APPLYING FITTS’ LAW IN A MOBILE AND PERVASIVE COMPUTING ENVIRONMENT

Gareth Batten
BSc in Mathematics and Computing
2004
APPLYING FITTS’ LAW IN A MOBILE AND PERVERSIVE COMPUTING ENVIRONMENT

Submitted by Gareth Batten

COPYRIGHT

Attention is drawn to the fact that copyright of this thesis rests with its author. This copy of the thesis has been supplied on condition that anyone who consults it is understood to recognise that its copyright rests with its author and no information derived from it may be published without the prior written consent of the author.

DECLARATION

This dissertation is submitted to the University of Bath in accordance with the requirements of the degree of Batchelor of Science in the Department of Computer Science. No portion of the work in this dissertation has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifically acknowledged, it is the work of the author.

Signed . . . . . . . . . . . . . . . . . . . . . . .

This thesis may be made available for consultation within the University library and may be photocopied or lent to other libraries for the purposes of consultation.

Signed . . . . . . . . . . . . . . . . . . . . . . .
Abstract

The computing environment of the future will be mobile and pervasive, involving a paradigm shift away from the desktop. Inspired by the beginnings of this shift, from desktop to mobile and pervasive computing, this project set out to look at whether mobile and pervasive computers were different to desktop computers. Specifically, its original aim was to investigate the practice of simply scaling the desktop interface up or down in order to implement it on large and small screen computers respectively. This investigation gave rise to the uncovering of a possible flaw in Fitts’ Law, the model intended for use in the comparison between mobile, pervasive and desktop computers. Following these two lines of investigation, it was found that small and large screens are significantly different from the medium, desktop screen and that Fitts’ Law is scale independent while the pointing task it models is not. This leads us to the conclusion that the Windows graphical user interface should not be shrunk to fit on to small screen displays or stretched to fill a large screen. We also conclude that although Fitts’ law has served as an accurate model in Human Computer Interaction, it has been faulty since its first application in the area. In a mobile and pervasive environment the problem of scale independence becomes more apparent and greatly reduces the predictive and descriptive powers of Fitts’ Law.
Acknowledgements

Thanks to the Lord Jesus Christ, who does not treat us as our sins deserve. He alone is my rock and my salvation.

Many thanks to my supervisor Eamonn O’Neill for his enthusiasm and direction.
## Contents

1 Introduction  

2 Literature Review  

2.1 The Changing Computing Environment  
2.1.1 Current Situation  
2.1.2 Mobile Computing  
2.1.3 Pervasive computing  

2.2 A Short History  

2.3 Interaction  
2.3.1 Direct manipulation  

2.4 Interface  

2.5 I/O  
2.5.1 Input Device  
2.5.2 Screen Size  

2.6 Summary  

2.7 Fitts’ Law  


2.7.1 Introduction .................................................. 19
2.7.2 Beginnings ..................................................... 19
2.7.3 Refining Fitts’ Law in HCI .............................. 20
2.7.4 Using Fitts’ Law To Make Comparisons ................. 23
2.7.5 Problem with Fitts’ Law .................................. 24
2.7.6 Factors to Consider When Designing a Fitts’ Task Experiment . 25

2.8 Summary ......................................................... 27

3 Experimental hypotheses .......................... 28

3.1 Screen Size ...................................................... 28
3.2 Scale Independency ........................................ 28

4 Method ............................................................ 30

4.1 Design ........................................................... 30
4.2 Pilot .............................................................. 31
4.3 Participants .................................................... 31
4.4 Apparatus ....................................................... 31
4.5 Procedure ...................................................... 32

5 Results ......................................................... 34

5.1 Movement Time Analysis ................................. 34
5.2 Error Analysis ................................................ 35
5.2.1 Undershoot Errors .................................... 35
5.2.2 Overshoot Errors ................................. 36
5.2.3 Overall Errors ................................. 36
5.3 Overshoot Analysis ................................. 37
5.4 Fit of the Model ................................. 37

6 Discussion 39

6.1 Screen Size ........................................ 39
6.2 Scale Independence / Movement Scale ................. 41
6.3 Movement Difficulty / ID ......................... 43
6.4 Methodology ........................................ 43
6.5 Fitts’ Law ........................................... 44
6.6 A Seamless Environment ........................... 45

7 Conclusions 47

References 49

A Allocation of Participants 55

B Selected participant comments 56

C Source Code 58

D Raw Data 59

D.1 Original Data Files ................................. 59
D.2 Collated Raw Data ............................................. 61

E Compiled Data for Analysis 62

F SPSS Outputs 64
List of Figures

2.1 Example of a WIMP GUI on a desktop computer . . . . . . . . . . . . . . 4
2.2 The HP iPAQ . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
2.3 The Windows Start Menu on an iPAQ . . . . . . . . . . . . . . . . . . . . 13
2.4 Fitts’ original experiment . . . . . . . . . . . . . . . . . . . . . . . . . . 20
2.5 Fitts’ pointing task in HCI . . . . . . . . . . . . . . . . . . . . . . . . . 21
2.6 Illustration of Movement Scale and Difficulty . . . . . . . . . . . . . . . 27
4.1 Screen shot of experiment . . . . . . . . . . . . . . . . . . . . . . . . . . 32
4.2 Screen shot of experiment (with error buttons) . . . . . . . . . . . . . . 33
6.1 Graphs of Movement Time and Overshoots against Screen . . . . . . . 40
6.2 Graphs of Undershoot and Overshoot errors against Screen . . . . . . . 41
6.3 Graphs of Movement Time and Errors against Amplitude . . . . . . . . 42
6.4 Graphs of Movement Time and Errors against Index Of Difficulty . . . . 44
6.5 Fitts’ Law in 2D . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 45
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Target Sizes (px)</td>
<td>31</td>
</tr>
<tr>
<td>5.1</td>
<td>Mean movement times</td>
<td>35</td>
</tr>
<tr>
<td>5.2</td>
<td>Regression Analysis</td>
<td>37</td>
</tr>
<tr>
<td>A.1</td>
<td>Allocation of participants to counterbalanced screen ordering</td>
<td>55</td>
</tr>
<tr>
<td>E.1</td>
<td>Adjusted data for participant 30</td>
<td>63</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

The 17” monitor has become the standard for desktop computers and people are accustomed to using them. Most people use a mouse to interact with the information displayed on them, and are continually pointing at and selecting objects on them. There is currently a lot of hype about mobile and pervasive computing, the advantages it will bring and the benefits of much smaller and much larger screens. However the current approach to these new devices is to shrink or expand desktop ideas on to the small and large screens. Input devices are also changing. The stylus is itself becoming ubiquitous in the world of mobile computing. While in the embedded environment there is no accepted, adequate large screen input device currently available.

This project will explore the mobile and pervasive computing models, concluding that we cannot assume to apply human computer interaction (HCI) desktop approaches in the new paradigm of mobile and pervasive computing. By looking at the current HCI approach to the desktop we find 3 properties of computing devices that could be used to compare mobile, desktop and pervasive devices.

These areas are: Interaction, Interface and Input-Output. Each of these is described and evaluated for its suitability as a property for comparing computing devices in the different environments. It is shown that interaction and interface are not suitable for making this comparison and input-output device is considered in detail. For practical reasons only screen size is selected for making the comparison between mobile, desktop and pervasive computers.

A method of comparing different size screens is required if we are to show whether the current approach of shrinking and expanding is useful. Fitts’ Law is presented for this purpose. Fitts’ law is shown to be a useful tool for (HCI) comparisons [9, 49] and we
suggest it as a suitable tool for making such a comparison. An overview of Fitts’ law and its development suggests that the law itself needs careful investigation and requires careful handling.

The experimental hypotheses are presented and the design of the experiment is discussed. Results are presented, discussed and conclusions drawn. Finally the project itself is analysed and concluded.
Chapter 2

Literature Review

2.1 The Changing Computing Environment

The way we use computers and where we use them is undergoing major change. We are seeing the emergence of mobile and pervasive computing, a very different paradigm from that of the desktop. Increasing divergence from the desktop bound computer to mobile or pervasive computers is evident in academic, business and personal computing. “The concept of pervasive computing is a paradigm shift from a traditional IT point of view, including to a large extent mobile IT” [31]. Here, we clarify the what the current situation is, then, what we mean by mobile and pervasive computing.

2.1.1 Current Situation

The desktop is the primary computing environment and paradigm at the current time. Computer users are familiar with, and have become accustomed to interacting with computers using the desktop paradigm. The prevalent implementation of the desktop paradigm is the Windows Icons Menus Pointers (WIMP) Graphical User Interface (GUI). An example showing some windows and icons in such a system is shown in Figure 2.1. The problem with desktop computers is that they are restricted - they are tied to the desk. “[The desktop] approach is intended to facilitate ones use of the system by making the manipulation of information in the system analogous to the manipulation of physical objects on a desktop. The choice of office objects in particular is intended to facilitate learning by capitalizing on users familiarity with such objects and with procedures involving them” [39]. This basic design feature of desktop computing has become its weakness, not only are users tied to their desk but “[the desktop] lead[s] us
[... ] towards making the tool the center of attention” [70]. “The desktop [... ] design has evolved on the basis that users are stationary that is, sitting at a desk” [45] and this is not always the case, users are often mobile as well as stationary within any given situation. This has lead to the suggestion of a new paradigm of mobile computing.

2.1.2 Mobile Computing

Mobile computing is the use of computers everywhere. It is being able to take a computer with you anywhere, find a computer anywhere and use a computer anywhere. It is “one of the fastest growth areas of computing” [12]. According to Carroll et. al the future of computing will be “ubiquitous, invisible, embedded, tangible, virtual, active, integrated, interconnected, interoperable and mobile” [17].

Mobile computing is currently implemented and embodied in personal digital assistants (PDAs), mobile phones, Tablet PCs and wearable computers. Companies such as Xybernaut [75] provide specialist wearable computers for the workplace, while PDAs and PDA-phones (e.g. the iPAQ, see Figure 2.2, [35]) make mobile computing available to many people.

A good model of the mobile paradigm is the paper notepad. It can be both small enough (e.g. A6) to fit in your pocket, or large enough (e.g. A4) to display a large amount of information on one page. In either case the notepad is extrememly mobile. It can be used inconspicuously anywhere, any time. Within a large office, forgetting your notepad at a meeting is not a disaster, you just get one from the nearest stationary cupboard. This is the vision of mobile computing, where computers are completely mobile and a commodity. Of course, a computer notepad can be much more powerful
than it’s paper counterpart, especially when used in an embedded environment. In that case, a new ‘notepad’ can automatically contain everything that is on the old one, regardless of where it is.

Why is mobile computing so important? Simply because humans, by their very nature, are mobile [43]. We require the use of computers in many different and varied locations and conditions. Pascoe et. al coin the term, Dynamic User Configuration to describe this. You may want to use your computer at your desk, while you are standing, walking or crawling and this should be facilitated by your computer or computers. In more detail, Pascoe et. al describe the use of a mobile computer in a fieldwork environment. They note that “handheld computing appliances are typically envisioned as tools within the businesspersons domain” and that “the world of the businessperson is far removed from the environment of the fieldworker”. However, in my view many of what are seen as unique requirements of the fieldworker’s situation by Pascoe, do apply to the “businessperson’s domain”. Dynamic User Configuration, Limited Attention Capacity, High-Speed Interaction and Context Dependency [62], are all applicable to situations in the “businessperson’s domain”.

Kristofferson and Ljungberg suggest three categories of mobility: travelling (going from one place to another in a vehicle), visiting (spending time in one place in order to carry out a specific task) and wandering (local mobility with no specific task). This coincides with and develops Pascoe’s Dynamic User Configuration.

This is an important point. Mobile computing is not just for the fieldwork environment, but for business and social environments as well.
This is only half of the paradigm shift that is taking place at the moment. Computing is also moving in a different direction at the same time, not towards smaller and more mobile computers but towards pervasive computers.

2.1.3 Pervasive computing

A definition of Pervasive Computing is given by Gupta and Moitra. Pervasive computing is

“[Saturating] an environment with computing and communication capability, yet having those devices integrated into the environment such that they disappear” [31].

Pervasive computing is embodied in embedded computing. We see pervasive computers as falling in to two categories: primary and secondary pervasive computers. Primary pervasive computers are those that the user interacts with directly, for example an interactive whiteboard. Secondary pervasive computers are those that are hidden from the user and augment or facilitate a task without the user’s knowledge. Current examples of secondary pervasive computers include those embedded in appliances such as washing machines and cookers, the engine management system in a car or wireless access points.

The trend in primary pervasive computing has been in the opposite direction to that taken in mobile computing. Primary pervasive computing is increasing the screen size, not reducing it. Pervasive computers are also therefore, fixed and immobile.

Large screen displays are already available in many environments such as pubs and offices. In social environments large screen displays provide a dynamic advertising feature, while in business they are increasingly being used for conferencing, group interaction and computer supported cooperative work (CSCW). Advances in plasma displays and new technologies such as light-emitting polymers are enabling manufacturers to provide larger displays at a more affordable cost.

There are a number of projects, currently being undertaken, implementing pervasive computing environments. These include, Project Oxygen at MIT [3], Aura at CMU [5], Protolano at Washington [4], Endeavour at Berkley [2], EasyLiving at Microsoft [54] and Cooltown at HP [36]. These involve a mixture of primary and secondary computers: large screens provide access away from the users’ own office and in meetings
while secondary computers support users’ own mobile devices providing, for example, wireless network capability.

Computers are already embedded in everyday appliances such as washing machines, cookers and cars and indeed they have moved from the desktop into our streets and buildings. Computers embedded in games machines and telephone boxes; pubs, cafes, hotels and shops, are providing embedded, wireless communication through projects such as myCloud [57] and BT Openzone [13].

We are seeing more and more large screen displays attached to embedded computers in social environments such as pubs and cafes, at football stadiums and in train stations. They are also increasingly being used in schools, for example in the form of electronic whiteboards. The implications of using large and small screens are even investigated for accident and emergency departments by O’Neill et. al [60], who state that “Information on a large and small, public and private scale is important in many situations”.

In fact, not only will large screen displays be used in public situations but authors such as Swaminathan and Sato “are convinced that it is simply a matter of time before large displays will be standard for home and office computers” [68].

We cannot necessarily apply the HCI approaches from the desktop paradigm to the new paradigms of mobile and pervasive computing, but what is the HCI approach in the desktop paradigm? A short history provides us with some answers.

2.2 A Short History

The HCI approach has undergone a large amount of change since the command line interface on computer terminals. As computers continue to change we look at the changes that have already taken place in order to understand those we are currently seeing.

The first step of note in user interfaces was made in 1963 by Ivan Sutherland at MIT. Sutherland’s Sketchpad, implemented on the TX-2, introduced the direct manipulation interface (See 2.3.1). It allowed users to grab, move, and resize objects using a light pen. It was the first graphical user interface. This was followed in 1965 by the development of the mouse. Although not widely in use until the 1980s the mouse replaced the expensive light pen, popular in the early 1980s and developed in 1957 [1]. This change in input device prompted researchers to compare old and new input devices, such as the mouse and lightpen, for example in [16]. The graphical user interface continued to
develop with multiple, tiled windows demonstrated in Engelbart’s NLS in 1968. Also in 1968, AMBIT/G pioneered iconic representations, gesture recognition, dynamic menus, selection of icons by pointing and moded and mode-free interaction [55].

Computer screens have also come a long way since Sutherland used a nine inch cathode ray tube (CRT) on TX-2. On personal computers, the first Apple Macintosh also came with a built in 9-inch black and white CRT screen in 1984 [20]. From then up to the 1990s, standard computer displays were limited to green letters on a black background. Colour screens were available, such as the 14” (640x480) monitor that came with the Macintosh II, released in 1987 [63]. Probably the first 19”, high resolution (1152x900) CRT came with the Sun SPARCstation 1, introduced in 1989.

In terms of small screens, the Apple Newton, released in August 1993, was the first popular hand held computer, it had a 366x240 pixel reflective Liquid Crystal Display (LCD). Today, Thin Film Transistor (TFT) and TFT LCD displays, with 240x320 pixel resolution are making hand held computers cheaper and lighter than before. Currently on the desktop, TFT and TFT LCD are taking over from CRT monitors. In the area of large screens, plasma displays and new technologies such as light-emitting polymers are creating a new class of flat-panel display that are thinner and lighter than ever before [52]. The size of screens is no longer a restricting factor. Manufacturers can create screens of any size, both small and large, and will soon be able to place them on almost anything.

Changes and inconsistencies in user interfaces caused IBM to try to standardise interaction across different machines. IBM’s Common User Access (CUA) guidelines provide a specification of interface components, techniques and their application. Three editions have been published in 1987, 1989 and 1991. Comparing these successive editions of the CUA guidelines, as in [11], provides a clearer insight into how user interfaces have had to change since 1987 and why. The original guidelines (CUA87) were aimed at “a world of personal computers intermixed with host-attached non-programmable terminals”. Their goal was consistency between non-programmable terminals and personal computers. These guidelines quickly became outdated as use of the personal computer proliferated. By 1989, it was necessary to select and develop those guidelines applying specifically to personal computers. The aim of CUA89 was to gain consistency between the different applications running on a single personal computer. It “established an application-oriented Graphical Model as the primary style for the PC environment.” The CUA guidelines of 1991 built on the developments of 1989 to implement tools and techniques that moved the user interface closer to its paradigm.

Slowly, computers have changed from shared to personal. Now they are changing from personal to mobile and pervasive. Are the new computers found in mobile and
pervasive environments actually different from each other and from the desktop? How are mobile and pervasive computers changing from the desktop? We have seen that there are 3 main ways in which computers have changed over the last 30 years. That is in: Interaction, Interface and Input/Output. Let us examine each of these in turn.

2.3 Interaction

The first area that is examined is interaction. The best way to explain the difference between interaction and interface or input/output is by example. We return to the earlier example of writing on a piece of paper. The input/output is found in the pen and pencil, interface whether you write a word or a picture to define an object and interaction is the action of putting the pen on the paper and watch the ink being left on the paper. In fact, this example also demonstrates the dominant interaction style: direct manipulation.

2.3.1 Direct manipulation

Direct manipulation was itself a paradigm shift from command line interaction to a transparent method of interacting with objects. Introduced by Sutherland, the primary method of communication between user and computer is direct manipulation. The user interface supports a dialogue, in some ways like a conversation between people, but between a user and a computer [11]. Direct manipulation is where the computer provides continuous feedback to the user’s inputs. The user can immediately see the effects of their actions. Objects are manipulated directly and the results are seen in real time. Schneiderman [22] gives a more complete definition, highlighting the following features

- Visibility of the objects of interest
- Incremental action at the interface with rapid feedback on all actions
- Reversibility of all actions, so that users are encouraged to explore without severe penalties
- Syntactic correctness of all actions, so that every user action is a legal operation
- Replacement of complex command languages with actions to manipulate directly the visible object.
Direct manipulation involves being immersed in the task, not the computer itself; its aim is for the tool to disappear.

Today there are two forms of technology: technology that we notice and technology that disappears. A pencil, paper or gas stove are all technologies, but we take them for granted. The computer is a technology that we notice. It is hard to ignore. Many homes own a personal computer and often it is the centre of attention in a room. When you are using a computer you are fully aware of the fact. This is most unlike other tasks, such as writing. One does not think “I need to use a piece of paper and a pencil to write a letter”, one simply picks them up and starts writing. This is the future that Norman argues for in [58].

Notice that all of the technologies that we take for granted above are all interacted with using direct manipulation. Here we find the goal of full direct manipulation: transparency. Transparency is best illustrated by two examples, one from primary, and one from secondary pervasive computing. (Incidentally, here we see again how the primary and secondary pervasive models can work together in one system).

The first example of transparency is in secondary pervasive computing in the Eurofighter Typhoon aircraft. In this aircraft, all of the pilot’s inputs to the flight controls are interpreted by the flight computer and translated into hundreds of tiny adjustments to the flight surfaces. The pilot is not, consciously, aware that this is happening. He has been taught that this is the only way that this aircraft is able to stay airborne, but he would not be able to tell otherwise. He simply flies the aircraft as if it were any other with wire or hydraulic interconnects. The computer and its actions are invisible.

The second, is also taken from the Eurofighter Typhoon. Here, transparency is demonstrated with a primary pervasive computer by the Head Up Display (HUD). The HUD is used to give the pilot information about the status and position of his and other aircraft. The pilot no longer has to look down in to the cockpit to check his flight instruments or tactical displays. The HUD allows him to concentrate on flying the aircraft, not keeping his heading correct or ensuring the correct weapon is selected. All of this information is presented in a way that is easily accessible and does not interfere with the task of flying the aircraft. If the pilot’s attention were to be drawn away from the task of flying to the tool - the results could be catastrophic.

This situation, where the user’s attention is drawn away from the task and towards the tool, is known as a “breakdown” [72]. The WIMP GUI is a relatively crude implementation of direct manipulation interaction; this stems from a lack of transparency and the fact that at times, it “breaks down”. “When the tool fails the user becomes conscious of its properties. This is a breakdown” [74]. The future, as we see it, is to
increase the transparency of the systems we are using. It has already been demonstrated that more transparent technology is already widely implemented and used by the military.

Another situation, where transparency is essential, is that of a fieldwork environment seen in chapter 2.1.2. Here the fieldwork requires total concentration, the fieldworker must not be distracted from their task. It is essential for everything about the computer to be transparent.

Kristoffersen and Ljungberg point out that different work contexts require different solutions. Direct manipulation, in its current state, is no use if the users hands are required for another task and are not free to manipulate the user interface or their eyes are not free to monitor the feedback. They argue that interaction should just “take place”. There should be no need for the user to have to stop what they are doing and “make place” for the interaction [42]. This is of course, exactly the need for more direct manipulation, for transparency.

Both Pascoe et. al and Kristoffersen and Ljungberg suggest alternatives to direct manipulation. Pascoe et. al suggest a specialised system called a MAUI (Minimal Attention User Interface). A MAUI is designed to “satisfy the needs of the fieldworker with respect to their characteristics of dynamic user configuration and low attention capacity”. Kristoffersen and Ljungberg have developed MOTILE [42], a system for mobile computers which are specifically being used in situations requiring: Little or no visual attention; structured, tactile input and the use of audio feedback.

As noted previously, the requirements of the fieldworker introduced by Pascoe, do apply in the businessperson’s domain. Dynamic User Configuration is certainly demonstrated by business people moving between offices, conversing in corridors or making their way to meetings. Limited attention capacity is applicable in meetings or presentations when taking notes or sending important messages. In such a context it is important to be able to do so without being distracted from the primary task. High speed interaction, accessing information to answer client enquiries, and context dependency also apply similarly.

Since we are all mobile by nature, these ideas apply to our everyday environments and activities. People are constantly moving between house, work and supermarket, needing access to traffic, weather and shopping information, and wanting it in an increasingly “on demand” world [37]. Transparency is a need of not just field workers, but anyone who uses a computer.
Primary pervasive, large screen interaction has received relatively little attention. Direct manipulation appears to be the accepted future as there is little literature looking at alternative approaches. Systems such as Reality Centers [65] and Powerwalls “allow several users to be immersed in a virtual environment” [32]. Such systems allow users to interact with objects on a larger scale, using a more direct manipulation of objects. Users are “immersed in [the] virtual environments, so [they] can explore, understand, and communicate about [the] data in ways not possible in the physical world” [65]. Cao and Balakrishnan present a review of many of the large screen systems available [15]. What we notice about the systems reviewed are that they all use direct manipulation. Cao and Balakrishnan suggest a new input device for large screens as part of a system that allows the selection and manipulation of objects on the display using the new input device. That is, a direct manipulation system of interaction.

We have suggested that the businessperson’s domain is not very different from that of the fieldworker. This suggests that the businessperson’s computing devices should use more direct manipulation than they currently do. Similarly the trend in pervasive computing is also towards more direct manipulation of objects. Direct manipulation is still a good method of interaction and the future, as we see it, is to move towards even more direct manipulation of objects. HUDs, reality centers and systems such as the MAUI attempt to move away from the GUI model towards more direct manipulation. Currently, the WIMP GUI is a relatively crude implementation of direct manipulation interaction, the next section looks at the desktop GUI in more detail.

2.4 Interface

The current interface presented to the user is the desktop. The desktop paradigm, and more specifically the GUI, appears to be a rather complex idea. What a GUI is, does not appear to have been comprehensively answered. A useful answer is found in [10], even though the task addressed was not to define the desktop. One of the main components of the GUI is the WIMP classification.

In addition to the WIMP GUI being a crude implementation of direct manipulation in the desktop environment, it was never intended for the mobile of pervasive environments.

Many, if not all, mobile devices have adopted the WIMP system, especially on current wearable and PDA releases (Xybernaut, iPAQ etc.). Manufacturers use cut down versions of the windows (Windows CE, embedded linux [6]) interface and operating system (see [53]). Figure 2.3 shows how the Windows start menu has been adapted for
the iPAQ. Schmidt et. al point this out in [64], “The software and especially the GUI supplied with these [wearable] computers [is] still the same as on desktop computers or developed in a very similar style, based on the WIMP paradigm”. They found that most of the non-desktop devices used a desktop metaphor and these interfaces were not “efficiently usable” in mobile situations. They implement a “wearable graphical user interface (wGUI)”. The design principles are based only on observations. Other interfaces have been proposed for small screen devices such as Peephole [76]. Peephole displays are spatially aware handheld computers where the non-dominant hand is used for navigating information spaces and the dominant hand is used for interaction/manipulation using a stylus.

Aimed at large screen interfaces, Denoue et. al present AttrActive Windows [21]. Moving away from the desktop metaphor, they take up the bulletin board paradigm set out in [29]. Here notices and posters can be ‘pinned’ to a virtual bulletin board, lifted to view underneath and moved to other parts of the board. Virtual sheets even flap in the ‘wind’. This is a result of trying to emulate the real world bulletin board model, making the computer version more natural to use. The result being an impression of transparency.

All of the new interfaces discussed here still rely on pointing and selecting objects. We concur with Weiser [70] in arguing that mobile and primary pervasive devices need a new kind of interface. The current approach is to shrink the desktop interface and transfer it, with a few minor changes, to palmtop computers, or expand it to large screen displays.

Although new interfaces have been suggested, they have all been developments within
direct manipulation interaction. It would not be useful, at this stage, to compare mobile and pervasive devices to the desktop by their interface, since we have no clear idea of what the interface is on a desktop, and because there are still so many interfaces in the first stages of implementation for mobile and pervasive computers.

2.5 I/O

So far no acceptable method of comparison between mobile, desktop and pervasive computers has been found. From the two previous sections we can see that interaction and interface are not properties of the devices themselves (e.g. desktop on PC and PDA) and therefore do not present suitable properties for making a comparison. In any case, currently, there is little difference between the different computers in the interactions and interfaces. We conclude that these areas do not present a suitable option. However, there are two things that almost all computers in mobile, desktop and primary pervasive environments share. The first, is a device for communicating information to the computer - an input device. The second a device for the computer to communicate information back to the user - the screen.

2.5.1 Input Device

Mobile and desktop computers already have standard, established input devices: the stylus and mouse respectively. There is no such established input device for primary pervasive computers yet.

The mouse has been associated with the desktop since the 1980s. The mouse itself has developed over the time it has been in use. Originally taking input from two perpendicular wheels, it has become more accurate, first using a ball and now with the introduction of the optical mouse [1].

By contrast there has been much investigation into different input devices for embedded computing but none has been adopted wholesale as yet. Statements such as “As large, wall-sized, multi-screen display systems become more ubiquitous, there is a need for better and simpler interaction tools” [18] come without supporting arguments or examples of what such tools might be. According to Kostakos and O’Neill, traditional desktop input techniques are inadequate [41] and do not transfer well to pervasive computing environment [59]. Suggestions for new input devices include visible and infrared pointers [18], wands [15], individual and group devices [32], hand gestures and utilising
mobile devices for gesture recognition [41] and 3D mice [69].

We have however, seen little development of the mobile computer stylus so far. The stylus is seen as the device of choice for use with mobile computers and has become generic. It has been assumed that the stylus is the best option, and this may well be true since it is modelled on the ubiquitous pen. There has been some research into improving particular aspects of stylus input, such as text entry: systems such as Directional Stroke Recognition [41] and Shorthand Writing on Stylus Keyboard [77] try to overcome the problems associated with writing on a touch screen with a stylus. However as Kostakos and O’Neill point out, “stroke recognition predates PDAs by quite a while”. There are efforts to develop completely different ways of inputting data to mobile computers such as TiltType [61], which uses the orientation of the device to select characters. There has also been little development of the stylus itself and no comparisons of mobile input devices to investigate whether the stylus is the best tool available.

Studies such as that by English et. al [23] point out that the “the pen must be held in the air while it is being used” leading to fatigue. Although it has been demonstrated that their findings are not entirely reliable, (see Section 2.7.4) this comment is backed up by MacKenzie et. al [49] who note the effect of “significant interaction between muscle and limb groups” which is not found when using a mouse. Although we know that the mouse and graphics tablet stylus are comparable (in a steering task), the direct nature of pointing with a touch screen stylus suggests that a comparison between the touch screen stylus and, at least the mouse, is necessary.

Although there has been no comparison of a range of available mobile input devices, there has been work in comparing only desktop devices [7, 16, 19, 24, 49] and only large screen devices [18]. In addition to there being no overall comparison between mobile input devices, there is no reported work comparing devices between environments. For instance English et. al [23] compared, among other devices, an early mouse and light pen in a desktop situation. They found that their mouse was faster than their light pen. Their results were affected by many problems due to the developmental stage of many of the devices they used. The observations made about their mouse are of questionable application to today’s mice because of the familiarity with, and developed stage of mice today. They recognised that the light pen was good for unexperienced users but this is more likely due to the fact that some of the users had not used a mouse before. The authors themselves point out that “There were some obvious defects in the particular devices tested” [23].

Input device presents a useful property that could be compared between computing devices. The stylus appears to be the pointer of choice for mobile computers, and the
mouse for the desktop. However, there is no such accepted generic pointer for large screen, primary pervasive computing devices. Furthermore, since the input device is interchangeable (interactive whiteboards can also use a ‘stylus’), comparison could only be drawn between input devices and not between the mobile, desktop and embedded computers themselves.

2.5.2 Screen Size

The final property considered is screen size. There has been little or no research into the direct effects of screen size. O’Neill, Woodgate and Kostakos demonstrate in [60] that information display on large and small, public and private scale is important in many situations. Small screens are essential in making mobile computers mobile, and embedded computers can have small, large (primary) or no screen at all (secondary). Large screens are useful especially in CSCW, and public displays. Large displays can be found in offices, bars, cafes, on train platforms, in football stadiums and in academia.

The ideas that exist, about designing for different size screens, are derived from a pragmatic approach, rather than from any quantifiable test results. Brewster presents a short argument that implementing the desktop interface on a mobile device does not work well. He presents more arguments than most, although those regarding black and white screens and power consumption are already dated [12].

When it comes to screens larger than those found on desktop computers (15”-21”) there is a similar lack of concrete data. Swaminathan and Sato report pragmatic design in their paper [68]. They record the lessons learned from designing and implementing a large screen based interaction system. They comment that they “thought of a large display as basically the same as a regular display, just larger! [After implementing our design] we no longer do so. We think that when a display exceeds a certain size, it becomes quantitatively different.”

Authors appear happy to make statements such as

“As the screen shrinks in size, and less information may be shown on it, […] the user will be required to increase the level of interaction with the device in order to get to desired information”. [40]

“physically shrinking everything including input and output devices does not create a usable mobile computer”. [41]
“document browsing in [palm-top computers] is not an easy task because of their limited screen size”. [67]

“The graphical techniques for designing interfaces on desktop systems do not apply well to handheld devices”. [12]

Similarly, authors are making statements about large screens such as

“As large, wall-sized, multi screen display systems become more ubiquitous, there is a need for better and simpler interaction tools”. [18]

“We thought of a large display as basically the same as a regular display, just larger”. [68]

There are many articles looking at how using a smaller screen affects browsing the web [44], [67], [73], reading [51] and how to use small screen space more efficiently [40]. Some of these provide basic arguments, while others are often based on the unwritten assumption that there is something wrong with the status quo: that is, the practice of shrinking the desktop interface down to fit on a mobile computer. Not only that, but manufacturers are largely ignoring these claims and assuming that small screen devices are the same desktops, continuing to develop desktop based programs to fit on the small devices.

Screen size provides the ideal property for comparison of the three different environments being investigated. Desktop computers at the moment have a narrow range of screen sizes (15-21”), mobile devices invariably come with small screens and embedded devices, more often than not, have large screens.

2.6 Summary

More and more questions are being asked about whether we should be using more developed interaction methods (more direct manipulation e.g. MAUI), different interfaces (e.g. Peephole, AttrActive Windows) or different input devices (e.g. TiltType, Wands) or perhaps all three. Questions are also being raised as to whether direct manipulation is the best way to interact with mobile and embedded computers. These are, to a large extent being ignored. However people are looking for new interaction methods, interfaces and input devices to use in the new computing environments. At this point,
assumptions are being made about the necessity of these investigations. It is required to show whether the computers in the different environments are different. Assumptions are being made about input and output devices making mobile and embedded computers different. Much of the arguments shown are based on changing usage. Ubiquitous and pervasive computers are used in different environments, for different tasks and in different ways and this makes them different. Other assumptions are based on apparent differences: different screen size for example.

These arguments are either disconnected from the actual devices (e.g. environment), superficial (screen size), or based only on experience in one area of application (field-worker).

For instance, the design of the interaction method seems to be dependent on the designers of the devices and whether the devices are designed for and being used in, a particular environment. This is useful for that situation, it does not show that a device is different. They also all use direct manipulation. Nor does it show that redesign is necessary in any but the demonstrated environments. The generic interface is very similar on many implementations of personal digital assistants to the desktop, for instance. This does not prove that the PDA is the same as the desktop.

How then, can we tell if ubiquitous and pervasive computers are different to personal computers? Screen size appears to yield a useful property for comparison within the current constraints. In order to vary only screen size we will test using direct manipulation, a mouse and a task that involves object selection, keeping these parameters constant across a range of screen sizes. Doing any more than this, for instance looking at a stylus as well as a mouse increases the complexity of the experiment too greatly and is not justifiable. Practically time and participants do not allow it. This project presents stage 1 of this investigation: keeping all but the screen size constant. Stage 2 should investigate input device. In conjunction these two properties uniquely identify the current computer devices.

In order to compare devices by their screen sizes we need some kind of model that can be used to provide experimental evidence. In the next section such a model is presented and investigated.
2.7 Fitts’ Law

2.7.1 Introduction

So far it has been argued that screen size would be a useful property to compare in order to see if mobile, desktop and embedded computers have any quantifiable differences. It is not enough to say that they are different by simply comparing their dimensions. A model that provides quantitative data is required.

Sections 2.3.1 and 2.4 demonstrated that direct manipulation and pointing and selection of objects are a mainstay of the current computing paradigm. “Today’s graphical interactive systems largely depend on pointing actions” [8]. It was also shown that this looks set to remain the case in the near future on some mobile and most embedded computers (The bulletin board paradigm, demonstrated in [29], for example). “Pointing undoubtedly remains the most universal interaction paradigm across diverse domains and contexts” [8]. It is therefore pertinent to look for a model that models pointing in a direct manipulation sense.

One of the best developed and researched models, modelling pointing and selection, in HCI is Fitts’ Law. In the following section the use and development of Fitts’ Law is reviewed together with its suitability as a possible method of comparison. It was found that Fitts’ Law has been well established in HCI since 1978 when Card et. al used it to compare a number of input devices. Analysis of Accot and Zhai’s [9] work shows that a common but possibly undesirable assumption is made and this is investigated in detail.

2.7.2 Beginnings

In 1954 Paul Fitts of the Ohio State University ran three experiments to test the specific hypothesis: *If the amplitude and tolerance limits of a task are controlled by the experimenter and the participant is instructed to work at his maximum rate, then the average time per response will be directly proportional to the minimum average amount of information per response demanded by the particular conditions of amplitude and tolerance* [27]. Of the three experiments we are interested in the first. This involved the participants tapping two rectangular metal plates alternately with a stylus. See Figure 2.4. The width of the plates and the distance between them was varied to control movement tolerance and amplitude respectively. Four error plates recorded undershoots and overshoots.
Fitts’ experiments demonstrated that the rate of performance increased uniformly as the amplitude decreased and the tolerance limits were extended, confirming the predictions set out in his hypothesis. As a result he formed his original equation, a measure for the binary index of difficulty $ID$, defined as:

$$ID = - \log_2 \left( \frac{W}{2A} \right) \text{bits/response,}$$

(2.1)

where $W$ is the width of plate (tolerance) and $A$ is the distance between the two plates (amplitude). According to MacKenzie [47], this comes from an analogy to physical communications.

Following further work with Peterson [26], the resulting law was presented:

$$MT = a + b \log_2 (ID); \quad ID = \log_2 \left( \frac{2A}{W} \right)$$

(2.2)

Where $MT$ is the movement time, $ID$ is the index of difficulty, $A$ is the amplitude of motion and $W$ is the target width or tolerance. This is Fitts’ Law in its original form.

### 2.7.3 Refining Fitts’ Law in HCI

The following developments of Fitts’ Law have all been made in HCI. Fitt’s law has been developed and accepted as a sound quantitative model of human computer interaction tasks. It is “One of the very few quantitative models applicable to HCI tasks” [9]. A review of some of the developments that have made Fitts’ Law the “robust” [49] model it is seen to be, follow.

---

Figure 2.4: Fitts’ original experiment
Fitts’ Law was first accepted in HCI after Card et al. used it to compare four input devices (see Section 2.7.4). Since then, the model itself has been refined within HCI.

According to Buxton [14], “Fitts’ Law was based on work in information theory by Claude Shannon”. This is backed up by MacKenzie, who points out further in [47], that Fitts based his model on an approximation of Shannon’s formula, by Goldman. MacKenzie argues that Fitts made an unnecessary deviation from Shannon’s work and proposes the following reformulation of the law, more accurately reflecting Shannon’s work:

\[ ID = \log_2 \left( \frac{A}{W} + 1 \right) \]  (2.3)

Where A is amplitude and W tolerance (width) (see Figure 2.5). The Shannon formula was argued for by MacKenzie for two reasons. First, in order to counter the fact that a negative index of difficulty was possible. This was first observed by MacKenzie and Buxton [48] in Card et. al’s experiment (Figure 6 in [16]). This is obviously counter intuitive. Secondly, MacKenzie suggests the Shannon formula as a formulation that more closely models the theory behind Fitts’ Law. The Shannon formula is the accepted standard of Fitts’ Law and is found in ISO9241-9.

Accot and Zhai [9] and MacKenzie and Buxton [48], point out that Fitts law is often applied to pointing actions in user interfaces. Unfortunately, Fitts’ Law is restrictively one-dimensional [48]. Further research to find a two dimensional Fitts’ Law was therefore necessary to deal with acquiring targets such as words or icons on a computer screen.
Bivariate Development

One of the criticisms levelled at Fitts’ Law is that it is a restricted, one dimensional model. The problems associated with pointing in 2-dimensions are demonstrated in [48]. Since bivariate “target acquisition [characterises] contemporary direct manipulation interfaces” [14] it is important to look at 2D pointing. Considerable research has shown that Fitts’ Law can be applied to bivariate pointing, culminating in the model put forward by Accot and Zhai in [9].

Crossman was perhaps the first to vary both target width and height. He correlated his data using: 

\[ MT = a + b \log_2 \left( \frac{A}{W} \right) + \log_2 \left( \frac{A}{V} \right) \]

where \( W \) is the tolerance in the direction of movement and \( V \) the tolerance perpendicular to the direction of movement. Crossman suggested: 

\[ MT = a + b \log_2 \left( \frac{A}{W} \right) + c \log_2 \left( \frac{A}{V} \right) \]

However, Crossman used only two participants in his experiment.

Hoffmann and Sheikh [33], building on the work of Crossman in particular, suggest: 

\[ MT = a + b \max (ID_H, ID_V) \]  

where \( ID_H \) is the Index of Difficulty in the direction of movement and \( ID_V \) is the Index of Difficulty perpendicular.

MacKenzie and Buxton [48] not only present a solution to the problem of negative ID, they also present a comprehensive analysis of the possible bivariate models. They consider four new models and compare them to the ‘status quo’ of just using horizontal extent.

Accot and Zhai [9], provide a new model that builds on this previous work. They hypothesise that “is it possible to find a model comparable to the original Fitts’ Law that would model the time to acquire a target of finite width and height” [9]. Accot and Zhai return to the mathematical basics in defining the notion of distance. Using the weighted \( \ell_p \) -norm they derive a formulation that fits the results of MacKenzie and Buxton’s investigation. This generalises nearly all previous suggestions to some form of: 

\[ ||x||_{p,w} = \left( \sum_{i=1}^{n} w_i |x_i|^p \right)^{\frac{1}{p}} \]  

Accot and Zhai then investigated models of the form:

\[ T = a + b \log_2 (||X||_{p,w} + 1) \]
where $a$ and $b$ are constants and $X$ is the vector:

$$X = \left( \frac{D}{W}, \frac{D}{H} \right) \quad (2.7)$$

$W$ here is the width of target pointed to and $H$ is the height of the target.

After testing on both their own data and data from Hoffmann and Sheikh’s earlier experiments they finally suggest:

$$T = a + b \log_2 \left( \sqrt{\left( \frac{D}{W} \right)^2 + \eta \left( \frac{D}{H} \right)^2} + 1 \right) \quad (2.8)$$

where $a \in [-50, 200]; b \in [100, 170]; \eta \in [1/7, 1/3]$ and $H$ the height of the target.

Accot and Zhai [9] demonstrate that it is possible to build two dimensional work on Fitts’ earlier one dimensional work.

### 2.7.4 Using Fitts’ Law To Make Comparisons

Fitts’ Law itself has developed much since it was first applied to HCI tasks. Its real usefulness is discovered when it is used as a predictive and descriptive tool [30]. In fact, its first application to HCI was in describing and predicting the differences between input devices. That is to say, in comparing input devices. The importance of an appropriate and adequate model in comparisons of computer devices is reflected in Card et. al’s comment [16] that English et. al’s study [23] “was more concerned with the evaluation of devices than with the development of models from which performance could be predicted”. Card et. al go on to test a number of devices and show that the Fitts’ Law model fits their results.

Comparing devices predates Card, English and Burr’s paper [16], but theirs was, according to [50], the first comparative evaluation of the mouse and, more importantly, the first use of Fitts’ Law in HCI. Card, English and Burr used the Welford formulation [71] of Fitts’ Law

$$ID = \log_2 \left( \frac{A}{W} + 0.5 \right) \quad (2.9)$$

They compared a mouse, a rate-controlled isometric joystick, step keys, and text keys in text selection tasks. They also attempted to give a theoretical account of their results, testing “The continuous movement devices […] against the predictions of Fitts’ Law”. The relevant findings of the experiment were that the mouse was the fastest device of those presented and that “For the continuous movement devices [ie. mouse and joystick], positioning time is given by Fitts’ Law”.

23
MacKenzie et. al [49] use Fitts’ Law and an extension of Fitts’ Law to “dragging” to replicate Card et. al’s conclusion that the mouse has very good performance for pointing. Their experiment also claimed to show that “Fitts’ Law can model [...] pointing tasks”.

Even after the developments in Fitts’ Law in the twenty five years since Card et. al, Fitts’ Law still provides an accurate model. MacKenzie and Soukoreff demonstrate this in light of the use of throughput and effective target width in [50].

Hence, we may conclude that Fitts’ Law is an accurate model for the prediction of movement time for different devices and provides a good way of comparing them. Studies have shown particularly that on a desktop environment the mouse is the fastest device (of those tested). Fitts’ Law is a powerful model for comparison in direct manipulation pointing tasks.

2.7.5 Problem with Fitts’ Law

A thorough reading of [9] presents a possible problem with Fitts’ Law. When recording the desirable properties of the model they are developing, Accot and Zhai [9] suggest Scale Independency, that is multiplying amplitude and tolerance (in their case horizontal and vertical tolerance) by a constant leaves movement time unchanged. In concluding [9] Accot and Zhai discuss the “interface design implications” of their results. They suggest factors that should be taken into account when pointing to objects and widgets, specifically buttons and the commonly found taskbar. Their suggestions could just as well be applied to icons, hyperlinks and other objects that require pointing for selection. Accot and Zhai suggest that widgets should be elongated in the direction of most common movement. This is a logical implication from their results. The unrecognised implication of their original assumption only becomes clear when larger amplitudes, such as those possible on a larger screen are considered. Here it becomes a hindrance.

In order to explain this claim we first ask two questions: Why do we use widgets? According to [34] one of the primary reasons for the introduction of icons was that they save space. Why do we want to use large screens? One answer is that the same information can be viewed from a greater distance. The second, more applicable in, for example CSCW is that more information can be presented on a large screen than its desktop counterpart.

The problem with scale independency is that, if it is applied literally, there is no
information capacity gain from using a larger screen. Since, as the size of the screen and therefore the amplitudes on it are increased, the widgets must increase in size by the same constant factor as the distance. In this case, no extra information is communicated by the screen. Everything is just bigger. This is not a problem if the screen is presenting football scores in a stadium, but in a CSCW environment this may cause problems, and certainly adds no value.

A review of previous literature reveals that this point has largely been ignored. This is likely due to the fact that scale independency is a property of Fitts’ original equation, the Welford formulation and even the Shannon formulation. This is due to the combination of amplitude and tolerance in a fraction, within the ID, and is easily demonstrated using, for example, the Shannon formulation:

\[
ID = \log_2 \left( \frac{D}{W} + 1 \right) = \log_2 \left( \frac{3D}{3W} + 1 \right) \tag{2.10}
\]

Here the 3s on the right hand side cancel to give the same ID. In fact, not only is this an undesirable property, it does not even model what actually happens in a Fitts’ Law pointing task. Gan and Hoffmann [28] demonstrate that in an original Fitts’ pointing task (that is using two metal plates and a stylus) “regardless of ID, there was a significant effect of amplitude on movement times”. In other words in an experiment, the equation modelled by equation 2.10 may not be an equality. In HCI, this is pointed out by Guiard [30], who says that although “Fitts’ Law predicts that MT, […] should be scale independent […] there is ample evidence that, in violation of the law, scale [amplitude independent of ID] affects MT quite substantially”.

The problem goes deeper than this, however. The assumption of scale independence is an end effect and not the cause which we will now investigate.

### 2.7.6 Factors to Consider When Designing a Fitts’ Task Experiment

The naive acceptance of scale independence comes from a disregard for the effect of amplitude. This is itself a result of poor experimental design. The effect of amplitude is not always fully considered in Fitts’ Law experiments. Card et. al [16] do vary distance and width and Mackenzie and Buxton [48] seem to make allusions to distance. English et. al [23] and more recently Accot and Zhai [9] do not mention taking amplitude into account. However Accot and Zhai’s analysis states that there was no significant effect of D (amplitude) when \( H > W \) (tall target) in terms of errors. Any effect, or not, of D in their movement time analysis is not mentioned. Amplitude errors are again mentioned
in the discussion: “The asymmetrical impact of amplitude and direction constraint is plausible in terms of the mechanisms underlying the pointing action”. That is, the difference in errors when targets were wide against tall, can be explained by the suggestion that “amplitude error [in the direction of motion] has to be momentarily controlled at the time of landing”. In their conclusion then make reference to the greater horizontal distance found on landscape displays. This is one example of a Fitts’ Law experiment where the effects of amplitude are mis-interpreted or ignored. Accot and Zhai suggest that the pointer has to be controlled at the time of landing, a valid extension of this idea may be that a greater distance, as on a horizontal display, could affect this 'landing time’. The effects of amplitude on movement time, are not considered however.

Guiard [30] shows that even when distance is 'fully’ considered, there are still problems with the experimental design. Experimental design dictates that independent variables (IVs) should be just that, independent and so, varying one should not have an effect on the other. However, in the reviewed literature [e.g. B3, mackenzie, fitts:1954, input:comparison, mouse:joystick], the convention of taking amplitude and tolerance as IVs contradicts this axiom. “D and W cannot work as [IVs] in a Fitts’ Law experiment: the effect of either D or W cannot be assessed separately because, as one is varied with the other kept constant, the ratio $D/W$ - and therefore the ID - is altered”. In experimental terms, the two variables confound one another.

The solution suggested by Guiard is based on the basic geometry of a rectangle. A 2D rectangle can be defined by: its shape (or aspect ratio), its size (the length of one side), its position ((x,y) co-ordinates for example) and its orientation. In the same way, the targets in a 1D Fitts’ Law task can be described by the figure shape (amplitude/tolerance) and size (e.g. amplitude). Defining a 1D target by it’s scale and shape provides a better alternative to using its width and amplitude. This gives us two independent variables: amplitude and ID, and hence two types of task: Movement Scale and Movement Difficulty that we can manipulate completely independently of one another. This is shown in Figure 2.6.

This design disentangles scale from ID and so amplitude and ID present truly independent variables. [30] points out that using amplitude (D) and tolerance (W) as independent variables and varying W with D constant involves only movement difficulty. Even more of a problem is when D is varied with constant W, this inadvertently co-varies movement difficulty and scale. Using amplitude and ID as independent variables avoids this and allows scale independence to be tested as an experimental hypothesis (see Chapter 3).
2.8 Summary

We propose that Fitts’ Law is a suitable model to test the differences between mobile, desktop and embedded computers in terms, at least, of their varying, typical screen sizes. However, we must be careful when designing the experiment that we do not cause the variables movement scale and difficulty to co-vary. By choosing movement scale and difficulty as our independent variables we avoid this problem. Using Fitts’ Law in this way, we can compare different sized screens while we also seek to confirm the argument that the pointing task being tested is not scale independent.

As we move towards a paradigm of mobile and pervasive computing systems, with much smaller and much larger screens than are normal in the desktop computing paradigm, it is useful to consider Fitts’ Law in these new environments and to consider these environments using Fitts’ Law. As long as the task still involves selecting widgets then Fitts’ Law will apply and this is likely since “Selection of objects is fundamental for interactions with objects” [66].
Chapter 3

Experimental hypotheses

We have argued that there is a need to compare the varying screen sizes that are synonymous with mobile, desktop and pervasive computing and that this will allow us to compare the devices from each individual paradigm. We have seen that it is useful to consider Fitts’ Law in these environments and that there may be a problem with Fitts’ Law in its current state. The following experimental hypotheses are used to test these proposals:

3.1 Screen Size

Experimental Hypothesis: Movement time is proportional to the relative distance travelled by the pointer. That is the distance the cursor travels across the screen. That is to say, large screens are slower and small screens faster.

3.2 Scale Independency

Experimental Hypothesis: Fitts’ law is not scale independent, in other words, movement time is affected by movement scale (amplitude) independently of movement difficulty (ID).

Order and carry-over effects will be controlled by asking the participants to take a short break, in a different room between screens. We will use a related subjects design to eliminate any individual differences. Any remaining order, carry over or learning
effects will be removed by counter-balancing the screens and randomising amplitude (movement scale) and ID (movement difficulty).
Chapter 4

Method

4.1 Design

We used a three-way (3x3x7) related samples factorial design. The independent variables were Screen (small, medium, large), Amplitude (200px, 400px, 600px. Measured between the nearest edges of the target.) and Index of Difficulty (2.58, 2.94, 3.46, 3.75, 3.93, 4.14, 4.39). Screen was counterbalanced and Amplitude and ID were randomised. Dependent variables were movement time (milliseconds), overall number of errors and number of overshoots (The pointer passed the furthest edge of the target without clicking). Errors were recorded as undershoot, overshoot and other errors.

The experiment included two sessions for each screen: a practice session and a data-collection session. The practice session lasted until participants reached an acceptable level of errors (less than 4 per condition). This took into account any warm up and practice effects by making sure all of the participants had reached a common standard of accuracy before beginning the experiment. The data-collection session consisted of participants testing the 63 conditions for Screen, Amplitude and ID. Within each condition, participants performed 10 trials. Subjects were allowed a rest every 210 trials, as there were 21 conditions per screen. Subjects were told to rest in a separate area between screens. This allowed for the effects of fatigue by causing them to rest their eyes, limbs and concentration before they became fatigued from the task.
<table>
<thead>
<tr>
<th>Amplitude (px)</th>
<th>ID (bits)</th>
<th>4.39</th>
<th>4.14</th>
<th>3.93</th>
<th>3.75</th>
<th>3.46</th>
<th>2.94</th>
<th>2.58</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td></td>
<td>10</td>
<td>12</td>
<td>14</td>
<td>16</td>
<td>20</td>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td>400</td>
<td></td>
<td>20</td>
<td>24</td>
<td>28</td>
<td>32</td>
<td>40</td>
<td>60</td>
<td>80</td>
</tr>
<tr>
<td>600</td>
<td></td>
<td>30</td>
<td>36</td>
<td>41</td>
<td>48</td>
<td>60</td>
<td>90</td>
<td>120</td>
</tr>
</tbody>
</table>

Table 4.1: Target Sizes (px)

4.2 Pilot

A pilot test was run using four participants. Participants were allowed to practice for as long as they felt necessary. Most of the participants gave up after a few practice clicks on the second or third screen in the pilot. The practice was therefore standardised in the actual experiment containing a fixed number of conditions (15) on all three screens. The pilot test also prompted the explicit measurement of overshoots in the actual experiment.

4.3 Participants

Overall there were 60 participants, 38 male and 22 female. 3 of the participants had taken part in the pilot. The participants ranged in age from 18 to 23. All participants were students at the University of Bath. Participants were allocated randomly to the conditions. Each used their preferred hand.

4.4 Apparatus

The experiment was conducted on a Xybernaut wearable PC (500 MHz Intel® Mobile Celeron®) connected to the screens by VGA cable. The machine ran Microsoft Windows 2000. It had an ATI Rage Mobility-M Graphics card. The mouse acceleration was set to default and was not changed during the experiment. The conditions were performed in full screen mode, with a grey background colour. Since the largest display had a wide screen aspect ratio it was set to normal screen ratio, leaving two dark gray bands at the edges of the screen. Mouse movement was restricted equally in all directions of movement using a circular mouse mat radius 95mm. Participants sat directly in front of the screen at a constant distance from the screens. The centre points of the screens were aligned at the same height.
The input device was a Microsoft IntelliMouse Optical USB.

The three screens used were:
Small: Xybernaut screen. (8.4”, 173x131mm, all-light readable display, 800 x 600 colour SVGA, FPD-00200) [75].
Medium: Ilyama screen. (18.1”, 362x290mm, dot pitch 0.28 mm, 1280 x 1024, AS4612UT BK) [38].
Large: NEC Plasma display. (61”, 1351 x 760 mm, 1014x768mm used, pixel pitch 0.99 mm square, 1365 vertical dots x 768 horizontal rows) [56].

The medium screen supplied an increased scale 2.1 times that of the small screen. The large screen was 5.9 times the size of the small screen. (ie. The area displaying output from the computer, not including the bands at the edges was 5.9 times larger). The output from the computer was constant at 800x600 pixel. The refresh rate was also constant and left at the default setting of 60Hz.

Movement time was calculated by taking the cpu clock time when the participant clicked a target and subtracting the time at the previous target click.

4.5 Procedure

Participants were instructed to sit and read the instructions that appeared on screen. They first completed a practice session and were instructed to begin whenever they were ready. When the practice session was complete the participant was shown the instructions again and instructed to begin when ready. The instructions specified that the participant should sit in an upright position and place the mouse mat in the most comfortable position on the desk. Participants were reminded to sit upright if they
The task was to point two square targets on the screen (see Figure 4.1). The active target was green, the passive target grey. The instructions requested participants to “click and release the mouse on the GREEN button as quickly and as accurately as possible”. When the participant clicked on the highlighted target, or missed, the colours swapped. This information was included in the on-screen instructions. When the colours swapped, this indicated that the clicking task was completed and that the participant was to move to the opposite target and select it. Clicking off the target recorded an error and passing the furthest edge of the target without clicking recorded an overshoot. There were four buttons arranged around the target to measure errors. These were the same colour as the background and were not visible. When clicked on they changed from light to dark grey indicating an error. The buttons were positioned: One to measure undershoots, one overshoots, one above the target and one below the target (under and overshoot buttons are shown in Figure 4.2). These extended the width of the target button. Clicking anywhere else on the screen recorded a general error. If an error was scored the time was not recorded, this ensured that only the movement time to select the target was recorded, since an error click was the same as clicking on a much larger target than the one presented. It also prevented errors such as clicking the same button twice from affecting the data. When each participant had completed the task on a screen they were instructed to sit in the doorway of a darkened room, facing in to the room and away from any computer screens. They remained there while the next screen was set up. This took three to four minutes.
Chapter 5

Results

The dependent variables measured in the experiment were movement time (ms), total number of errors and number of overshoots. A significance level of .05 was used for all statistical tests.

Error trials were discarded from time calculations since these trials did not record the movement time to select a target, but a larger area around the target. The mean of the remaining trials was taken and times more than two standard deviations away from each conditions mean were discarded to prevent outliers having an effect. This prevented occurrences such as participants pausing during a trial to adjust the position of their hand on the mouse mat affecting the results. The mean was recalculated and this adjusted mean was taken as the movement time for the condition.

The adjusted data was analysed using a three-way analysis of variance (ANOVA) for related samples, with Screen, Amplitude and Index of Difficulty the independent variables. The sphericity of the variables was measured using Mauchly’s test of Sphericity\(^1\). ie. ANOVA for a 3x3x7 related design.

5.1 Movement Time Analysis

There was a significant main effect of: Screen \(F(2,118) = 58.73, p < .001\). Sphericity was violated for all other variables. All of the corrected values were significant so

\(^1\)Sphericity refers to the equality of variances of the differences between treatment levels. Mauchly’s test of sphericity was used to test the hypothesis that the variances of the differences between conditions were equal.
<table>
<thead>
<tr>
<th>Screen</th>
<th>Amplitude 200</th>
<th>Amplitude 400</th>
<th>Amplitude 600</th>
<th>ID 200</th>
<th>ID 400</th>
<th>ID 600</th>
<th>ID 200</th>
<th>ID 400</th>
<th>ID 600</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>4.39</td>
<td>950.82</td>
<td>828.15</td>
<td>808.58</td>
<td>871.49</td>
<td>746.84</td>
<td>757.69</td>
<td>936.80</td>
<td>820.15</td>
</tr>
<tr>
<td>Medium</td>
<td>4.14</td>
<td>884.36</td>
<td>762.68</td>
<td>766.80</td>
<td>805.12</td>
<td>731.45</td>
<td>722.64</td>
<td>852.18</td>
<td>783.46</td>
</tr>
<tr>
<td>Large</td>
<td>3.93</td>
<td>819.74</td>
<td>741.67</td>
<td>735.32</td>
<td>767.13</td>
<td>709.29</td>
<td>686.35</td>
<td>824.82</td>
<td>741.12</td>
</tr>
<tr>
<td>3.75</td>
<td>789.52</td>
<td>704.50</td>
<td>725.56</td>
<td>737.55</td>
<td>653.80</td>
<td>670.20</td>
<td>757.18</td>
<td>710.82</td>
<td>726.51</td>
</tr>
<tr>
<td>3.46</td>
<td>723.00</td>
<td>656.92</td>
<td>672.59</td>
<td>680.55</td>
<td>627.45</td>
<td>644.55</td>
<td>716.08</td>
<td>676.30</td>
<td>693.55</td>
</tr>
<tr>
<td>2.94</td>
<td>635.098</td>
<td>587.77</td>
<td>608.00</td>
<td>615.41</td>
<td>573.17</td>
<td>570.06</td>
<td>653.78</td>
<td>624.15</td>
<td>631.27</td>
</tr>
<tr>
<td>2.58</td>
<td>573.50</td>
<td>560.39</td>
<td>576.88</td>
<td>566.55</td>
<td>530.65</td>
<td>555.19</td>
<td>605.87</td>
<td>573.65</td>
<td>608.69</td>
</tr>
</tbody>
</table>

Table 5.1: Mean movement times

The conservative Greenhouse-Geisser corrected values of the degrees of freedom are reported.

There was a significant main effect of Amplitude (movement scale) $F(1.62, 118) = 171.18, p < .001$, ID (movement difficulty) $F(3.64, 214.93) = 879.94, p < .001$, Screen x ID $F(9.34, 550.88) = 4.05, p < .001$, Amplitude x ID $F(8.62, 508.70) = 17.33, p < .001$.

The effect of Screen x Amplitude (conservative Greenhouse-Geisser=.052 was used although the more liberal Huynh-Feldt was .049) and of Screen x Amplitude x ID was not significant. The cell mean movement times are shown in table 5.1.

A post-hoc pairwise comparison for the main effect of Screen with a Bonferroni adjustment was used. This reveals that the significant main effect of screen reflects a significant difference ($p < .001$) between levels 1 and 3 (small and medium), and 2 and 3 (large and medium).

The significant main effect of Amplitude reflects a significant difference between levels 1 and 2 (200px and 600px), 1 and 3 (200px and 400px), ($p < .001$) and 2 and 3 (600px and 400px), ($p < .05$) The significant main effect of ID reflects a significant difference between all pairwise level comparisons ($p < .001$).

5.2 Error Analysis

5.2.1 Undershoot Errors

Screen and Amplitude violated Sphericity. All of the corrected values were significant. The Greenhouse-Geisser corrected values show that there was a significant main effect of
Screen $F(1.78, 105.26) = 26.27, p < .001$ and of Amplitude $F(1.83, 108.03) = 3.36, p < .05$. There was a significant main effect of ID $F(6, 354) = 2.96, p < .01$ and Screen x Amplitude $F(4, 236) = 7.75, p < .001$.

There was no significant main effect of Screen x ID or Screen x Amplitude x ID. The Greenhouse-Geisser values show a significant main effect of Amplitude x ID $F(8.97, 529.16) = 2.19, p < .05$.

A post-hoc pairwise comparison shows a significant difference between all levels, small and medium ($p < .001$), medium and large ($p < .01$) and small and large ($p < .001$).

### 5.2.2 Overshoot Errors

There was a significant main effect of screen $F(2, 118) = 3.16, p < .05$. The Greenhouse-Geisser corrected values of the degrees of freedom show a significant main effect of Amplitude $F(1.17, 101.05) = 42.17, p < .001$. There was a significant main effect of ID $F(6, 354) = 6.82, p < .001$, Screen x Amp $F(4, 236) = 3.37, p < .01$. Screen x ID was not significant.

There was a main effect of Amplitude and ID, and Screen x Amplitude x ID, the conservative Greenhouse-Geisser corrected values are reported: $F(7.76, 457.8) = 2.43, p < .05$ and $F(14.38, 848.47) = 1.79, p < .05$ respectively.

A post-hoc pairwise comparison shows that there was a significant difference between the small and large screens ($p < .05$).

### 5.2.3 Overall Errors

The three-way ANOVA of errors showed that there was a significant main effect of Screen $F(2, 118) = 9.51, p < .001$, ID $F(6, 354) = 11.59, p < .001$, Amplitude $F(2, 118) = 20.37, p < .001$ and Amplitude x ID $F(8.39, 495.01) = 4.78, p < .001$. (Amplitude and Amplitude x ID values are Greenhouse-Geisser corrected.

There was no significant main effect of Screen x Amplitude, Screen x ID or Screen x Amplitude x ID.

A post-hoc pairwise comparison shows there was a significant difference between 200 and 400 px, and 200 and 600 px amplitudes ($p < .001$). There was no significant
5.3 Overshoot Analysis

The three-way ANOVA of overshoots showed that there was a significant main effect of Screen $F(2, 118) = 70.72, p < .001$, ID $F(6, 354) = 166.10, p < .001$, Screen x Amplitude $F(4, 236) = 3.2, p < .05$, Screen x ID $F(12, 708) = 3.68, p < .001$. Sphericity was violated for Amplitude and Amplitude x ID. All of the corrected values were significant so the conservative Greenhouse-Geisser corrected values of the degrees of freedom are reported. There was a significant main effect of Amplitude $F(1.40, 82.8) = 153.62$, Amplitude x ID $F(9.53, 562.13) = 4.22, p < .001$. Screen x Amplitude x ID was not significant.

A post-hoc pairwise comparison of screen reveals that the significant main effect reflects a significant difference ($p < .001$) between all levels. The significant main effect of Amplitude also reflects a significant difference ($p < .001$) between all levels. The significant main effect of ID reflects a significant difference between all levels ($p < .001$) except levels 1 and 2 ($p < .01$), levels 2 and 3 which were not significant and levels 3 and 4 ($p < .05$).

5.4 Fit of the Model

Using a linear regression model the values of $a$ and $b$ in $MT = a + bID$ were calculated for the three screens. “In regression analysis we fit the predictive model to our data”. By doing this, we can “use that model to predict values of the dependent variable from one or more independent variables” [25]. We used the Shannon formulation, in this

<table>
<thead>
<tr>
<th>screen</th>
<th>$r$</th>
<th>$R^2$</th>
<th>Intercept a (ms)</th>
<th>Slope b (ms/bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>.63</td>
<td>.396</td>
<td>187.728</td>
<td>144.511</td>
</tr>
<tr>
<td>Medium</td>
<td>.63</td>
<td>.39</td>
<td>196.709</td>
<td>133.399</td>
</tr>
<tr>
<td>Large</td>
<td>.65</td>
<td>.421</td>
<td>221.035</td>
<td>140.754</td>
</tr>
</tbody>
</table>

Table 5.2: A regression analysis using the Shannon formulation of ID showing correlation ($r$), % variation accounted for by ID ($R^2$), intercept (a) and slope (b).
case, because this is currently the accepted standard of Fitts’ Law. The results are presented in table 5.2
Chapter 6

Discussion

6.1 Screen Size

The small and large screens were significantly slower than the medium screen and not significantly different from each other (see Fig 6.1 (a)). Thus, against the prediction, there was no linear relationship between distance across the screen and movement time. The evidence from literature [40, 12, 68] and intuition suggests that the small and large screens will be different from medium sized ones. Further analysis of the data indicates that the large and small screens were different to the medium screen. The results show that movement time has allowed us to quantify some of the differences between the screens. They also show that errors and overshoots are useful in this task of comparing screen sizes. The findings of this project are thus consistent with the speculation of previous literature that small and large screens are different to desktop sized screens.

The implications are that people are so used to using a desktop that they find it difficult to adjust to different size screens. The movement time findings are combined with the undershoot/overshoot errors and overshoots in an attempt to understand the reasons for this. The increase in undershoot errors as screen size decreased (Figure 6.2 (a)) and the increase in overshoots and overshoot errors as screen size increased (Figures 6.2 (a) and 6.1 (b)) suggest that participants saw what they perceived to be a shorter distance on the small screen and so moved the mouse a shorter distance, even though the distance relative to the screen, and hence the mouse movement required, was the same. (200px = 1/4 screen width, 400px = 1/2 screen width etc.). The decrease in overshoots as screen size decreased supports this, and suggests that participants overestimated the distance on the large screen. Post-experimental interviews (see Appendix B) by the experimenter revealed that some of the participants also noticed this effect. However it
was not always attributed to the right cause. An analogy may be drawn with driving someone else’s car - the different characteristics of the clutch and brake, especially the ‘bite’ point, mean that each car is different and requires a careful approach when driving a new car that the driver is not used to. It is unknown how well or how quickly participants adjusted. That they were aware of the problem is a problem as it contravenes the goal of transparency. In fact this is a “breakdown” [72]. We recall from section 2.3.1 that a breakdown occurs when “the tool fails [and] the user becomes conscious of its properties” [74]. This is a major complication in the trend towards transparency. These results suggest that further research into how participants adapt would be fruitful, allowing us to test if this is a permanent problem or one that can be overcome with learning.

This problem is likely to be further influenced by the different screen setups that may be used in the future. Swaminathan and Sato [68] envisage a number of different ways of using large screens. Distant-contiguous, desktop-contiguous and dis-contigous setups. Swaminathan and Sato define these terms as: Distant-contiguous: “consisting of a large contiguous display placed [...] at a greater distance than a typical 17” monitor”, desktop-contiguous: a single contiguous display at a standard reading distance and dis-contigous: many different, discontinuous “display surfaces”, possibly at different distances from the user. Further work should investigate how the effects of screen size that have been found in this study work on these different setups and whether any new problems are manifested.

The limited time and resources for this project prevented us from testing more than one screen size in either direction from the desktop norm. Further work should seek to establish whether different ratios within smaller or larger screens also have an effect.
For instance, testing a desktop screen against a PDA sized screen and a wearable sized screen to see if movement time is proportional between non-desktop screens. It has only been possible to show that small screens are different from desktop screens. From the current data there is no way of telling if a PDA sized screen would be slower than the wearable.

In conclusion, the findings suggest that small and large screens are statistically different from standard desktop sized screens, possibly due to users unfamiliarity with them. Our original aim, of finding a method of comparing mobile and embedded computers to the desktop is satisfied. It has been shown that screen size is a quantifiable method of doing this, and that by application, devices with small and large screens are different from desktops.

### 6.2 Scale Independence / Movement Scale

The results are consistent with the second experimental hypothesis: as movement scale varied, the movement time also varied significantly (see Figure 6.3 (a)). This is contrary to the majority of literature \[9, 48, 27, 49, 16\], since varying movement scale is equivalent to multiplying amplitude and tolerance by the same factor. However, it is in agreement with \[30\] and \[28\]. We conclude that the Fitts’ Law task is indeed not scale independent (see Section 6.5). The implications for design are that objects are the edges of the screen do not have to be increased at the same scale as the increase in amplitude. It has not been possible to assertain whether targets at greater (or smaller) amplitudes should be increased at a smaller or larger factor compared to the amplitude
increase, this is an area that requires further investigation.

An examination of Figure 6.3 (a) suggests that a deeper investigation of the affect of movement scale is necessary. The relationship of 400 to 600 px is intuitive, the movement time increasing with amplitude. However the change for 200px appears counter-intuitive. We would expect movement time to decrease as the amplitude decreases. Cross referencing with Figure 6.3 (b) casts some light on the anomaly. It is shown that the error rate increase was statistically significant from 400 to 200px, but statistically constant from 400 to 600px. Participants were going slower and making more errors at the 200px amplitude than at either of the other two amplitudes. We must ask what caused these increases at 200px. There is little literature relating to this phenomenon. We conjecture that a third factor affects accuracy at 200px and that this increase in errors caused participants to slow down. Suppose that the pointing task became a ballistic task due to the short duration and amplitude of movement at 200px. Ballistic movements are defined by Gan and Hoffmann as “a rapid voluntary movement which is motor programmed and in which visual feedback for path correction is not possible” [28]. The lack of visual feedback due to the ballistic nature of the task at 200 px explains the greater number of errors. MacKenzie also states that “[Fitts’] tapping tasks were highly ballistic” [47]. The context of this quote indicates that he is also referring to ballistic as a task with no feedback (although the feedback he is primarily referring to is haptic). This seems a plausible explanation, since the 200 px task would have invited a short and fast movement, which would be ballistic. The lack of visual feedback and hence adjustment would cause additional errors, in turn leading to participants slowing down in order to be more accurate.

This is certainly an area that would benefit from further research. If mouse, or indeed
stylus, movements become ballistic at a certain amplitude, the design implications would be far reaching and the limiting amplitude should be ascertained. We have already seen that visual feedback is essential in direct manipulation interfaces.

6.3 Movement Difficulty / ID

Participants’ movement time increased significantly as ID or movement difficulty, increased (see Figure 6.4 (a)). In other words, smaller targets took longer to point to. The findings show that pointing was also more erroneous for smaller targets (higher ID, see Figure 6.4 (b)). Investigation of the interaction effect of ID and Screen shows that there was a significant increase in time on the large screen, contrary to the overall trend between ID levels 5 and 6 at 200px and 400px, and levels 2 and 3 at 600px. There is the possibility that this could be anomalous data due to the small number of data points. We conjecture however, that there is a best target size (ie. ID) for each amplitude on the large screen. This is supported by the significant interaction between Amplitude and ID which means that the effect of amplitude was different depending on the Index of Difficulty. This has important connotations for graphical object design.

Further investigation of this result is required to see if there is a way of modelling the optimal size for objects on a large screen. This would allow us to size objects on large screens in an efficient way, balancing the amount of data being presented and the speed of the user in selecting the objects. This could reduce the possible problems that arise from not having a scale independent task. Having to increase targets by a factor of 3 as the amplitude increases by a factor of 2, in order to maintain movement time would have negative connotations. ie. we would have less screen space to use at the edges because objects here would have to be much larger than those in the middle of the screen. If, however, there was an optimal size that required a much smaller increase in size this would be of great importance. Of course, scale dependency may imply, in the opposite direction that we do not have to increase objects at the same rate as amplitude increases and this would be of great benefit. Of course, the implications also need to be considered for small screens and how the conclusions presented affect smaller objects and amplitudes.

6.4 Methodology

By varying movement scale and movement difficulty we have kept the effects of these two variables separate. Guiard’s description [30] of these variables is an apt one, since
it was found that as movement difficulty increased (Amplitude constant, ID increased, tolerance decreased) the pointing task became slower and more erroneous, ie. more difficult. The effects of scale are more complicated and have already been discussed. In the data from this experiment, the Shannon index of difficulty only accounts for 40% of the variation in Movement Time in the equation $MT = a + bID$. This supports the previous suggestion [28] that Amplitude and ID should be varied as fully independent variables and not Amplitude and tolerance.

### 6.5 Fitts’ Law

There are a number of problems that have arisen from our investigation into Fitts’ Law. These are: Fitts’ Law is scale independent, it is one dimensional and it models repetitive movements. These are all contrary to the pointing task on a computer which is not scale independent, is two dimensional and is more often than not, a discrete movement. We have already discussed the scale independence of Fitts’ Law. Here we discuss the dimensional restraints and the movement type of the Fitts’ Law task.

It has been shown that Fitts’ Law is restrictively one dimensional [48] (see Section 2.7.3). In fact the bivariate problem breaks down into two areas, demonstrated in Figure 6.5. The first is caused by varying approach angle, demonstrated in Figure 6.5(a) at 0° and 90°. The second, in Figure 6.5(b) comes from rotating the target.

A review of the literature shows that the first has so far been overlooked by those investigating or using at Fitts’ Law. Further investigation is required to discover whether
pointing in different directions has similar properties. The effects of (b) are investigated by Accot and Zhai, amongst others. Although their results show good correlation, their model suffers from the scale independence assumption. This means that further work is necessary to find a two dimensional, scale dependent model.

It has not been possible, due to the time restraints of this project to discuss the applications to objects on different sized screens. The fact that the pointing task is not scale independent has important implications for such objects. Since objects do not need to be scaled up by the same constant as the amplitude, investigation is required in order to find the fastest pointing ratios of distance to amplitude. There has been some investigation into icons on small screens and the question ‘How small can we go?’ needs to be addressed on both small and large screens alike. Due to its scale independence, Fitts’ Law is not a relevant tool in this debate, suggesting that further empirical investigation and a new tool is required.

6.6 A Seamless Environment

“The vision of ubiquitous computing [...] holds the promise of yet another interaction paradigm shift” [22]. New models, such as the Bulletin board, are already emerging. As they do, it will be important to develop techniques of comparing them to each other and to the desktop. Without empirical data, there will be no way of influencing users and manufacturers as to which are the best solutions for them.
It will also be essential to ensure that users can move seamlessly from operating on a small screen device to a large screen one and back again. This is the situation that Kostakos and O’Neill investigate in [41]. It can be seen from the results of this study that this is likely to cause problems, especially while multi-device unaware and inexperienced users exist.

Section 2.2 noted that similar abilities were required when there was a mix of mainframe terminals and PCs. User interface guidelines were required to align different types of access point to a standard, enabling users to use both with ease. This was facilitated by the Common User Access (CUA) Guidelines. It is entirely foreseeable that for the mobile, desktop and embedded environments to be used seamlessly they must be exactly that - seamless. To do this would require further research and the development of Multi-Environment CUA Guidelines to gain consistency between the myriad of personal, mobile and embedded computers that are becoming increasingly available. An example of this type of research exists in [41]. Not only will this be required for inputting as in [41] but also for the presentation of data so that users can migrate from one computer to another without substantial problems.
Chapter 7

Conclusions

The original problem that we set out to investigate was whether we can simply scale the desktop up on to large screens and shrink it down on to small screens. The data collected suggests that this is not possible. Even with the standard direct manipulation, desktop interface we have shown that people use small and large screens differently to how they use desktop sized screens. Coupled with the fact that the Fitts’ pointing task is not scale independent, we conclude that scaling the desktop to smaller and larger screens is not the correct approach to mobile and pervasive HCI, especially as it reduces the, already poor, transparency of the desktop.

After analysis of the task tested, it was found that the task was, in fact a two dimensional task. The use of square targets means that, according to Accot and Zhai [9] there would have been some effect of target height. The analysis of the statistics also failed to take throughput and effective width into account. These developments in Fitts’ Law do not suggest that the results in this project are flawed. Effective width takes speed and errors into account but is not applied to experiments attaining an error rate of 4% [46]. Since the overall error rate in our experiment was 4% we can safely say that this is not a major omission. Throughput (TP) is a new performance measure [50] calculated using $TP = ID/MT$. This formulation has no effect on the problems caused by varying amplitude and tolerance, suggesting that it does not insert any doubt into our assertions of scale dependence. However, further investigations should take these developments into account.

Many lessons have been learned from my investigation into psychological experimentation. It is noted that investigations in HCI do not always choose the most careful experimental designs. I should not have re-used 3 of the participants from the pilot experiment. Fortunately, this should not have a large effect on the results, since the
number of participants in the experiment was high. Large numbers of IVs and levels and small numbers of participants are problems that have arisen in both this project and some of the preceding literature.

It has been argued that screen size may be used to characterise different computers and provides a way of comparing differences between them. Experimentation has shown that there is a significant difference between desktop and non desktop sized screens. It is not outrageous to attribute this to an over familiarity with the desktop and inexperience of using different sized displays. Furthermore it has been shown that the apparently robust Fitts’ Law model is not as robust as was first thought. The irregularities between the model and the actual problem lead to the suggestion that Fitts’ Law has served its purpose. In a world of mobile and embedded computers, where amplitude and index of difficulty are more applicable than ever before, a model that can be used to influence design based on hard empirical evidence is required.

Questions such as ‘How small can we make objects’ need to be answered and Fitts’ Law has been shown not to be up to the task. The reasoning for this comes not only from our results, but also from an investigation of the theory behind Fitts’ Law. Drawing analogy from Shannon’s work on communication systems to the two dimensional, multi-environment, multi-device human computer interaction of the future is riddled with problems. As the environments and devices continue to change and grow less like the desktop, the predictive and descriptive capabilities of Fitts’ Law can only continue to lose touch with what is happening in the real world. If interaction is to be standardised between multiple environments and devices with some sort of multi-environment CUA, a model that can be applied to, and has predictive power in, all of the environments and devices is essential.
References


[64] Albrecht Schmidt, Hans-W. Gellersen, Michael Beigl, and Ortwin Thate. Developing user interfaces for wearable computers - don’t stop to point and click.


Appendix A

Allocation of Participants

<table>
<thead>
<tr>
<th>1st Screen</th>
<th>2nd Screen</th>
<th>3rd Screen</th>
<th>Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Medium</td>
<td>Large</td>
<td>1, 7, 15, 22, 28, 31, 42, 47, 51, 58</td>
</tr>
<tr>
<td>Small</td>
<td>Large</td>
<td>Medium</td>
<td>4, 8, 17, 23, 25, 35, 38, 44, 50, 56</td>
</tr>
<tr>
<td>Medium</td>
<td>Small</td>
<td>Large</td>
<td>5, 9, 13, 20, 26, 33, 39, 43, 53, 60</td>
</tr>
<tr>
<td>Medium</td>
<td>Large</td>
<td>Small</td>
<td>3, 11, 18, 24, 30, 36, 41, 45, 49, 55</td>
</tr>
<tr>
<td>Large</td>
<td>Medium</td>
<td>Small</td>
<td>2, 12, 16, 21, 27, 32, 37, 48, 54, 61</td>
</tr>
<tr>
<td>Large</td>
<td>Small</td>
<td>Medium</td>
<td>6, 10, 14, 19, 29, 34, 40, 46, 52, 59</td>
</tr>
</tbody>
</table>

Table A.1: Allocation of participants to counterbalanced screen ordering
Appendix B

Selected participant comments

5 Commented that the small screen was better because you could see the whole screen in your field of vision
7 Asked if the reason was to look at icon sizes on screens, also mentioned that people are used to using the medium sized display
12 Commented going from the large to the medium screen that it “felt different” and asked if it was the same mouse
13 Commented that the smallest screen was the best because it was “crisper”, commented when using the large screen that it felt like a different mouse
14 Preferred the normal screen because that was what she was used to
25 Said she wanted to sit further back when sitting down at the large screen
26 Immediately sat back from the large screen and wanted to sit further back
28 Commented, “I like this (the medium) screen” wrt to reading the text, also “Don’t like this (the large) screen, I hate it”
30 Thought that the sensitivity had been put up on the large screen. Said it was “ultra sensitive”
36 Commented that the mouse moved further than you expected it to on the large screen than the small. Small screen too small, large screen headache, had to move head. Medium just right. Used to medium.
39 Commented that the mouse was more sensitive on the large screen
40 Found it fatiguing.
41 “woah! This is odd!” of the large screen
43 Smaller boxes on the medium screen and normal boxes on small screen meant more concentration was needed, big screens caused carelessness. Large gave perception that the mouse sensitivity was greater. And found that he adjusted. Got better on large screen for this reason but stayed constant or depreciated slightly due to fatigue on other screens
Found using the mouse for length of time fatiguing on wrist. Found large harder than small, moved eyes not head. Set up the mouse to act the same on different size as to 17. Every time he increased size he found himself overshooting. Even small to medium.

Commented that the mouse sensitivity was much reduced on the small screen and increased on the large screen. Thought investigation was in to mouse sensitivity, screen size. Mouse was over sensitive on the large and under sensitive on the small. Medium stopped mouse from jumping across the screen which it did on the big screen. Consciously noticed that he was over shooting and adjusted.

Pointer moved faster on the large screen. More used to the medium, found the small to small, but would prefer to use the large for working. Commented that it took a little time for the brain to calibrate to the mouse on the large screen.
Appendix C

Source Code

The source code for the testing program can be found at:
http://students.bath.ac.uk/ma1gjb/fitts/code
Appendix D

Raw Data

D.1 Original Data Files

Raw data was output into a separate file for each participant. Each line corresponded to one condition. The condition properties are printed first (condition number; amplitude; target size; screen; resolution width; resolution height), then the time for each trial, followed by the error totals for each type of error (under; over; high; low; other; wrong button; overshoots). The data collected for participant 30 is shown below. The complete set of data files for all of the participants are available at:

http://students.bath.ac.uk/ma1gjb/fitts/data/raw

2,200,14,0.7,1,800,600,0,881,661,661,731,1302,892,1251,1042,831,0,0,1,0,0,0,7
12,400,60,3.0,1,800,600,601,491,620,611,451,461,440,541,531,0,0,0,0,0,0,2
7,400,20,1.0,1,800,600,972,691,811,781,871,701,802,811,551,600,0,0,0,0,0,2
19,600,90,4.5,1,800,600,661,511,611,481,531,520,451,521,450,0,0,0,0,0,0,0
8,400,24,1.2,1,800,600,0,972,881,801,671,931,862,981,681,721,1,1,0,0,0,0,4
0,200,10,0.5,1,800,600,861,841,0,1041,1032,761,1222,1001,991,882,0,1,0,0,0,0,3
1,200,12,0.6,1,800,600,871,902,871,791,952,881,911,831,581,851,0,0,0,0,0,0,4
4,200,20,1.0,1,800,600,741,480,671,1222,661,871,791,671,831,0,0,0,0,0,0,4
13,400,80,4.0,1,800,600,701,471,500,481,421,400,461,471,500,451,0,0,0,0,0,0,1
10,400,32,1.6,1,800,600,711,751,570,511,701,801,711,591,701,821,0,0,0,0,0,0,4
18,600,60,3.0,1,800,600,821,490,711,732,540,541,661,601,681,521,0,0,0,0,0,0,4
11,400,40,2.0,1,800,600,821,491,821,511,541,600,451,631,691,521,0,0,0,0,0,0,1
14,600,30,1.5,1,800,600,661,731,771,751,651,561,761,811,691,631,0,0,0,0,0,0,0
15,600,36,1.7999,1,800,600,811,661,541,560,661,591,591,0,1261,1112,0,1,0,0,0,0,4
20,600,120,6.0,1,800,600,471,1402,661,631,661,530,551,501,571,470,0,0,0,0,0,0,0
D.2 Collated Raw Data

The raw data for all participants was then collated in to a master spreadsheet available at:
http://students.bath.ac.uk/ma1gjb/fitts/data/alldata.xls
Appendix E

Compiled Data for Analysis

The mean times and errors for each condition were then calculated as described in 5. The data for participant 30 is shown below. The compiled data for analysis for all participants is available at:
http://students.bath.ac.uk/ma1gjb/fitts/data/sdadjusted.xls
<table>
<thead>
<tr>
<th>participant</th>
<th>age</th>
<th>gender</th>
<th>height</th>
<th>weight</th>
<th>smoking</th>
<th>alcohol</th>
<th>exercise</th>
<th>adjusted score</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>2</td>
<td>male</td>
<td>200</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>824.56</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>12</td>
<td>male</td>
<td>400</td>
<td>60</td>
<td>1</td>
<td>2</td>
<td>491.7</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>7</td>
<td>male</td>
<td>400</td>
<td>20</td>
<td>1</td>
<td>2</td>
<td>699.1</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>19</td>
<td>male</td>
<td>600</td>
<td>90</td>
<td>1</td>
<td>0</td>
<td>461.78</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>8</td>
<td>male</td>
<td>400</td>
<td>24</td>
<td>1</td>
<td>4</td>
<td>753.33</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>male</td>
<td>200</td>
<td>10</td>
<td>1</td>
<td>3</td>
<td>861.11</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>1</td>
<td>male</td>
<td>200</td>
<td>12</td>
<td>1</td>
<td>4</td>
<td>759.1</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>4</td>
<td>male</td>
<td>200</td>
<td>20</td>
<td>1</td>
<td>4</td>
<td>630.89</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>13</td>
<td>male</td>
<td>400</td>
<td>80</td>
<td>1</td>
<td>1</td>
<td>411.67</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>10</td>
<td>male</td>
<td>400</td>
<td>32</td>
<td>1</td>
<td>4</td>
<td>604.8</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>18</td>
<td>male</td>
<td>600</td>
<td>60</td>
<td>1</td>
<td>4</td>
<td>577.8</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>11</td>
<td>male</td>
<td>400</td>
<td>40</td>
<td>1</td>
<td>1</td>
<td>555.8</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>14</td>
<td>male</td>
<td>600</td>
<td>30</td>
<td>1</td>
<td>2</td>
<td>638.9</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>15</td>
<td>male</td>
<td>600</td>
<td>36</td>
<td>1</td>
<td>4</td>
<td>552</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>20</td>
<td>male</td>
<td>600</td>
<td>120</td>
<td>1</td>
<td>0</td>
<td>508.56</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>9</td>
<td>male</td>
<td>400</td>
<td>28</td>
<td>1</td>
<td>2</td>
<td>639.9</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>17</td>
<td>male</td>
<td>600</td>
<td>48</td>
<td>1</td>
<td>2</td>
<td>587.56</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>6</td>
<td>male</td>
<td>200</td>
<td>40</td>
<td>1</td>
<td>1</td>
<td>412.78</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>16</td>
<td>male</td>
<td>600</td>
<td>41</td>
<td>1</td>
<td>4</td>
<td>563.38</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>3</td>
<td>male</td>
<td>200</td>
<td>16</td>
<td>1</td>
<td>4</td>
<td>649.78</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>5</td>
<td>male</td>
<td>200</td>
<td>30</td>
<td>1</td>
<td>1</td>
<td>456.6</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>20</td>
<td>male</td>
<td>600</td>
<td>120</td>
<td>2</td>
<td>0</td>
<td>508.56</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>18</td>
<td>male</td>
<td>600</td>
<td>60</td>
<td>2</td>
<td>3</td>
<td>603.9</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>12</td>
<td>male</td>
<td>400</td>
<td>60</td>
<td>2</td>
<td>3</td>
<td>551.7</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>male</td>
<td>200</td>
<td>10</td>
<td>2</td>
<td>8</td>
<td>934.78</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>17</td>
<td>male</td>
<td>600</td>
<td>48</td>
<td>2</td>
<td>3</td>
<td>605.33</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>15</td>
<td>male</td>
<td>600</td>
<td>36</td>
<td>2</td>
<td>6</td>
<td>668.38</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>2</td>
<td>male</td>
<td>200</td>
<td>14</td>
<td>2</td>
<td>7</td>
<td>878.3</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>11</td>
<td>male</td>
<td>400</td>
<td>40</td>
<td>2</td>
<td>4</td>
<td>671</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>16</td>
<td>male</td>
<td>600</td>
<td>41</td>
<td>2</td>
<td>0</td>
<td>642</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>1</td>
<td>male</td>
<td>200</td>
<td>12</td>
<td>2</td>
<td>5</td>
<td>659.78</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>19</td>
<td>male</td>
<td>600</td>
<td>90</td>
<td>2</td>
<td>0</td>
<td>494.7</td>
<td></td>
</tr>
</tbody>
</table>

Table E.1: Adjusted data for participant 30
Appendix F

SPSS Outputs

Statistics were calculated using SPSS 10 for Windows. Selected, relevent statistical outputs, from the program, are presented below. The comprehensive collection, including SPSS input files, is available at:
http://students.bath.ac.uk/ma1gjb/fitts/spss