That one there! Pointing to establish device identity

Colin Swindells [*] *Kori M. Inkpen* [†] *John C. Dill* [*] *Melanie Tory* [*]

[*] Centre for Systems Science
Simon Fraser University
Burnaby, BC V5A 1S6, Canada
Tel: +1-604-291-4371
E-mail: swindell@cs.ubc.ca, {dill, mktory}@cs.sfu.ca

[†] Faculty of Computer Science
Dalhousie University
Halifax, NS B3H 1W5, Canada
Tel: +1-902-494-2093
E-mail: inkpen@cs.dal.ca

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. This paper describes a 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. As ubiquitous computing environments become more pervasive, people will rapidly switch their focus between many computing devices. The results of our work demonstrate that our gesturePen method can improve the user experience in ubiquitous environments by facilitating significantly faster interactions between computing devices.

KEYWORDS: Pointing, gesturing, gesturePen, identification, handheld computer, PDA, infrared (IR) tag.

1 MOTIVATION

As the use of short-range wireless technologies such as IEEE 802.11 and Bluetooth grows, new possibilities for ubiquitous computing are continually 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.

With advancements in ad-hoc mobile networking, environments are becoming increasingly populated with computing devices that automatically identify each other when they come in proximity of one another. Automatic identification 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, these advancements in computer-computer identification may cause human-computer identification to be more difficult.

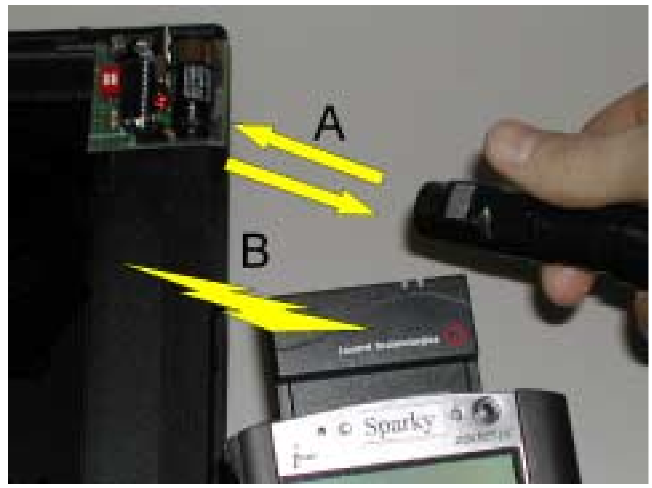

Figure 1: Line-of-sight identification with tag and gesturePen. A: The user points to a tag with a special pen to uniquely identify a laptop (e.g. its network address). B: Information can then be transferred over a wireless network to the laptop

When we interact in short-range wireless networks, we can often physically see the device we want to interact with. 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 [1], the ability to gesture in conjunction with a graphical interface provides a more natural interaction than typing symbols.

This paper describes our implementation and user evaluation of the gesturePen, a line-of-sight, tag-based identification system (see Figure 1). This system facilitates information sharing in mobile computing environments by allowing users to identify devices by simply pointing at

UIST'02, October 27-30, 2002, Paris, FRANCE.

them. We begin with an example scenario to present existing methods of data transfer between devices and briefly illustrate how the gesturePen can overcome problems with these methods. Next, we describe previously published research in the area and how it relates to our work. Following this, we present the implementation details of our system and the user study we conducted to gain feedback on the interaction style. We then discuss knowledge gained from this study and 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, we present directions for future work in this area.

2 Example scenario: Sharing documents at an airport

A colleague, Marie, and you meet up while waiting for a flight at the airport. Having recently returned from a conference in Paris, Marie wants to give you a copy of the paper she presented at the conference. The paper is stored on her handheld computer and connectivity is available in the airport through a wireless network.

There are several ways that Marie could transfer the document from her handheld to your laptop. Strengths and limitations of two common techniques (infrared and wireless networking) are presented below.

2.1 Infrared communication

Using infrared communication, Marie could select the document on her handheld, then point to your laptop's infrared port, and 'beam' the data (i.e. send using line-of-sight IR communications). 'Beaming' is advantageous because the target device is implicitly identified by pointing towards it. However, beaming is problematic because 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. In addition, data can only be transferred between two devices at any given time.

2.2 Wireless network communication

Using wireless networking, Marie could transfer the document by asking for your laptop's host name and domain name (e.g. sparky.sfu.ca), then selecting these names from a graphical list of currently accessible computing devices. After selecting your device as the target (from a list of potentially hundreds of devices), she could then copy the document over the wireless network. Separating the identification and data transfer tasks resolves the problem of needing to maintain a line-of-sight connection, as well as the limitation of being able to transfer to only one device at a time. However, in most network environments, users perform the identification task by selecting a target device from a graphical user interface. 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 will 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 [14] found that users took more time to find information when its spatial location varied than when the information was in a consistent location.

2.3 Our Solution: Identification via pointing

Our solution utilizes both infrared and wireless communication. We developed the gesturePen, a line-of-sight, tag-based identification system for identifying target device(s) (see Figure 1). Identification through a pointing gesture can be easier and more natural than selecting the device's name from a list or entering its IP address into a text box. We also maintain the separation of pointing and data transfer tasks, so that devices can be identified through a pointing gesture, and then data can be subsequently transferred over the wireless network.

This solution combines the benefits of both infrared and wireless network communication and helps overcome the weaknesses of these approaches. Using a line-of-sight technology such as infrared for the identification, users can uniquely identify a device through a gesture and avoid searching through potentially long GUI lists. Then, once the device has been identified, the transfer occurs seamlessly over the wireless network. Users are free to move about and work on other tasks while data is being transferred. Additionally, data can be transferred to multiple devices simultaneously.

3 RELATED RESEARCH

A number of tag-based systems have been presented in previous work. We compare our gesturePen method with similar tag based information transfer mechanisms and technologies used to promote social interactions.

3.1 Information transfer mechanisms

The notion of transferring a small identifier (e.g. a URL) to a device which then triggers additional data to be retrieved is not a new concept. HP Lab's E-squirt technology [3] is a commercial endeavor that uses small transmitters to 'squirt' URLs into nearby devices. The devices then display the appropriate multimedia content retrieved from the Internet. Want *et al.* [17] 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; it 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.

A recent study utilizing E-squirt technology supports the difficulties hypothesized above [6]. For example, usability concerns with the beacons were noted when using E-squirt technology for identifying museum exhibits. Specifically, "the most common case involved accidentally reading a beacon soon after reading deliberately." [6, p.18] Fleck *et al.* suggested that "requiring users to press a button when picking up a beacon might help, but our PDAs didn't have an ergonomically suitable button that users could find and press accurately." [6, p.18] Our gesturePen method addresses these concerns by utilizing a pen form factor with an ergonomically placed button, and two-way handshaking to avoid accidentally reading a tag.

Significant research achievements that addresses the seamlessness of transferring information between computing devices using a physical surrogate have been explored in the Pick-and-Drop [11] and mediaBlocks [15] projects. 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, information can be 'dropped' to a device that is out-of-reach, which is not possible with the Pick-and-Drop method.

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. 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 significantly from a mediaBlock. Instead of perceiving information storage, the user 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.

Instead of using active tag technology like E-squirt or gesturePen, Ljungstrand and Holmquist [7] 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.

3.2 Encouraging interaction between users

The Meme Tags project [2] 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 a couple 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 light-weight computing devices. However, data is transferred over a wireless network using our gesturePen method – not over an infrared link. We also believe our gesturePen form affords more controlled, conscious identification of target devices because Myers *et al.'s* [8] laser pointer research showed that a pen form was the fastest, most accurate way to single-handedly identify a target.

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 [16].

While this technology is similar to ours, gesturePen tags are visible so users can perceive their existence and interact with them.

4 DESIGN AND IMPLEMENTATION

Our goal was to develop a custom stylus (the gesturePen) and tags as stand-alone computing devices to create a flexible, generic solution. The form-factor of a pen was chosen because pointing with a pen is a fairly natural gesture. Recent work by Myers *et al.* [8] demonstrated that pointing with a pen-type device was more accurate than other devices, such as a PDA, when interacting with one hand. The stand-alone pen also provides a similar interaction style for larger computing devices, such as laptops and desktops, that are obviously difficult to pick up and point. A two-way line-of-sight communications link was chosen to facilitate interaction with devices in dense computing environments from distances of 1 m or more. Fewer, if any, undesired tags will be 'pinged' using directional communications (e.g. infrared) compared to omni-directional communications (e.g. radio frequency).

To identify a device with a pointing gesture, we developed a custom stylus – the gesturePen (initially described in [13]). We added an IrDA (InfraRed Data Association) compliant infrared transceiver into the gesturePen and developed tags that can be fixed to active and passive devices such as computers and walls, and communicate with the stylus. The tags are stand-alone devices based on tags developed by Poor [9]. The gesturePen prototype can be used by any computing device with an RS-232 serial communications port, such as a handheld or laptop. The gesturePen needs only a serial communications link between itself and its 'host'. This solution maximizes device, platform, and application independence. Furthermore, tags can be put on inanimate objects that are

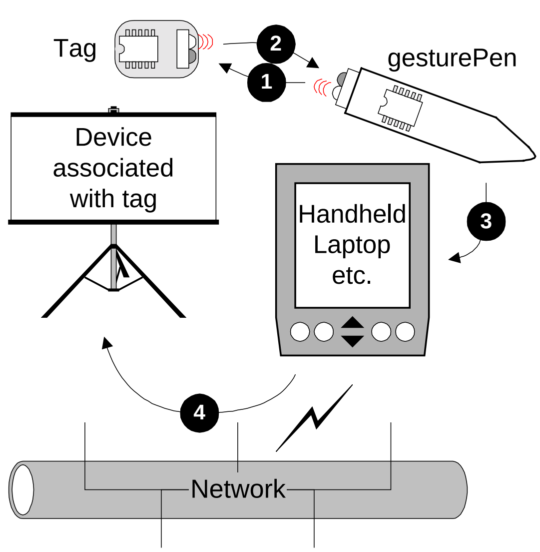

Figure 2: Communications flow between the gesturePen, a tag, and the tag's associated device.

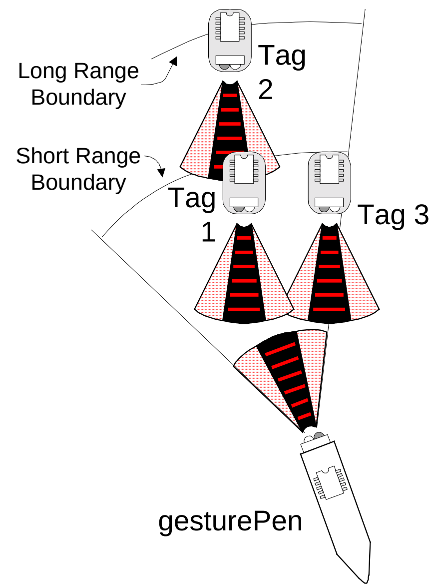

Figure 3: Effects of different transceiver ranges and viewing angles. A narrow viewing angle is illustrated with a darkly shaded beam, while a wide viewing angle also includes the lightly shaded area. Short and long ranges are illustrated by the thin arcs near and far from the gesturePen, respectively

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.

4.1 Communications flow using the gesturePen

Figure 2 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 target device) 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 serial communications link to the device attached to the gesturePen (e.g. a handheld).
4. Selected information is transferred over a network to the device associated with the tag. For example, a document is transferred from a handheld to a wall display.

4.2 Transceiver range and angle considerations

Range and angle of the infrared beam are important features affecting the usability of a pointing device 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. 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. We use a viewing angle of 30° (i.e. a viewing angle of 15° from all directions of the transceiver's central line-of-sight).

Figure 3 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 3), 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.

4.3 Hardware development

Figure 4 shows a component diagram of our gesturePen and tag. The tag has three main components: an infrared transceiver, voltage regulator, and microcontroller. The HSDL 3000 infrared transceiver uses an IrDA compliant wavelength of 880 nm, and range of 1.5+ metres. The gesturePen is simply a tag with an RS-232 driver microchip and slightly different firmware. To provide user feedback, a red light emitting diode (LED) was added to the tags which blinks when the tag receives a message.

Most current handhelds have infrared transceivers that function up to ~30 cm. To have link distances over 1 m, we built the tags and stylus using the components shown in Figure 4. We integrated the stylus parts into an old whiteboard marker. The IR transceiver was mounted on the end of the stylus to facilitate pointing like a laser pointer or remote control. A standard iPAQ stylus was placed along the marker's core, so the gesturePen can be used both as an IR identification device and as an ordinary stylus.

We added a button for users to press when pointing the gesturePen towards a target tag. We also added an RS-232 driver chip to create a serial communications link between the iPAQ and gesturePen microprocessors. A Bluetooth enabled gesturePen capable of wirelessly communicating with its host device would have been preferable to a RS-232 serial communications link, however such technology was immature when we constructed our prototype.

4.4 Software integration

Software for a handheld computer was developed to retrieve information from the gesturePen, and then relay this information to a desired application. Specifically, the Java 2 Standard Edition (J2SE) and javax.comm serial communications extension was used. This enables our current implementation to run on any computer with any

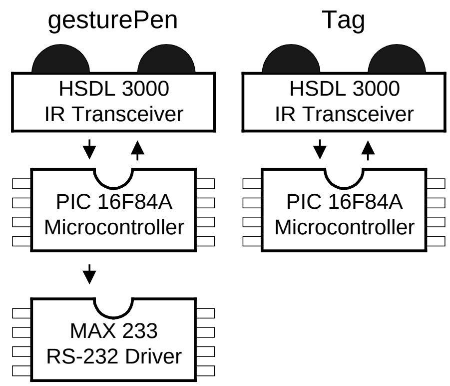

Figure 4: Main hardware components and communication paths for the gesturePen and tag. (Power management electronics such as voltage regulators were used, but are not shown)

operating system that has RS-232 serial port communications capability, and supports the J2SE virtual machine.

The gesturePen interaction style was incorporated into the software from one of our related projects, WindowSpaces [12]. 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 – using Java's JINI technology – so they could be 'pasted' to the laptop and wall display.

5 USER STUDY

Our user study compares the "naturalness" of identifying a device using a pointing gesture compared to using a dominant GUI technique. Although technologies similar to the gesturePen method have been developed (such as E-squirt [3] and Pick-and-Drop [11]), these methods have not been compared to traditional GUI techniques or analyzed using standard qualitative methodologies. Thus, our user study compares the speed and accuracy of a pointing method (our gesturePen method) to a dominant GUI method (list selection). Using these results plus formal qualitative analysis, we then analyze the environments and situations that are best suited to each method.

We obtained qualitative measures based on the social computing framework that Dryer *et al.* developed for mobile computing systems [5]. The user study questions were based on usability guidelines suggested by Dryer *et al.* [5] and conducted according to structured observation techniques suggested by Dray [4]. In addition to our qualitative data, we compared the performance of our gesturePen prototype to a graphical list method by timing users as they selected devices using each method.

5.1 Study design

Figures 5 and 6 show our two experimental configurations.

5.1.1 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:

- Reading an IP address label on the tag, and selecting the appropriate IP address from a graphical list on their handheld computer
- Pointing towards the appropriate tag with the gesturePen and clicking the button on the pen to select the tag

After selecting a tag, the participant returned to the puzzle while the experimenter chose the next two target tags (from a set of 20). Each participant performed 12 tag selections. For consistency, the screen was set up with the puzzle filling the top 50 % of the screen, while the trial 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.

5.1.2 Phase 2: Mobile environments task

The second phase of this study was to explore the effectiveness of the gesturePen for device identification in mobile ad-hoc environments. As shown in Figure 6, the participants were required to select target computing devices as if they had walked into a room and needed to transfer information to 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 labeled 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). Times were measured from the time the user was told which target to select until they correctly identified the device.

5.1.3 Steps to prevent experimental biases

The 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.

5.1.4 Qualitative data collection

Qualitative data related to user interaction issues associated with our gesturePen method was gathered 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

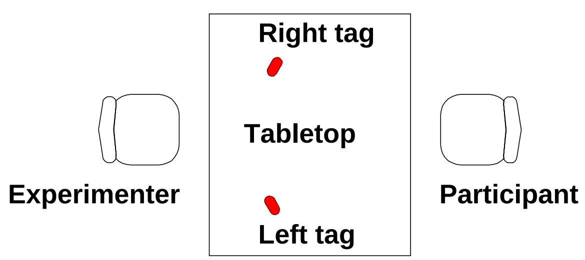

Figure 5: Experimental setup for the cognitive load task. The participant played with a jigsaw puzzle on a handheld computer, and then was interrupted by the experimenter to select a tag with a pointing gesture or from a graphical list on their handheld.

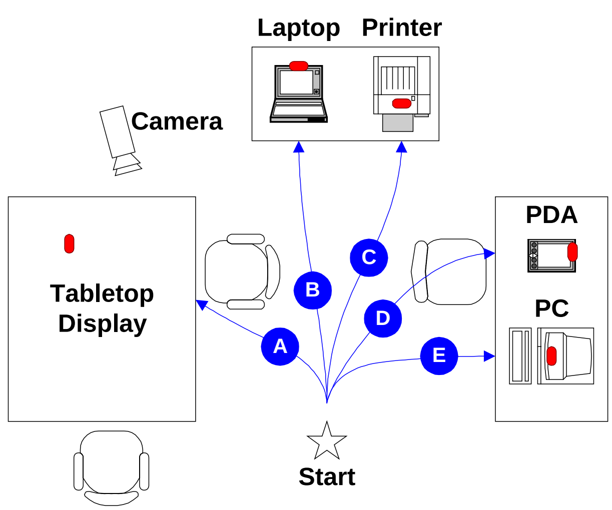

Figure 6: Experimental setup to test identification in new, ad-hoc environments. The participant stood over a star on the floor, and then selected one of 5 computing devices by either gesturing to the device with the gesturePen or reading a label on the device and selecting an IP address from a graphical list on their handheld computer.

strongly agree to *strongly disagree*. The questions addressed the usability criteria in Table 2 adapted from Dryer *et al.* [5]. Participants were also encouraged by the experimenter to explain their rankings. The experimenter asked questions according to established structured observation techniques [4], and made hand-written notes and audio/video records.

5.2 Results

The results of this study revealed that both the gesturePen and graphical list methods were easy to understand and use. There were however important differences which impact the effectiveness of each strategy for particular environments. We did not notice or analyze differences based on groups such as culture, gender, or religion. Our qualitative results are organized according to key qualitative measures suggested by Dryer *et al.* [5]. All rankings given in this section refer to a 7 point Likert scale where 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.

Table 2: Qualitative measures used in our study [5]

Measure	***Descriptive phrase***
Accessibility	Easy to understand?
Familiarity	Familiar and appropriate form?
Input sharing	Enables nonusers to share input?
Output sharing	Enables nonusers to share output?
Relevance	Appears useful to user?
Appeal	Comfortable? Attractive?
Disruption	Disrupts user's natural behaviours?
User distraction	Low cognitive load on user?
Identification	Inclusive technology for all users?
Pervasiveness	Mobile and convenient?

5.2.1 Qualitative measures

The qualitative results and analysis from these inquiries are discussed below. Mean and standard deviation (SD) rankings for the questions used during the structured inquiry are shown in Tables 3 and 4.

Accessibility was high for both the graphical list and gesturePen methods. Participants ranked the ease of use an average of 1.4 for the list and 1.6 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, 1.5, and 2.6, respectively; whereas, they ranked their ability to select a device using the gesturePen an average of 1.4. The main difficulty with the gesturePen was its range. At times, participants wanted to select devices from across the room (i.e. 6 – 10 m) instead of the 1.5 m range of our prototype.

Familiarity was also high for both the list and gesturePen. Participants ranked their ability to understand how to use the devices as 1.4 for the list and 1.3 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. This could be attributed to the fact that participants tended to relate the gesturePen to a remote control and all participants reported that they often used a remote control. However this metaphor was also misleading because most participants expected the gesturePen to have a longer range like a remote control.

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 favoured the gesturePen. For example, participants had difficulty observing the screen of another person who was using a graphical list on a handheld next them. Conversely, participants could usually deduce a user was pointing by looking at their 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 device selected by another person an average of 2.8, 3.3, and 3.6 for graphical lists of 5, 10, and 20 items, respectively, whereas they ranked the gesturePen 2.1.

Relevance appeared to vary widely depending on the environment. Participants believed the graphical list would be better in more static computing environments 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 public places or someone else's office.

Appeal ratings were an average of 2.6 for the list and 1.4 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.

Disruption was similar for the list and gesturePen – ranked 1.8 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 would feel uncomfortable if it appeared that another person was pointing at them with the gesturePen.

User Distraction was evaluated by asking participants how much attention they needed to select using the graphical list or gesturePen. Participants reported needing more attention for the graphical lists. Rankings were 2.2, 2.6, and 3.1 for graphical lists of 5, 10, and 20 items, respectively, whereas the gesturePen was ranked 2.0. Scrolling and switching windows were the most distracting elements of the graphical list method. During the cognitive load task, we noticed that participants focused less on the target tag (to read a tag’s name) when they were interrupted from their puzzle and asked to point 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.

Identification was perceived to be similar for the graphical list and gesturePen. Both were perceived as inclusive of most communities (e.g. both can be used by left- or right-handed people). Although all participants had good eye-sight, 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.

Pervasiveness was mainly dependent on the ‘host’ computer such as the handheld, not the identification method. Participants gave an average ranking of 1.4 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 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. Several participants had 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 for more details).

Table 3: Qualitative inquiry rankings (1 = most positive ranking and 7 = least positive ranking)

	Questions	*Mean*	*SD*
1	I understand how to use the gesturePen	**1.38**	*0.52*
2	I understand how to use the GUI list	**1.25**	*0.71*
3	I can use the gesturePen easily	**1.38**	*0.74*
4	I can use the GUI list easily	**1.63**	*0.92*
5	I can select a computing device easily using the gesturePen	**1.38**	*0.74*
6	I can select a computing device easily using a GUI list with 5 items	**1.13**	*0.35*
7	… with 10 items	**1.50**	*0.76*
8	… with 20 items	**2.63**	*1.41*
9	I like using the gesturePen	**1.38**	*0.52*
10	I like using the GUI list	**2.63**	*1.51*
11	I would feel comfortable using the gesturePen in public	**1.88**	*0.84*
12	I would feel comfortable using the GUI list in public	**1.63**	*1.06*
13	I would feel comfortable using the trial handheld computer (Sparky) in public	**1.75**	*0.89*
14	I can move freely using the trial handheld computer (Sparky)	**1.38**	*0.74*
15	I need almost all my attention to use the gesturePen	**5.00**	*1.60*
16	I need almost all my attention to use the GUI list with 5 items	**4.75**	*1.49*
17	… with 10 items	**4.38**	*1.69*
18	… with 20 items	**3.88**	*1.81*
19	I always knew the target device when the experimenter used the gesturePen	**2.13**	*1.55*
20	I always knew the target device when the experimenter selected from a GUI list with 5 items	**2.75**	*1.58*
21	… with 10 items	**3.25**	*1.91*
22	… with 20 items	**3.63**	*2.07*

Table 4: Summary of inquiry ranking averages (1 = most positive ranking and 7 = least positive ranking)

	Graphical List	*gesturePen*
Usability (Questions 1-4, 9-12)	1.78	1.50
Performance (Questions 5-8, 15-22)	2.88 / 3.04 / 3.38	2.83

5.2.2 Identification times

Tables 5 and 6 show the mean and standard deviation times participants took to identify a device within the cognitive load and mobile environments tasks, respectively. One participant's phase 1 data was not collected due to software difficulties.

Table 5 shows that participants were able to identify a device significantly faster using the gesturePen than a graphical list within the environment illustrated in Figure 5. 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 graphical list, users would need to release some or all focus from their current task to show a list of possible devices.

Table 6 shows a wide variance in time to identify a device using the gesturePen within the environment illustrated in Figure 6. 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 30º. 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 pointed towards a tag and pressed the pen's button.

Table 5: Phase 1 – Cognitive load task times

	Graphical List	*gesturePen*
Mean (s)	4.4	2.9
SD (s)	1.8	1.8
Significance	$F(1,6)=19.48$, $p=0.005$	

Table 6: Phase 2 – Mobile environments task times

	Graphical List	*gesturePen*
Mean (s)	3.8	4.4
SD (s)	1.8	3.9
Significance	$F(1,7)=0.998$, $p=0.351$	

5.2.3 Slips during identification

Any misidentification of the correct tag because of an execution mistake was labeled as a 'slip'. Different types of execution mistakes (or 'slips') were found for the graphical list and gesturePen methods. As shown in Table 7, no participants selected the wrong tag during phase 1 (the cognitive load task). A few participants did, however, accidentally identify the wrong device during phase 2 (the mobile environments task). We believe comments from the post-trial inquiry explain why participants made these slips. Using the graphical list, the seven misidentified devices all had a device name 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. This increased accuracy would likely take more time and/or require a higher cognitive load than gesturing within a wider target range.

Table 7: Total number of misidentified devices

	Graphical List	*gesturePen*
Cognitive Load Task	0 *(0 %)*	0 *(0 %)*
Mobile Environ. Task	7 (2.9 %)	3 *(1.3 %)*

6 FUTURE WORK AND CONCLUSIONS

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 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. Also, a dial on the gesturePen case to adjust the pen's range and/or beam angle was suggested.

Redström *et al.* [10] describe how we rapidly form and break different communities of devices and people in our daily interactions. 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. Thus, portable computing technologies, such as the gesturePen, must be fast and flexible in their ability to change their range between small and large transmission distances and angles. Additionally, because the content can change rapidly during a meeting – especially an ad-hoc meeting – devices must be able to adapt quickly to the changing needs of 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.

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. This approach enables users to focus on their tasks with minimal distraction from other devices in their environment.

Even though no participants had used our gesturePen before, they were comfortable using the gesturePen and found our identification technique very easy to learn. Participants suggested a graphical list would be more useful in typical 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.

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 visible on the network.

ACKNOWLEDGMENTS

We thank the EDGE Lab members, Dr. Lyn Bartram, Michael Blackstock, Dr. Jason Harrison, Prof. Kellogg Booth, and Prof. Shahram Payandeh for their comments and suggestions. We are also grateful for the financial support from Colligo Networks, Inc., the BC Advanced Systems Institute, the Natural Sciences and Engineering Research Council, and Simon Fraser University.

REFERENCES

1. Bolt, R. (1980) Put-That-There: Voice and Gesture at the Graphics Interface. In *Proc. of SIGGRAPH '80*.
2. 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 *Proc. of CSCW 1998*.
3. Bucholtz, C. (2000) *Software Behind HP Labs' CoolTown*. Available at: http://www.interex.org/hpworldnews/hpw005/03news.html
4. Dray, S. (2001) *Structured observation*. Tutorial presented at CHI 2001.
5. 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.
6. Fleck, M., Frid, M., Kindberg, T., O'Brien-Strain, E., Rajani, R., and Spasojevic, M. (2002) From Informing to Remembering: Ubiquitous Systems in Interactive Museums. In *IEEE Pervasive Computing*.
7. Ljungstrand, P. and Holmquist, L.E. (1999) WebStickers: Using Physical Objects as WWW Bookmarks. In *Extended Abstracts of CHI 1999*.
8. 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 *Proc. of CHI 2002*.
9. Poor, R.D. (1999) *The iRX 2.1*. Available at: http://www.media.mit.edu/~r/projects/picsem/irx2/.
10. Redström, J., Dahlberg, P., Ljungstrand, P., Holmquist, L.E. (1999). Designing for Local Interaction. In *Proc. of MANSE 1999*.
11. Rekimoto, J. (1997) Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments. In *Proc. of UIST 1997*.
12. Shu, K., Swindells, C., Chai, D., and Inkpen, K.M. (2002) WindowSpaces to share our digital media. *Tech Report CMPT2002-04*. Simon Fraser University. Available at: http://www.cs.sfu.ca/
13. Swindells, C. and Inkpen, K.M. (2002) What is that? Gesturing to determine device identity. Poster Presentation in *Extended Abstracts of CHI 2002*.
14. Teitelbaum, R.C., and Granda, R.E. (1983) The effects of positional constancy on searching menus for information. In *Proc. of CHI 1983*.
15. Ullmer, B. and Ishii, H. (1999) mediaBlocks: Tangible interfaces for online media. In *Proc. of CHI 1999*.
16. Want, R. and Borriello, G. (2000) Survey on Information Appliances. *IEEE Computer Graphics and Applications*, May/June 2000.
17. Want, R., Fishkin, K.P., Gujar, A., and Harrison, B.L. (1999) Bridging Physical and Virtual Worlds with Electronic Tags. In *Proc. of CHI 1999*.