USE THAT THERE! POINTING TO ESTABLISH DEVICE IDENTITY

by

Colin Edward Swindells

B.A.Sc., Simon Fraser University, 2000

Thesis submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE

in the School

of

Computing Science

SIMON FRASER UNIVERSITY

July 2002

Approval

Name: Colin Edward Swindells
Degree: Master of Science
Title of Thesis: Use that there! Pointing to establish device identity

Examining Committee:

Dr. Torsten Möller, Chair
Assistant Professor
School of Computing Science
Simon Fraser University

Dr. Kori Inkpen, Senior Co-Supervisor
Associate Professor
Faculty of Computing Science
Dalhousie University

Dr. John Dill, Senior Co-Supervisor
Professor
School of Engineering Science
Simon Fraser University

Dr. Shahram Payandeh, Supervisor
Professor
School of Engineering Science
Simon Fraser University

Dr. Kellogg Booth, Examiner
Professor
Department of Computer Science
University of British Columbia

Date Approved: July 18, 2002

SIMON FRASER UNIVERSITY

PARTIAL COPYRIGHT LICENSE

Title of Thesis/Project/Extended Essay

Use that there! Pointing to Establish Device Identity

Author:

(signature)

Colin Edward Swindells

(name)

June 11, 2002

(date)

:kj

Abstract

Computing devices within current work and play environments are relatively static. As the number of 'networked' devices grows, and as people and their devices become more dynamic, situations will commonly arise where users will wish to use 'that device there' instead of navigating through traditional user interface widgets such as lists and trees. Our method of interacting with 'that device there' is composed of two main parts: identification of a target device, and transfer of information to or from the target device. By decoupling these processes, we can explore the most effective way to support each part. This thesis describes our process for identifying devices through a pointing gesture using custom tags and a custom stylus called the gesturePen. Implementation details for this system are provided along with qualitative and quantitative results from a formal user study. The results of this work indicate that our gesturePen method is an effective method for device identification, and is well suited to dynamic computing environments that are envisioned to be commonplace in the near future.

Acknowledgments

The author wishes to thank his supervisors Dr. Kori Inkpen, Dr. John Dill and Dr. Shahram Payandeh for their guidance. For their constructive suggestions and support, he also thanks the entire EDGE Lab. In particular, Felix Lau, Keith Shu, Arman Danesh, Denis Chai, and Mark Hancock shared their knowledge of networking, Java, and general ad hoc groupware development skills. Dr. Lyn Bartram and Regan Mandryk suggested usability and overall design critiques. Melanie Tory helped brainstorm and proof-read numerous drafts of documents related to this research. Dr. Jason Harrison, Melanie Tory, and Mark Hancock aided the creation of a demonstration video and gave valuable critiques of this research. As external examiner, Dr. Kellogg Booth provided valuable insights into the thesis research and how it could be enhanced. The author is additionally grateful for the financial support and technical suggestions from Colligo Wireless, Inc., the Natural Sciences and Engineering Research Council, the BC Advanced Systems Institute, and Simon Fraser University.

Table of Contents

APPROVAL .... **i**

ABSTRACT .... **iii**

LIST OF TABLES .... **viii**

LIST OF FIGURES .... **ix**

GLOSSARY .... **x**

ACRONYMS .... **xi**

PRIMARY JOURNALS, PROCEEDINGS, AND ASSOCIATIONS .... **xii**

CHAPTER 1 INTRODUCTION **1**

- 1.1. MOTIVATION 1
- 1.2. USAGE SCENARIO AND CONTEXT 1
- 1.3. RATIONALE 4
- 1.4. THESIS OUTLINE 5

CHAPTER 2 WHAT IS A SHARED SPACE? **7**

- 2.1. INTRODUCTION 7
- 2.2. EXAMPLE APPLICATION: SHARING AND VIEWING PHOTOS 7
  - *2.2.1. Introduction* *7*
  - *2.2.2. Sharing and Viewing Physical Photos* *8*
  - *2.2.3. Sharing and Viewing Digital Photos on a Single PDA* *9*
  - *2.2.4. Sharing and Viewing Digital Photos on a PDA with Line-of-Sight Communications Capability* *11*
  - *2.2.5. Sharing and Viewing Digital Photos on a PDA with Ad Hoc Wireless Networking Capability* *13*
  - *2.2.6. A Possible Scenario for Sharing Photos in an Ad Hoc Network* ... *14*
  - *2.2.7. Scenario Notes* *16*
  - *2.2.8. Scenario Analysis* *17*
- 2.3. SHARED SPACES 19
  - *2.3.1. What is a Shared Space?* *19*
  - *2.3.2. Using Activity Theory to Define a Shared Space* *19*
  - *2.3.3. An Example of Activity Theory Analysis* *22*

CHAPTER 3 RELATED RESEARCH 25
3.1. IMPORTANCE OF GESTURING 25
3.2. AMPLIFIED VS. VIRTUAL REALITY 27
3.3. INFORMATION TRANSFER MECHANISMS 30
3.4. ENCOURAGING INTERACTION BETWEEN USERS 33
3.5. DIRECT, LINE-OF-SIGHT INTERACTION TECHNIQUES 34
3.6. CURRENT TECHNOLOGIES 36
CHAPTER 4 DESIGN AND IMPLEMENTATION 38
4.1. OVERVIEW 38
4.2. COMMUNICATIONS FLOW OF THE GESTUREPEN 38
4.3. RANGE AND ANGLE CONSIDERATIONS 40
4.4. HARDWARE 41
4.5. SOFTWARE 46
4.5.1. Trial Administrator Software 46
4.5.2. Tag Software 50
4.5.3. Integration with mobile media transfer systems 53
CHAPTER 5 USER STUDIES 55
5.1. EXPERIMENTAL DESIGN 55
5.1.1. Overview 55
5.1.2. Experimental Rationale 59
5.1.3. Phase 1: Cognitive load task 59
5.1.4. Phase 2: Mobile environment task 62
5.1.5. Steps to prevent experimental biases 62
5.1.6. Qualitative data collection 63
5.1.7. Quantitative data collection 65
5.2. RESULTS AND DISCUSSION 65
5.2.1. Qualitative results 66
5.2.2. Identification times 73
5.2.3. Slips during identification 75
CHAPTER 6 FUTURE WORK AND CONCLUSIONS 77
6.1. SUMMARY OF CONTRIBUTION 77
6.2. FUTURE WORK 79
6.3. CONCLUSIONS 81
REFERENCES 84
CORPORATIONS 84
PAPERS AND BOOKS 85
APPENDIX A: EXPERIMENTER'S USER STUDY SCRIPT 92

APPENDIX B: TRIAL QUESTIONNAIRES 95

APPENDIX C: RUNNING THE USER STUDY SOFTWARE 106

List of Tables

Table 1: Summary of activity theory descriptions.. 21
Table 2: Amplified and virtual reality spectra.. 27
Table 3: Summary of user study trial methods.. 56
Table 4: Social computing checklist used for the post-trial inquiry 64
Table 5: Summary of participant general preferences.................................. 67
Table 6: Summary of participant graphical list length preferences 67
Table 7: Phase 1: Cognitive load task times... 75
Table 8: Phase 2: Mobile environment task times.. 75
Table 9: Total numbers of misidentified devices 76
Table 10: Comparison of information sharing methods................................ 78

List of Figures

Figure 1: Identification and transfer using our custom tag and gesturePen...... 3
Figure 2: Example GUI components for an ad hoc meeting.......................... 16
Figure 3: Shared spaces discipline domain.. 19
Figure 4: Activity theory relationships 20
Figure 5: Sensor fusion deduction of context.. 29
Figure 6: Communications between a gesturePen, tag, and target device...... 39
Figure 7: Effects of different IR transceiver ranges and viewing angles........ 41
Figure 8: Screen capture of the trial administrator software.......................... 42
Figure 9: Software flow for the iPAQ trial administrator software 43
Figure 10: Software flow for the infrared tag.. 44
Figure 11: Software flow for the gesturePen... 45
Figure 12: Screen capture of WindowSpaces – a media sharing prototype ... 47
Figure 13: Schematic of the infrared tag 49
Figure 14: Photo of a tag.. 51
Figure 15: Photo of the gesturePen... 52
Figure 16: Schematic of the gesturePen 53
Figure 17: Experimental set-up for the cognitive load task (phase 1)............ 57
Figure 18: Experimental set-up for the mobile environment task (phase 2) .. 58

Glossary

802.11b. A wireless microwave radio communications standard for low-power computing devices such as desktop computers, personal digital assistants, and phones.

ad hoc meeting. An informal, task-related meeting of two or more people where the tasks, participants, and meeting venue vary with considerable fluidity. Ad hoc meetings are usually unplanned and seldom have an agenda.

beam. To send data from one personal digital assistant to another via a line-of-sight transmission (e.g. infrared communications).

Bluetooth. A protocol specification designed for short-range (~10 m) radio frequency communications between portable computing devices such as phones, personal digital assistants, and laptops.

firmware. Computer programming instructions contained in read-only memory or other semi-permanent memory hardware.

gesture. Motion of one or more body parts to express thought.

groupware. Software and hardware that allows one or more people to work in the same information space towards a common goal (i.e. performing Computer Supported Collaborative Work (CSCW)).

information appliances. Simple objects that collectively form a rich information space utilizing computational awareness of themselves, and the world.

IrDA. A protocol specification designed for short-range (~1 m) infrared communications between portable computing devices such as phones, personal digital assistants, and laptops.

tag. A small passive or active object that identifies another object within close proximity.

ubiquitous computing. A collection of [small] computing devices that seemlessly interact with people and computing devices. Ubiquitous computing devices meld into the user's environment and support context switching.

Acronyms

ACM. Association of Computing Machinery

AAAI. American Association for Artificial Intelligence

CON1. A proprietary, non-standard communications port on iPAQ computers

CSCW. Computer Supported Cooperative Work

EEPROM. Electrically Erasable Programmable Read-Only Memory

HCI. Human-Computer Interaction

IEEE. Institute of Electronic and Electrical Engineers

iPAQ. A personal digital assistant developed by Compaq

IR. InfraRed

IrDA. Infrared Data Association

J2ME. Java 2 Micro Edition

J2SE. Java 2 Standard Edition

PAN. Personal Area Network

PCMCIA. Personal Computer Memory Card International Association

PDA. Personal Digital Assistant

RF. Radio Frequency

UBICOMP. Ubiquitous Computing

URL. Uniform Resource Locator

VM. (Java) Virtual Machine

Primary Journals, Proceedings, and Associations

ACM. Association of Computing Machinery

CHI. ACM Conference on Computer-Human Interaction

Computer Graphics. IEEE Computer Graphics and Applications

Concurrency. IEEE Concurrency

Consumer Electronics. IEEE Transactions on Consumer Electronics

COOP. International Conference on the Design of Cooperative Systems

CPA. Canadian Psychology Association

CSCW. ACM Conference on Computer Supported Cooperative Work

GI. Graphics Interface

GROUP. ACM Conference on Supporting Group Work

GW. International Gesture Workshop

HCR. International Communication Association (ICA) Journal of Human Communication Research

HUC. International Symposium on Handheld and Ubiquitous Computing

IEEE. Institute of Electrical and Electronics Engineers

Int J Psychol. Psychology Press International Journal of Psychology

INTERACT. International Federation for Information Processing (IFIP) Conference on Human-Computer Interaction

IUI. International Conference on Intelligent User Interfaces

J Exp Psychol. Journal of Experimental Pyschology

MANSE. Managing Interactions in Smart Environments

SIGCHI. ACM Special Interest Group on Computer-Human Interaction

SIGGRAPH. ACM Special Interest Group on Graphics

TOCHI. ACM Transactions on Computer-Human Interaction

UBICOMP. International Symposium on Ubiquitous Computing

UIST. ACM Conference on User Interface Software Technology

Chapter 1
Introduction

1.1. MOTIVATION

Current computing systems couple the processes of device identification and information transfer even though users will often conceptualize identification as a spatial task and information transfer as a non-spatial task (or vice versa). For example, a user may wish to use 'that device there', but may not care about the organization of the underlying infrastructure that enables him or her to interact with 'that device'. Consequently, separating identification and information transfer, using appropriate technology for each process, can support more flexible and useable systems. This thesis focuses on the identification of 'that device there' using a pointing gesture. To clarify the kind of environments and situations where we envision our research could apply, we present the following scenario.

1.2. USAGE SCENARIO AND CONTEXT

Suppose a newborn baby is in a hospital room with several other newborns. While monitoring the babies, a nurse named Keith notices that one baby requires medical attention. Keith approaches the baby and begins to examine her; then, he points his handheld computer towards an active tag on the baby's wrist. The baby's name, Britney, along with current biometric information is sent from the tag to Keith's handheld. Additionally, Britney's medical records are transferred to his handheld from the hospital's central database. Unable to make a diagnosis, Keith phones Kori – a medical doctor who happens to be chatting with a few residency students in her office. Keith points his handheld towards his cell phone to identify the current receiver (Kori) as a recipient for a copy of the medical records that are on his handheld. Since Kori's primary active device is the laptop in her office, the medical records currently visible on Keith's handheld are transferred to Kori's laptop. While conversing with Keith on the phone, Kori points with a special stylus to her office wall display. Britney's medical records are subsequently displayed on the wall display for Kori's residency students, and they quickly join the discussion. Keith, Kori, and the residency students discuss Britney's prognosis for a few minutes before deciding that some additional tests are needed. Kori closes Britney's medical files and continues her discussion with her residency students. Meanwhile, Keith takes Britney to another area of the hospital for tests.

Several new interactions in a ubiquitous computing environment are described within the above scenario. Instead of selecting a desired computing device using a user interface widget, such as typing Britney's name into an edit box widget or selecting Kori's wall display from a graphical list of available displays, Keith and Kori identified devices by pointing towards them. Information was then transferred to the desired display such as Keith's handheld or Kori's wall display.

This thesis describes our implementation and user evaluation of the gesturePen, a line-of-sight, tag-based identification system. As shown in Figure 1, a person uses our system by pointing to a tag with a special pen (called the gesturePen) to uniquely identify a computing device such as a laptop (e.g. obtain its network address). Information can then be transferred over a wireless or wired network to the computing device. As described in the hospital scenario above, a user could also identify a device using an infrared port on their handheld instead of using a separate device such as the gesturePen. However, a separate device such as the gesturePen may offer a more convenient form factor, particularly when attached to larger devices such as laptops. We chose to develop and use the gesturePen because it enables more research flexibility such as experimenting with different ranges, activation angles, and communications protocols. An earlier discussion of this work was presented at the ACM Conference on Human Factors in Computing Systems [96]. By allowing users to identify devices by simply pointing at them, our system facilitates information sharing in mobile computing environments. In other words, our gesturePen method is composed of two parts: ***identification*** and ***transfer***. The user first identifies one or more desired devices using a line-of-sight pointing gesture. Subsequently, information

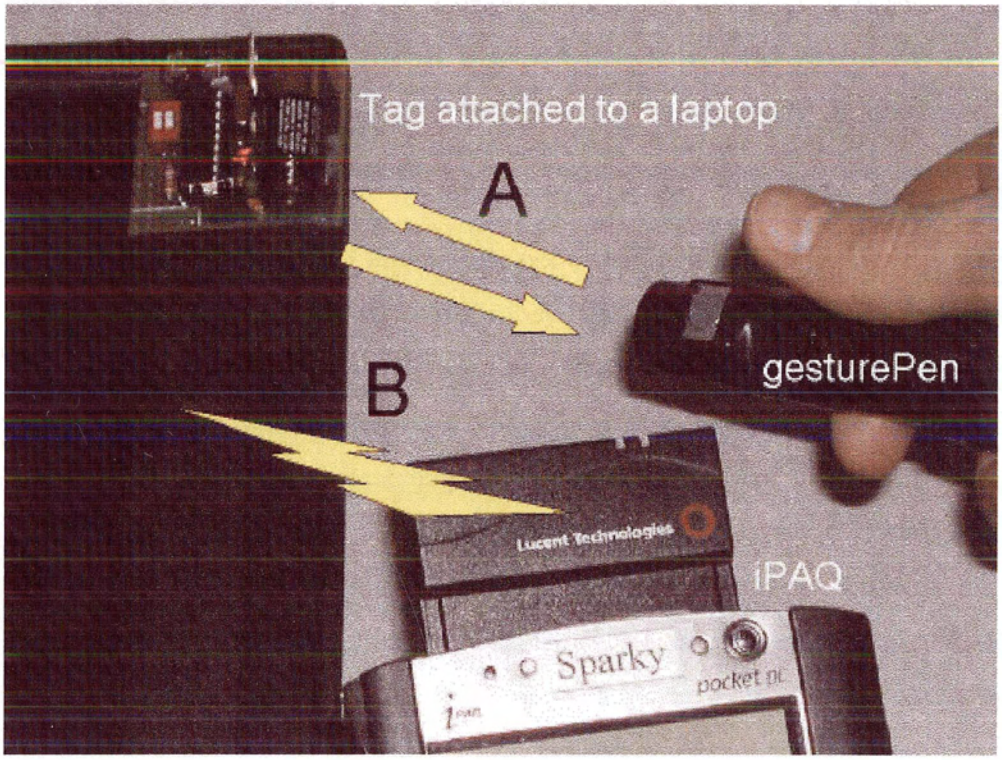


Figure 1: Line-of-sight identification with our custom tag and gesturePen. A: The user points to a tag with a special pen to uniquely identify a laptop (e.g. its network address). B: Information can then be transferred over a wireless or wired network to the laptop

can be transferred from the source device associated with the gesturePen to the target device(s) over a wireless or wired network. A three minute long .mpg format movie that motivates and demonstrates our gesturePen method can be found at the *Theses* section of http://www.cs.sfu.ca/.

1.3. RATIONALE

As the use of short-range wireless technologies such as IEEE 802.11 and Bluetooth continues to grow, new possibilities for ubiquitous computing are emerging. These networks enable a multitude of devices to be interconnected in more flexible ways than ever before. Such technological advances will undoubtedly change the way people interact, both with technology and with each other; it is therefore essential to understand how to effectively support users' interactions in these environments.

Environments are becoming increasingly populated with computing devices that automatically discover each other when they come in proximity of one another. With advances in ad hoc mobile networking, automatic discovery is an attractive feature because it enables a large number of devices to seamlessly join and leave the network. However, users may have trouble identifying devices in these environments. Selecting a target device from a list of networked devices can become a daunting task when the environment contains dozens or even hundreds of devices – many with non-descriptive names. This problem is expected to escalate as computing devices become more integrated into our environments because large numbers of devices will be accessible within the range of a wireless network node. Consequently, this advance in computer-computer discovery makes human-computer identification more difficult.

When we interact in short-range wireless networks, we can often physically see the device with which we want to interact. It seems counter-intuitive to search through a graphical user interface to find the name of a device when we know it is 'that one there'. As suggested by Bolt [30], the ability to gesture in conjunction with a graphical interface provides more natural interaction than typing symbols. Gesturing is ingrained into our sub-conscious during early childhood, and is frequently used in non-computing situations to identify things. Infants as young as two months old have been observed gesturing for attention by extending their index finger [33]. By adulthood, people support approximately 75% of dialogue with gestures [63].

1.4. THESIS OUTLINE

Chapter 2 defines how a shared space is formed during a face-to-face engagement, and supports this definition with a detailed scenario that compares and contrasts existing methods of identification and information transfer to our method. Chapter 3 describes previously published research and how this research relates to our work. Chapter 4 details the development of custom hardware and software to facilitate the user studies, and form a basis for a commercially viable system. Additionally, design considerations are examined in an effort to motivate the current and future systems development. Chapter 5 summarizes the experimental design for qualitative and quantitative user studies to address both cognitive and physical issues with our gesturePen system. Results are then listed, and followed by analysis of knowledge gained from this study. For example, we suggest situations where our method is well suited – as well as situations that are better suited to more traditional graphical user interface (GUI) methods. Finally, concluding remarks and directions for future work are described in Chapter 6.

Chapter 2
What is a Shared Space?

2.1. INTRODUCTION

This section examines the environments and situations in which we envision our gesturePen method will be used. We first examine an example application of sharing and viewing photos using a variety of methods – both traditional and technology-supported. This variety of methods forms a broad context in which we can compare the gesturePen method to other methods, and assess the main advantages and disadvantages of these methods. After exploring several interaction methods and describing a detailed scenario, we use the concepts of shared spaces and activity theory to help define the relationships that people and their devices will form while using our gesturePen method.

2.2. EXAMPLE APPLICATION: SHARING AND VIEWING PHOTOS

2.2.1. Introduction

Suppose Felix meets a group of friends while walking down a hallway. Felix and his friends spontaneously share photos from their ski trip last weekend. What are the various ways they can share the photos?

Whether they view the photos as physical prints or digitally on their PDAs (Personal Digital Assistants) will alter many of their social interactions and discussions. As extensively described by McLuhan [62], the medium will alter both the flow of content and how content is perceived. Three classes of media technology for sharing photos are described below: physical prints, PDAs with line-of-sight communications capability, and PDAs with ad hoc wireless networking capability. Quintessential pros and cons of each medium are subsequently summarized. A possible scenario for sharing and viewing digital photos is then described. Finally, we analyze the current technological issues that arose in the scenario.

2.2.2. Sharing and Viewing Physical Photos

If Felix's photos were in the form of physical prints, he could remove the prints from an envelope and share them with his friends in many ways including the following:

- Felix could hold the stack of prints in his hand such that his friends could all crowd around him and see the photos. Felix could then cycle through the prints one-by-one and discuss the currently visible photo.

- Instead of huddling around Felix, his friends could arrange themselves in a circle. Felix could then pass the top photo on the stack to a nearby friend who then passes the photo to their neighbour and so on until the photo again reaches Felix. Because multiple photos would be visible at one time, group discussion would likely change topics more readily, and the conversation would dominate around a single photo less frequently. There could also be a temporal lag between when a friend hears Felix's description of the photo and when a friend views a photo (or vice-versa).

- Several stacks of photos could be distributed by Felix to several friends and viewed in a random, free-flowing manner. Various sub-groups could seamlessly form and dissolve as one or more friends simultaneously view, exchange and discuss the sub-set of photos within their control.

The main advantages of sharing and viewing physical photos are:

- Photos are instantly available and very easy-to-use.
- High-definition visual, haptic, and audio feedback are seamlessly integrated with the photos and the environment. For example: the number of photos in a stack can be estimated by the thickness of the stack, subtle tactile and audio feedback are available when a photo is picked up, and visual context is preserved when a photo is physically given to a friend.

The main disadvantages of sharing and viewing physical photos are:

- Physically bulky.
- Cannot give copies of photos to a friend without making more prints.
- Photos are not instantly available from a camera (except for Polaroid prints).
- Difficult for larger groups (3+) of people to simultaneously view the same photo without resorting to other technologies such as a slide projector.

2.2.3. Sharing and Viewing Digital Photos on a Single PDA

Suppose Felix wished to show his digital photos to a group of friends who did not have a PDA (or, a PDA that did not have any communications functionality). Felix could show the digital photos in several ways including the following:

- Felix could load a desired photo on his PDA, then he could physically pass his PDA to his friends as if it was a bulky paper photo. After everyone had viewed the photo, Felix or one of his friends could load another photo on the same PDA. The PDA could then be passed again to other interested viewers in the group.

- Felix could hold the PDA in front of him such that his friends could crowd around him and see the photos. Felix could then cycle through the prints one-by-one as if they were a physical stack, and discuss the currently visible photo.

The main advantages of sharing and viewing photos using a single PDA are:

- Friends and other people do not need to currently possess any computing devices to view the photos.

- Photos are more secure because they are not copied to other devices before they are viewed. Thus, a person can remember the photo, but it is difficult for a user to make a copy.

The main disadvantages of sharing and viewing photos using a single PDA are:

- There is a limited amount of screen real-estate for others to view photos. For example, if several people try to crowd around Felix's display, the screen may be difficult to see because of the observer's viewing angle or distance from the display.

- Copies of photos can not be easily distributed to friends.

2.2.4. Sharing and Viewing Digital Photos on a PDA with Line-of-Sight Communications Capability

If Felix's photos were digitally stored on a PDA with line-of-sight communications capability (e.g. a Palm Pilot or Compaq iPAQ with an infrared (IR) port), and his friends also had access to a device equipped with a compatible line-of-sight communications capability, Felix could take out his PDA and share digitally stored photos in many ways including the following:

- Instead of huddling around Felix to view photos on his PDA screen, his friends could arrange themselves in a cycle. Felix could then beam a copy of the currently visible photo on his PDA to his neighbouring friend's PDA who could then beam the photo to his or her neighbour. Because the photos are copied from friend to friend instead of moved, friends who have not seen a certain photo can obtain the photo from *any* person who has already seen the photo – not just the *last* person who has seen the photo. Unless each person explicitly deletes a photo from their device after they beam a photo to another device, the photo viewing session will likely change into a more random, free-flowing exchange of photos. Essentially, the inability to instantly copy physical photos helps maintain a certain amount of structure in the way physical photos are viewed. When viewing digital photos, loss of this structure could lead to a more unstructured exchange of photos and conversation than is possible with physical photos. Alternatively, more structure could result. For example, an ad hoc slideshow could occur where multiple people look at the same photo (each on their own PDA) while listening to the same description.

- Felix could beam the entire folder of photos to each of his friends and discuss the photos using either of the two ways just described above.

The main advantages of sharing and viewing photos using line-of-sight communications between two PDAs are:

- Copies of photos can be easily distributed to friends.
- Photos can be transferred between devices using a simple physical gesture that is visible to others in the group. However, most current infrared transmitters used by PDAs can only send data up to 30 cm.
- Target devices are implicitly identified by pointing towards them and attempting to establish a line-of-sight communications link.

The main disadvantages of sharing and viewing photos using line-of-sight communications between two PDAs are:

- Photos can only be transferred between two devices at any given time.
- Copies of photos become the property of whoever's device they are stored. In other words, once a photo is beamed to another device, the sender loses all control over that copy. This could pose a security risk with sensitive photos.
- PDA screen sizes are often lower resolution, lower contrast, and smaller than physical prints.
- Everyone must have PDAs with compatible software and hardware technologies.
- Infrared line-of-sight communications typically support lower bandwidths than wired or radio frequency (RF) communications links.

- Both devices must remain stationary with no disruption to the line-of-sight infrared connection during the complete data transfer. In practice, remaining motionless is quite awkward, particularly for large files. If the line-of-sight link breaks, the user must typically try to re-send the file.

2.2.5. Sharing and Viewing Digital Photos on a PDA with Ad Hoc Wireless Networking Capability

If Felix's photos were digitally stored on a PDA with ad hoc wireless networking communications capability (e.g. a Handspring or Compaq iPAQ with 802.11b or Bluetooth facilities), he could share digitally stored photos in ways similar to an un-networked PDA with line-of-sight communications, but with several important differences:

- Felix could send copies of photos to all or certain sub-sets of friends with one action (i.e. photos can be transferred between devices in a one-to-many way, not just a one-to-one manner).

- Felix, or another friend, could moderate a slideshow type presentation remotely from a PDA (i.e. photos could be simultaneously displayed on the friends PDAs with the initiation of one person).

The main advantages of sharing and viewing photos using an ad hoc computer network are:

- Can exchange collections between several people at any given time utilizing fast, non-directional communications.

- Can remotely manage a friend's device (e.g. have all devices obtain and display the next picture in Felix's album). Thus, the high control afforded with physical photos is still possible without loss of the flexibilities afforded with digital media.

The main disadvantages of sharing and viewing photos using an ad hoc computer network are:

- Current wireless networking technologies such as 802.11b and Bluetooth have poor security (i.e. encryption development is not yet in a mature state).

- Additional ways of viewing and transporting data leads to a greater possibility of information overload and management problems. As the number of computing devices accessible in a network environment grows, searching for a device name in a graphical list becomes more difficult. The set of devices will constantly change as people carry their cell phones, handheld computers, and other devices in and out of the wireless network's range. Locating an item in the dynamically changing list can be challenging because a user's memory of the item's previous location in the list will be of little or no help in the current search. Teitelbaum and Granda found that users took more time to find information when its spatial location varied than when the information was in a consistent location [98].

- The overhead of forming an ad hoc group of devices. When several people meet in a hall, they all automatically know the boundaries of their group through subtle social cues and previous knowledge; however, devices cannot easily interpret these cues.

2.2.6. A Possible Scenario for Sharing Photos in an Ad Hoc Network

Felix walks down the hall and encounters two friends – Regan and Mark. Felix, Regan, and Mark all have PDAs. Because they are in close proximity to one another, they are automatically added to the same wireless network – along with other devices within close proximity. Since other devices within the vicinity are all added to the network, Felix, Regan, and Mark separate their devices from others in the vicinity by forming a personal area network (PAN). Felix creates the PAN with a gesture of his stylus (the gesturePen) towards Regan's PDA, then towards Mark's PDA. Felix, Regan, and Mark have now formed a group of three people with their devices (i.e. a PAN). During each gesture, the gesturePen obtains the IP address that uniquely identifies another person's device, and sends this data to Felix's PDA. This information is then relayed to others in the PAN via the wireless network infrastructure. Thus, the PDAs of Felix, Regan, and Mark have formed two levels of network connectivity by simply walking up to each other and having one member gesturing with his stylus towards the others. Depending on the level of security required, gesturing could also be used to set up trust relationships in a PAN. For example, Felix could select a folder of digital photos on his PDA, then give Regan and Mark access to these photos by gesturing to their devices with the gesturePen.

Once the PAN has been formed, all three PDAs display a common *file service* icon that may contain files such as images, documents, or presentations. (e.g. the folder named *Felix's Cypress Ski Photos* shown in Figure 2). Each file is physically stored on its creator's hard disk, but is available to others in a way that is conceptually the same as a 'symbolic link' in UNIX or 'shortcut' in Microsoft Windows. Consequently, files that are physically stored on the individual devices of Felix, Regan, and Mark can be quickly discovered, shared, and processed. If they are currently connected to the Internet, they can also work with other devices such as their home desktop computers. The PDAs additionally show *display service* icons that enable one to display the contents of a file, such as a photo, on another person's display.

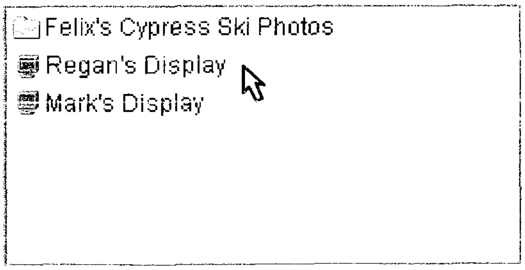


Figure 2: Example GUI component for an ad hoc meeting

Felix uses a file browser to find his most recent photos and he drags the files into the common file service – labeled *Felix's Cypress Ski Photos* in Figure 2. Regan and Mark both begin viewing photos on their individual displays while chatting with each other and Felix. One of Regan's comments triggers Felix to remember a photo from another ski trip that he would now like to share with Regan. Felix searches for the file on his device, copies the file, and then gestures towards Regan's device to immediately display the photo on Regan's display. Alternatively, Felix could have displayed the photo on his device, but Regan's display is larger and has better resolution. Felix, Regan, and Mark continue their conversation for a few more minutes, then walk away.

2.2.7. Scenario Notes

How is the above scenario different from beaming a folder of photos to each friend?

Beaming involves a one-to-one file transfer between two devices. Once a personal area network is established, multiple files can be copied to or from several devices with a single action.

How is the above scenario different from viewing digital photos on a web page?

The end result is very similar; although, our scenario does not require the overhead of setting up a web page, and does not require a connection to the Internet. Also, because URLs for many on-line photo banks are often 50 characters or longer, they are poorly suited for manual entry into a web browser. Alternatively, Felix could:

- E-mail the URL to Regan and Mark
- Beam the URL to Regan, then Mark
- Place the URL in a *file service* icon and use the same software as in the scenario

2.2.8. Scenario Analysis

Returning to the example of sharing physical photos, suppose Felix and his friends automatically form an ad hoc group by moving into close proximity to one another. There is a common conscious awareness of both the 'group' of friends and group of photos they are viewing. One physical photo does not have any knowledge of its relation to other photos or its users; however, a group of photos together form a context greater than the sum of individual photos. For example, the height of the stack of photos gives a sense of how many pictures can be viewed. Furthermore, because each photo is comprised of pigments on a single sheet of photo paper, the data 'stored' by the pigments is permanently linked to the display medium (i.e. the data *is* the display). In contrast, PDAs encode digital photo information as a bit sequence on a storage medium such as a hard drive, and require special software to decode the bit sequence then show the photo on the PDA's display. The display is separated from the data, and, more importantly, there is a one-to-many relationship between PDA displays and photo data – each display is responsible for showing multiple photos. This abstraction of the data from the display provides more flexible viewing options, but requires devices to have knowledge of their context in addition to the basic knowledge of how to decode the photo data (i.e. what data is to be displayed, when is it to be displayed, etc.). The added flexibility gained by using digital media on a PDA should not hinder the ability of users to perform their task (e.g. view and discuss photos). In general, the PDA acts as an *information appliance* [45],[100],[69],[72],[103], and should enable a user to easily interact with other people and computing devices in a wide variety of situations.

All the technologies previously described in the photo scenario for ad hoc networks (see section *2.2.6*) are currently available, or in mature phases of research and development; however, the technologies are in an early developmental stage in terms of ubiquitous computing. Creating a truly seamless and invisible infrastructure is a massive undertaking requiring significant standardization, additional development, and usability analysis. Current concepts such as a database of friends, string of text, or security settings, rarely transfer between devices or applications in a completely natural way. For example, computing infrastructure is rarely as interchangeable as in the physical-world – a toolbox is a reasonably good doorstop although it was not designed for such a purpose. Seamless computing infrastructure is currently maturing with the development of technologies such as Bluetooth, IrDA, and Jini [2],[3],[8],[18],[71],[95]. However, widespread interoperability and seamlessness is still many years from reality.

2.3. SHARED SPACES

2.3.1. What is a Shared Space?

We define a shared space as the physical *and* psychological set of information created when two or more individuals perform a common task. Research on shared spaces melds theoretical and applied work of Computer Science, Engineering Science, and Psychology disciplines as visualized in Figure 2. Describing a shared space helps form a conceptual foundation for ad hoc, mobile environments that we envision our gesturePen method to enhance. Example contribution areas to the concept of a shared space are:

Computer Science: networking, databases, artificial intelligence, and design.

Engineering Science: embedded systems, communications, and design.

Psychology: cognitive and social interactions between computers and people.

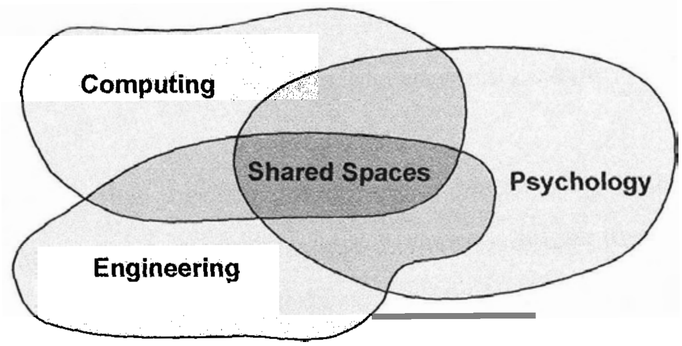


Figure 3: Shared spaces discipline domain

2.3.2. Using Activity Theory to Define a Shared Space

Applied activity theory described by researchers, such as Engeström [41], Kuuti [59], and Ræithel [78], can help define a theoretical framework for a shared space. Activity theory expands the primary relationships between people, tools, and objects into a more detailed relationship involving rules, communities, and divisions of work as illustrated in Figure 4 [99].

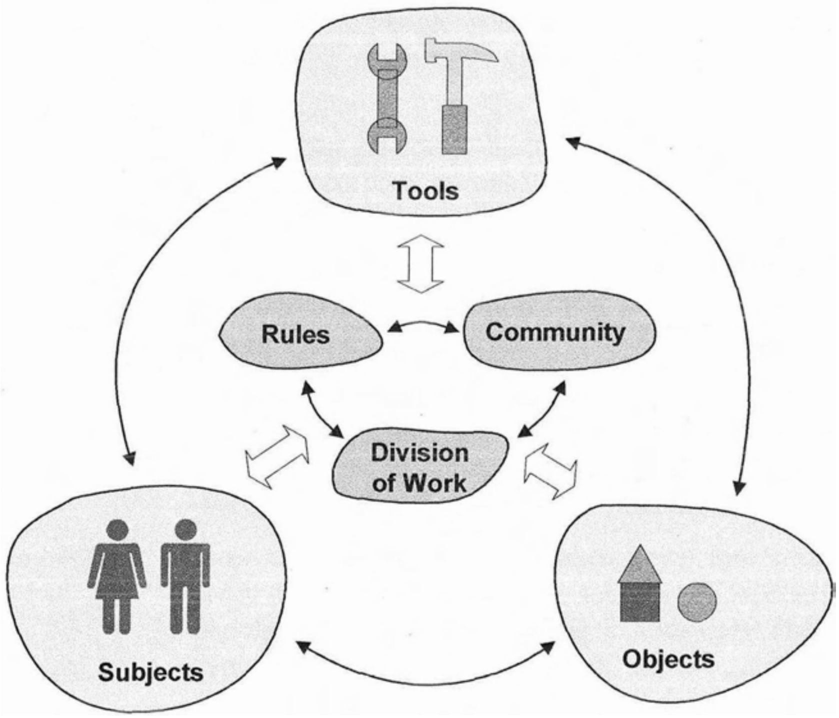


Figure 4: Activity theory relationships

Figure 4 shows that subjects interact with objects using tools. Arrows in the figure represent the many relationships between subjects, tools, and objects. For example, several subjects (people) may use tools according to a set of rules to collectively perform work on an object. Labour can be divided among subjects, and a community can be defined as a group of related tools, subjects with similar goals, or another grouping. Essentially, subjects, tools, and objects each have rules of use/behaviour, form a community, and are divided/combined in various ways as summarized in Table 1.

Table 1: Summary of activity theory descriptions

Item	***Description***
Subjects	Two or more people working together to solve a common problem. For example, a friend showing photos to some friends.
Tools	Intermediate devices (objects) used by subjects to interact with objects. For example, a pen, personal digital assistant (PDA), mouse, pointer, or a fingertip.
Objects	Matter and/or ideas (either physical or virtual) that subjects are collectively acting upon. For example, a shirt, physics theorem, or computer icon.
Rules	Any kind of limitation or guideline. For example, the shirt must be green, the principles of special relativity must be obeyed, or an n-dimensional navigation task should be performed with an n-dimensional pointing device.
Division of Work	How tasks, tools, etc. are broken down into smaller sub-tasks, and divided among the subjects. For example, the people responsible for building a house are divided into various trades such as carpenters, masons, architects, and interior designers. This division includes the scheduling of all work and people.
Community	A related physical or psychological grouping. For example, a community of retail sales clerks, or a collection of computer mice.

Our gesturePen method (previously illustrated in Figure 1 and summarized on page 5) is designed to enhance human-computer interaction within shared spaces. A shared space is a formal way to define the components and relationships that we envision will arise while using our gesturePen method. Essentially, the concept of a shared space sets a foundation for analyzing how our gesturePen method fits into the users' environment.

2.3.3. An Example of Activity Theory Analysis

To clarify the concepts of activity theory and how they relate to our gesturePen method, this section gives examples of the theory classifications *subjects*, *tools*, *objects*, *rules*, *division of work*, and *community* that were introduced in Table 1. All the examples refer to the scenario of Felix sharing his photos with two friends, Regan and Mark.

Subjects are two or more people working together to solve a common problem. For example, Felix, Regan, and Mark are a group of primary subjects. Secondary subjects include other people physically around Felix, Regan, and Mark, people in the photos, and people working on devices that are accessible to the group. The group has an uneven distribution of power and knowledge – an important distinction from much collaborative ubiquitous computing research studied (e.g. Borovoy *et al.* [31] and Gutwin and Greenberg [51]). Specifically, Felix has *a priori* knowledge of the photo contents that Regan and Mark do not possess, and Felix has control of what photos he shares and who may view the photos. During the course of the scenario, much of Felix's power and knowledge is shared (e.g. Regan and Mark may copy photos). Other power differences could exist if Felix was a boss or sub-ordinate of Regan or Mark.

Tools are intermediate devices used by subjects to interact with objects. A PDA, stylus, file sharing software, and wireless networking infrastructure are all examples of primary ubiquitous computing tools. As stand-alone devices, each of these tools is less useful, but collectively, they provide an increasingly sophisticated information space. Their power is in their ability to provide information relative to other tools and objects – hopefully seamlessly.

Objects are matter and/or ideas (either physical or virtual) that subjects are collectively acting upon. For example, photos are the primary objects in the scenario. Other significant objects include the mental models created by Felix, Regan, and Mark that represent the information space containing the photos.

Rules are any kind of limitation or guideline. For example, rules in the scenario can be divided into two main categories: social protocols and usage of tools/objects. Social protocols include the general niceties such as greetings, and more complicated relationships such as power differences between Felix and Regan or taboos. For example, if one photo depicts a friend flirting with others on the slopes, Felix may decide to withhold this picture from Regan and Mark. Felix may also rely on his friends to avoid re-arranging common shared folders or displays – such as access to a certain directory or wall display, respectively. Alternatively, etiquette could be enforced by formal rules within the software.

Other rules relate to the organization of ubiquitous computing devices. For example, the wireless network should not have a range that is too small or large – a 50 cm range wouldn't be recognizable to a friend 1 m away, but a 2000 m range would create information overload because Felix may find devices from people across the hall or in other rooms. Also, devices need certain rules (such as 802.11b or Bluetooth protocols, and physical constraints) before they can communicate with one another.

Division of Labour defines how tasks, tools, etc. are broken down into smaller sub-tasks, and sub-divided among the subjects. Proper division of labour between humans and an underlying ubiquitous computing infrastructure enables the computing devices to become seamless. Ad hoc groups often have a dynamic, non-hierarchical structure that causes the division of labour to rapidly transfer to and from other humans and their devices. For example, Felix may act as a group presenter one moment, then Regan may act as a group presenter a few minutes later. Furthermore, division of labour is controlled – or delegated – by the underlying software rules as well as by humans. For example, can Regan copy a file onto Mark's device? Regan should have permission from the file's creator, in addition to the system capabilities to copy the resource as she intends.

Instead of automating or replacing human tasks, the author believes that the best computing infrastructure creates minimal cognitive load while supporting human needs and social collaboration. Others disagree. For example, Cassell *et al.* have developed a software real-estate agent that has artificial intelligence and a graphical, human-like body [35]. An example from the aerospace industry is the different approaches taken by Boeing and Airbus as perceived by a pilot [104]. One pilot preferred a Boeing plane because he felt that he was put in control of many sophisticated, computer-aided tools; whereas, an Airbus plane made him feel more of a supervisor of many automated tasks.

Community is a logical grouping of physical or psychological items. Several important example communities exist within the scenario. These include the set of computing devices, the group of Felix, Regan, & Mark, the other people in the hallway, etc. Additional communities include people who can and do know how to access the technology needed to view digital photos, the subjects within the photos, and any other group of subjects, tools, or objects. In general, ad hoc groups are short-lived communities that rely on a stable foundation of mature communities for their success. For example, people who form a short-lived community often rely on mature communities such as their society's social structure. Also, a short-lived 'community' of five photos relies on a large collection of digital photos that are in a similar, widely accessible format.

Chapter 3
Related Research

In this chapter, we discuss research that relates either physically or conceptually to our gesturePen system (previously introduced on page 3). First, we discuss relevant psychology research related to gesturing. Second, we explore research related to the mobile environments in which we envision users will use our gesturePen method. Next, we compare and contrast our gesturePen method with similar tag-based information transfer mechanisms, technologies to promote social interactions, and other gesture computing research. Finally, we summarize current technologies that are related to our gesturePen method and the environments that we envision our method will be used.

3.1. IMPORTANCE OF GESTURING

"...body gestures are a living language which we all have learned to read." – Paul Zucker [108, p. 43]

Gesturing is a very elementary, natural way for people to communicate. Infants as young as two months old have been observed gesturing for attention by extending their index finger. By 12 months, infants can point towards an object to direct their mother's gaze [33]. By adulthood, people support about 75% dialogue with gestures [63].

In addition to communicating with others, gesturing supports the speaker's encoding of information during speech [35]. Thus, gesturing provides constructive feedback to aid the speaker's thought process as well as acting as a direct communication medium intended for the others. Many people have argued that supporting the speaker's thought process is the *primary* function of gesturing [46], [58], [82].

Other research suggests that people rely more heavily on gestures if their ability to communicate via other means is hindered. For example, Rogers [83] found that people increased their dependence on multiple modalities when communicating in noisy environments. Thus, the speaker's auditory feedback is hindered in a noisy environment, so he or she relies more heavily on visual and haptic feedback from gesturing. Additionally, the intended recipients of the speaker's words benefit from the increased visual and haptic communication because their auditory senses are reduced in a noisy environment.

Gesturing is equally important to a person's interaction with inanimate objects. According to Reeves & Nass [80], "given the slightest chance, humans will attribute social responses, behaviours, and internal states to computers." Thus, proper integration of gesturing into ubiquitous computing environments could aid the seamless communication of people and computing devices. Unstructured social protocols used in ad hoc meetings are among the most difficult to understand, and provide a challenging barrier to the invisibility of ubiquitous computing devices. Humans constantly change their communication style, and other behaviours, depending on subtle changes in their perceived context. For example, less rich communication may be used when communicating with close friends because they are – presumably – more familiar with each other's communication nuances. Compared to a complete stranger, much more information chunking can be assumed, and coarse communication can be effectively used. Korteum *et al.* [56] are beginning to analyze rules for collaboration in unstructured encounters within the context of ubiquitous computing environments. Their research centres around the classification of user profiles and encounters between people and ubiquitous computing devices.

3.2. AMPLIFIED VS. VIRTUAL REALITY

"Virtual reality focuses an enormous apparatus on simulating the world rather than on invisibly enhancing the world that already exists." – Mark Weiser [105, p. 94]

Falk *et al.* [42] states that computing devices adhere to one of two categories: amplified reality or virtual reality (summarized in Table 2). Virtual reality is classified as a *private* experience where individuals are immersed into a world that is *superimposed* on top of the existing world – creating an *impression* for the user. In contrast, amplified reality is a *public* experience where computing devices are *embedded* into our existing world – enabling people to *express* themselves using the computing environment. The term amplified reality is therefore a similar concept to Weiser's invisibility [107], and supports natural collaboration without altering or mediating communication channels as suggested by Moran *et al.* [67] and Redström *et al.* [79]. Furthermore, subjects should be able to create a common cognitive environment that represents the collective information space of the group [57].

Table 2: Amplified and virtual reality spectra[1]

Amplified Reality	***Virtual Reality***
expression	impression
public	private
embedded	superimposed

---

[1] Table 2 is adapted from a Figure by Falk *et al.* *[42]*. Falk *et al* use the term *augmented reality* instead of *virtual reality*. However, augmented reality systems are often defined as any (usually computerized) augmentation to one's environment; so, we have used the term *virtual reality* to better communicate the intent of Falk *et al.*

Creating an environment as close to reality as possible is one of the greatest challenges with virtual reality, whereas the seamless support of interaction among people is one of the primary challenges with amplified reality design. For example, ad hoc meetings contain a rich array of social interactions [28]:

- Different people may join and leave during various stages of the meeting.
- The meeting location may move from room-to-room, into the hall, etc.
- Topics will rise and fall in an unplanned way.
- Multiple topics may be discussed in parallel.

Difficulties that arise as a result of these rich social interactions are described by Redström *et al.* [79]. For example, we constantly change our behaviour based on the context of our surroundings. One minute, we may whisper to the person next to us, the next minute we may shout across the room. In a similar way, portable computing technologies, such as the gesturePen, must be fast and flexible in their ability to change their range between small and large distances. Additionally, because the content can change rapidly during a meeting – especially an ad hoc meeting – devices must be able to adapt quickly to the rapidly changing needs of the participants. We seamlessly change from one group to another, and from one topic to another, in ways that are difficult for current computing technologies because the ability to easily swap between different software and hardware components is usually a secondary design consideration.

Marshall McLuhan's statement "the medium is the message" [62] applies equally well to new computing devices within virtual reality environments as it does to McLuhan's intended media such as radios, televisions, and books. Essentially, the nature of different computing media, such as graphical user interface widgets and the gesturePen, affect the flow of information content and how it is perceived by the user in the same way that radio and television media affect the delivery of a newscast (i.e. the content of television and radio newscasts usually differ because the two media are communicated to their audiences in different ways).

Dey *et al.* [37] and Schmitt *et al.* [87], focus heavily on the software and hardware infrastructure required to give computing devices the contextual information required to function effectively in unstructured social environments. The crux of their research is the fusion of cues from multiple sensors to create a dynamic representation of the context during a social interaction (see Figure 5). Areas of interest from the environment are sensed as users interact with the environment. Areas of interest are subsequently processed to form a collection of cues that collectively form an overall context.

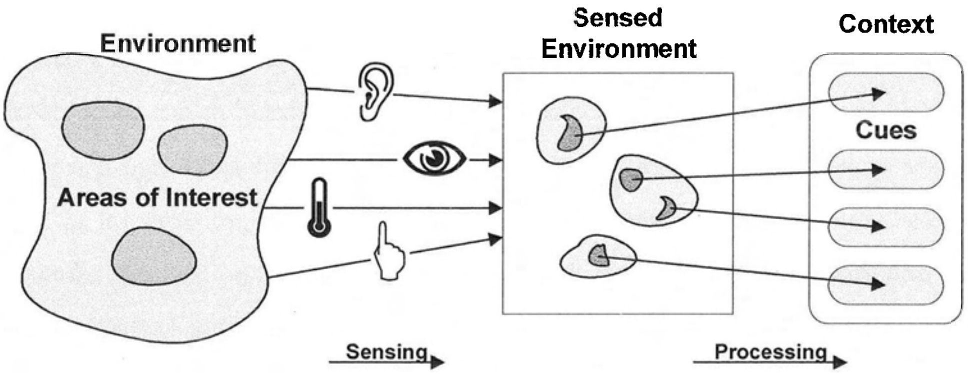


Figure 5: Sensor fusion deduction of context. The user senses their environment using vision, hearing, touch, taste, and smell to establish a context for their interactions

3.3. INFORMATION TRANSFER MECHANISMS

"In their current form, computer interfaces are very nearly the embodiment of complexity." – Brygg Ullmer & Hiroshi Ishii [100, p.1]

Pick-and-Drop [81] and mediaBlocks [100] represent significant research achievements that address the seamlessness of transferring information between computing devices – the same goal of our research. Pick-and-Drop explores the concept of 'storing' information in a pen such that information on a computer display appears to be stored in the pen (picked) and then dropped onto a different computer display. For example, text can be picked up from a handheld computer's text editor and dropped into a brainstorming application on a wall display. Unlike the gesturePen method, Pick-and-Drop users physically interact with both the sending device and receiving device. This has the benefit that users can accurately control the location of the 'dropped' information on the receiving device. For example, touching a photo icon on a handheld, then touching a wall display in its top-right corner, will 'pick' a photo from the handheld and 'drop' it in the top-right corner of the wall display. Our gesturePen method uses a similar 'pick' of information, but information can only be 'dropped' to a device, not on a particular location on the device. However, with the gesturePen, information can be 'dropped' to a device that is out-of-reach, which is not possible with the Pick-and-Drop method. Future versions on the gesturePen could communicate areas of a screen such as 2-D co-ordinates, active windows, or applications; but, these issues were not within the scope of this thesis.

Because users point with a gesturePen towards a tag associated with a computing device – not the device itself – our method better facilitates interaction with abstract objects such as wall displays or room lights. This abstraction enables users to interact where they 'interpret' computers to be. Thus, the Pick-and-Drop and gesturePen interaction methods could complement each other.

mediaBlocks explore information storage, as well as transportation and organization issues. For example, the contents of a whiteboard surface can be 'transferred' to a physical block that a user can take with them to another computing device such as a laptop. Barrett and Maglio [28] addressed similar issues as Ullmer and Ishii using a floppy disk instead of a physical block. Instead of transferring a large amount of information from the whiteboard to the block then to the laptop, the block only contains a unique ID. Consequently, the laptop can gain access to the whiteboard information from a central server or the whiteboard itself.

The gesturePen utilizes a similar philosophy to mediaBlocks since the gesturePen only obtains a very small amount of data, such as an IP address, while the user perceives a large amount of data is being transferred. Thus, the gesturePen acts as a kind of mediaBlock, but the mental model of its usage differs from a mediaBlock. With a mediaBlock, the user perceives information storage, while the user of the gesturePen perceives information transfer. In other words, the main concept of mediaBlocks is the illusion that data is stored in an object, whereas the main concept of the gesturePen is illusion that information is transferred along a line-of-sight.

HP Lab's E-squirt technology [32] and Want *et al.* [101],[102] also utilize small tags similar to those designed for the gesturePen. E-squirt technology uses small transmitters to "squirt" URLs into nearby devices. The devices then display the appropriate multimedia content retrieved from the Internet. Want *et al.* use tags that broadcast to receivers attached to devices such as tablet computers. Although these research projects use hardware that is very similar to the gesturePen, the interaction methodologies are quite different. Such technology focuses on tags broadcasting their information like a beacon. Instead of acting as a beacon, our tags for the gesturePen are only activated when 'pinged' by the gesturePen. Thus, our gesturePen method is an 'on-demand', two-way interaction technique that theoretically enables users to selectively interact with a high density of target devices surrounding the user over distances up to several metres. Conversely, broadcasting technology could overwhelm the user with conflicting information from many beacons. Also, broadcasting would require dense clusters of beacons to have very small transmission ranges (a few centimeters), and force the user to distinguish between nearby beacons from a user interface or by moving closer to the desired beacon – two usability concerns that our gesturePen method is designed to avoid.

Xenote [20] is a commercial product based on the concept of storing (or appearing to store) information in a tangible medium using non-directional tag technology similar to Want *et al.* [103]. A Xenote is a key chain with one button and a radio frequency transceiver that records a time-stamp and the radio-station a person is listening to whenever the button is pressed. Thus, active people such as car drivers or joggers can easily obtain more information about a radio program or advertisement at a more convenient time. For example, the Xenote could be synchronized with the user's personal computer once she returns home.

Instead of using active tag technology like E-squirt or gesturePen, Ljungstrand and Holmquist [60] tag computing devices with passive bar codes, then identify the target computing device with a bar-code reader. Data is then transferred over a network in a manner similar to our gesturePen method. Passive bar codes have the advantage of requiring no power, but bar code readers typically have a range of only a few centimetres, so selecting a computing device across a table or room is not feasible. Bar code systems also rely exclusively on the surrounding network infrastructure for their utility. Conversely, active tags can dynamically adapt to their environment and perform processing such as encryption.

3.4. ENCOURAGING INTERACTION BETWEEN USERS

"How can we communicate if we do not use the language of people?" – Tarcisio Della Senta [90, p. 1]

The Meme Tags project [100] utilized a large collection of simple computing devices to encourage interaction between people in ad hoc environments. Meme Tags store a small number of quotes of interest, and these quotations can be exchanged via an infrared transceiver embedded within each Meme Tag. When tested at an academic conference, participants exchanged quotations as an ice-breaker. Personal mini-networks consisting of two people were constantly created as participants met, and broken as they left to talk with others. These mini-networks fit into a larger, more stable information space of all conference participants. Although, hundreds of computing devices and people were constantly networking with one another, the computing infrastructure was conceptually simple and transparent to the users.

Our gesturePen could facilitate dynamic ad hoc communication in similar ways as the Meme Tags. Like Meme Tags, our gesturePen relies on a line-of-sight identification between two lightweight computing devices. Pointing to a target implicitly identifies a target without the need for a user interface widget such as a list, tree, or command prompt. The pen form of our gesturePen affords more controlled, conscious identification of target devices than Meme Tags. Also, once the target device has been identified, the data is transferred over a radio frequency (RF) wireless network using our gesturePen method – not over an infrared (IR) link. An RF wireless communications link is usually faster than an IR communications link. It also enables a source device to simultaneously transfer information to multiple target devices. Furthermore, transferring large amounts of information over an RF link is more reliable than an IR link because an IR link requires the two communicating devices to remain stationary to avoid breaking the line-of-sight condition.

Other research utilizes small computing devices and a sophisticated supporting infrastructure to create an environment where devices have a sense of how they fit into their environment (i.e. contextual awareness). Such research aims to hide tags more invisibly into the user's environment than our gesturePen tags. Want and Borriello discuss embedding tags and sensing capabilities into everyday objects to create information appliances [101]. While this technology is similar to our gesturePen method, either our tags or the devices to which they are affixed must be visible so users can perceive their function and interact with them.

3.5. DIRECT, LINE-OF-SIGHT INTERACTION TECHNIQUES

"...an important, if not the most important, aspect of a part's performance is how well it interacts with other parts to affect the performance of the whole." – Russell L. Ackoff [24, p. 12]

Research examples more directly related to gesturing are the finger mouse by Queck *et al.* [77], finger print recognition by Sugiura and Koseki [93], Bodytalk [76], Java Rings [10], and laser pointers [68]. The finger mouse uses video and image processing to determine a person's hand configuration and where their index finger points. Thus, a direct mapping between the finger and a cursor in a virtual reality environment was explored. However, gesturing between people in a 'real-world' environment was not addressed.

Sigiura and Koseki used fingerprints instead of the hardware tags such as those researched by Want and Russell [103], and Ullmer and Ishii [100]. For example, a one-button mouse with a fingerprint reader could perform left-, middle-, and right-button actions depending on which finger activated the mouse button. Fingerprint readers are not as technologically mature as tags; however, fingerprint readers have the huge advantage that no external tool is required for user interaction. Although, not explored by Sigiura and Koseki, we believe finger print recognition could prove very beneficial in many collaborative environments. For example, a shared tabletop or wall display could recognize and adapt to many people in parallel without the need for external tools such as pens or mice. MEMS technology could also be utilized to embed many small fingerprint reader pixels into a large shared display; however, this technique is currently technologically challenging and expensive.

Bodytalk is another vision-based system that identifies and categorizes body movements such as hand gestures, arm gestures, or leg gestures using hidden Markov model analysis of each frame of video. Because video was used, no tools were needed by the user to interact with the system, and the system could react to complex gestures. However, video resolution constraints, camera placement difficulties, and physical obstructions prevent high reliability and speed of vision systems in ad hoc environments.

Java rings combine the reliability of tagging technology with more natural gesturing interaction. Essentially, two people can identify one another after shaking hands if both participants are wearing Java rings. Each Java ring is shaped like a standard jewelry ring, and contains a short-range RFID tag that can communicate with other tags as well as close devices such as the user's PDA.

Myers *et al.* [68] analyzed the speed and accuracy of participants using laser pointers in various computing environments. This research relates to our gesturePen selection method because the gesturePen was designed such that users would point to a target as if they were using a laser pointer. Their results indicate that participants were most accurate when holding the laser pointer with two hands. They also analyzed six different ways to hold the laser pointer with one hand – these included holding like a pen, gun, PDA, and rod. The most accurate way to hold a pointer with one hand was holding the pointer like a rod. Thus, the research by Myers *et al.* suggests that the gesturePen form helps to maximize accuracy when the user points with one hand. Specifically, they found users pointing at targets 1.5, 3.0, and 4.5 metres had average angular deviations of 0.17°, 0.16°, and 0.13°, respectively. These angular deviations corresponded to average distance deviations of 0.5, 0.8, and 1.1 centimetres, respectively, on the target display.

3.6. CURRENT TECHNOLOGIES

We conclude this chapter with a summary of computing infrastructure technologies that are related to our gesturePen method and the environments in which we envision the method will be used. Most PDAs can be added to 802.11b or Bluetooth wireless networks with small hardware and software additions such as a PCMCIA card (e.g. Compaq iPAQ [4] or Handspring Visor [9]). Cholesteric liquid crystal displays developed by Kent Displays [11], and Gyricon displays by Xerox [21] are examples of high contrast display technologies that more closely resemble paper both physically and visually compared to current displays such as cathode ray tubes and normal liquid crystal displays. Active tags have been developed by companies such as Microchip [12], embedded into products such as key chains by Xenote [20], and applied in ubiquitous computing environments by several researchers such as Want, Ishii, and Winograd [45],[100],[103]. Additionally, one-way infrared communications have been used for decades in consumer products such as TV and radio remote controls. The two-way infrared communications described in the scenario and implemented for the user studies described in this thesis are conceptual extensions to this remote control technology [1]. Bluetooth styli, such as the ChatPen [7], enable styli to be associated with a specific user and/or device. Thus, a gesture with a stylus towards a tag could transfer information between the tag and stylus via a line-of-sight infrared link and then relay this information to its associated device via an omni-directional wireless system such as Bluetooth. Consequently, merging the ChatPen technology with our gesturePen technology could better enable users to freely pass the pen to others because the pen would not be physically tethered to a device with a wire. Such ad hoc exchanges are commonly done when a stranger or friend approaches you and asks to borrow your pencil to jot down a personal note on a piece of paper.

Chapter 4
Design and Implementation

4.1. OVERVIEW

To mark a device and enable its identification, we developed custom tags and a custom stylus – the gesturePen. We added IrDA (InfraRed Data Association) compliant infrared transceivers into the gesturePen and tags. We developed the tags such that they can be fixed to active and passive objects such as computers and walls, and communicate with the stylus. The tags are stand-alone devices based on tags developed by Poor [75]. The gesturePen prototype can be used by any computing device with an RS-232 serial communications port, such as a handheld or laptop. It needs only a one-way serial communications link from itself and to its handheld 'host'. This solution maximizes device, platform, and application independence. Furthermore, tags can be put on inanimate objects that are logically related to a computing device but not physically connected to the device (e.g. a projected wall display). The infrared transceiver technology we used supports link distances of at least 1.5 m, whereas most current transceivers in handhelds and laptops fail beyond 0.3 m.

4.2. COMMUNICATIONS FLOW OF THE GESTUREPEN

Figure 6 illustrates communication flow between the gesturePen, a tag, and the tag's associated device. Steps required to identify and transfer information to a target device are:

1. The user points the gesturePen towards a tag, and presses a button on the pen's case. A microcontroller within the gesturePen detects the button event and pings the tag using its infrared transceiver.
2. The tag's microcontroller receives the ping message, blinks its light, and then sends its identity information (e.g. host name and domain name of the device to which it is attached, or a URL) back to the gesturePen.
3. The gesturePen receives the identity information and checks the message validity with a cyclic redundancy check. If the identity information from the tag is correct, it is sent via a standard (wire) serial communications link to the device attached to the gesturePen (e.g. a handheld). Otherwise, no further information transfer is performed.
4. Selected information is transferred over a network to the device associated with the tag. For example, a photo is transferred from a handheld to a wall display using a wireless network.

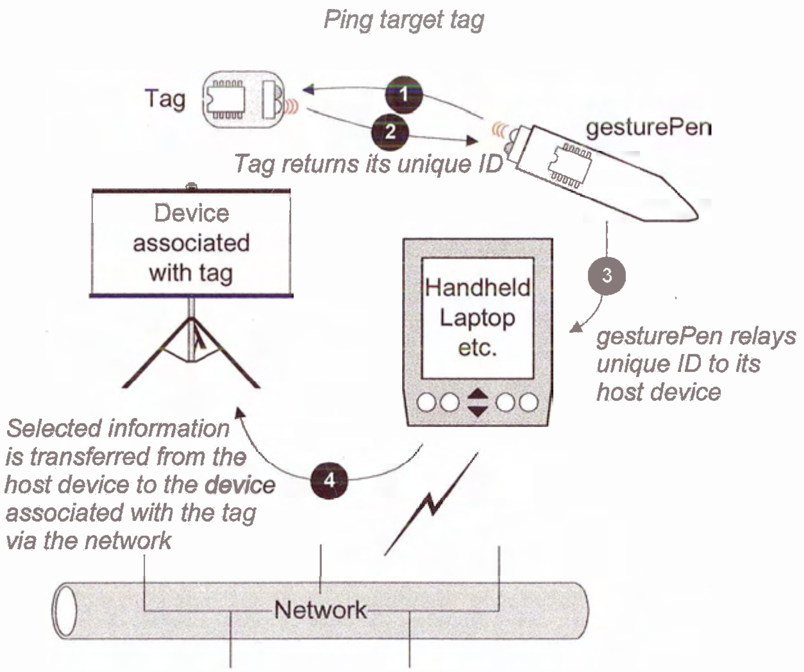


Figure 6: Communications flow between the gesturePen, tag, and the tag's associated device such as a wall display

4.3. RANGE AND ANGLE CONSIDERATIONS

Range and angle of the infrared beam are significant features affecting the usability of a stylus such as the gesturePen. Wider viewing angles allow the user to point in the general direction of a target tag; however, the chance of additionally selecting nearby tags increases as the viewing angle widens. Also, a narrow viewing angle and short range improves the security of the communications link because there is less physical space available for untrustworthy people and their devices to intercept the communications broadcasts. Increasing the range enables a user to point longer distances (e.g. across a room to a printer), but the effective beam span at the maximum distance can become quite wide. Specifically, for an infrared transceiver with maximum transmission distance d and viewing angle θ the beam span at the maximum transmission distance is given by equation 1. Our gesturePen has a viewing angle of $\theta = 30°$ (i.e. a viewing angle of 15° from all directions of the transceiver's central line-of-sight).

$$beam\_span \approx 2\,d\,Sin\left(\theta/_{2}\right) \qquad (1)$$

Figure 7 illustrates the effects of different ranges and angles of an infrared beam. If the infrared transceivers have a short range and narrow viewing angle (darkly shaded beams in Figure 7), only tag 1 will be selected. Keeping a short range, while increasing the viewing angle (including lightly shaded areas), will activate both tags 1 and 3. Alternatively, keeping a narrow viewing angle, while increasing the range, will activate tags 1 and 2. Note that the range of the tags and gesturePen should be the same because we are establishing a two-way communications link.

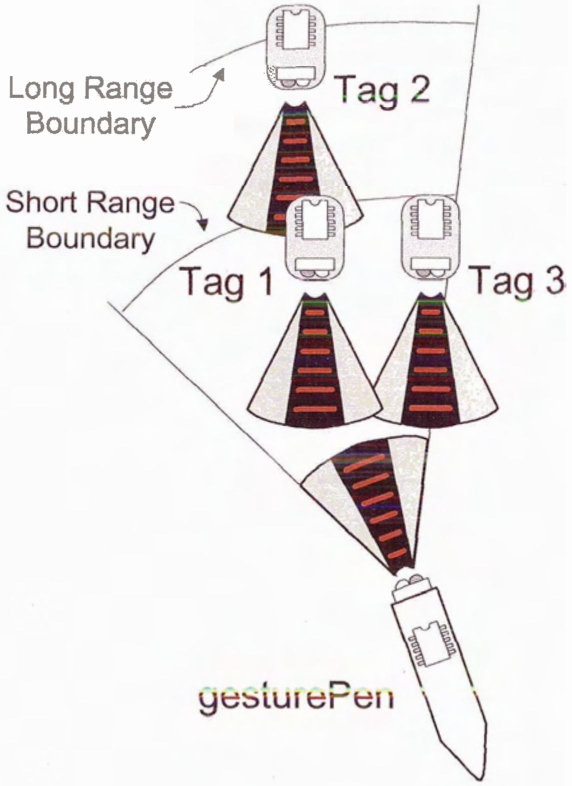


Figure 7: Effects of different transceiver ranges and viewing angles. A narrow viewing angle is illustrated with a darkly shaded beam, while a wide viewing angle also includes the lightly shaded area. Short and long ranges are illustrated by the thin arcs near and far from the gesturePen, respectively

4.4. HARDWARE

Figure 8 and Figure 9 show a schematic and photo, respectively, of our custom tag that stores and transmits a unique identification code (e.g. IP address). The tag is composed of three main components: an infrared transceiver, voltage regulator, and microcontroller. The HSDL 3000 IR transceiver was chosen because of its small size, low susceptibility to noise, IrDA compliant wavelength of 880 nm, and ranges of 1.5 m or greater. The LM78M05CT voltage regulator was chosen because it can supply a stable 5 V voltage and current bursts up to 500 mA for a low price. The PIC 16F84A microcontroller was chosen because it is cheap and easy to rapidly program prototypes (e.g. it has re-programmable FLASH memory, and many publicly available software examples.).

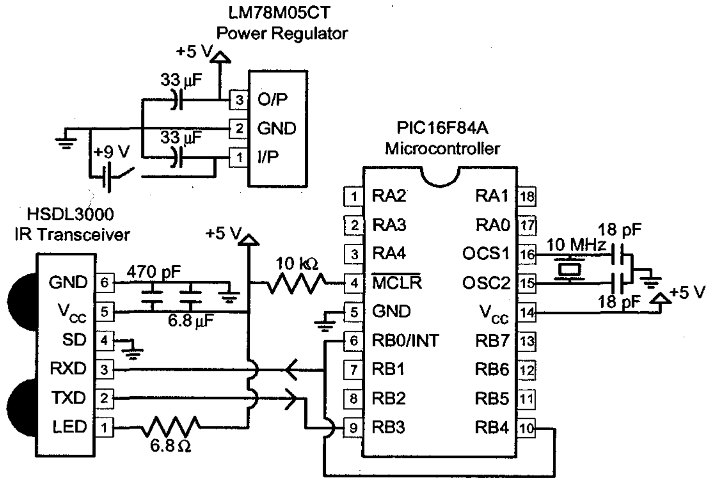


Figure 8: Schematic for the infrared tag

To maximize the flexibility of our gesturePen, we designed the tags as stand-alone computing devices. Tags can be easily attached to new devices or removed from existing devices. Also, our tags can be attached to, or removed from, inanimate objects that users associate with a computing device. By placing tags on inanimate objects, tags can act as physical aliases to computing devices. For example, to identify a front-projected wall display, a user could place a tag on the wall instead of the computer or projector controlling the wall. In this situation, the user does not need to know what computer is controlling the wall display.

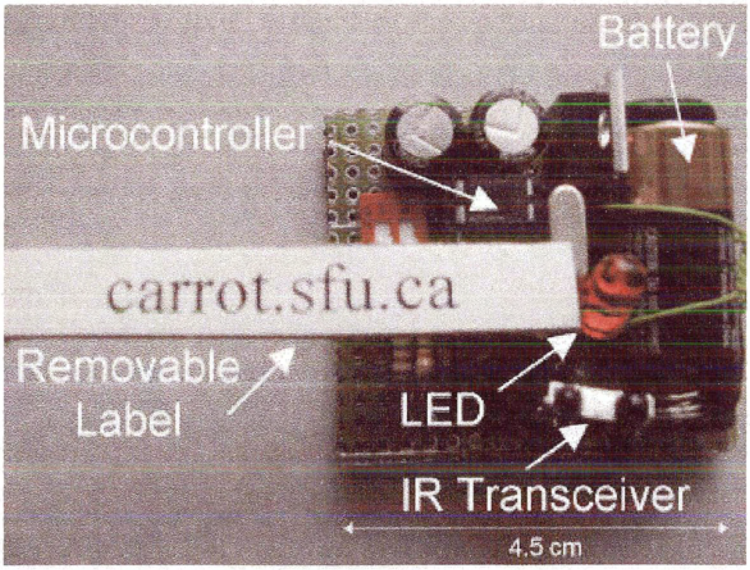


Figure 9: Life-size photo of a tag. The tag measures approximately 4 cm x 4.5 cm and has Velcro backing for easy placement on objects. Removable labels are also fixed to the tags with Velcro. The IR transceiver enables communication with the gesturePen.

Current PDAs, such as the Compaq iPAQ, Handspring Visor, and Palm Pilot, have infrared transceivers that function up to distances of 30 cm, but we hypothesized that ranges greater than 1 m were needed for users to comfortably point to other devices. Furthermore, given the form factor of the PDA's stylus and the fact that users would probably have the stylus in their hand when using the PDA, we hypothesized that pointing with a stylus was preferable to pointing with the whole body of a handheld device. Thus, users can point with the gesturePen towards a target device in a similar way to pointing to a TV with a remote control, or presentation screen with a laser pointer. We created the gesturePen prototype by integrating a modified tag into an old whiteboard marker as shown in Figure 10. One end of the gesturePen has an infrared transceiver for communicating with tags while the other end has the tip of an ordinary iPAQ stylus. Pressing the button on the stylus sends a message towards a tag. The desired tag will then respond with information (e.g. an IP address such as carrot.sfu.ca) about the device with which the tag is associated. We could have integrated the modified tag into the body of the PDA instead of a separate stylus (i.e. the gesturePen). Our main reason for choosing to develop the gesturePen was because we could easily experiment with other computing devices such as laptops (i.e. picking up and pointing with a laptop is much more difficult than pointing with the gesturePen).

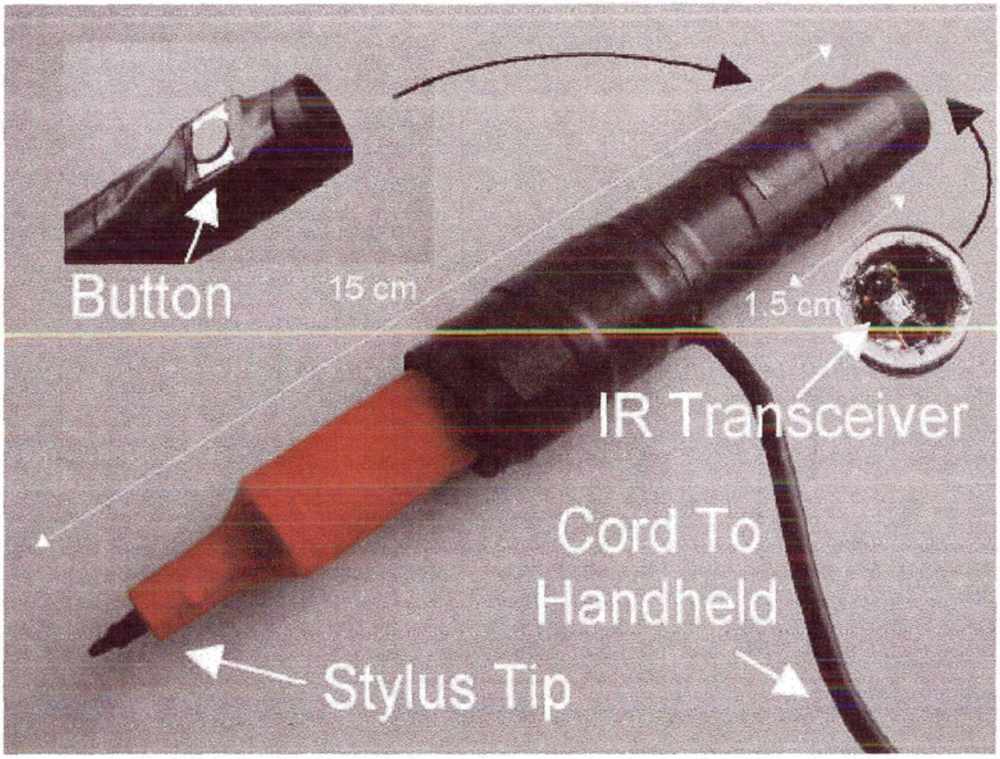


Figure 10: Photo of the gesturePen (¾ size). Our gesturePen prototype incorporates electronics into the case of a whiteboard marker and includes a standard stylus tip that is designed for touch displays such as iPAQ and Palm Pilot handheld devices. Pressing the button on the gesturePen will attempt to establish communications with a tag

The gesturePen is a modified tag that is electronically the same as the tag show in Figure 8 and Figure 9 except an RS-232 driver chip was added to mediate communications between the PDA and tag microprocessors. An adapter was purchased to link the iPAQ CON1 port to the RS-232 driver chip because the iPAQ has a proprietary, non-standard serial communications port. Additionally, the IR transceiver was mounted on the end of the PDA's stylus to hopefully facilitate more comfortable pointing. Figure 11 illustrates a schematic of the modified tag for the iPAQ. A Bluetooth enabled stylus capable of wirelessly communicating with its host device would have been preferable to a physical RS-232 serial communications link, however such technology was immature when this prototype was constructed.

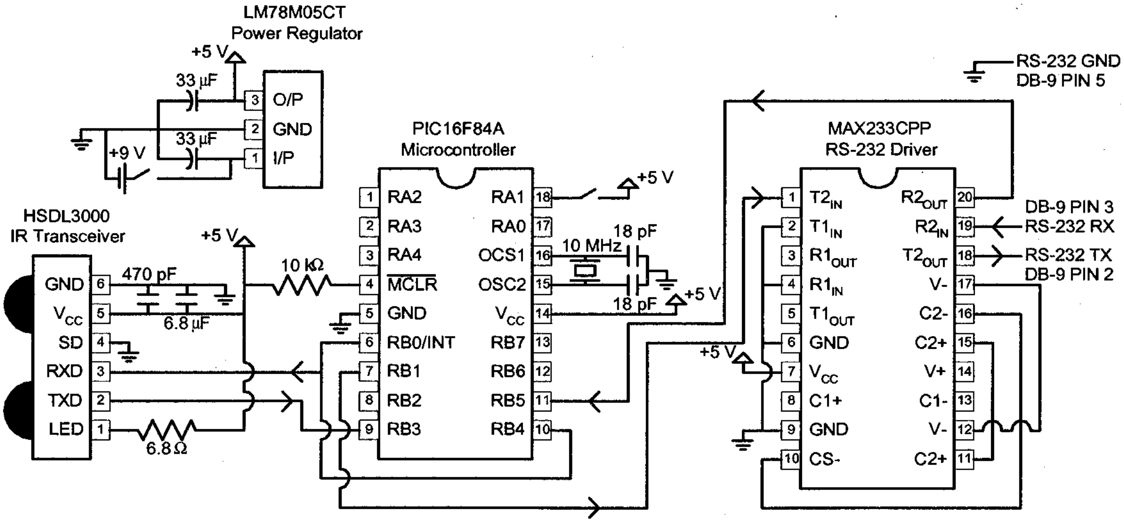


Figure 11: Schematic for the gesturePen. The gesturePen uses the same electronic components as the tag, except a button is attached to pin 18 of the microcontroller, an RS-232 driver is added, and no LED is onboard the gesturePen

The infrared port on a PDA could have been used instead of the gesturePen, but our gesturePen was a more flexible configuration for experimental prototyping. For example, most PDAs only have IR ports with ranges up to 30 cm, but our gesturePen IR transceiver has a range of 1.5 m. In addition, because our gesturePen is a stand-alone device, it can be used with any device with a serial port such as a laptop or desktop computer. These devices often have IR ports, but picking up and pointing such a computer would be considerably more difficult than pointing with a gesturePen. A stand-alone device also facilitates future research such as experimenting with different IR ranges and beam angles.

4.5. SOFTWARE

4.5.1. Trial Administrator Software

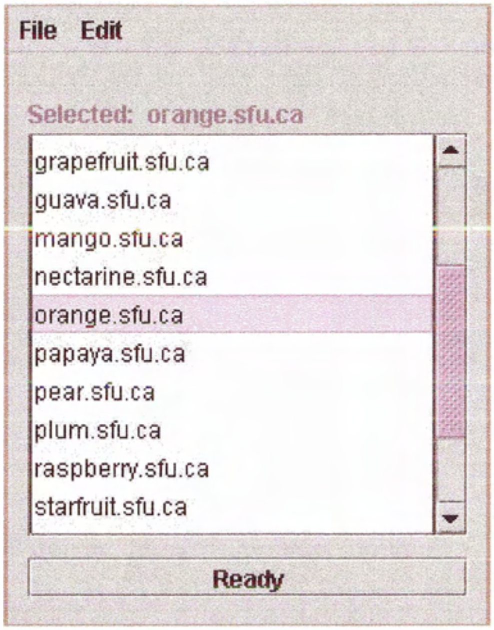
Software for a handheld computer was developed to evaluate the effectiveness of our gesturePen method compared to traditional identification using a GUI list. The software enables users to select an IP address text string using one of two methods:

- Selecting an item from a graphical list interface on an iPAQ display.
- Pointing towards a physical tag using a custom stylus – the gesturePen.

This software generates the trials for the user study and logs the time a user takes to make a selection. The software creates an ASCII text file to record each method invoked by the user and the number of milliseconds taken to select an IP address using the chosen method. Formats of the input and output log files are summarized in *Appendix C*. Figure 12 illustrates a snapshot of the GUI. Figure

13 illustrates the high-level flow of the application, and the trials are described in more detail within *Chapter 5 User Studies* starting on page 55. The following steps outline the intended use of the software:

1. Subject clicks the *Ready* button on the iPAQ handheld to begin a new trial.
2. Experimenter verbally indicates a physical computing device (e.g. "laptop" or "computer to your left")
3. Subject selects an IP address using one of the two methods described above (i.e. selecting from a graphical list or pointing towards a physical tag).



Selected: orange.sfu.ca

Figure 12: Screen capture of the trial administrator software. Pressing the ready button refreshes the graphical list and logs a timestamp. When a tag is selected using the gesturePen, or clicking the corresponding list item (such as "orange.sfu.ca"), the selection is logged and indicated at the top of the display. No list items are displayed during trials when the participant is instructed to use the gesturePen selection method

The software was written using the Forte 4j integrated development environment installed with Java SDK 1.3.0.04 and the javax.comm communications API 2.0 extension [18]. At the time of development, the Win CE operating system did not support the Java 2 Standard Edition (J2SE); thus, the SavaJe XE operating system and Java virtual machine was installed on a Compaq iPAQ PDA. Because the software is a stand-alone application written with Swing GUI widgets, the application can run on any computer with any operating system that:

- Has RS-232 serial port communications capability
- Supports the J2SE VM

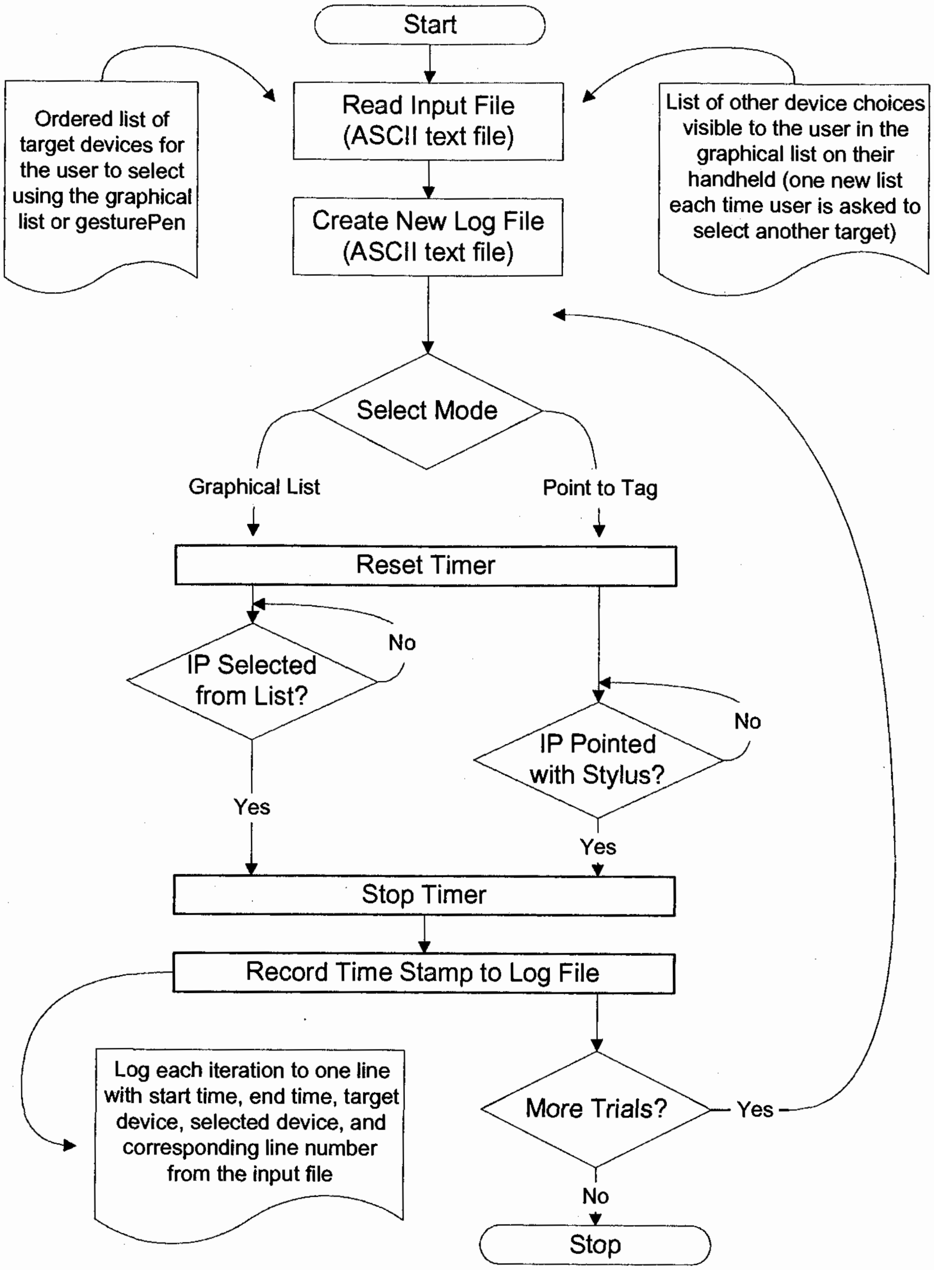


Figure 13: iPAQ trial administrator software flow. The software was written using Java 2.0 Standard Edition and can execute on many operating systems such as Microsoft Windows, Linux, MacOS, and SavaJe

4.5.2. Tag Software

The tag software was written in PCM C (developed by Custom Computer Services [5]) and PIC mid-range assembler (developed by Microchip [12]) for the PIC 16F84A microcontroller. It was derived from irbeacon.c and irda.c prototypes developed by Robert Poor [75].

Figure 14 illustrates the software flow for the infrared tag that was attached to a target device, and Figure 15 illustrates the software flow for the modified infrared tag that was attached to the sending device (i.e. gesturePen attached to the iPAQ handheld). Sending and receiving messages was based on the IrDA 1.2 standard [23] with the following exceptions:

- No bit stuffing or bit insertion was performed.
- The Cyclic Redundancy Check (CRC) byte was an 8 bit byte instead of a 16 bit byte, and was the successive XOR of each byte in the entire message instead of the CRC polynomial in equation 2.

$$CRC(x) = x^{16} + x^{12} + x^{5} + 1 \qquad (2)$$

- Frames of data were not handled.
- All communications were assumed to be 9600 baud, no parity, and no stop bit.

Transmission of data over the infrared link between our gesturePen and one tag was very reliable (i.e. > 99 % at 1 m), so the CRC was not needed to obtain reliable data for a single link. However, if the gesturePen's message was simultaneously received by two or more tags, both tags would reply with messages for the gesturePen. Using our current protocol and timing structure, we were unable to recover multiple messages in such a situation. Consequently, the CRC byte was primarily used to disregard attempts to select two or more tags at the same time.

A fully IrDA compliant tag could be created by linking an HSDL 7001 IrDA modulation/demodulation chip to the tag's infrared transceiver, or purchasing/coding a full-featured IrDA stack for the microcontroller. Additionally, more complicated message passing could be accomplished by developing more sophisticated communications software.

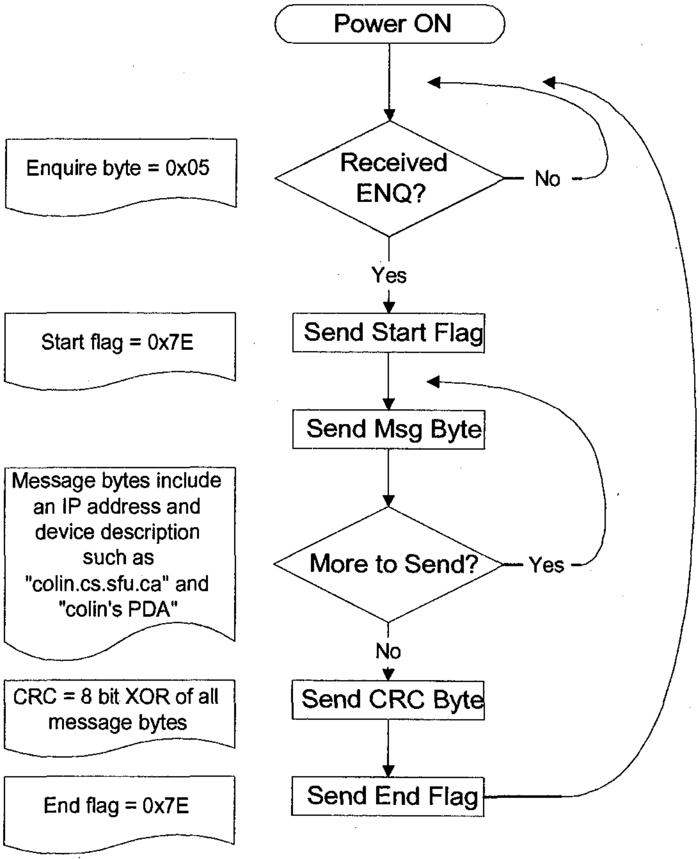


Figure 14: Software flow for the infrared tags. The infrared tag software was written using PCM C and Pic assembler, and burned into the Pic microcontroller on each tag

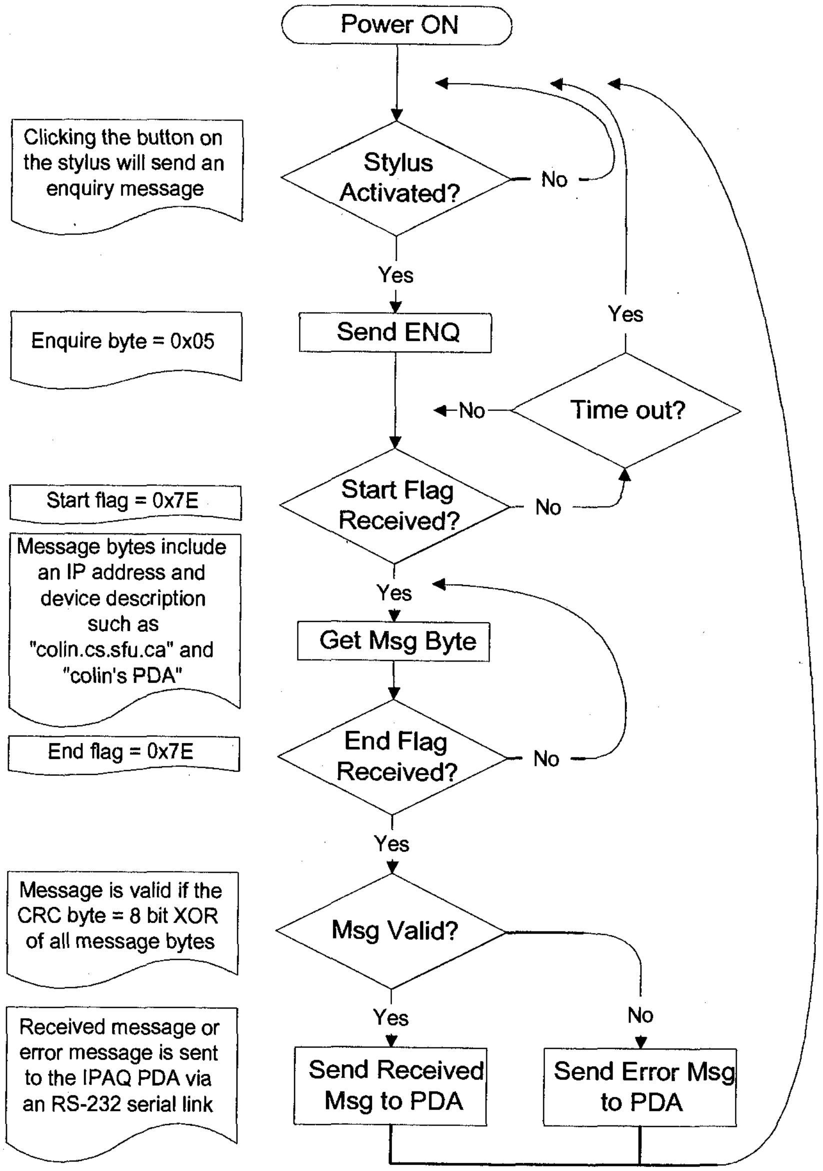


Figure 15: Software flow for the gesturePen. The software was written using PCM C and Pic assembler, and burned into the Pic microcontroller inside the gesturePen

4.5.3. Integration with mobile media transfer systems

Our gesturePen identification method integrates into software from a related project called WindowSpaces [91]. The goal of the WindowSpaces project is the seamless sharing of experiences embodied in digital media between multiple participants within dynamic contexts. For example, WindowSpaces can facilitate sharing digital photos between several people at the same time on different computing devices and operating systems. Using WindowSpaces and the gesturePen, a user could copy a photo to a wall display and a laptop by gesturing to the devices – each device would be equipped with a tag and WindowSpaces client software. The gesture would trigger each tag to send its name to the WindowSpaces client, which would then route the photos over the network so they could be 'pasted' to the laptop and wall display.

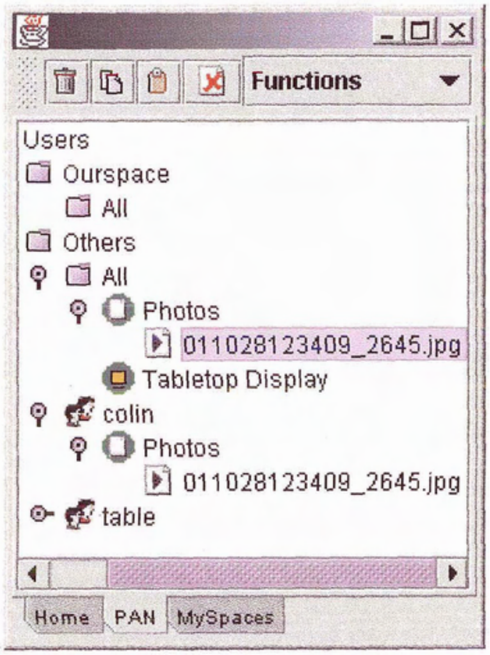


Figure 16: Screen capture of WindowSpaces – a media sharing prototype. WindowSpaces enables users to form ad hoc groups using network technologies such as 802.11b to quickly exchange data and access services of nearby computing devices

Swing graphical user interface widgets and JINI networking components were utilized to develop the WindowSpaces prototype illustrated in Figure 16 [91]. WindowSpaces currently runs on an IEEE 802.11b wireless network, and can successfully transfer information between laptops running Microsoft Windows, desktops running Linux, and Compaq iPAQs running SavaJe.

Chapter 5
User Studies

5.1. EXPERIMENTAL DESIGN

5.1.1. Overview

We performed a user study with qualitative and quantitative measures to analyze the usability of the gesturePen and the graphical list selection methods. Table 3 presents a summary of these methods. We compared and contrasted our gesturePen prototype to interaction using a standard graphical list widget. Our main objective was to obtain feedback on the appropriateness of this technique. We gathered qualitative measures based on Dryer *et al.*'s social computing framework developed for mobile computing systems [39]. The user study questions were also based on usability guidelines suggested by Dryer *et al.* [39] and conducted according to structured observation techniques suggested by Dray [38]. Information regarding user preferences was obtained via written questionnaires and audio/video taped discussions with the experimenter. In addition to our qualitative data, we compared the performance of our gesturePen prototype to a graphical list method by recording timing and error data as participants selected devices using each method.

The taped discussions were mainly intended as a permanent record to support the experimenter's field notes. Audio was recorded from a microphone on the table in addition to two small microphones attached to the experimenter's and participant's shirts, respectively. These three audio sources were mixed with video footage into a single VHS tape for each participant as he or she performed the trials. A script of what the experimenter told each participant during the user studies is included in *Appendix A*.

Table 3: Summary of trial methods

Method	***Description***
Select from graphical list	Select an IP address (such as *plum.cs.sfu.ca*) by clicking an item in a graphical list using the stylus tip on the gesturePen. This method was repeated for each of the following menu lengths: 5, 10, and 20. List lengths did not dynamically change as the participants attempted to make a selection.
Gesture with gesturePen	Select a target computing device by pointing the end of the gesturePen towards a tag on the desired device and pressing the button on the gesturePen. The tag fixed to the computing device sends the IP address of its host device to the trial administrator software.

We set up two experimental configurations – phase 1 and phase 2 as illustrated in Figure 17 and Figure 18, respectively. Phase 1 was a cognitive load test where a participant played with a graphical jigsaw puzzle on a handheld computer. While playing the puzzle, participants were interrupted by the experimenter to select a tag with a gesture using the gesturePen or from a graphical list on their handheld. Phase 2 tested identification of computing devices in mobile environments that would be unfamiliar to the user – such as a mall, airport, or another person's office. A participant stood over a star on the floor, and then selected one of 5 computing devices by either gesturing to the device with the gesturePen or reading the label on the device and selecting the matching IP address from a graphical list on their handheld computer.

Participants were instructed to familiarize themselves with the gesturePen and graphical list selection methods to minimize learning effects during the experiment and to become comfortable in their environment. Specifically, the experimenter loaded the trial administrator software for each participant to perform several selections using the graphical list and gesturePen methods. Once each participant felt comfortable with both selection methods (usually 2 – 3 selections with each method), the experimenter addressed any questions the participant had about the selection methods or experimental procedure.

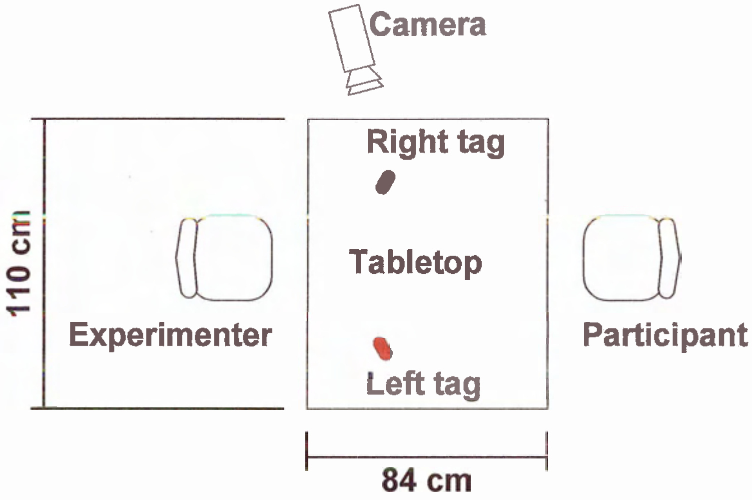


Figure 17: Experimental set-up for cognitive load task (phase 1). The participant played with a jigsaw puzzle on a handheld computer, and was interrupted by the experimenter to select a tag with the gesturePen or from a graphical list

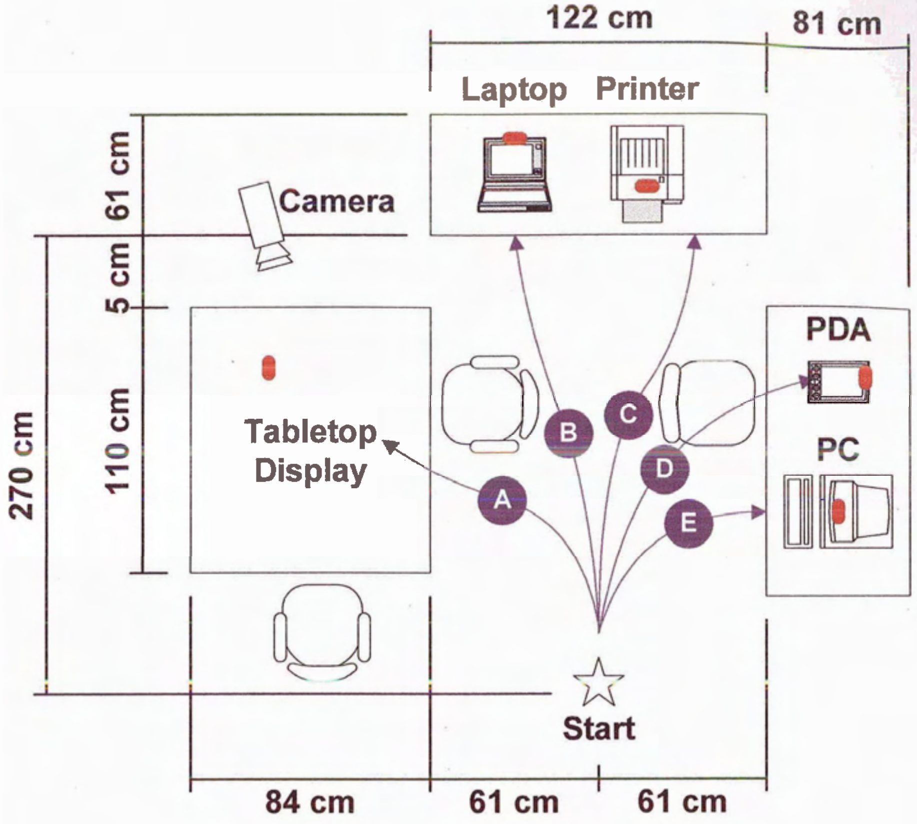


Figure 18: Experimental set-up for the mobile environment task (phase 2). The participant stood over a star on the floor, and then selected one of 5 computing devices by either pointing to the device with the gesturePen or reading a label on the device and selecting an IP address from a graphical list on their handheld computer

The iPAQ handheld (including custom hardware and a wireless PCMCIA card sleeve) had a width of 8.5 cm, height of 13 cm, and depth of 6 cm. Its display measured 6 cm wide by 8 cm high. The tags were 4.5 cm wide and 4 cm high. Additionally, the gesturePen was connected to the iPAQ handheld by a 50 cm long 2-wire microphone cable – chosen for its flexibility. This cable was attached to the gesturePen 6 cm down from the pen's IR transceiver. To minimize the possibility of accidentally selecting two tags at once, the two tags on the laptop and printer, on the PDA and PC, and on the tabletop display, were all separated by approximately 65 cm.

5.1.2. Experimental Rationale

We compared our gesturePen method to graphical lists because graphical lists are currently the standard way to select networked computing devices. A list or hierarchical tree is often used the first time a user selects a computing device (i.e. before setting up short-cuts, shell scripts, or other time saving methods). Consequently, a graphical list is a likely alternative to our gesturePen method when users select a computing device in an unfamiliar network structure or environment. In less familiar environments, users would typically rely on predefined lists of most frequently used devices or lists organized according to a particular topic. Even if the environment is a familiar one, such as an office, a set of frequently used devices will often be in the form of a short list. An example of such a short list would be a group of five desktop computers used by five workers in a common workgroup. During their daily activities, various combinations of two or three workers may form sub-groups and exchange files between their computers without bothering the whole group. With this in mind, we compared our gesturePen method to short lists of five items in addition to analyzing longer lists.

5.1.3. Phase 1: Cognitive load task

This phase of the study was to explore how easily people could use the gesturePen while engaged in a task. Each participant was instructed to play a jigsaw puzzle game on an iPAQ handheld computer using the gesturePen as a normal stylus. The puzzle was chosen because it was a fun, easy task that quickly engaged participants and no text entry was needed. Every 30 seconds, the participant was distracted by the experimenter and asked to select the tag on their left or right using one of the following methods:

- Read an IP address label on the tag, and select the appropriate IP address from a graphical list on their handheld computer.
- Point towards the appropriate tag with the gesturePen and click the button on the pen to select the tag.

After selecting a tag, the participant returned to their puzzle while the experimenter chose the next two target tags from a selection of 20 labels with Velcro backing. Each participant performed 12 tag selections. For consistency, each participant's screen was setup with the puzzle filling the top 50 % of the screen, while the trial administrator software filled the bottom 50 % of the screen. Thus, participants did not move or alter any interface windows during the experiment. Times were measured from the time the user was distracted until they selected the appropriate tag. To accurately mark the start times participants were asked to click a graphical button in the trial administrator immediately after being distracted by the experimenter. To ensure participants consistently clicked this button after every cognitive load task, the experimenter began each statement with "Click the ready button and select …".

The table and chair were set up for the participant to rest the iPAQ computer (if desired), and to fill out pre-trial and post-trial questionnaires. Most participants preferred to hold the iPAQ closer to their face than the table, but they rested their elbows on the table for greater stability while using the iPAQ. The experimenter sat across from the participant so the experimenter's instructions (i.e. what IP address to select or device to point towards) could be clearly heard. The tag labels were 1 cm x 5 cm and were written using 18 pt. Times New

Roman font. Two tags were placed on the table and separated by 65 cm to prevent the gesturePen from attempting to simultaneously communicate with both tags. To avoid exceeding the manufacturer's recommended link distance (1.5 m) [27] for the infrared transceivers used within the gesturePen, we placed each tag 60 cm from the participant. Thus, the participant and two tags formed an isosceles triangle with the equidistant edges from the participant to the tag measuring 60 cm. The other edge of this triangle was composed of the two tags separated by 65 cm.

We also videotaped each participant's full hand movement as their hand left the handheld computer and gestured towards a tag. Thus, we were able to record participant activities such as:

- Rotating the gesturePen in their hand to align its infrared transceiver with a tag's transceiver.
- Amount of eye contact focused on a target tag when gesturing with the gesturePen versus selecting from a list on the handheld computer.
- Any unexpected hand movements. For example, one participant always switched the gesturePen from his right hand to his left hand before gesturing towards a tag. (This participant was right-handed. When asked by the experimenter about this action, he remarked that he was not consciously aware of his action and could not suggest why he switched hands.)

Because of glare from the iPAQ display, we were not able to visually record participants actions such as playing the jigsaw puzzle or selecting from a graphical list. However, every selection using the graphical list (correct or incorrect) was logged and taps to the screen were indirectly recorded on video because the stylus was always visible to the camera.

5.1.4. Phase 2: Mobile environment task

The second phase of the study was to explore the effectiveness of the gesturePen for device identification in mobile ad hoc environments. As shown in Figure 18, the participants were required to select target computing devices as if they had just walked into a room and needed to transfer information to or from one of the devices in the room. Participants were asked by the experimenter to select one of five devices: a tabletop display, laptop, printer, Palm Pilot, or computer monitor. As in the cognitive load task, the participant either read the IP address label on the target device's tag and selected the device from a graphical list, or pointed to the tag using the gesturePen. Each participant made a total of 30 device selections according to a randomly ordered set of computing devices. Since we hypothesized the difficulty of selecting from a list would depend on the list's length, we used three different lengths for the list selection portion of the experiment: 5 items, 10 items, and 20 items (typical handhelds can display 10 items on a list without requiring scrolling). Selection times were measured from the time the user was told which target to select until they correctly identified the device. Timestamps for correct and incorrect selections with the gesturePen and graphical list were logged using the same start and end cues as in phase 1.

Participants were instructed to familiarize themselves with each method to minimize learning effects during the experiment. Specifically, participants could try out each method (usually 2-3 selections with each method) and ask the experimenter any questions about the methods or experimental procedure.

5.1.5. Steps to prevent experimental biases

Participants were four male and four female right-handed students previously unknown to the experimenter. Each phase of the study was counterbalanced such that half the participants (2 male and 2 female) selected a set of tags first using the gesturePen, then using a standard graphical list. The other half of participants (2 male and 2 female) selected devices using the list first, then the gesturePen.

5.1.6. Qualitative data collection

In order to understand qualitative user interaction issues associated with our gesturePen method, we gathered feedback using:

- **Pre-trial questionnaires:** users were asked how often they use, and how comfortable they are with various computing and pointing devices.

- **Post-trial questionnaires:** users were asked to write short answers regarding the advantages, disadvantages, ease of use, and preferences for the gesturePen and list methods.

- **Post-trial inquiries:** users were asked to rank 22 questions according to a 7-point Likert scale from strongly agree to strongly disagree. The questions addressed the usability criteria in Table 4 adapted from Dryer *et al.* [39]. Participants were also encouraged by the experimenter to explain their rankings. The experimenter asked questions according to established structured observation techniques [38], and made hand-written notes and audio/video records.

These questionnaires, along with data from each participant, are reported in *Appendix B.*

Table 4: Social computing checklist used to structure the post-trial inquiry

Item	***Description***
Accessibility	Do non-users believe that they could use the device easily, and do they understand easily how it works?
Appeal	Is the device something that the user is comfortable being seen using, and do non-users find the device attractive?
Communication	Does the device make communication among people easy, especially the sharing of important social information such as appointments and contact information?
Disruption	Does the device disrupt an individual's natural social behaviors, such as referring to shared information while interacting?
Familiarity	Is the form of the device one that is familiar and appropriate for the context of its use?
Identification	Does the device appear to include or exclude the user from certain communities, and do non-users see themselves as persons who would use the device?
Input sharing	Does the device allow non-users to input information easily?
Output sharing	Does the device allow non-users to easily perceive and understand output?
Perceiver distraction	Does using the device create noise or otherwise create a distraction for non-users?
Pervasiveness	Is the device mobile and convenient to use in social settings?
Power	To what extent does use of the device put one person more "in charge" than another person, and to what extent does using the device communicate a difference in status?
Relevance	Does the device appear to non-users to be useful to the user and to the non-user?
Social application	Does the device support rich social interactions, such as through interest matching, meeting facilitation, or social networking?
User distraction	Does the device place a high cognitive load on the user during use or otherwise create a distraction?

5.1.7. Quantitative data collection

Timing data and error data were recorded to quantitatively analyze the two interaction methods. Times were recorded every time a participant clicked an item in the trial handheld's graphical list or clicked the button on the gesturePen while pointing towards a tag. Participants were instructed to repeat a selection if the trial software recorded the wrong target device. After selecting a device, text feedback was provided on the PDA screen indicating the selected IP address. If the participant pointed towards a tag, he or she would also receive feedback from the tag's red LED. Whenever a participant selected the wrong target device, they always recognized the mistake and subsequently selected the correct device (i.e. we did not observe any user-feedback problems). Error data was determined by reviewing user study logs and comparing the desired and selected target computing devices using both selection methods. For the few cases where a user first selected the wrong item and then selected the correct item, the time until the first selection was used (see section *5.2.3 Slips during identification* on p. 75 for a more detailed discussion of accidental item selection).

5.2. RESULTS AND DISCUSSION

Both the gesturePen and graphical list methods were easy to understand and use. We did not notice or analyze differences based on groups such as culture, gender, or religion. We organized and analyzed participant responses according to key qualitative measures suggested by Dryer *et al.* [39]. All rankings given in this section refer to a 5-point scale in the pre-trial questionnaire or a 7-point scale in the post-trial inquiry. For the pre-trial questionnaire, rankings for the scales were 1 = strongly agree and 5 = strongly disagree. For the post-trial inquiry, rankings for the scales were 1 = strongly agree and 7 = strongly disagree. Some measures were not directly measured with ranking questions, so not all categories have ranking data. Next, timing results for both user study phases are described. Then, common mistakes – or 'slips' – using the gesturePen and graphical list methods are analyzed. Copies of the questionnaires that were used during the trials and the complete responses for each participant are provided in *Appendix B*.

5.2.1. Qualitative results

During the post-trial inquiry, the experimenter asked each participant 22 questions (See *Appendix B* for a list of these questions and more detailed results). After ranking each question, the experimenter asked the participant to describe the reasoning for their response. The discussions from these questions helped to elucidate the qualitative analysis described in the social computing items described throughout this section.

Table 5 and Table 6 summarize the qualitative results according to the 7-point scales of the post-trial inquiry. Table 5 summarizes average rankings of how strongly participants agreed or disagreed with four general questions about the gesturePen and graphical list methods. Table 6 summarizes average rankings from three questions that compared the gesturePen method to the graphical list method with different list lengths. The trial question numbers in the tables correspond to the post-trial inquiry questions detailed in *Appendix B*. The lower rankings for the gesturePen method suggest that users slightly favoured our gesturePen method to the graphical list method. Also, the trend of increasingly higher rankings for selecting from lists of 5, 10, and 20 items suggests that longer list lengths are more difficult for users to navigate.

Table 5: Summary of participant general preferences
(1 = most positive ranking and 7 = least positive ranking)

	Graphical List	***gesturePen***
Mean	1.78	1.50
Standard Deviation	1.16	0.67
Trial Questions	*2, 4, 10, 12*	*1, 3, 9, 11*

Table 6: Summary of participant preferences related to graphical list length
(1 = most positive ranking and 7 = least positive ranking)[2]

	Graphical List			***gesturePen***
	5 items	*10 items*	*20 items*	
Mean	2.38	2.79	3.46	2.17
Standard Deviation	1.53	1.74	1.82	1.46
Trial Questions	*6, 16, 20*	*7, 17, 21*	*8, 18, 22*	*5, 15, 19*

Accessibility was high for both the graphical list and gesturePen methods. Participants ranked the ease of use an average of 1.4 / 7 for the list and 1.6 / 7 for the gesturePen. Small text size and scrolling were the main accessibility problems with the list. Participants ranked their ability to select a device using lists of 5, 10, and 20 items 1.1 / 7, 1.5 / 7, and 2.6 / 7, respectively; whereas, they ranked their ability to select a device using the gesturePen an average of 1.4 / 7. The main gesturePen difficulty was its range. Participants wanted to select devices from across the room (i.e. 6 – 10 m) instead of the 1.5 m range of our prototype.

---

[2] The most positive responses to questions 15-18 were to strongly disagree with these questions. Consequently, in Table 6, averages and standard deviations for questions 15-18 were calculated for ratings 8 *minus* the rating ranked by the participant (e.g. 6 would be mapped to 8 – 6 = 2).

Appeal ratings were an average of 2.6 / 7 for the list and 1.4 / 7 for the gesturePen. Participants liked the form factor and the direct interaction of the pen. One participant preferred the gesturePen because it saved screen real-estate. Some participants remarked that they would feel 'cool' using the handheld device and gesturePen, while others believed it would make them look like a 'geek'. Several participants also mentioned that any social stigmas associated with using the gesturePen or graphical list on a handheld would fade over time in a manner similar to cellular phone appeal.

Communication is supported well with both the graphical list and gesturePen methods. As mentioned in the *accessibility* sub-section above, participants favourably ranked both the graphical list and gesturePen methods. Thus, both the graphical list and gesturePen methods could facilitate easy sharing of information between people in social settings *if* the methods were used with well-designed 'host' application software and hardware.

Disruption was similar for the list and gesturePen – ranked 1.8 / 7 when using either in public. Participants were more concerned about psychological disruptions than physical ones such as noise or light. For example, several participants mentioned they would wonder what a person selected if the person clicked on a graphical list. Also, because it is socially unacceptable to point at people, participants stated they would feel uncomfortable if it appeared that another person was pointing at them with the gesturePen.

Familiarity was also high for both list and gesturePen. Participants ranked their ability to understand how to use the devices as 1.4 / 7 for the list and 1.3 / 7 for the gesturePen. Thus, even though the gesturePen was a new device that no participant had used before, participants were able to quickly understand its use – probably because participants tended to relate the gesturePen to a remote control.

All participants used a remote control often (1.3 / 5), but none of the participants used a laser pen often (4.3 / 5). Most participants expected the gesturePen to have a longer range like a remote control.

Identification was perceived to be similar for the graphical list and gesturePen. Both were identified as inclusive of most communities (e.g. both can be used by left- or right-handed people). Although all participants had good eyesight, one participant speculated that the gesturePen would be better for people with poor eyesight because users could point instead of reading a list with small fonts.

Input Sharing varied mainly with respect to the computing device being used. For example, a handheld computer affords input sharing with both graphical list identification and gesturePen methods because the handheld can be easily given to others. Participants noted that our gesturePen prototype was bulkier than a standard handheld computer stylus. Thus, it did not afford input sharing as well as the graphical list because the handheld and gesturePen were more difficult to give to another person than just a handheld with a stylus mounted inside its case. However, a wireless gesturePen could afford greater input sharing in some circumstances because it could be given to another person without its 'host' device such as a handheld computer.

Output Sharing slightly favours the gesturePen. For example, participants had a difficult time observing the handheld's screen when the experimenter used a graphical list while sitting next to the participant. Conversely, participants could usually deduce where the experimenter was pointing by looking at the experimenter's arm and hand. An exception was when several possible targets were close together, and were therefore not easily distinguishable to the observer. Participants ranked their ability to deduce the computing device selected by another person an average of 2.8 / 7, 3.3 / 7, and 3.6 / 7 for graphical lists of 5, 10, and 20 items, respectively, whereas they ranked the gesturePen 2.1 / 7.

Perceiver distraction slightly favoured the graphical list over the gesturePen because several participants noted that they would be uncomfortable if they perceived a stranger was pointing towards them (i.e. a social taboo in most societies). Conversely, using the handheld computer display creates a 'clicking' sound when a user taps the display with their stylus (i.e. a quiet clicking noise from the stylus touching the hard surface, and, depending on how the handheld is configured, possibly loud feedback from the handheld's speaker after every stylus action). This could potentially be distracting for nearby non-users (i.e. just like a laptop keyboard is distracting in some environments such as classrooms and meeting rooms). However, no participants mentioned the 'clicking' noise as distracting. Perceiver distraction seems most affected by the computing device with which the gesturePen or graphical list methods are used. For example, several participants remarked that the more flashy design of the iPAQ made it stand out more than the Palm Pilot (a Palm Pilot was the target PDA for phase 2 of the user study).

Pervasiveness was mainly dependent on the 'host' computer such as the handheld, not the identification method. Participants gave an average ranking of 1.4 / 7 for their ability to move freely with the handheld computer and gesturePen. None of the participants felt their mobility was reduced by the wire connecting the gesturePen to the handheld. However, several participants desired a gesturePen that is smaller and less bulky than our prototype (the size of a typical whiteboard marker). They also complained that the iPAQ was bulky because it had a sleeve for a wireless network card and additional hardware for the gesturePen. These size issues could be easily addressed by embedding hardware directly into the handheld computer. For the list selection method, several participants had greater difficulty selecting from a list while standing because they could not rest their elbow. These comments were supported by a greater number of errors when participants selected from a list while standing compared to sitting (see section *5.2.3 Slips during identification* on page 75 for more details).

Power was slightly greater for the graphical list than the gesturePen. Because participants perceived the list to be more private, the user could have a greater sense of power over others. The gesturePen was perceived as being more public because one can see where the user is pointing. Several participants felt the iPAQ handheld had an associated status because it was expensive, and thus gave them a sense of power. More status was given to the iPAQ than the Palm Pilot. All participants felt the power and status associated with the 'host' device such as a handheld far outweighed the power conferred by the gesturePen.

Relevance appeared to vary widely depending on the environment. Participants believed the graphical list would often be better in more static computing environments and/or with more knowledgeable users. For example, a person working in their own office would usually remember the name of their printer and could easily create a short-cut or default setting for their printer. Furthermore, their printer may not be directly in front of them, and even if it was in front of them, it may be more distracting to stop focusing on a computer monitor, gesture to a printer, and then re-establish focus on the monitor. Also, one participant was comforted seeing all available computing devices on a list. Conversely, other participants found long lists of devices overwhelming, and believed the gesturePen would be more useful in dynamic computing environments (i.e. "unknown, new environments"). For example, participants suggested they were more likely to identify objects as 'that one' in places such as an airport or someone else's office.

Social application was similar for both methods because the functionality of the graphical list and gesturePen methods are equivalent – both enable users to select devices in ad hoc mobile environments. Thus, both methods can potentially facilitate rich social interactions. We hypothesized that the gesturePen would slightly enhance social interactions because users could context switch between another person and the target device faster than context switching between another person and the handheld's screen. We reasoned that participants would have more difficulty with the low contrast graphical objects on the handheld's screen and experience greater re-focusing of their eyes when context switching between the handheld and another person. We did not however observe any consistent and significant instances of participants gesturing to a target tag without looking directly at the tag before, during, and after the gesture. Maybe after users had a chance to become more comfortable with the gesturePen and gain an intuitive grasp of its range, they would be more adept at gesturing to a target device without making extended eye contact with the device. Current file sharing applications require significant attention to a computer display, and poorly facilitate rapid establishment and relinquishment of a user's focus to a computing device while interacting in a social context. Until such applications improve, interaction methods such as the gesturePen method will only marginally reduce the overall percentage of time a user needs to focus on a computing device. However, the combination of many small social applications, such as using the gesturePen method, could total a significant overall improvement.

User Distraction was evaluated by asking participants how much attention they needed to select using the graphical list or gesturePen. Participants needed more attention for the graphical lists. Rankings were 2.2 / 7, 2.6 / 7, and 3.1 / 7 for graphical lists of 5, 10, and 20 items, respectively, whereas the gesturePen was ranked 2 / 7. Scrolling and switching windows were the most distracting elements of the graphical list method. During the cognitive load task, we noticed participants did not need to look up (to read a tag's name) when they were interrupted from their puzzle and asked to gesture to a tag on their left or right. Thus, the participants were able to stay more focused on the task at hand than when they selected from a graphical list. By contrast, some participants still looked up from their puzzle towards the desired tag while gesturing. Some participants said they used the blinking light as feedback that they had performed the task correctly, but others said they ignored the blinking light and relied exclusively on the handheld display for feedback.

5.2.2. Identification times

Table 7 and Table 8 show the mean and standard deviation times that participants took to identify a device during the cognitive load and mobile environment tasks, respectively.[3] The graphical list results include an equal number of lists containing 5, 10, and 20 items. These lists were randomly distributed among the trials.

Table 7 shows results for phase 1 of the user study (i.e. the cognitive load task where two tags were placed in front of the participant). The table results suggest that participants were able to identify a device significantly faster using the gesturePen than a graphical list within the static environment during the cognitive load task (previously illustrated in Figure 17). Since the graphical list was always displayed on the participant's screen during this task, this result represents a 'best case' scenario for graphical list selection. Consequently, we believe the gesturePen would outperform graphical list selection by greater margins during common computing tasks. For example, several participants said that they would spend more time selecting from a graphical list that was not currently visible on their screen. In other words, in most computing situations, to select from a

---

[3] One participant's phase 1 (cognitive load task) data was not collected due to software difficulties.

graphical list, users would need to release focus from their current task to show a list of possible devices.

Table 8 summarizes results from phase 2 of the user study (i.e. the mobile environment task where users walked towards a computing device and identified it). No significant difference was found for the time it took to identify a device using either the gesturePen or the graphical list method. As shown in Table 8, a wide variance was shown in the time participants took to identify a device using the gesturePen within the environment (previously illustrated in Figure 18). After discussions with participants during our qualitative analysis and reviews of our video logs, we believe the large standard deviation was due to the gesturePen's range of 1.5 m and beam dispersion angle of 30°. If a target device was within a participant's line-of-sight, most participants had a strong desire to point directly to the device from their current position. Thus, they desired a gesturePen with a longer range and narrower beam angle. Participants often attempted to point to a tag that was out of range, then walk closer to the tag and point again. Conversely, when selecting from a graphical list, participants only needed to walk close enough to a tag such that they could read its name. In other words, participants received constant feedback of when their eyes were 'in range' (i.e. could read a tag), but received feedback from the gesturePen only when they were within range of a tag and pressed the pen's button. Thus, we believe that increasing the gesturePen's range to ~6 – 10 m would significantly improve the speed with which users could identify a device. However, in environments with many computing devices, the longer range may make it more difficult to discriminate between computing devices because the gesturePen could potentially communicate with many more devices. One could solve the problem of non-continuous feedback by adding a laser pen to the gesturePen, or continually illuminating tags when they are within range of a gesturePen.

Table 7: Phase 1 – Cognitive load task times

	Graphical List	***gesturePen***
Mean	4.4	2.9
Standard Deviation	1.8	1.8
	F(1,6)=19.48, p=0.005	

Table 8: Phase 2 – Mobile environment task times

	Graphical List	***gesturePen***
Mean	3.8	4.4
Standard Deviation	1.8	3.9
	F(1,7)=0.998, p=0.351	

5.2.3. Slips during identification

Using the graphical list and gesturePen methods resulted in different 'slips'. Norman [69, p. 105] states that slips "result from automatic behavior, when subconscious actions that are intended to satisfy our goals get waylaid en route". Thus, we labeled any misidentification of the correct target tag because of an execution mistake as a slip. Specifically, while intending to select the correct tag, some participants accidentally selected an unintended item in a graphical list, or pointed to an unintended computing device with the gesturePen.

As shown in Table 9, no participants selected the wrong tag during phase 1 (i.e. the cognitive load task). A few participants did, however, accidentally identify the wrong device during phase 2 (i.e. the mobile environment task where users walked towards a computing device and identified it). We believe comments from the post-trial inquiry explain why participants made these slips. Using the graphical list, the seven misidentified devices were all directly above or below the target name in the list. Several participants noted that selecting from a list was more difficult when they were standing because they could not stabilize their hand by resting their elbow on an object such as a table. Using the gesturePen, the three misidentified device names all belonged to a tag neighbouring the target tag. Comments from our post-trial inquiry suggest that slips could be reduced by increasing the font size in a graphical list, and narrowing the beam angle for the gesturePen. However, increasing the font size of list items would increase the average amount of scrolling required to select a desired item. Narrowing the beam angle of the gesturePen would require users to gesture towards a tag with increased accuracy.

Table 9: Total numbers of misidentified devices

	Graphical List	***gesturePen***
Phase 1 – Cognitive load task	0	0
Phase 2 – Mobile environment task	7	3

Chapter 6
Future Work and Conclusions

6.1. SUMMARY OF CONTRIBUTION

Our gesturePen method separates the task of sharing information into two main processes:

1. Identification (of the target user(s) and device(s))
2. Transfer (of the data)

In other words, we uniquely identify a tagged device by pointing to the tag with our gesturePen, and then copy desired information over a network to or from the device associated with the tag. This separation of identification and transfer utilizes the main benefits of two common data sharing methods: infrared 'beaming' and network copying. Specifically, infrared 'beaming' involves copying information, such as a document, over a line-of-sight infrared link – often performed between two handheld devices. Network copying involves selecting a computing device from a user interface widget, such as a graphical list, and then transferring information over a network – often performed by File Transfer Protocol (FTP) applications. Table 10 summarizes the main advantages and disadvantages of infrared 'beaming', network copying, and our gesturePen method.

Identification is inherently a directional task, thus it is best performed using a line-of-sight infrared communications link. Conversely, transferring information is inherently a non-directional task, thus it is best performed using an omni-directional radio frequency communications link. Once a source and target has been identified, the actual data transmission path is irrelevant to the user as long as information is routed from the desired source to the desired destination. Both the qualitative and quantitative results from our user study support the claims in Table 10 that our gesturePen method combines the benefits of a directional communications link for identification, and a non-directional communications link for transfer of data.

Table 10: Comparison of information sharing methods

	Infrared Beaming	*Network Copying*	*gesturePen Method*
Advantages			
Target device is implicitly identified by pointing towards it	✓		✓
Can simultaneously transfer data to many devices		✓	✓
Disadvantages			
Both devices must stay stationary to maintain infrared link	✓		
Data can only be transferred between two devices at once	✓		
Target device(s) must be selected from user interface widgets such as a list, tree, or command prompt		✓	
Requires more infrastructure (minor cost problem that can be addressed through economy-of-scale)			✓

6.2. FUTURE WORK

Future versions of the gesturePen could incorporate sensors, such as accelerometers, and intelligence to react to more subtle user movements. For example, instead of pressing a button on the gesturePen, the user could flick their wrist, or squeeze a pressure sensor, after pointing to a tag. Additionally, different gestures could be used to identify different sets of tags within dense computing environments. For example, a sweeping gesture could activate several tags within the sweep. During the user study, participants suggested moving the infrared transceiver and button to the same end of the gesturePen as the stylus tip. This would eliminate the need to swivel the gesturePen in the user's hand before gesturing to a tag. A dial on the gesturePen case to adjust the pen's range and/or beam angle was also suggested.

Most current applications, such as file explorers and word processors, are designed for the user to keep focusing on the computing device display for an extended amount of time. Conversely, ubiquitous computing environments will require users to rapidly establish and relinquish focus with a computing device just like one establishes and relinquishes focus with a stack of photos while talking to other people. Future work could be done to research how the gesturePen and other interaction methods facilitate rapid re-focusing between multiple computing devices and people. To copy data from one computing device to another, users will often switch their context several times in rapid succession. For example, a user may switch their focus from another person, to a wall display, to their handheld, and then back to the person with whom they were initially engaged. Further research could analyze if users switch contexts differently while using the gesturePen and graphical widget selection methods.

Different densities of tagged devices would probably affect the optimal range and beam angle of the gesturePen and tags. A gesturePen or tag with a narrower beam angle such as 5° might be more effective for environments with very dense clusters of tagged computing devices. Likewise, a gesturePen or tag with a longer range or wider beam angle might be better suited for more open, sparsely arranged clusters of tagged computing devices. Developing and performing user studies with a new gesturePen and tags with variable ranges and beam angles would allow scalability issues to be addressed.

Feedback methods other than our blinking LED would also be interesting to evaluate. For example, more sophisticated protocols could be developed such as having the tag's LED blink once when activated, and twice when the gesturePen communicated a successful transfer of information. We could study the effects of continuous feedback by adding a laser pointer next to the IR transceiver on the end of the gesturePen. The laser pointer would provide feedback to the user of their current target (i.e. a laser dot on the tag) before pressing the button on the gesturePen to request the device's identity. Since some participants in our user study looked at their handheld computer instead of the tag for feedback, we could also experiment with different feedback in the user interface on the gesturePen's host computing device.

Although not addressed in this thesis, security and authentication are important considerations for any commercial use of an interaction method such as our gesturePen method. Encryption could be added to the communications between the gesturePen and tags. Sharing of physical and software public keys could also be explored. During the identification stage of our two step *identification* and *transfer* method of sharing information, one could include many types of authentication. Does this person have authority to identify or access the device? Is he or she only allowed to identify the device, or can he or she transfer information to it as well? Does the person only have access to a certain class of devices such as printers? How can security and authentication be managed with minimal user effort? If I'm trusted to use a computing device, and I trust my friend, should my friend be allowed to use the computing device too?

We could also experiment with different uses of the tags. One idea would be to change the communications protocol between the gesturePen and tags to include spatial information such as 2-D co-ordinates, active graphical windows, or software applications. Such spatial information would enable users to control where they interact and with what they interact while identifying a target computing device. For example, users could identify a location on a wall display in a way similar to Rekimoto's Pick-and-Drop research [81]. Since our gesturePen and tags utilize a two-way communications link, tags could also be dynamically re-programmed using the gesturePen. Thus, users could leave messages for people in tags just like using a Post-It Note or voice mail. Also, several tags could each represent a different group of people or objects. For example, during an ad hoc meeting, people could point towards tags to re-program them. Names of the people or objects in the group, or aliases of computing devices that are not within line-of-sight, could be stored. Thus, the tags would act as physical icons shared among the group of people and their computing devices.

6.3. CONCLUSIONS

Our user study results suggest the gesturePen method is well suited to the dynamic, ubiquitous computing environments that are envisioned to be commonplace in the near future. Since our tags are two-way communication devices – not beacons – our tags only communicate information when requested by a user. Our approach reduces the cognitive load associated with using devices. Users can more easily focus on their current tasks without being distracted by nearby devices broadcasting for their attention.

Even though no participants had used our gesturePen before, they were comfortable using the gesturePen and found our identification technique very easy to learn. Improvements to our initial prototype, such as increasing the range of the gesturePen and improving feedback, would result in a faster, more usable system.

Participants suggested a graphical list would be more useful in current office environments. For example, users in their own offices will typically know the name of their favourite printer. Identifying the printer from a list enables a user to select it without being within its line-of-sight. Conversely, participants believed our gesturePen method would be better in mobile environments. For example, selecting from a list while holding a handheld computer steady is more difficult and error prone than using our gesturePen. Also, users are much less likely to know or desire the name of the device in front of them when in more mobile settings such as malls, airports, or foreign offices.

The feeling of being overwhelmed that is associated with many technologies, such as desktop computing, does not occur with most interactions in our daily lives. Many researchers have discussed ways to mitigate such overwhelming feelings [37],[45],[69],[88],[106],[107]. For example, most people do not experience information overload or stress when viewing physical photos, or opening a door. We do not consciously think about things or technologies that are extremely usable. As Mark Weiser stated, "The most profound technologies are those that disappear."[105] Our gesturePen method is a small step towards the goal of ubiquitous computing. The concept of using a line-of-sight pointing gesture to identify a device, then transferring information over a wireless network, can facilitate more flexible computing environments.

As people and their devices become increasingly mobile, more situations will arise where users will want to transfer information to 'that device there' instead of navigating through traditional graphical widgets such as lists. Items within these widgets will constantly fluctuate as new devices join and leave a network, but the complexity of selecting with the gesturePen will stay relatively constant regardless of the number of computing devices on the network.

References

CORPORATIONS

[1] Agilent Technologies http://www.agilent.com.

[2] Bluefish Technologies http://www.bluefish-tech.com.

[3] Colligo Wireless http://www.colligo.com.

[4] Compaq http://www.compaq.com.

[5] Custom Computing Services http://www.ccsinfo.com.

[6] Dynasys Technologies http://www.dyna-sys.com.

[7] Ericsson http://www.ericsson.com.

[8] eXI Wireless Systems http://www.exi.com.

[9] Handspring http://www.handspring.com.

[10] iButton.com http://www.ibutton.com.

[11] Kent Displays http://www.kentdisplays.com.

[12] Microchip http://www.microchip.com.

[13] Microsoft Research http://research.microsoft.com/.

[14] Palm http://www.palm.com.

[15] Smartboard http://www.smarttech.com/smartboard.

[16] Softboard http://www.softboard.com.

[17] Sony Computer Science Laboratories http://www.csl.sony.co.jp/.

[18] Sun Microsystems http://www.sun.com/jini.

[19] Texas Instruments http://www.ti.com/tiris/main.htm.

[20] Xenote http://www.xenote.com.

[21] Xerox PARC http://www.parc.xerox.com.

[22] Xicor http://www.xicor.com.

PAPERS AND BOOKS

[23] _____ (1997) *Infrared Data Association Serial Infrared Physical Layer Link Specification.* Version 1.2.

[24] Ackoff, R. (1994) *System thinking and thinking systems.* Available at: http://courses.lib.odu.edu/commhealth/pstepano/ackoffart.ppt

[25] Adams, N., Gold, R., Schilit, B., Tso, B., and Want, R. (1993) An Infrared Network for Mobile Compters. In *Proceedings of USENIX Symposium on Mobile and Location-Independent Computing.* Boston, Massachusetts, 1-3 August 1993.

[26] Agilent Technologies (2001) *HSDL 3201/7001 Specifications.*

[27] Agilent Technologies (2001) *Agilent HSDL-3000#007/017 IrDA Data Compliant 115.2 kbps Infrared Transceiver Data Sheet.*

[28] Barrett, R. and Maglio, P.P. (1998) Informative Things: How to attach information to the real world. In *Proceedings of UIST 1998.* San Francisco California, 1-4 November 1998.

[29] Bergqvist, J., Dahlberg, P., Ljungberg, F. and Kristoffersen, S. (1999) Walking Away from the Meeting Room: Exploring Mobile Meetings. In Proceedings of *ECSCW 1999*, Copenhagen, Denmark, 12 September 1999.

[30] Bolt, R.A. (1980) Put-That-There: Voice and Gesture at the Graphics Interface. *ACM Computer Graphics.* Vol. 14(3): 262-270.

[31] Borovoy, R., Martin, F., Vemuri, S., Resnick, M., Silverman, B., and Hancook, C. (1998) Meme Tags and Community Mirrors: Moving from Conferences to Collaboration. In *Proceedings of CSCW 1998.* Seattle, Washington, 14 November 1998.

[32] Bucholtz, C. (2000) *Software Behind HP Labs' CoolTown.* Available at: http://www.interex.org/hpworldnews/hpw005/03news.html

[33] Butterworth, G. and Grover, L. (1990) Joint Visual Attention, Manual Pointing, and Preverbal Communication in Human Infancy. *Attention and Performance XIII: Motor Representation and Control.* Jeannerod, M. (Ed.). ISBN: 0805805656.

[34] Cassell, J., and Stone, M. (1999) Living Hand to Mouth: Psychological Theories about Speech and Gesture in Interactive Dialogue Systems. In *Proceedings of AAAI 1999 Fall Symposium on Narrative Intelligence.* North Falmouth, Massachusetts, 5-7 November 1999.

[35] Cassell, J., and Bickmore, T. (2001) A Relational Agent: A Model and Implementation of Building User Trust. In *Proceedings of CHI 2001*. Seattle, Washington, March 31-April 5 2001.

[36] Compton, J. (2001) *Groupware grows up – finally*. ZD Inc. Available at: http://netscape.zdnet.com/zdnn/stories/news/0,4586,2698269,00.html.

[37] Dey, A.K., Abowd, G.D., and Salber, D. (1999) A Context-Based Infrastructure for Smart Environments. In *Proceedings of MANSE 1999*. Dublin, Ireland, 13-14 December 1999.

[38] Dray, S. (2001) *Structured observation*. Tutorial presented at ACM CHI 2001. Seattle, Washington, ?-? April 2001.

[39] Dryer, D.C., Eisbach, C., and Ark, W.S. (1999) At what cost pervasive? A social computing view of mobile computing systems. *IBM Systems Journal*, vol. 38(4): 652-677.

[40] Efron, D. (1941) *Gesture and Environments*. King's Crown Press, Morningside Heights, New York.

[41] Engeström, Y., Miettinen, R. and Punamäki, R-L. (Eds.) (1998), *Perspectives on activity theory*. ISBN 052143730X.

[42] Falk, J., Redström J., and Björk, S. (1999) Amplifying Reality. In *Proceedings of HUC 1999*. Karlsruhe, Germany, 27-29 September 1999.

[43] Fishkin, K., Gujar, A., Harrison, B., Moran, T. and Want, R. (2000) Embodied User Interfaces for Really Direct Manipulation. *Communications of the ACM*. vol.43(9):75-80.

[44] Fitts, P. and Peterson, J. (1964) Information capacity of Discrete Motor Responses. *Journal of Experimental Psychology*. vol. 67:103-112, 1964.

[45] Fox, A., Johnston, B., Hanrahan, P., and Winograd, T. (2000) Integrating Information Appliances into an Interactive Workspace. *IEEE Computer Graphics*, May/June 2000.

[46] Freedman, N. (1972) The analysis of movement behavior during the clinical interview. In Siegman, A. and Pope, B. (Eds.) *Studies in Dyadic Communication*. ISBN 0080158676.

[47] Goffman, E. (1971) *Relations in public; microstudies of the public order*. Basic Books.

[48] Goffman. E. (1966) *Behavior in Public Places: Notes on the Social Organization of Gatherings*. ISBN 0029119405.

[49] Graham, J. A. and Argyle, M. (1975) A Cross-Cultural Study of the Communication of Extra-verbal Meaning by Gestures. *International Journal of Psychology*, vol. 10(1):57-67.

[50] Gutwin, C. and Greenburg, S. (1999) The effects of workspace awareness support on the usability of real-time distributed groupware. In *ACM Transactions on Computer-Human Interaction.* vol. 6(4):243-281.

[51] Gutwin, C., and Greenburg, S. (2000) The Mechanics of Collaboration: Developing Low Cost Usability Evaluation Methods for Shared Workspaces. *IEEE 9th International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET-ICE'00).* Gaithersburg, Maryland, 14-16 June 2000.

[52] Hahn, J., and Kim, J. (2000) Why Are Some Diagrams Easier to Work With? Effects of Diagrammatic Representation on the Cognitive Integration Process of Systems Analysis and Design. In *ACM Transactions on Computer-Human Interaction.* vol. 6(3):181-213.

[53] Hinckley, K., Pierce, J., and Horvitz, E. (2000) Sensing Techniques for Mobile Interaction. In *Proceedings of UIST 2000.* San Diego, California, 5-8 November 2000.

[54] Hutchins, E. (1990) The Technology of Team Navigation. In *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work.* Galegher, J., Kraut, R.E., and Egido, C. (Eds.). ISBN: 0805805346.

[55] Kidd, C.D., Orr, R.J., Abowd, G.D., Atkeson, C.G., Essa, I.A., MacIntyre, B., Mynatt, E., Starner, T.E., Newstetter, W. (1999) The Aware Home: A Living Laboratory for Ubiquitous Computing Research. In *Proceedings of the International Workshop on Cooperative Buildings (CoBuild).* Pittsburgh, Pennsylvania, 1-2 October 1999.

[56] Kortuem, G., Segall, Z., Cowan G.T. (1999) Close Encounters: Supporting Mobile Collaboration through Interchange of User Profiles. In *Proceedings of HUC 1999.* Karlsruhe, Germany, 27-29 September 1999.

[57] Krauss, R., and Fussell, S. (1990) Mutual knowledge and communicative effectiveness. *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work.* Galegher, J., Kraut, R.E., and Egido, C. (Eds). ISBN 0805805346.

[58] Krauss, R., Morrel-Samuels, P., and Colasante, C. (1991) Do conversational hand gestures communicate? *Journal of Personality and Social Psychology*, 61(5):743–754.

[59] Kuuti, K. (1996) Activity Theory as a potential framework for Human-Computer Interaction research. *Context and Consciousness: Activity Theory and Human-Computer Interaction.* Nardi, B.A. (Ed.). ISBN: 0262140586.

[60] Ljungstrand, P. and Holmquist, L.E. (1999) WebStickers: Using Physical Objects as WWW Bookmarks. In *Conference Abstracts of CHI 1999.* Pittsburgh, Pennsylvania, 15-20 May 1999.

[61] McGee, D.R. and Cohen, P. (1998) Exploring Handheld, Agent-based, Multimodal Collaboration. In *Proceedings of CSCW 1998.* Seattle, Washington, 14 November 1998.

[62] McLuhan, M. and Fiore, Q. (1967) *The Medium is the Message.* ISBN: 0671689975

[63] McNeil, D. (1992) *Hand and Mind: What Gestures Reveal about Thought.* ISBN: 0226561321.

[64] Microchip, Inc. (2001) *PIC 16F870 Specifications* . Available at: http://www.microchip.com/

[65] Moore, D.J., Want, R., Harrison, B.L., Gujar, A., and Fishkin, K. (1999) Implementing Phicons: Combining Computer Vision with InfraRed Technology for Interactive Physical Icons. In *Proceedings of UIST 1999.* Asheville, North Carolina, 7-10 November 1999.

[66] Moran, D.B., Cheyer, A.J., Julia, L.E., Martin, D.L. and Park, S. (1997) Multimodal user interfaces in the Open Agent Architecture. In *Proceedings of IUI 1997.* Orlando, Florida, 6-9 January 1997.

[67] Moran, T.P., Chiu, P., Harrison, S., Kurtenbach, G., Minneman, S. and van Melle, W. (1996) Evolutionary engagement in an ongoing collaborative work process: A case study. In *Proceedings of CSCW 1996.* Boston, Massachusetts, 16-20 November 1996.

[68] Myers, B.A., Bhatnagar, R., Nichols, J., Peck, C.H., Kong, D., Miller, R. and Long, A.C. (2002) Interacting at a Distance: Measuring the Performance of Laser Pointers and Other Devices. In *Proceedings of CHI 2002.* Minneapolis, Minnesota, 20-25 April 2002.

[69] Norman, D. (1990) *The Design of Everyday Things.* ISBN 0385267746.

[70] Norman, D (1998). *The Invisible Computer: Why Good Products Can Fail, the Personal Computer Is So Complex, and Information Appliances Are the Solution.* ISBN 0262640414.

[71] Olstad, L., Ramirez, J., Brady, C., and McHollan, B. (1999) *Jini Technology: Impromptu Networking and its Impact on Telecommunications.* Group Thesis Proceeding. University of Colorado at Boulder.

[72] Pascoe, J., Ryan, N., and Morse, D. (1999) Issues in Developing Context-Aware Computing. In *Proceedings of HUC 1999.* Karlsruhe, Germany, 27-29 September 1999.

[73] Pederson, E. R., McCall, K., Moran, T. P., and Halasz, F. G. (1993) Tivoli: An Electronic Whiteboard for Informal Workgroup Meetings. In *Proceedings of CHI 1993*, Amsterdam, Netherlands, 24-29 April 1993.

[74] Perrow, C. (1984) *Normal Accidents: Living with High-Risk Technologies.* ISBN: 0691004129.

[75] Poor, R.D. (1999) *The iRX 2.1 ...where atoms meet bits.* Available at: http://www.media.mit.edu/~r/projects/picsem/irx2/

[76] Project Bodytalk (2001) Multimedial presentation system as an example for gesture based human-computer-interaction. Available at: http://ls7-www.cs.uni-dortmund.de/bodytalk/.

[77] Queck, F. (1996) Unencumbered Gesture Interaction. *IEEE MultiMedia.* 3(4):36-47, Winter 1996.

[78] Ræithel, A. (1992) Activity theory as a foundation for design. In *Software Development and Reality Construction.* Floyd C., Züllighoven, H., Budde, R. and Keil-Slawik, R. (Eds.) ISBN 354054349X.

[79] Redström, J., Dahlberg, P., Ljungstrand, P., Holmquist, L.E. (1999). Designing for Local Interaction. In *Proceedings of MANSE 1999.* Dublin, Ireland, 13-14 December 1999.

[80] Reeves, B. and Nass, C. (1996) *The Media Equation: How People Treat Computers, Television and New Media like Real People and Places.* ISBN: 1575860538.

[81] Rekimoto, J. (1997) Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Evironments. In *Proceedings of UIST 1997.* Banff, Alberta, 14-17 October 1997.

[82] Rimé, B. (1982) The elimination of visible behavior from social interactions: effects of verbal, nonverbal and interpersonal variables. *European Journal of Social Psychology*, 12:113–129.

[83] Rogers, W.T. (1978) The contribution of kinesic illustrators towards the comprehension of verbal behavior within utterances. *Human Communication Research*, 5:54-62.

[84] Rubine, D. (1991) *The Automatic Recognition of Gestures.* Ph.D. Dissertation, Carnegie-Mellon University.

[85] Schilit, B.N., Adams, N.I., and Want, R. (1994) Context-Aware Computing Applications. In *Proceedings of the Workshop on Mobile Computing Systems and Applications*. IEEE Computer Society. Santa Cruz, California, December 1994.

[86] Schmidt, A. (1999) Implicit human-computer interaction through context. In *Proceedings of INTERACT 1999*, Edinburgh, Scotland, 30 August – 3 September 1999.

[87] Schmidt, A., Aidoo, K., Takaluoma, A., Tuomela, U., Van Laerhove, K., and Van de Velde, W. (1999) Advanced Interaction in Context. In *Proceedings of HUC 1999*. Karlsruhe, Germany, 27-29 September 1999.

[88] Schmidt, K. and Simone, C. (2000) Mind the gap! Towards a unified view of CSCW. In *Proceedings of COOP 2000*. Sophia Antipolis, France, 23-26 May 2000.

[89] Senta, T.D. (2000) *Made as a speaker in the "Television Challenge" panel in the Opening Plenary of the United Nations World Television Forum 2000*. Available at: http://www.tvforum.org/asp2/forum_2000.asp#challengeSpka.

[90] Shneiderman, B. (2000) Creating Creativity: User Interfaces for Supporting Innovation. In *ACM Transactions on Computer-Human Interaction*. vol. 7(1):114-138.

[91] Shu, K., Swindells, C., Chai, D., and Inkpen, K.M. (2002) WindowSpaces to share our digital media. *Tech Report CMPT2002-04*. School of Computing Science, Simon Fraser University. Available at: http://www.cs.sfu.ca/

[92] Sibert, J.L., and Gokturk, M. (1997) A Finger-Mounted, Direct Pointing Device for Mobile Computing. In *Proceedings of UIST 1997*. Banff, Alberta, 14-17 October 1997.

[93] Sugiura, A., and Koseki, Y. (1998) A User Interface Using Fingerprint Recognition – Holding Commands and Data Objects on Fingers. *In Proceedings of UIST 1998*. San Francisco, California, 1-4 November 1998.

[94] Sun Microsystems (2001) *Jini Network Technology: An Executive Overview*. External corporate whitepaper.

[95] Suvak, D. (2000) *IrDA and Bluetooth: A Complementary Comparison*. Extended Systems, Inc. Tech Note. Available at: http://www.extendedsystems.com.

[96] Swindells, C. and Inkpen, K.M. (2002) What is that? Gesturing to determine device identity. In *Extended Abstracts of CHI 2002*. Minneapolis, Minnesota, 20-26 April 2002.

[97] Swindells, C., Inkpen, K.M., Dill, J.C., Tory, M. (2002) Use that there! Gesturing to determine device identity. In *Proceedings on User Interface Software and Technology (UIST).* Paris, France, 27-30 October 2002.

[98] Teitelbaum, R.C., and Granda, R.E. (1983) The effects of positional constancy on searching menus for information. In *Proceedings of CHI 1983.* Boston, Massachusetts, 12-15 December 1983.

[99] Uden, L., and Willis, N. (2001) Designing User Interfaces using Activity Theory. In *Proceedings of the 34th Hawaii International Conference on Systems Sciences (HICSS-34).* Wailea Maui, Hawaii, 3-6 January 2001.

[100] Ullmer, B. and Ishii, H. (1999) mediaBlocks: Tangible Interfaces for Online Media. In *Conference Abstracts of CHI 1999.* Pittsburgh, Pennsylvania, 15-20 May 1999.

[101] Want, R. and Borriello, G. (2000) Information Appliances. *IEEE Computer Graphics*, May/June 2000.

[102] Want, R. and Borriello, G. (2000) Survey on Information Appliances. *IEEE Computer Graphics*, May/June 2000.

[103] Want, R., and Hopper, A. (1992) Active Badges and Personal Interactive Computing Objects. *IEEE Transactions of Consumer Electronics.* vol. 38(1):10-20.

[104] Warrior, A. (1996) Author's verbal discussions with a pilot.

[105] Weiser, M. (1991) The computer for the twenty-first century. *Scientific American.* vol. 265(3):94-104, September, 1991.

[106] Weiser, M. (1993) Some computer science issues in ubiquitous computing. *Communications of the ACM.* vol. 36(7):75-85, July 1993.

[107] Weiser, M. (1994) The World is not a Desktop. *ACM Interactions*; vol. (1)1:7-8, January 1994.

[108] Zucker, P. (1963) *Styles in painting: a comparative study.* Dover Press, New York. ISBN 0486207609.

Appendix A: Experimenter's User Study Script

Hi, I'm Colin, and I am going to ask you to do a user study with two parts. In the first part, I will ask you to sit here [gesture to seat in front of the tabletop display] and play a jigsaw puzzle game. While you are playing the jigsaw puzzle, I will distract you and ask you to select one of these two tags [point to left and right tags on the tabletop display]. First, I will ask you to select one of the tags by pointing with this [gesturePen] to the tag and pressing this button [press the button while pointing to the tag and the LED on the tag blinks][4]. After you have done this a few times, I will ask you to select the tag in a different way. Instead of pointing to the tag with this pen, you will read the label on the tag [point to label on tag] and select the corresponding item from a list like this [show the list on the iPAQ handheld computer]. Do you have any questions? [Answer questions. Let the participant try selecting tags using the gesturePen and selecting items from the list. Start to describe phase II when the participant is ready.]

Now, I will describe the second part. Just like before, I will ask you to select a tag using either this pen or this graphical list. However, I will ask you to first stand on this star [point to star on the floor], then go over to one of these five objects [experimenter points to each while speaking]: tabletop display, laptop, printer, Palm Pilot, or monitor, and select it's tag. Then, I will ask you to return back here and stand on the star before I ask you to select an object again. Do you have any questions? [Answer questions. Let the participant try selecting tags using the gesturePen and selecting items from the list.]

---

[4] Because the trials were counterbalanced such that half the participants started with the gesturePen method and half the participants started with the graphical list method, the experimenter altered the script to reflect the order in which the current participant would be performing the tasks.

Now please read and complete these forms [ethics forms]. I will be video taping this experiment and audio recording your comments, but I will not make these records public without your permission and I will not associate your name with any of the results. Do you have any questions about these forms or the experiment? [Participant fills out forms]

[Start video taping and audio taping]

[Ask participant to fill out the pre-trial questionnaire]

[Start phase I of the user study and remind the participant to press the ready button before making a selection. The experimenter repeatedly states one of the following two commands.]

- Click the ready button and select the tag on your left
- Click the ready button and select the tag on your right

[Start phase II of the user study and remind the participant to press the ready button before making a selection. The experimenter repeatedly states one of the following five commands.]

- Click the ready button and select the tabletop display.
- Click the ready button and select the laptop.
- Click the ready button and select the printer.
- Click the ready button and select the Palm Pilot.
- Click the ready button and select the monitor.

[Ask the participant to sit in the chair beside the tabletop display and fill out the post-trial questionnaire. Once the participant finishes filling the post-trial questionnaire the experimenter asks him or her to wear a microphone on their shirt and checks the audio level. The experimenter first asks the participant if he or she has any questions or comments about the post-trial questionnaire. Next, the experimenter dictates each question on the post-trial inquiry form and asks the participant to rank their level of agreement with the question on a scale of 1 = strongly agree to 7 = strongly disagree. Once the participant ranks a question, the experimenter asks the participant to explain why they chose the ranking. The experimenter jots down notes while listening to the participant, and probes for clarification and more information using the least judgmental language possible. After finishing the post-trial questionnaire, the experimenter asks the participant if he or she has any remaining comments.]

[The experimenter pays the subjects CDN $15 and asks them to sign a receipt of payment form. The experimenter then thanks the participant and escorts him or or her out of the lab.]

Appendix B:
Trial Questionnaires

Pre-Trial Questionnaire

For each of the questions below, please indicate the extent of your agreement or disagreement by checking **one** choice for **each device**

1. I use the following devices on a regular basis:

Device	Strongly Agree (use often)	Slightly Agree	Neutral	Slightly Disagree	Strongly Disagree (never used)
Laptop Computer					
Desktop Computer					
Remote Control					
Laser Pointer					
Handheld e.g.Palm Pilot, IPAQ					

2. I feel **comfortable** using the following devices:

Device	Strongly Agree	Slightly Agree	Neutral	Slightly Disagree	Strongly Disagree
Laptop Computer					
Desktop Computer					
Remote Control					
Laser Pointer					
Handheld e.g.Palm Pilot, IPAQ					

Post-Trial Questionnaire

1. Briefly describe the most significant advantages and disadvantages for the menu and gesture interaction techniques:

GUI List

Advantage ______________________________

____

Disadvantage ______________________________

____

____

gesturePen

Advantage ______________________________

____

Disadvantage ______________________________

____

____

____

2. Which did you find **easiest** to use? **GUI List** or **gesturePen**
circle **one**

Why?

3. Which did you **prefer** to use? **GUI List** or **gesturePen**
circle **one**

Why?

4. When would you prefer to use a…

Write a phrase for **each** interaction technique for a total of **2 phrases**

GUI List

gesturePen

5. Please write any other comments on the back of this sheet.

Likert-type Scales

	For each of the statements below, please indicate the extent of your agreement or disagreement by placing a check in the appropriate column	Strongly Agree	Agree	Somewhat Agree	Neither Agree nor Disagree	Somewhat Disagree	Disagree	Strongly Disagree
	Accessibility							
1	I understand how to use the gesturePen							
2	I understand how to use the GUI list							
3	I can use the gesturePen easily							
4	I can use the GUI list easily							
5	I can *select* a computing device easily using the gesturePen							
6	I can *select* a computing device easily using a GUI list with 5 items							
7	I can *select* a computing device easily using a GUI list with 10 items							
8	I can *select* a computing device easily using a GUI list with 20 items							
	Appeal							
9	I like using the gesturePen							
10	I like using the GUI list							
11	I would feel comfortable using the gesturePen in public							
12	I would feel comfortable using the GUI list in public							
13	I would feel comfortable using the trial handheld computer (Sparky) in public							
14	I can move freely using the trial handheld computer (Sparky)							
	Distraction							
15	I need almost all my attention to use the gesturePen							
16	I need almost all my attention to use the GUI list with 5 items							
17	I need almost all my attention to use the GUI list with 10 items							
18	I need almost all my attention to use the GUI list with 20 items							
	Awareness *(Experimenter 1st selects w/ the pen & from a GUI in front of the subject)*							
19	I always knew the target device when the experimenter used the gesturePen							
20	I always knew the target device when the experimenter selected from a GUI list with 5 items							
21	I always knew the target device when the experimenter selected from a GUI list with 10 items							
22	I always knew the target device when the experimenter selected from a GUI list with 20 items							

Pre-trial Questionnaire Responses

1. "I use the following devices on a regular basis"

Participant	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*Mean*	*StdDev*
laptop	1	4	4	4	4	3	3	5	**3.50**	*1.195*
desktop	3	1	1	1	1	1	1	1	**1.25**	*0.707*
remote control	1	1	1	1	1	2	1	2	**1.25**	*0.463*
laser pointer	5	4	3	3	5	4	4	5	**4.25**	*0.707*
handheld	5	5	4	4	5	4	4	5	**4.50**	*0.535*

Rankings: 1 = strongly agree / 3 = neutral / 5 = strongly disagree

2. "I feel comfortable using the following devices"

Participant	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*Mean*	*StdDev*
laptop	1	3	2	1	1	1	1	2	**1.50**	*0.756*
desktop	1	2	1	1	1	1	1	1	**1.13**	*0.354*
remote control	1	1	1	1	1	1	1	2	**1.13**	*0.354*
laser pointer	3	4	2	1	3	1	1	2	**2.13**	*1.126*
handheld	5	5	4	2	4	2	1	3	**3.25**	*1.488*

Rankings: 1 = strongly agree / 3 = neutral / 5 = strongly disagree

Post-trial Questionnaire Responses (Written)

Post-trial questionnaire responses from each of the eight participants are transcribed below as numbered list items, 1, 2, 3, ... for the 1st, 2nd, 3rd, ... participants, respectively.

1a. GUI List

Advantages

1. Can see all the devices your computer can connect to
2. Got to hold pointer in same position. If I'm sitting down working its less effort to use GUI
3. All the options are right in front of the user
   Easy to select
4. You don't need to worry about line-of-sight or selecting a device because it is close to the one you want, or find the device when it is far away.
5. Don't have to move, and can see all other's that are available.
6. The list was easy to read and scroll through.
7. Alphabetical sorted (clear font used)
   Scroll bar (understandable) (pick up time quick)
   Sometimes short list (not requiring scroll bar) is good (quick)
8. Able to se list of all available machines
   Don't need to move close to machine to select it.

Disadvantages

1. Have to change windows
2. While standing It's harder to use GUI because screen is so small and I'm like trying not to fall while walking too! This is hard man!
3. Easy to make a mistake since everything is so close together.
4. You have to remember names and search through a possibly large list.
5. Its mixed with other id tags and you might click the wrong one. You have to scroll through the list.
6. None really, the pen was just easier – more user friendly.
7. Sometimes the list get too long that I have to use the scroll bar (require my hand to move to another place accurately with another skill (dray in the device)
8. Small screen in combination with pen makes it difficult to make the proper selection (easy to pick the machine above / below on list)

1b. GesturePen

Advantages

1. Speed, quicker access, don't need to change programs (i.e. windows) to select device, but just point.
2. Good for things far away cause you didn't have to try and read them. And I have bad distance eyesite.
3. Objects are far away from each other, there is clear, visible distinction.

1. No device names to remember.
   No list to search through / it might be large.
2. No scrolling, just point & click.
3. Very easy to walk over or point to the machine and just click. Point & click – I like it.
4. Fast to locate a target (like shooting gun).
   I like using button to activate an event.
5. Easy to see which machine is being selected (don't accidently select the wrong machine)

Disadvantages

1. Cumbersome that the range of the gesturePen was so short and that you needed to acknowledge the connection by such a big ready button. [in trial s/w]
2. While sitting I had to change position of pen in hand and this sucked and slowed down my puzzle game.
3. User can be disoriented because they have to gesture at things located at different places.
   Need to wait for light to indicate a tag has been selected.
4. Might select one close to the one you want.
   Line-of-sight.
   Have to locate the device physically.
5. Range appears to be limited, some required multiple clicks.
6. Sometimes the pen did not want to pick up the device right away and I had to click a few times.
7. Short range
   Sometimes targets get too close that I can't target accurate enough.
8. Need to move within ~0.5 m in order meter in order for the pen to recognize machine.

2. Easiest (GUI List or gesturePen)

1. gesturePen – The gesturePen was faster, and didn't make me switch contexts too much in comparison to the GUI list.
2. GUI List / gesturePen – mostly because it's harder to stand and point at a little screen without my elbows stabilized.
3. GUI List – When doing the puzzle pointing to GUI list is much faster. It is a simple list format so I was accustomed to it. It's also easier to switch back to your previous application (puzzle).
4. GUI List – I didn't have to walk around to the device in order to select it.
5. gesturePen – It was simply a point & click, very easy to do.
6. gesturePen – because all you had to do was point and click, not have to remember names.
7. GUI List – the pen's range is too short that you have to see if you have shot the target by moving my head to the screen (have to keep moving my head to make sure the task is done) annoying.
8. gesturePen – No mistakes in selecting the correct machine (easy to slip using the GUI)

3. Prefer (GUI List or gesturePen)

1. gesturePen – same reason as above.
2. GUI List / gesturePen – the same as above and I am lazy (future bum) and loving it.
3. GUI List – It was easiest to use.
4. gesturePen – Didn't have to look through a list (sometimes long to find a machine)
5. I could see the list and it didn't require movement, I wasn't really that distracted from what I was doing.
6. gesturePen – because it is just easier if the devices were not labelled it would be easier to point to them to add them to the network than try and figure out what they are called to add them. Highly user friendly.
7. gesturePen – If that pen has long range & the interface has a solution for missed shots problem. That would be quick and comfortable for me to shoot things instead of using the list within a very small screen. (that screen annoys me too)
8. GUI List – liked being able to see entire list of available machines. Didn't need to move close to desired machine.

4a. Prefer GUI List when

1. When there are a large number of devices in the room (i.e. would not have to hunt around)
2. While sitting on my duff (ass)
3. Choosing something within the machine you are using.
4. When there are few devices and their names are known to me.
5. If a large number of computers / palm pilots etc were on the network
6. If there were only a few device to choose from, it would be quite easy to use the list.
7. When the list is small (quick to locate)
   Targets change list moving around (1st experiment)
8. When I already know which name is associated with each machine.

4b. Prefer gesturePen when

1. Any other time except above.
2. While cruising around picking up chics and having to change interfaces this is better.
3. Interaction with another machine.
4. When there are many devices or new devices with names I don't know.
5. If the computers / palm pilots etc were easily accessible as in close range to where I was.
6. If there were many unlabelled devices it would be easier to point to one than look through a list for it.
7. When there are many targets while I'm used to the environment (targets not changing (2nd experiment)
8. When I can see the machine I want to select, but don't know its name.

Post-trial Questionnaire Responses (Likert-type)

The experimenter asked each of the participants to rank following 22 questions on a scale from 1 to 7 – 1 = strongly agree and 7 = strongly disagree. After answering each question, each participant was encouraged to explain their ranking.

Individual Participant Responses

Participant	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	***Mean***	***StdDev***
Question 1	2	1	1	2	1	1	2	1	**1.38**	*0.52*
2	1	1	1	1	1	1	1	3	**1.25**	*0.71*
3	1	1	2	1	1	1	3	1	**1.38**	*0.74*
4	1	2	1	1	1	1	3	3	**1.63**	*0.92*
5	1	1	3	1	1	1	2	1	**1.38**	*0.74*
6	1	1	1	1	1	1	1	2	**1.13**	*0.35*
7	1	1	1	2	1	1	2	3	**1.50**	*0.76*
8	2	2	2	2	2	2	6	3	**2.63**	*1.41*
9	1	1	2	2	1	1	1	2	**1.38**	*0.52*
10	5	2	1	2	2	2	5	2	**2.63**	*1.51*
11	3	1	2	2	1	2	1	3	**1.88**	*0.84*
12	2	1	1	1	1	1	4	2	**1.63**	*1.06*
13	1	1	3	2	1	2	1	3	**1.75**	*0.89*
14	2	1	1	1	1	1	3	1	**1.38**	*0.74*
15	7	6	4	4	6	6	2	5	**2.00**	*1.60*
16	3	7	5	3	6	6	4	4	**2.25**	*1.49*
17	3	7	4	3	6	6	3	3	**2.63**	*1.69*
18	1	7	4	3	5	5	3	3	**3.13**	*1.81*
19	1	1	1	2	1	4	5	2	**2.13**	*1.55*
20	2	2	3	2	2	1	4	6	**2.75**	*1.58*
21	3	2	4	2	2	1	6	6	**3.25**	*1.91*
22	3	2	5	2	2	2	7	6	**3.63**	*2.07*

Distribution of Responses

The frequency of rankings for each post-trial inquiry question are shown below. Individual questions are colour coded to correspond to the summary at the bottom of the page. Green indicates gesturePen questions and orange indicates graphical list questions. A ranking of 1 = Strongly Agree and a ranking of 7 = Strongly Disagree.

	Questions Rankings:	1	2	3	4	5	6	7	Mean	Std Dev
1	I understand how to use the gesturePen	5	3	0	0	0	0	0	**1.38**	*0.52*
2	I understand how to use the GUI list	7	1	0	0	0	0	0	**1.25**	*0.71*
3	I can use the gesturePen easily	6	1	1	0	0	0	0	**1.38**	*0.74*
4	I can use the GUI list easily	5	1	2	0	0	0	0	**1.63**	*0.92*
5	I can select a computing device easily using the gesturePen	6	1	1	0	0	0	0	**1.38**	*0.74*
6	I can select a computing device easily using a GUI list with 5 items	7	1	0	0	0	0	0	**1.13**	*0.35*
7	I can select a computing device easily using a GUI list with 10 items	5	2	1	0	0	0	0	**1.50**	*0.76*
8	I can select a computing device easily using a GUI list with 20 items	0	6	1	0	0	1	0	**2.63**	*1.41*
9	I like using the gesturePen	5	3	0	0	0	0	0	**1.38**	*0.52*
10	I like using the GUI list	1	5	0	0	2	0	0	**2.63**	*1.51*
11	I would feel comfortable using the gesturePen in public	3	3	2	0	0	0	0	**1.88**	*0.83*
12	I would feel comfortable using the GUI list in public	5	2	0	1	0	0	0	**1.63**	*1.06*
13	I would feel comfortable using the trial handheld computer (Sparky) in public	4	2	2	0	0	0	0	**1.75**	*0.89*
14	I can move freely using the trial handheld computer (Sparky)	6	1	1	0	0	0	0	**1.38**	*0.74*
15	I need almost all my attention to use the gesturePen	0	1	0	2	1	3	1	**5.00**	*5.40*
16	I need almost all my attention to use the GUI list with 5 items	0	0	2	2	1	2	1	**4.75**	*5.51*
17	I need almost all my attention to use the GUI list with 10 items	0	0	4	1	0	2	1	**4.38**	*5.31*
18	I need almost all my attention to use the GUI list with 20 items	1	0	3	1	2	0	1	**3.88**	*5.19*
19	I always knew the target device when the experimenter used the gesturePen	4	2	0	1	1	0	0	**2.13**	*1.55*
20	I always knew the target device when the experimenter selected a 5 item list	1	4	1	1	0	1	0	**2.75**	*1.58*
21	I always knew the target device when the experimenter selected a 10 item list	1	3	1	1	0	2	0	**3.25**	*1.91*
22	I always knew the target device when the experimenter selected a 20 item list	0	4	1	0	1	1	1	**3.63**	*2.07*

Summary	Questions		
gesturePen	1, 3, 9, 11	**1.50**	*0.67*
List (general)	2, 4, 10, 12	**1.78**	*1.16*
gesturePen	5, 15, 19	**2.17**	*1.46*
List (5 items)	6, 16, 20	**2.38**	*1.53*
List (10 items)	7, 17, 21	**2.79**	*1.74*
List (20 items)	8, 18, 22	**3.46**	*1.82*

Appendix C:
Running the User Study Software

The trial administrator software for the user study can be run with the command:

```
java -classpath comm.jar;. <executable> <input filename>
<tag map filename> <output filename> <operating system>
```

An example command is given below:

```
java -classpath comm.jar;. commtest.commtest samplein.txt
tagmap.txt sampleout.txt windows
```

`<executable>` is the Java executable name (i.e. commtest.commtest)

`<input filename>` is the name of an ASCII text file describing the graphical list items to be displayed to participants during a user study. Each line in the input file should have the format: `["add" | "match" | "remove" | "end"] <target string>`. Valid example lines include `add plum.sfu.ca` and `end`. `add` and `remove` will add and remove, respectively, the specified target string to the graphical list that the participant interacts with during the user study. `match` will cause the trial administrator software to wait and log the participant's selections until the target string is selected by tapping the graphical list *or* activating gesturePen towards a tag. `end` should be the last line in the text file. `<target string>` is a text string representing a device name.

`<tag map filename>` is the name of an ASCII text file that re-maps the text string sent from a tag to a different text string. This enables the same tag to be quickly changed between two or more 'host' computing devices. Each line in the input file should have the format: `[<tag string> | "end"] <target`

`string>`. Valid example lines include `tag1 plum.sfu.ca` and end. `<tag string>` is the text string sent by the custom tag, and `<target string>` is the text string that the string will used in the software instead of `<tag string>`.

<output filename> is the name of an ASCII text file where the participant logs will be recorded. If the file already exists, output will be appended to the existing file. Each line in the input file will have the format: `<line> ["match:" | "miss:"] <selected string> <target string> <start time> <end time>`. An example line of output is `16 miss: carrot.sfu.ca starfruit.sfu.ca 1008262313346 1008262436846`. `<line>` is the line number in the input file that corresponds to the data on this output line. `"match:"` and `"miss:"` refer to whether the participant selected the correct or incorrect target string, respectively. `<selected string>` and `<target string>` represent the name of the computing device the participant selected and the name of the computing device that the participant was supposed to select, respectively (These should match if the participant selected the correct target device). <start time> and <end time> represent the times in milliseconds since 0:00 1 January 1970 UTC when the participant pressed the *ready* button on the user interface to begin the current matching task, and when the participant selected the `<selected string>` using graphical list or the gesturePen methods.