazy ads to new heights. *Wall Street Journal*, 10(10), 2005.
- [48] Peter Frhlich, Rainer Simon, Lynne Baillie, Joi Roberts, and Roderick Murray-Smith. Mobile spatial interaction. In *CHI '07 extended abstracts on Human factors in computing systems*, San Jose, CA, USA, 2007. ACM.
- [49] G. Fritz, C. Seifert, P. Luley, L. Paletta, and A. Almer. Mobile vision for ambient learning in urban environments. *Mobile Learning Anytime Everywhere*, pages 77–80, 2005.
- [50] Peter Froehlich, Lynne Baillie, and Rainer Simon. Feature: Realizing the vision of mobile spatial interaction. *interactions*, 15(1):15–18, January 2008.
- [51] Peter Froehlich, Lynne Baillie, and Rainer Simon. Realizing the vision of mobile spatial interaction. *Interactions*, 15(1), 2008.
- [52] Hans Gellersen, Carl Fischer, Dominique Guinard, Roswitha Gostner, Gerd Kortuem, Christian Kray, Enrico Rukzio, and Sara Streng. Supporting device discovery and spontaneous interaction with spatial references. *Personal and Ubiquitous Computing*, 13(4), 2008.
- [53] Tao Gu, Hung Keng Pung, and Da Qing Zhang. A service-oriented middleware for building context-aware services. *Journal of Network and computer applications*, 28(1):1–18, 2005.
- [54] Uday G. Gujar and I. V. Nagendra. Construction of 3d solid objects from orthographic views. *Computers and Graphics*, 13(4):505–521, 1989.
- [55] Andy Harter and Andy Hopper. A distributed location system for the active office. *Network, IEEE*, 8(1):62–70, 1994.
- [56] Jeffrey Hightower and Gaetano Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57–66, 2001.
- [57] Steve Hinske. *Human-Computer Interaction. Interaction Platforms and Techniques*, chapter Pulling Digital Data from a Smart Object: Implementing the PullMe-Paradigm with a Mobile Phone. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 306-310 edition, 2007.

- [58] Paul Holleis, Enrico Rukzio, Friderike Otto, and Albrecht Schmidt. Privacy and curiosity in mobile interactions with public displays. In *Workshop Mobile Spatial Interaction at CHI'07*, 2007.
- [59] Andreas Holzinger, Gig Searle, and Alexander Nischelwitzer. On some aspects of improving mobile applications for the elderly. In Constantine Stephanidis, editor, *Universal Access in Human Computer Interaction. Coping with Diversity*, volume 4554 of *Lecture Notes in Computer Science*, pages 923–932. Springer Berlin Heidelberg, 2007.
- [60] M. Hossain, Jorge Parra, Pradeep Atrey, and Abdulmotaleb El Saddik. A framework for human-centered provisioning of ambient media services. *Multimedia Tools and Applications*, 44(3):407–431, 09 2009.
- [61] SK Hossain, ASM Rahman, and A.E. Saddik. Bringing virtual events into real life in second life home automation system. In *Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS), 2011 IEEE International Conference on*, pages 1–5. IEEE, 2011.
- [62] SK Alamgir Hossain, A S M Mahfujur Rahman, and A El Saddik. Haptic based emotional communication system in second life. In *Haptic Audio-Visual Environments and Games (HAVE), 2010 IEEE International Symposium on*, pages 1 –1, 2010.
- [63] SK Alamgir Hossain, AbuSalehMd Mahfujur Rahman, and Abdulmotaleb El Saddik. Interpersonal haptic communication in second life. In *Haptic Audio-Visual Environments and Games (HAVE), 2010 IEEE International Symposium on*, pages 1 –4, 2010.
- [64] S.K.A. Hossain, A.S.M.M. Rahman, and A. El Saddik. Bridging the gap between virtual and real with second life client in a virtual home automation system. In *Electrical and Computer Engineering (CCECE), 2011 24th Canadian Conference on*, pages 001212–001217. IEEE, 2011.
- [65] IEEE-LTSC-LOM. Ieee standard for learning technology-extensible markup language (xml) schema definition language binding for learning object metadata. *IEEE Std 1484.12.3-2005*, pages 1–46, 2005.

- [66] H. Ishii and B. Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 234–241. ACM, 1997.
- [67] Alejandro Jaimes and Nicu Sebe. Multimodal human-computer interaction: A survey. *Computer Vision and Image Understanding*, 108(1-2):116–134, November 2007.
- [68] Wang Jinqiao, Fang Yikai, and Lu Hanqing. Online video advertising based on users attention relevancy computing. In *IEEE International Conference on Multimedia and Expo*, pages 1161 –1164, April 26 2008.
- [69] Ali Karime, M Anwar Hossain, ASM Mahfujur Rahman, Wail Gueaieb, Jihad Mohamed Alja’am, and Abdulmotaleb El Saddik. Rfid-based interactive multimedia system for the children. *Multimedia Tools and Applications*, 59(3):749–774, 2012.
- [70] Murad Khan, Sadia Din, Sohail Jabbar, Moneeb Gohar, Hemant Ghayvat, and SC Mukhopadhyay. Context-aware low power intelligent smarthome based on the internet of things. *Computers & Electrical Engineering*, 52:208–222, 2016.
- [71] T. Kindberg, J. Barton, J. Morgan, G. Becker, D. Caswell, P. Debaty, G. Gopal, M. Frid, V. Krishnan, H. Morris, et al. People, places, things: Web presence for the real world. *Mobile Networks and Applications*, 7(5):365–376, 2002.
- [72] Golden Krishna. *The Best Interface Is No Interface: The simple path to brilliant technology*, volume 1 of *Voice That Matter Series*. New Riders, Feb 2015.
- [73] E.P. Kukula, M.J. Sutton, and S.J. Elliott. The human biometric-sensor interaction evaluation method: Biometric performance and usability measurements. *Instrumentation and Measurement, IEEE Transactions on*, 59(4):784 –791, 2010.
- [74] S. Kumar, J. Chhugani, C. Kim, D. Kim, A. Nguyen, P. Dubey, C Bienia, and Y. Kim. Second life and the new generation of virtual worlds. *Computer*, 41(9):46–53, 2008.
- [75] Linden Lab. Second life: <http://lindenlab.com>.
- [76] Linden Lab. Second life. Technical report, http://en.wikipedia.org/wiki/Second_Life, 2010.

- [77] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, et al. Place lab: Device positioning using radio beacons in the wild. *Pervasive Computing*, pages 301–306, 2005.
- [78] Marc Langheinrich. A privacy awareness system for ubiquitous computing environments. In *UbiComp 2002: Ubiquitous Computing*, pages 237–245. Springer, 2002.
- [79] Rensis Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 140:1–55, 1932.
- [80] Huiying Liu, Shuqiang Jiang, Qingming Huang, and Changsheng Xu. A generic virtual content insertion system based on visual attention analysis. In *Proceeding of the 16th ACM international conference on Multimedia*, pages 379–388, New York, NY, USA, 2008. ACM.
- [81] Yue Liu, Ju Yang, and Mingjun Liu. Recognition of qr code with mobile phones. In *Control and Decision Conference, 2008. CCDC 2008. Chinese*, pages 203–206. IEEE, 2008.
- [82] A. Lugmayr, B. Stockleben, T. Risse, and J. Kaario. Editorial: Re-thinking the future of semantic ambient media. *Multimedia Tools and Applications*, pages 1–4, 2011.
- [83] A. Madhavapeddy, D. Scott, R. Sharp, and E. Upton. Using camera phones to enhance human-computer interaction. In *Adjunct Proc. 6th Int’l Conf. Ubiquitous Computing (UbiComp 2004)*, 2004.
- [84] A. Mangent. Hypertext fiction reading: haptics and immersion. *Research in Reading*, 31(4):404–419, nov. 2008.
- [85] Jani Mantyjarvi, Fabio Paterno, Zigor Salvador, and Carmen Santoro. Scan and tilt: towards natural interaction for mobile museum guides. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, pages 191–194, Helsinki, Finland, 2006. ACM.
- [86] D. Marr and T. Poggio. *A Computational Theory of Human Stereo Vision*, volume 204, pages 301–328. The Royal Society, 1979.

- [87] P Misra and P Enge. Special issue on global positioning system. *Proceedings of the IEEE*, 87(1):3–15, 1999.
- [88] Alice Mitchell and Carol Savill-Smith. The use of computer and video games for learning: A review of the literature. 2004.
- [89] B.A. Myers, H. Stiel, and R. Gargiulo. Collaboration using multiple pdas connected to a pc. In *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, pages 285–294. ACM, 1998.
- [90] Y. Nakazato, M. Kanbara, and N. Yokoya. Localization system for large indoor environments using invisible markers. In *In Proceedings of the 2008 ACM symposium on Virtual reality software and technology*, Bordeaux, France, 2008. ACM.
- [91] Giles John Nelson. *Context-aware and location systems*. PhD thesis, Ph. D. Theses, University of Cambridge, Computer Lab., UK, 1998.
- [92] Charith Perera, Arkady Zaslavsky, Peter Christen, and Dimitrios Georgakopoulos. Context aware computing for the internet of things: A survey. *IEEE Communications Surveys & Tutorials*, 16(1):414–454, 2014.
- [93] Trevor Pering, Rafael Ballagas, and Roy Want. Spontaneous marriages of mobile devices and interactive spaces. *Commun. ACM*, 48(9):53–59 devices and interactive spaces
- [94] LSL Portal. http://wiki.secondlife.com/wiki/lsl_portal. Technical report, Linden lab, Accessed December 2010.
- [95] S Poslad. *Ubiquitous Computing*. Wiley Online Library, 2009.
- [96] A S M M Rahman and A.E. Saddik. Remote rendering based second life mobile client system to control smart home appliances. In *Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS), 2011 IEEE International Conference on*, pages 1–4. IEEE, 2011.
- [97] A S M M Rahman and A.E. Saddik. Virtual rendering based second life mobile application to control ambient media services. In *4th International Workshop on Semantic Ambient Media Experience (NAMU Series), SAME 2011*, Brisbane, Australia, June 2011. IEEE.

- [98] A S M Mahfujur Rahman, M. Anwar Hossain, Jorge Parra, and Abdulmotaleb El Saddik. Motion-path based gesture interaction with smart home services. In *Proceedings of the seventeen ACM international conference on Multimedia*, Beijing, China, 2009. ACM.
- [99] A S M Mahfujur Rahman, S K Alamgir Hossain, and Abdulmotaleb El Saddik. *Haptic Interaction and Avatar Animation Rendering Centric Telepresence in Second Life*, chapter 28, pages 717–739. *Multimedia Image and Video Processing*. CRC Press, 2nd edition, 2011.
- [100] A S M Mahfujur Rahman, SK Alamgir Hossain, and Abdulmotaleb El Saddik. Bridging the gap between virtual and real world by bringing an interpersonal haptic communication system in second life. In *IEEE International Symposium on Multimedia (ISM2010)*, Taiwan, 2010.
- [101] Abu Saleh Md Mahfujur Rahman, J. Cha, and Abdulmotaleb El Saddik. Authoring edutainment content through video annotations and 3d model augmentation. In *EEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems (VECIMS)*, pages 370–374, Hong Kong, China, May 2009.
- [102] Abu Saleh Md Mahfujur Rahman and Abdulmotaleb El Saddik. Mobile pointme-based spatial haptic interaction with annotated media for learning purposes. *Multimedia Systems*, pages 1–19, 2013.
- [103] Abu Saleh Md Mahfujur Rahman, M Anwar Hossain, and Abdulmotaleb El Saddik. Spatial-geometric approach to physical mobile interaction based on accelerometer and ir sensory data fusion. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, 6(4):28, 2010.
- [104] A.M. Rahman and A.E. Saddik. Touch me interaction paradigm for physically browsing personal learning spaces. In *Multimedia and Expo (ICME), 2010 IEEE International Conference on*, pages 436–441. IEEE, 2010.
- [105] A.S.M.M. Rahman and A. El Saddik. A 3d notice board metaphor for visualizing, categorizing, and sharing the learning objects. In *The Second International Conference on Systems ICONS 2007*, Martinique, French Caribbean, 2007.

- [106] A.S.M.M. Rahman and A. El Saddik. Hkiss: Real world based haptic interaction with virtual 3d avatars. In *Multimedia and Expo (ICME), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.
- [107] A.S.M.M. Rahman and A. El Saddik. Mobile pointme based pervasive gaming interaction with learning objects annotated physical atlas. In *Multimedia and Expo (ICME), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.
- [108] Rathika Rajaravivarma. A games-based approach for teaching the introductory programming course. *ACM SIGCSE Bulletin*, 37(4):98–102, 2005.
- [109] B. Rao and L. Minakakis. Evolution of mobile location-based services. *Communications of the ACM*, 46(12):61–65, 2003.
- [110] K. R. Rao, Z. S. Bojkovic, and D. A. Milovanovic. *Multimedia Communication Systems: Techniques, Standards, and Networks*. Prentice Hall PTR, 2002.
- [111] Omer Rashid, Will Bamford, Paul Coulton, Reuben Edwards, and Jurgen Scheible. Pac-lan: mixed-reality gaming with rfid-enabled mobile phones. *Comput. Entertain.*, 4(4), 2006.
- [112] P. Rashidi and D.J. Cook. Keeping the intelligent environment resident in the loop. In *2008 IET 4th International Conference on Intelligent Environments*, pages 1–9, 2008.
- [113] D. Reilly, M. Rodgers, R. Argue, M. Nunes, and K. Inkpen. Marked-up maps: combining paper maps and electronic information resources. *Personal and Ubiquitous Computing*, 10(4):215–226, 2006.
- [114] M. Rohs. *Linking physical and virtual worlds with visual markers and handheld devices*. PhD thesis, Dissertation, Swiss Federal Institute of Technology, 2005.
- [115] M. Rohs. Visual code widgets for marker-based interaction. In *25th IEEE International Conference on Distributed Computing Systems Workshops*, pages 506–513, 2005.
- [116] Michael Rohs. Real-world interaction with camera-phones. In *2nd International Symposium on Ubiquitous Computing Systems*. Springer, 2004.

- [117] Michael Rohs and Beat Gfeller. *Using Camera-Equipped Mobile Phones for Interacting with Real-World Objects*. Austrian Computer Society (OCG), 2004.
- [118] Enrico Ruzkio, Gregor Broll, Karin Leichtenstern, and Albrecht Schmidt. Mobile interaction with the real world: An evaluation and comparison of physical mobile interaction techniques. In *AmI-07: European Conference on Ambient Intelligence*, Darmstadt, Germany, 2007. Springer.
- [119] M. Satyanarayanan. Pervasive computing: Vision and challenges. *Personal Communications, IEEE*, 8(4):10–17, 2001.
- [120] Gerhard Schall, Erick Mendez, Ernst Kruijff, Eduardo Veas, Sebastian Jungmanns, Bernhard Reitinger, and Dieter Schmalstieg. Handheld augmented reality for underground infrastructure visualization. *Personal and ubiquitous computing*, 13(4):281–291, 2009.
- [121] Bill Schilit, Norman Adams, and Roy Want. Context-aware computing applications. In *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on*, pages 85–90. IEEE, 1994.
- [122] Albrecht Schmidt, Matthias Kranz, and Paul Holleis. Interacting with the ubiquitous computer: towards embedding interaction. 2005.
- [123] Peter Schoo and Massimo Paolucci. Do you talk to each poster? security and privacy for interactions with web service by means of contact free tag readings. In *First International Workshop on Near Field Communication*, pages 81–86. IEEE Computer Society, 2009.
- [124] Wilbur Schramm. *Television in the lives of our children*. Stanford University Press, 1961.
- [125] SecondLife. Second life open source portal. Technical report, http://wiki.secondlife.com/wiki/Open_Source_Portal, Accessed 10 June 2010.
- [126] H. Shen, Y. Lu, F. Wu, and S. Li. High-performanance remote computing platform. In *PERCOM '09 Workshop*, pages 1–6, 2009.
- [127] Dong-Her Shih, Po-Ling Sun, David C Yen, and Shi-Ming Huang. Taxonomy and survey of rfid anti-collision protocols. *Computer communications*, 29(11):2150–2166, 2006.

- [128] Sanni Siltanen, Charles Woodward, Seppo Valli, Petri Honkamaa, and Andreas Rauber. *User Interaction for Mobile Devices*. Springer US, 33, 2008.
- [129] Rainer Simon, Peter Frohlich, and Thomas Grechenig. Geopointing: evaluating the performance of orientation-aware location-based interaction under real-world conditions. *J. Locat. Based Serv.*, 2(1):24–40, 2008.
- [130] S. Siorpaes, G. Broll, M. Paolucci, E. Rukzio, J. Hamard, M. Wagner, and A. Schmidt. Mobile interaction with the internet of things. In *Poster at 4th International Conference on Pervasive Computing (Pervasive 2006), Dublin, Ireland*. Citeseer, 2006.
- [131] Kay M. Stanney, Mansooreh Mollaghasemi, Leah Reeves, Robert Breaux, and David A. Graeber. Usability engineering of virtual environments (ves): identifying multiple criteria that drive effective system design. *Int. J. Hum.-Comput. Stud.*, 58:447–481, April 2003.
- [132] Wen Sun, Yan Lu, and Shipeng Li. Redi: an interactive virtual display system for ubiquitous devices. In *Proceedings of the international conference on Multimedia, MM '10*, pages 759–762, New York, NY, USA, 2010. ACM.
- [133] Craig W Thompson and Fran Hagstrom. Modeling healthcare logistics in a virtual world. *Internet Computing, IEEE*, 12(5):100–104, 2008.
- [134] Martin Tomitsch, Richard Schl, Thomas Grechenig, Christoph Wimmer, and K Thomas. Accessible real-world tagging through audio-tactile location markers. In *5th Nordic conference on Human-computer interaction: building bridges*, Lund, Sweden, 2008. ACM.
- [135] Dzmitry Tsetserukou, Alena Neviarouskaya, Helmut Prendinger, Mitsuru Ishizuka, and Susumu Tachi. ifeel.im: innovative real-time communication system with rich emotional and haptic channels. In *CHI EA '10: Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*, pages 3031–3036, New York, NY, USA, 2010. ACM.
- [136] Timo Tuomisto, Pasi Välikkynen, and Arto Ylisaukko-oja. Rfid tag reader system emulator to support touching, pointing and scanning. 2005.

- [137] P. Välikkynen, I. Korhonen, J. Plomp, T. Tuomisto, L. Cluitmans, H. Ailisto, and H. Seppä. A user interaction paradigm for physical browsing and near-object control based on tags. In *Mobile HCI, Physical Interaction Workshop on Real World User Interfaces. Udine, Italy*, pages 31–34, 2003.
- [138] Pasi Valkkynen, Marketta Niemel, and Timo Tuomisto. Evaluating touching and pointing with a mobile terminal for physical browsing. 2006.
- [139] Pasi Valkkynen and Timo Tuomisto. Physical browsing research. In *Pervasive Mobile Interaction Devices*. Springer, 2005.
- [140] Paul Van Schaik and Jonathan Ling. Design parameters of rating scales for web sites. *ACM Trans. Comput.-Hum. Interact.*, 14(1):4, 2007.
- [141] Qifei Wang, Gregorij Kurillo, Ferda Ofli, and Ruzena Bajcsy. Evaluation of pose tracking accuracy in the first and second generations of microsoft kinect. In *Health-care Informatics (ICHI), 2015 International Conference on*, pages 380–389. IEEE, 2015.
- [142] R. Want, A. Hopper, V. Falcão, and J. Gibbons. The active badge location system. *ACM Transactions on Information Systems (TOIS)*, 10(1):91–102, 1992.
- [143] Roy Want, Kenneth P. Fishkin, Anuj Gujar, and Beverly L. Harrison. Bridging physical and virtual worlds with electronic tags. In *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit*, pages 370–377, Pittsburgh, Pennsylvania, United States, 1999. ACM.
- [144] M. Weiser. The computer for the 21st century. *Scientific American*, 265(3):94–104, 1991.
- [145] Mark Weiser. Ubiquitous computing. *Computer*, 26(10):71–72, 1993.
- [146] Evan Welbourne, Karl Koscher, Emad Soroush, Magdalena Balazinska, and Gaetano Borriello. Longitudinal study of a building-scale rfid ecosystem. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, pages 69–82. ACM, 2009.
- [147] Wikipedia. Xml-encryption. Technical report, http://en.wikipedia.org/wiki/XML_Encryption, Accessed, 10 July 2010.

- [148] A. Wilson and N. Oliver. Multimodal sensing for explicit and implicit interaction. In *11th International Conference on Human-Computer Interaction (HCI International 2005), Las Vegas, Nevada, USA, 2005*.
- [149] Andrew Wilson and Steven Shafer. Xwand: Ui for intelligent spaces. In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 545–552, New York, NY, USA, 2003. ACM.
- [150] X10. <http://www.x10.com/homepage.htm>. Technical report, Accessed December 2010.
- [151] Lin Yang, Longyu Zhang, Haiwei Dong, Abdulhameed Alelaiwi, and Abdulmotaleb El Saddik. Evaluating and improving the depth accuracy of kinect for windows v2. *IEEE Sensors Journal*, 15(8):4275–4285, 2015.
- [152] Jiao Yu, Wei-Shinn Ku, Min-Te Sun, and Hua Lu. An rfid and particle filter-based indoor spatial query evaluation system. In *Proceedings of the 16th International Conference on Extending Database Technology*, pages 263–274. ACM, 2013.
- [153] Bin Zhang, Pei-Luen Patrick Rau, and Gavriel Salvendy. Design and evaluation of smart home user interface: effects of age, tasks and intelligence level. *Behaviour and Information Technology*, 28(3):239–249, 2009.
- [154] Martina Ziefle and Susanne Bay. How older adults meet complexity: aging effects on the usability of different mobile phones. *Behaviour & Information Technology*, 24(5):375–389, 2005.