Conferences in Research and Practice in Information Technology

Volume 139

User Interfaces 2013

Australian Computer Science Communications, Volume 35, Number 5
USER INTERFACES 2013

Proceedings of the Fourteenth Australasian User Interface Conference (AUIC 2013), Melbourne, Australia, Adelaide, Australia, 29 January – 1 February 2013

Ross T. Smith and Burkhard C. Wünsche, Eds.

Volume 139 in the Conferences in Research and Practice in Information Technology Series. Published by the Australian Computer Society Inc.

Published in association with the ACM Digital Library.
Table of Contents

Proceedings of the Fourteenth Australasian User Interface Conference (AUIC 2013), Adelaide, Australia, 29 January – 1 February 2013

Preface .................................................................................................................................................. vii
Programme Committee ...................................................................................................................... viii
Organising Committee ....................................................................................................................... ix
Welcome from the Organising Committee ......................................................................................... x
CORE - Computing Research & Education ........................................................................................ xi
ACSW Conferences and the Australian Computer Science Communications ........................................... xii
ACSW and AUIC 2013 Sponsors ............................................................................................................ xiv

Contributed Papers

Tangible Agile Mapping: Ad-hoc Tangible User Interaction Definition .................................................. 3
James A. Walsh, Stewart von Itzstein and Bruce H. Thomas

vsInk Integrating Digital Ink with Program Code in Visual Studio .................................................... 13
Craig J. Sutherland and Beryl Plimmer

Supporting Informed Decision - Making under Uncertainty and Risk through Interactive Visualisation 23
Mohammad Daradkeh, Clare Churcher and Alan McKinnon

Metadata Manipulation Interface Design ............................................................................................. 33
Stijn Dekeyser and Richard Watson

Understanding the Management and Need For Awareness of Temporal Information in Email ............ 43
Nikash Singh, Martin Tomitsch and Mary Lou Maher

An Online Social-Networking Enabled Telehealth System for Seniors A Case Study ....................... 53
Jaspaljeet Singh Dhillon, Burkhard C. Wünsche and Christof Lutteroth

Validating Constraint Driven Design Techniques in Spatial Augmented Reality ................................ 63
Andrew Irlitti and Stewart von Itzstein

Music Education using Augmented Reality with a Head Mounted Display ........................................ 73
Jonathan Chow, Huoyang Feng, Robert Amor and Burkhard C. Wünsche

A Tale of Two Studies .......................................................................................................................... 81
Judy Bowen, Steve Reeves and Andrea Schweer

Making 3D Work: A Classification of Visual Depth Cues, 3D Display Technologies and Their Applications ................................................................. 91
Mostafa Mehrabi, Edward M. Peek, Burkhard C. Wünsche and Christof Lutteroth

An Investigation of Usability Issues in AJAX based Web Sites .......................................................... 101
Chris Pilgrim

Determining the Relative Benefits of Pairing Virtual Reality Displays with Applications .................. 111
Edward M. Peek, Burkhard C. Wünsche and Christof Lutteroth
Contributed Posters

An Ethnographic Study of a High Cognitive Load Driving Environment

Robert Wellington, Stefan Marks

Experimental Study of Steer-by-Wire and Response Curves in a Simulated High Speed Vehicle

Stefan Marks, Robert Wellington

3D Object Surface Tracking Using Partial Shape Templates Trained from a Depth Camera for Spatial Augmented Reality Environments

Kazuna Tsuboi, Yuji Oyamada, Maki Sugimoto, Hideo Saito

My Personal Trainer - An iPhone Application for Exercise Monitoring and Analysis

Christopher R. Greeff, Joe Yang, Bruce MacDonald, Burkhard C. Wünsche

Interactive vs. Static Location-based Advertisements

Moniek Raijmakers, Suleman Shahid, Omar Mubin

Temporal Evaluation of Aesthetics of User Interfaces as one Component of User Experience

Marlene Vogel

Author Index
Preface

It is our great pleasure to welcome you to the 14th Australasian User Interface Conference (AUIC), held in Adelaide, Australia, January 29th to February 1st 2013, at the University of South Australia. AUIC is one of 11 co-located conferences that make up the annual Australasian Computer Science Week.

AUIC provides an opportunity for researchers in the areas of User Interfaces, HCI, CSCW, and pervasive computing to present and discuss their latest research, to meet with colleagues and other computer scientists, and to strengthen the community and explore new projects, technologies and collaborations.

This year we have received a diverse range of submission from all over the world. Out of 31 submitted papers, 12 papers were selected for full paper presentations and 6 were selected for posters. The breadth and quality of the papers reflect the dynamic and innovative research in the field and we are excited to see the international support.

Accepted papers were rigorously reviewed by the community to ensure high quality publications. This year we are excited to announce that all AUIC publications will now be indexed by Scopus (Elsevier) to help increase their exposure and citation rates.

We offer our sincere thanks to the people who made this years conference possible: the authors and participants, the program committee members and reviewers, the ACSW organizers, Scopus and the publisher CRPIT (Conference in Research and Practice in Information Technology).

Ross T. Smith
University of South Australia

Burkhard Wünsche
University of Auckland

AUIC 2013 Programme Chairs
January 2013
Programme Committee

Chairs
Ross T. Smith, University of South Australia, Australia
Burkhard C. Wünsche, University of Auckland, New Zealand

Web Chair
Stefan Marks, AUT University, New Zealand

Members
Mark Apperley, University of Waikato, New Zealand
Robert Amor, University of Auckland, New Zealand
Mark Billinghurst, HITLab, New Zealand
Rachel Blagojevic, University of Auckland, New Zealand
Paul Calder, Flinders University, Australia, Australia
David Chen, Griffith University, Australia
Sally Jo Cunningham, University of Waikato, New Zealand
John Grundy, Swinburne University of Technology, Australia
Stewart Von Itzstein, University of South Australia, Australia
Christof Lutteroth, University of Auckland, New Zealand
Stuart Marshall, Victoria University of Wellington, New Zealand
Masood Masoodian, University of Waikato, New Zealand
Christian Müller-Tomfelde, CSIRO, Australia
Beryl Plimmer, University of Auckland, New Zealand
Gerald Weber, University of Auckland, New Zealand
Burkhard C. Wünsche, University of Auckland, New Zealand
Organising Committee

Chair
Dr. Ivan Lee

Finance Chair
Dr. Wolfgang Mayer

Publication Chair
Dr. Raymond Choo

Local Arrangement Chair
Dr. Grant Wigley

Registration Chair
Dr. Jinhai Cai
On behalf of the Organising Committee, it is our pleasure to welcome you to Adelaide and to the 2013 Australasian Computer Science Week (ACSW 2013). Adelaide is the capital city of South Australia, and it is one of the most liveable cities in the world. ACSW 2013 will be hosted in the City West Campus of University of South Australia (UniSA), which is situated at the north-west corner of the Adelaide city centre.

ACSW is the premier event for Computer Science researchers in Australasia. ACSW 2013 consists of conferences covering a wide range of topics in Computer Science and related area, including:

- Australasian Computer Science Conference (ACSC) (Chaired by Bruce Thomas)
- Australasian Database Conference (ADC) (Chaired by Hua Wang and Rui Zhang)
- Australasian Computing Education Conference (ACE) (Chaired by Angela Carbone and Jacqueline Whalley)
- Australasian Information Security Conference (AISC) (Chaired by Clark Thomborson and Udaya Pampalli)
- Australasian User Interface Conference (AUIC) (Chaired by Ross T. Smith and Burkhard C. Wünsche)
- Computing: Australasian Theory Symposium (CATS) (Chaired by Tony Wirth)
- Australasian Symposium on Parallel and Distributed Computing (AusPDC) (Chaired by Bahman Javadi and Saurabh Kumar Garg)
- Australasian Workshop on Health Informatics and Knowledge Management (HIKM) (Chaired by Kathleen Gray and Andy Koronios)
- Asia-Pacific Conference on Conceptual Modelling (APCCM) (Chaired by Flavio Ferrarotti and Georg Grossmann)
- Australasian Web Conference (AWC2013) (Chaired by Helen Ashman, Michael Sheng and Andrew Trotman)

In addition to the technical program, we also put together social activities for further interactions among our participants. A welcome reception will be held at Rockford Hotel’s Rooftop Pool area, to enjoy the fresh air and panoramic views of the cityscape during Adelaide’s dry summer season. The conference banquet will be held in Adelaide Convention Centre’s Panorama Suite, to experience an expansive view of Adelaide’s serene riverside parklands through the suite’s seamless floor to ceiling windows.

Organising a conference is an enormous amount of work even with many hands and a very smooth cooperation, and this year has been no exception. We would like to share with you our gratitude towards all members of the organising committee for their dedication to the success of ACSW 2013. Working like one person for a common goal in the demanding task of ACSW organisation made us proud that we got involved in this effort. We also thank all conference co-chairs and reviewers, for putting together conference programs which is the heart of ACSW. Special thanks goes to Alex Potanin, who shared valuable experiences in organising ACSW and provided endless help as the steering committee chair. We’d also like to thank Elyse Perin from UniSA, for her true dedication and tireless work in conference registration and event organisation. Last, but not least, we would like to thank all speakers and attendees, and we look forward to several stimulating discussions.

We hope your stay here will be both rewarding and memorable.

Ivan Lee
School of Information Technology & Mathematical Sciences
ACSW2013 General Chair
January, 2013
CORE welcomes all delegates to ACSW2013 in Adelaide. CORE, the peak body representing academic computer science in Australia and New Zealand, is responsible for the annual ACSW series of meetings, which are a unique opportunity for our community to network and to discuss research and topics of mutual interest. The original component conferences - ACSC, ADC, and CATS, which formed the basis of ACSW in the mid 1990s - now share this week with eight other events - ACE, AISC, AUIC, AusPDC, HIKM, ACDC, APCCM and AWC which build on the diversity of the Australasian computing community.

In 2013, we have again chosen to feature a small number of keynote speakers from across the discipline: Riccardo Bellazzi (HIKM), and Divyakant Agrawal (ADC), Maki Sugimoto (AUIC), and Wen Gao. I thank them for their contributions to ACSW2013. I also thank invited speakers in some of the individual conferences, and the CORE award winner Michael Sheng (CORE Chris Wallace Award). The efforts of the conference chairs and their program committees have led to strong programs in all the conferences, thanks very much for all your efforts. Thanks are particularly due to Ivan Lee and his colleagues for organising what promises to be a strong event.

The past year has been turbulent for our disciplines. ERA2012 included conferences as we had pushed for, but as a peer review discipline. This turned out to be good for our disciplines, with many more Universities being assessed and an overall improvement in the visibility of research in our disciplines. The next step must be to improve our relative success rates in ARC grant schemes, the most likely hypothesis for our low rates of success is how harshly we assess each others’ proposals, a phenomenon which demonstrably occurs in the US NFS. As a US Head of Dept explained to me, “in CS we circle the wagons and shoot within”.

Beyond research issues, in 2013 CORE will also need to focus on education issues, including in Schools. The likelihood that the future will have less computers is small, yet where are the numbers of students we need? In the US there has been massive growth in undergraduate CS numbers of 25 to 40% in many places, which we should aim to replicate. ACSW will feature a joint CORE, ACDICT, NICTA and ACS discussion on ICT Skills, which will inform our future directions.

CORE’s existence is due to the support of the member departments in Australia and New Zealand, and I thank them for their ongoing contributions, in commitment and in financial support. Finally, I am grateful to all those who gave their time to CORE in 2012; in particular, I thank Alex Potanin, Alan Fekete, Aditya Ghose, Justin Zobel, John Grundy, and those of you who contribute to the discussions on the CORE mailing lists. There are three main lists: csprofs, cshods and members. You are all eligible for the members list if your department is a member. Please do sign up via http://lists.core.edu.au/mailman/listinfo - we try to keep the volume low but relevance high in the mailing lists.

I am standing down as President at this ACSW. I have enjoyed the role, and am pleased to have had some positive impact on ERA2012 during my time. Thank you all for the opportunity to represent you for the last 3 years.

Tom Gedeon
President, CORE
January, 2013
The Australasian Computer Science Week of conferences has been running in some form continuously since 1978. This makes it one of the longest running conferences in computer science. The proceedings of the week have been published as the *Australian Computer Science Communications* since 1979 (with the 1978 proceedings often referred to as *Volume 0*). Thus the sequence number of the Australasian Computer Science Conference is always one greater than the volume of the Communications. Below is a list of the conferences, their locations and hosts.

**2014. Volume 36. Host and Venue - AUT University, Auckland, New Zealand.**

**2013. Volume 35. Host and Venue - University of South Australia, Adelaide, SA.**

**2012. Volume 34. Host and Venue - RMIT University, Melbourne, VIC.**

**2011. Volume 33. Host and Venue - Curtin University of Technology, Perth, WA.**

**2010. Volume 32. Host and Venue - Queensland University of Technology, Brisbane, QLD.**

**2009. Volume 31. Host and Venue - Victoria University, Wellington, New Zealand.**

**2008. Volume 30. Host and Venue - University of Wollongong, NSW.**

**2007. Volume 29. Host and Venue - University of Ballarat, VIC. First running of HDKM.**

**2006. Volume 28. Host and Venue - University of Tasmania, TAS.**

**2005. Volume 27. Host - University of Newcastle, NSW. APBC held separately from 2005.**


**2002. Volume 24. Host and Venue - Monash University, Melbourne, VIC.**

**2001. Volume 23. Hosts - Bond University and Griffith University (Gold Coast). Venue - Gold Coast, QLD.**


**1999. Volume 21. Host and Venue - University of Auckland, New Zealand.**

**1998. Volume 20. Hosts - University of Western Australia, Murdoch University, Edith Cowan University and Curtin University. Venue - Perth, WA.**


**1995. Volume 17. Hosts - Flinders University, University of Adelaide and University of South Australia. Venue - Glenelg, SA.**


**1992. Volume 14. Host and Venue - University of Tasmania, TAS. (ADC held separately at La Trobe University).**


**1990. Volume 12. Host and Venue - Monash University, Melbourne, VIC. Joined by Database and Information Systems Conference which in 1992 became ADC (which stayed with ACSW) and ACIS (which now operates independently).**

**1989. Volume 11. Host and Venue - University of Wollongong, NSW.**


**1987. Volume 9. Host and Venue - Deakin University, VIC.**

**1986. Volume 8. Host and Venue - Australian National University, Canberra, ACT.**


**1983. Volume 5. Host and Venue - University of Sydney, NSW.**

**1982. Volume 4. Host and Venue - University of Western Australia, WA.**

**1981. Volume 3. Host and Venue - University of Queensland, QLD.**

**1980. Volume 2. Host and Venue - Australian National University, Canberra, ACT.**

**1979. Volume 1. Host and Venue - University of Tasmania, TAS.**

**1978. Volume 0. Host and Venue - University of New South Wales, NSW.**
Conference Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Conference Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACDC</td>
<td>Australasian Computing Doctoral Consortium</td>
</tr>
<tr>
<td>ACE</td>
<td>Australasian Computer Education Conference</td>
</tr>
<tr>
<td>ACSC</td>
<td>Australasian Computer Science Conference</td>
</tr>
<tr>
<td>ACSW</td>
<td>Australasian Computer Science Week</td>
</tr>
<tr>
<td>ADC</td>
<td>Australasian Database Conference</td>
</tr>
<tr>
<td>AISC</td>
<td>Australasian Information Security Conference</td>
</tr>
<tr>
<td>APCCM</td>
<td>Asia-Pacific Conference on Conceptual Modelling</td>
</tr>
<tr>
<td>AUIC</td>
<td>Australasian User Interface Conference</td>
</tr>
<tr>
<td>AusPDC</td>
<td>Australasian Symposium on Parallel and Distributed Computing (replaces AusGrid)</td>
</tr>
<tr>
<td>AWC</td>
<td>Australasian Web Conference</td>
</tr>
<tr>
<td>CATS</td>
<td>Computing: Australasian Theory Symposium</td>
</tr>
<tr>
<td>HIKM</td>
<td>Australasian Workshop on Health Informatics and Knowledge Management</td>
</tr>
</tbody>
</table>

Note that various name changes have occurred, which have been indicated in the Conference Acronyms sections in respective CRPIT volumes.
We wish to thank the following sponsors for their contribution towards this conference.

CORE - Computing Research and Education, www.core.edu.au

Australian Computer Society, www.acs.org.au

AUT University, www.aut.ac.nz

University of South Australia, www.unisa.edu.au/
CONTRIBUTED PAPERS
Tangible Agile Mapping: Ad-hoc Tangible User Interaction Definition

James A. Walsh, Stewart von Itzstein and Bruce H. Thomas
School of Computer and Information Science
University of South Australia
Mawson Lakes Boulevard, Mawson Lakes, South Australia, 5095
james.walsh@setoreaustralia.com, stewart.vonitzstein@unisa.edu.au, 
bruce.thomas@unisa.edu.au

Abstract
People naturally externalize mental systems through physical objects to leverage their spatial intelligence. The advent of tangible user interfaces has allowed human computer interaction to utilize these skills. However, current systems must be written from scratch and designed for a specific purpose, thus meaning end users cannot extend or repurpose the system. This paper presents Tangible Agile Mapping, our architecture to address this problem by allowing tangible systems to be defined ad-hoc. Our architecture addresses the tangible ad-hoc definition of objects, properties and rules to support tangible interactions. This paper also describes Spatial Augmented Reality TAM as an implementation of this architecture that utilizes a projector-camera setup combined with gesture-based navigation to allow users to create tangible systems from scratch. Results of a user study show that the architecture and our implementation are effective in allowing users to develop tangible systems, even for users with little computing or tangible experience.

Keywords: Tangible user interfaces, programming by demonstration, organic users interfaces, proxemic interactions, authoring by interaction.

1 Introduction
All of us utilize the physical affordances of everyday objects to convey additional information about some mental ‘cognitive system’, as a means of reducing errors compared to when we simulate the system mentally (Myers, 1992). If a teacher were explaining the reactions between different chemicals, they would pick up different objects to represent elements, moving them closer together to indicate a reaction, changing the chemicals’ state. Despite the simple rules for these interactions, complex configurations can easily be created. Mentally tracking these roles and states, however, introduces a cognitive overhead for both primary and collaborative users. Tangible User Interfaces (TUIs) help this problem by designing a system that utilizes the affordances of physical objects. However, despite research into TUIs existing for some years, the adoption of TUIs is only recently being seen in the consumer market. For example, the Sifteos commercial product followed on from the Siftables research project (Merrill et al., 2007).

Rapid reconfiguration of workspaces is required in many tasks, for example to facilitate different users’ workflows and task switching (Fitzmaurice, 1996). The ability for a user to fully customize their system is difficult to achieve. By allowing the system to be at least partially (re)defined during use, the system can compensate for user diversity. The user already has the knowledge regarding how they want to interact with the system and what they want it to do. However, despite research into TUIs and customization, no systems currently exist that support the tangible ad-hoc definition of objects and functionality.

Currently, designing and using tangible and augmented systems involves three main steps; calibration of the system, authoring the content (both models and logic), and interacting with the system. This process does not support a natural workflow, where objects, roles and functionality need to rapidly change in an ad-hoc nature.

This paper presents our investigations into the merging of the authoring and interacting stages to create a form of Authoring By Interaction (ABI), where system content and functionality can be defined through normal methods of system interaction.

The architecture described in this paper, called Tangible Agile Mapping (TAM), works towards ABI, enabling previously unknown objects to be introduced into the system during the interaction phase. Through these interactions, users can introduce new objects and define their properties and functionality. This allows the authoring of new virtual content and systems, using only normal interactions with the system – lowering the threshold for developing TUIs. This allows novice users to develop TUI applications in the same vain GUI toolkits enabled the development of desktop user interfaces (UIs) by a wider audience. TAM also allows for tangible systems created by users to be saved for future reuse and distribution. This research is driven by the questions:

- How can the system enable users to easily develop TUIs when the system has no context of what the user is trying achieve?
- How can the system support definition of these interfaces inside the environment when there are no existing UI components outside the tangible realm (i.e. no mouse, keyboard or IDE)?
Whilst there exists systems that investigate interactions with ad-hoc objects, the separating factor and key contribution of this work is its focus as a primary means of enabling ad-hoc UI and functionality definition in a tangible manner. The implementation of the architecture as an example system supports the design of a tangible chemistry set and basic tangible war game, amongst others. To the authors’ knowledge, this generalized, program-as-you-go approach to TUIs offers a new application, outside previous works’ focus on application specific development.

This paper makes the following contributions:

- An architecture to support the conceptual model of tangible ABI systems based on previous literature and pilot study.
- A functioning system demonstrating this architecture, incorporating an encapsulating UI.
- The evaluation of the functioning system through the results of its implementation and a user study.

As the focus of this work is on the ad-hoc development of systems in a purely tangible manner, this work does not purport to contribute or focus on the visual tracking, gesture interaction, or programming by demonstration fields. As such, this work makes the assumptions that such a system has six degree-of-free (6DOF) tracking and object mesh generation capabilities, ideally enabling the formation of organic user interfaces (OUI) (Holman and Vertegaal, 2008). Recent advances illustrate that these are not unreasonable assumptions (Izadi et al., 2011).

The remainder of the paper is structured as follows: related work is discussed, identifying a number of challenges. A pilot study conducted to evaluate how users would interact with such a system is described, which precedes a description of an architecture to support ad-hoc tangible interaction. The implementation of this architecture is explored in an example system and evaluated through a user study with regards to the success and user experience. We then conclude with future work and final thoughts.

2 Related Work

Our research follows previous work in HCI, more specifically TUIs and OUIs, as well as sharing similarities with programming by demonstration/example systems.

TUIs afford physical objects, spaces and surfaces as a coupled interface to digital information (Ishii and Ullmer, 1997). The Bricks system (Fitzmaurice et al., 1995) explored graspable UIs, as a predecessor to TUIs, enabling manipulation of digital elements through 6DOF tracked ‘bricks’, exploring the advantages of spatial-based interaction. The affordances of graspable UIs provided a number of advantages in bimanual and parallel interaction, utilization of spatial reasoning, externalization of interfaces for ‘direct’ interactions, and support for collaborative workspaces (Fitzmaurice et al., 1995).

The MetaDESK (Ullmer and Ishii, 1997) explored rendering maps based on the location of miniature building surrogates, with appropriate warping of the map to ensure correct map alignment. Alongside this work, the authors suggested a number of conceptual equivalencies between traditional GUIs and TUIs.

URP (Underkoffler and Ishii, 1999) and other similar projects explored tangible spatial and environmental configuration and interaction. The use of physical props enabled the user to control system variables through tangible interaction. URP allowed users to quickly experiment with different configurations, by experimenting with different spatial layouts of structures. These systems supported bi-directional communication regarding system configuration, allowing the user to receive feedback regarding their interactions.

Tangible Tiles (Waldner et al., 2006) allowed interaction with projected elements using gestures (‘scooping’ them up and ‘sliding’ them off). Of note was that users could create copies of the digital content. Rekimoto and Saitoh (1999) explored UI inheritance as not being intuitive for novice users, leading to the question of whether a TUI should utilize shallow or deep copies when managing virtual properties. Travers’ (1994) direct manipulation system, supported both shallow and deep object copies, noting that whilst inheritance can have a high pedagogical value, it can cause issues for users who have no concept of inheritance.

Ullmer (2002) explored the GUI model-view-controller (MVC) equivalency in TUIs, identifying three distinguishing categories of TUIs; interactive surfaces, constructive assemblies, and tokens+constraint (TAC). The TAC category allowed constraints to be imposed on the TUI based on the physical constraints of the object, utilizing their natural affordances for logical constraint.

Holman and Vertegaal (2008) introduced OUIs as non-planar displays that are the primary means of both output and input, allowing them to ‘become’ the data they are displaying. This follows closely with the real world with little distinction between input and output, with perhaps the closest equivalent being cause and effect (Sharlin et al., 2004).

Papier-Mâché (Klemmer et al., 2004) explored the abstraction of sensor fusion to provide generic inputs to the system, allowing programs to be developed without managing low-level input. In a similar line, Kjeldsen et al. (2003) abstracted vision based inputs. Applications requiring inputs would ask the middleware for a specific input (such as a button), which is dynamically generated and mapped by the system based on the available inputs. More recently, the Proximity Toolkit (Marquardt et al., 2011) abstracted multi-device hardware to provide a set of program events for proxemic interactions.

Both VoodooIO (Villar et al., 2006) and Phidgets (Greenberg and Fitchett, 2001) explored reconfigurable physical toolkits, supporting rapid development via plug-and-play hardware. Similarly, the iRos/iStuff (Borchers et al., 2002) system provided a patch-panel framework for functionality. Despite offering reconfiguration, the systems only looked at mapping controls.

Bill Buxton coined the term Programming By Example (PBE) as systems that require the user to specify every
system state (Myers, 1986), allowing the user to work through a specific example of the problem. Halbert (1984) characterized them as “do what I did”, whereas inferential Programming By Demonstration (PBD) (Dey et al., 2004) systems are “do what I mean”. However, inferential systems create procedures that are both complex and unstructured (Myers, 1986). Myers (1986) noted that PBD/PBE systems must provide support (even implicitly) for conditional and iterative operations. Whilst you can only demonstrate one branch at a time, it was noted that demonstrational interfaces would be appropriate in scenarios where users possess high level domain knowledge that could be represented using low level commands repeatedly or in an interface with limited options that the user wants to customize. Following the impact of GUI toolkits, visual programming systems allowed non-programmers to create moderately complex programs with minimal knowledge (Halbert, 1984). Hacker (1994) explored the psychology of tangible problem solving and task completion through goal attainment using action regulation theory. Following this, since pragmatic interactions can reveal new information (Kirsh and Maglio, 1994), system interactions should enable trial-and-error with a low cost of speculative exploration (Sharlin et al., 2004).

As highlighted, this work builds on the concepts present in a number of different fields. TAM explores the application of PBE to TUIs, building on preceding work in HCI and GUI design, abstraction and interaction. Despite work on abstracting interactions, developing the interactions and content is still isolated from the use of the system. Through TAM, a number of these fields are brought together in the hope of enabling ABI.

3 Derived Challenges

Despite the previous research on tangible user interfaces, digital augmentation as well as PBD/PBE systems, a number of problems still exist for TUI developers and users alike, creating significant scope for further research. This creates a number of derived challenges:

1. Tangible systems must be designed specifically for their application. There is no generic architecture for developing tangible systems available for developers, which in turn makes few systems available to users.

2. Augmented tangible systems involve a number of sub-components: high level object tracking and spatial relationships, utilizing 3D models of the objects for augmentation and the logic for managing the actual interactions. These all must be either developed from scratch or heavily re-worked to support ad-hoc functionality (the Proximity Toolkit did however start to explore proxemic abstraction).

3. Most importantly, tangible systems are not accessible to end users and cannot be customized beyond their original purpose despite a clear benefit.

4 Exploratory Interview

Following early discussions regarding the development of a system to support ABI, exploratory interviews were conducted with six participants to gain a better understanding of how users think about interacting with tangible systems, as well as how they would envision extending them. A tangible version of the Fox, Duck and Grain game was used to explore how users, in an ideal scenario, would interact and communicate such a game to the system for tangible replication. The game involves two banks of a river, a boat and a fox, duck and a bag of grain. Users must get all the items to the other river bank, without leaving the fox alone with the duck or the duck alone with the grain. The boat can only carry one object at a time. This game was chosen as it involves a number of key concepts:

- Defining object roles;
- Defining common groups/types;
- Defining physical and quantitative restraints;
- Defining virtual entities/regions (for use as the river banks);
- Defining interactions both between individual objects and groups of objects, as well as interacting with virtual objects (regions).

In an ideal world, the user could convey such a system to the computer as if it were another person. However, the user will always have to restate their problem definition in a structure and language that the computer understands. This creates two problems, the first is having the user reformulate their program to match the system’s structure, with the second being the transfer of this knowledge into the system, creating two points of failure where the user is managing the problem in two different cognitive structures. To explore this, the questions discussed in the study were:

1. How would users ideally like to communicate instructions for a tangible system to the computer? This involved participants physically and vocally describing the system, step-by-step;

2. What kind of interactions do users expect the system to be able to support? This involved having participants actually ‘playing-out’ this game in a tangible sense;

3. How should the user communicate with the system to perform instructions for ‘learning’ versus ‘playing’? This involved having the participants explain how they would like to communicate a change in task focus.

The interview was conducted with six people (two female, four male), two of which had a background in computer science. Participants separated interactions into two groups; good (legal) and bad (illegal) moves. Participants would program an interaction, e.g. leaving the fox and duck alone on the riverbank, and identify that as being a ‘bad’ interaction, asking the system to highlight the objects in red to convey the error state. However, when programming a ‘good’ interaction, they wanted a different operation, even though the only difference is that a ‘good’ operation highlights in green. They did not intuitively abstract this operation to ‘here is the interaction’, ‘here is the output’. One non-technical
participant did note that they wanted to generalize an interaction to support the substitution of objects.

Whilst programming interactions into the system, most users created regions for ‘accepted’ and ‘unaccepted’ interactions. To program rules into the system, users would move objects next to one another in the appropriate regions. The remainder wanted the system to project an ‘accepted/unaccepted’ or ‘yes/no’ menu next to the objects.

For identifying objects, users preferred text names/abbreviations and/or colours. However, the use of colour was then overloaded by the fact that participants suggested its use to define groups/types of objects, which could be defined based on proximity, drawing a line around them or holding all the objects in their hands.

For feedback about incorrect moves, all but one participant wanted feedback to be local to the offending interaction. The other wanted the whole system to provide feedback, referring to “tilting a pinball machine”. A different participant (one without a computer science background) wanted a log or system tray so that they could keep track of their interaction history to see “what worked and what didn’t”.

Physical constraints for an object’s location were defined by pointing or drawing a line, then moving the object along that path. Quantitative constraints (i.e. only one object allowed on the boat) were defined by performing the interaction, then writing the legal number of objects near the boat.

Most participants chose the use of a virtual button to switch tasks to program rules or start playing the game, with the remainder wanting to use a thumbs-up gesture.

The final component of the interview involved asking the participants if their expectations of such a system would be addressed based on our initial thoughts regarding such a system. This involved verbally describing the system history to see “what worked and what didn’t”.

5.1 Architecture

To enable a flexible, ad-hoc environment, a certain level of complexity is required within the architecture to enable adaptation. Any TUI system that wants to enable ad-hoc functionality will need to support the following functions at a high level:

- **Definition of core objects**, which could be either physical or virtual.
- **Define types/groups** of objects that enable substitution.
- **Properties** that can be used to describe those objects.
- **Support associations** between those properties (including many-to-many associations).
- **Define rules** for those objects which in-turn can make any number of changes to objects.
- **Support sequential interactions**.

It is important to realize that this complexity is hidden from the user, as they are working at a higher level. Using these functions, there are four different scenarios that can occur:

**Table 1: Interaction scenarios**

<table>
<thead>
<tr>
<th>Isolated Updates</th>
<th>Common Updates</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Isolated interactions</strong></td>
<td><strong>Common updates</strong></td>
</tr>
<tr>
<td>Scenario 1: e.g. “two different interactions affecting two different objects”</td>
<td>Scenario 2: e.g. “two different interactions affecting the same objects”</td>
</tr>
<tr>
<td>Scenario 3: e.g. “two different interactions involving some of the same objects, but affecting two different objects”</td>
<td>Scenario 4: e.g. “two different interactions involving some of the same objects and affecting the same objects”</td>
</tr>
</tbody>
</table>

Despite Scenario 1 being the primary method of interaction, the system needs to support all four scenarios.

To address these requirements, our architecture consists of six classes (Figure 1) to define the model component of Ullmer’s TUI equivalent to the MVC. Our implementation, to be described later, follows this architecture, as we believe the features described in this architecture are core to any tangible ad-hoc system. The remainder of this section describes our architectural support for the defined high-level functions.

5.1.1 Definition of Core Objects

All objects are defined using InteractionObjects (InObjs), which are further described by Properties. These InObjs trigger Actions (rules) as part of an Interaction, which performs a copying from/to Properties as defined by a PropertyMap. We also require a core application to update the system.

![Figure 1: Relationships between the core components of the architecture](Image 310x158 to 554x240)

We use an inheritance structure for representing objects and properties. To interact with both physical objects and digital systems, we use a core InObj class. The class serves as the base that all objects (physical or virtual)...
inherit from. This class provides basic capabilities such as system names, IDs, tracking information, references to properties as well as providing virtual functions for drawing and managing the object’s properties.

5.1.2 Define types/groups
Each InObj also contains two sets of references to InObjs, one for listing groups to which this InObj is a member, the other, to define other InObjs which are a group member of this object. As such, InObjs can be grouped in a hierarchical manner. A group object on its own is a single, non-physical InObj with child members; however each of those child InObj members may have their own children, creating a tree. This means any interaction occurring with an InObj can easily be substituted by any other InObj (including those acting as groups), allowing any of those objects to act as a substitute. This is crucial when defining generalized interactions for substitution. For example, when defining interactions for a game’s playing pieces, you do not want to specify the same rules for all playing pieces. You just create a group (i.e. a new InObj) containing all playing pieces, and create rules based on that group. The use of groups enables the definition of types for both physical and virtual objects.

5.1.3 Properties
To describe objects, we can apply any number of Properties. All properties extend from a base Property, which (like InObj) constitutes a collection of sub-properties in addition to possibly holding its own data. For example, a location Property may constitute Position and Orientation Properties. This allows Properties to be created in a hierarchical nature. To capture an object’s current configuration, Property needs to support the creation of deep-copies for later retrieval. This is used to capture the before state of an object when performing ‘toggled’ interactions. To enable deep-copies, all properties must provide a method for returning a new Property of the same type.

Defining Rules and Associations
To enable interactivity, TAM manages interactions through a set of objects for each interaction. Interaction tracks the Action (rule set) that triggers the interaction; based on a set of InObjs (stored in Interaction) that can trigger it, as well as what properties need to be updated to what value. Interactions essentially contain a set of rules (defined by an Action), what objects can break them (stored in Interaction) and the changes that occur (stored in a PropertyMap, discussed later). The Action class takes a set of InObjs from Interaction and evaluates if the Action has been triggered and what objects triggered it, and is overridden by each specific Action (rule) supported by the system. This is important as when members of a group trigger an action, we need to know which one(s) were involved (i.e. which playing piece out of all playing pieces actually triggered it). This function is over-ridden by action-specific handlers (e.g. proximity, orientation, etc.). Multi-Action interactions (e.g. based on location and orientation) are managed as two separate Interactions (for location and orientation) which serve as prerequisites to a single Interaction that performs the actual property updates.

To handle updating Properties, Interaction stores a PropertyMap. PropertyMap is responsible for defining a set of ‘From’ and ‘To’ Properties, defining where object values need to be retrieved from and which objects to copy to. To support complex mappings, PropertyMap also contains a MappingAction to define how to handle non-1:1 mappings. PropertyMap also has Push/PopHistory functions, which capture and store a deep copy of all Properties involved for later retrieval (as is needed for toggled interactions).

5.1.4 Sequential Interactions
To support updates to common properties, i.e. scenarios 2 and 4 in Table 1, where the rules dictate that the same property must have two different values, Interaction contains a set of references to other Interactions that must occur as pre-requisites (either simultaneously or at any time previously). As a result, we dictate the precedence for the order of execution of interactions, as well as enabling staged interactions, i.e. A must occur before B.

We believe this architecture is sufficiently flexible to support a wide range of complex interactions, beyond simple TUIs. For example, whilst not explored in this paper, the architecture does support functionality such as the user editing core system components and functions as part of their interactions, as discussed in the next section.

5.2 Implementation
The implementation of the architecture was tailored towards its use in Spatial Augmented Reality (SAR), and was aptly titled SAR-TAM. The implementation extended InObjs into PhysicalInteractionObject (PhInObj) and VirtualInteractionObject (ViInObj). PhInObj offers attributes for the object’s location and bounding box functionality (as handled internally by the system). The application of ViInObjs allows the system to interact with existing virtual systems and remains as future work – however we envisage little-to-no modification will be needed.

One of the core features of TAM is the ability to assign any existing property to any other existing property (e.g. you can easily assign an object’s colour based on its position). As such, the implementation only has single ‘core’ property (i.e. a property that actually stores data), IntProperty, which stores a single integer value. All other properties such as Colour, Position, Orientation, Outline, etc. all consist of a generic property that has IntProperties as children. Sub-Properties may override a Draw() method that actually ‘applies’ the property to the system. For example, in this implementation ColourProperty has four IntProperty children (for the RGBA colour channels). Colour’s Draw() function calls OpenGL’s glColor() function, passing the values of the four IntProperty children. The hierarchical nature means any property can encapsulate any other property, e.g. the Location property just contains Position and Orientation properties, which are themselves collections of IntProperties.
In our implementation, PropertyMap contains multiple PropertyMapping objects, each containing the individual one-to-one mappings for Properties as a result of an interaction. By default when assigning a property, the system attempts to do a linear assignment of properties to copy from/to, i.e. Colour 1’s Red will be copied to Colour 2’s Red, etc (based on their order in the set). In the case of mismatches (i.e. mapping three elements to a single element), the PropertyMapping being applied has an attribute to handle such situations, e.g. use the max/min or averaged values. Values can also be assigned as relative to the current value or absolute, as well we being capped at either/or a max/min value or operate in a modulus fashion – this allows for interactions over a variably defined range, beyond simple boolean interactions. Whilst this many-to-one functionality exists in a working manner architecturally, modifying this attribute has not been explored in the TAM UI and remains as future work.

Action provides two evaluation functions, EvaluatingIncludingGroups and EvaluateExcludingGroups, used to evaluate either with or without objects’ children/group members. Currently, only one type of action is supported in TAM, ProximityAction. Action support will grow in the future to include orientation and virtual actions (to support input from existing digital systems) as well as temporal based interactions.

Whilst not explored in this paper, SAR-TAM does support the possibility of the user editing core system functions as part of their interactions. One example could be the use of physical objects (e.g. blocks) to create custom keyboards. Using blocks, the user could define an interaction where upon placing a block on a keyboard key, the key becomes ‘virtually glued’ to the block, allowing the position of the system-managed, virtual entity to be modified at run-time using a custom interaction defined by the user. The user could also create interactions to change the letters on the keys, creating custom keyboards and layouts for other languages. One suggested use by someone trialling the system was to create a tangible video editor so that film editing can leverage the user’s spatial skills.

6 System Overview

SAR-TAM uses two Microsoft Kinects and an OptiTrack 6DOF system (which enables object tracking using retroreflective markers and infrared light). One Kinect is downward facing and is used for object detection and enabling touch interactions. The other Kinect faces the user for skeleton tracking to enable gesture input. The OptiTrack is used to track objects once they have been registered with the system, as the Kinect cannot reliably track objects between frames whilst being handled.

The system runs at 20fps due to delays in updating both Kinects and detecting the user’s skeleton contour. SAR-TAM utilises a state machine, with poses (static gestures) the primary method of navigation, allowing users to start with a ‘blank slate’ with no system-specific input tools. All poses inherit from a base pose, which returns a boolean based on a skeleton provided by the OpenNI framework. Users must hold a posture for at least 500ms to help prevent false positives. Feedback for pose detection is provided using a graphical component on the table showing the user’s contour with a skeleton overlay. Skeletal bones with less than 70% accuracy are rendered in red. Upon performing a pose, the user’s contour colour is changed to white, instead of the previously randomly assigned pastel colour. A description of the matched pose is displayed underneath.

6.1 Using the System

To use the system, users interact solely through visual and audio cues. The projected UI is designed to be minimal (Figure 3a) to allow users to develop completely custom, immersive systems. The current state (either ‘Interacting’, ‘Introducing Object’, ‘Defining Group’ or ‘Defining Interaction’) is displayed on the top of the display area as well as a brief set of instructions for the current step. Upon changing states, instructions are updated on the display and read aloud using a text-to-speech engine.

To introduce objects, users place an object on the surface and perform a ‘one arm open’ gesture, as if to say “here is a new object” (Figure 3a). The system then prompts users for an object name and highlights the outline of the object detected each frame with the Kinect (Figure 3b). Users enter a name using a simplified projected keyboard displayed at a fixed location in front of the user.

Upon pressing ‘Confirm’ (replacing the ‘Enter’ button) users then select a default colour for the object from a ‘linear’ style colour chart (Figure 3c). The object is augmented using the object’s contour and projecting the colour. Given contours update each frame, they are subject to jitter.

When the desired colour is selected, users place their forearms vertically parallel as if they were about to take a photo to ‘capture’ the current configuration. This is known as the Confirm pose. Any objects that have been defined are now shown in their selected colour, with the name projected along side. New objects can be introduced at any stage.

Once an object is formally introduced, the system starts to match objects between the Kinect and OptiTrack system each frame. Each PIO is attached to a single

Figure 2: The SAR-TAM tabletop with projector, tabletop and pose Kinects and OptiTrack system (red cameras)
OptiTrack marker. Every frame, TAM locates the markers and matches the contours detected by the Kinect, allowing objects to be tracked frame-by-frame. Currently, this is based on a simple proximity test.

Once users have defined at least two objects, they can either create groups/sets of objects (to enable object substitution using groups in Interactions) or define an interaction rule set. To define a group, users extend both arms out towards the table as if to gesture, “here are a group of objects”. The system then projects a virtual keyboard, and asks users to point to objects in the group and enter a name for the group. Tracked objects are now only identified by their projected name until users ‘point’ at them by placing their finger within 5 cm of the object, at which point the object is highlighted using its default colour. Users then enter a group name and press ‘Confirm’. Creation of the group is confirmed by voice prompt.

To create a set of rules for an interaction, users place their forearms at 90° (Figure 4a). The system then prompts users to point to objects involved in the interaction and perform the Confirm pose. Should any of those objects be members of a group, the system will prompt users to resolve this ambiguity by displaying a menu next to each object with group memberships. The menu shows the object’s name as the default option, with the object’s groups’ names as options (Figure 4c). Users are prompted to select which group the object can be substituted by and perform the Confirm pose. The system then prompts users to perform the interaction (at the moment this is limited to arranging objects based on proximity relative to each other) and then perform the Confirm pose. The system plays the sound of a SLR camera taking a photo to provide feedback that the arrangement was ‘captured’. The system then prompts users to highlight which objects change during that interaction and presents the colour chart, allowing selected objects to have their colour changed. Once this is done, users perform the Confirm pose and the system goes back to the normal state, allowing users to trigger interactions or continue defining new objects/groups/rules.

7 Evaluation

Regardless of the level of flexibility offered by an architecture, there will always remain a level of adaptation imposed on the user due to individual differences; however we seek to minimize this. As such, for a user to be able to use TAM, there are two things that must occur for the user to fully externalize their internal thoughts into a tangible system:

1. Users must adapt their view/architecture of system to match that supported by the adaptive system.
2. Users must translate that knowledge into the system.

As a result, our evaluation was designed to evaluate:
- How easily can users grasp the concepts involved in TAM to convert a scenario into the required structure?
- How easily can users then communicate that structure to the system?

This was evaluated by means of an exploratory user study. We employ an experimental design similar to Scott et al. (2005) to gain an understanding of how users engage with and understand the system. The study consisted of the participants first being seated at a desk, watching a video that explained the system and demonstrated a single, group-based interaction between three objects to simulate the reaction between hydrogen and chlorine to form hydrogen-chloride (two hydrogen atoms were defined, and a group created so either hydrogen could trigger the reaction). To ensure equal training, all participants could only watch the video once, but could pause and ask questions at any time.

After the video, participants were asked to create three scenarios, each scenario extending the last, to demonstrate a minimal game. The tasks consisted of:
1. A country that was being invaded by an attacking army. Upon the attacking army reaching the country, they would change its national colour to red.
2. Same as Task 1 except with a defending army. Upon both the defending and attacking armies reaching the country, the defending army defeats the attacking and changes the national colour to green in mourning of those lost.
3. Same as Task 2, but with two defending and two attacking armies. No matter which combination reach the country (as long as there is at least one attacking army...
and one defending), the attacking army is defeated and the colour changed to green in remembrance of those lost.

The country and armies were represented by numerous large, white pencil erasers. Upon the participant moving the objects beyond the participant-defined limit, the system would trigger the associated change in colour. This allowed the user to physically move the different armies and see the resulting state of the country object.

Participants were provided with a reference sheet showing each pose and its corresponding function in the system. This was provided as we were not interested in evaluating the particular poses, rather the functionality that they linked to. Participants were read aloud from a copy of each task, and then given a printed copy. All participants were video recorded. At the conclusion, participants filled out a questionnaire focusing on the system’s intuitiveness and ease of use. The questionnaire was a mix of visual analogue scales and qualitative questions. The questions focussed on how easy, intuitive and appropriate they found each sub-section of the system to function (introducing objects, defining groups, etc.). They were also asked about the level of guidance provided by the system and how the system functioned versus how they would explain the problem to another person. The study concluded with an informal discussion.

7.1 Results

The user study conducted consisted of 21 participants (2 female, 19 male, mean age of 24), nine of which had experience with tangible UIs. Despite the increased member of male participants, none of the tasks featured gender-influenced decisions. All participants successfully completed all three tasks. All participants successfully completed the first scenario with their first interaction attempt; the mean number of interactions/groups can be seen in Table 2. Four of the participants created groups; however this was unnecessary (average of 0.19 groups/participant) as the scenario did not require them. Sixteen participants completed the second scenario in their first interaction, four participants on their second interaction, with the remaining taking three interactions (average of 1.29 attempts/participant). Six participants falsely utilized groups in this scenario, believing groups were needed for interactions with more than two objects.

For the final scenario 15 participants succeeded with their first interaction attempt, with five taking two attempts and one taking three. A mean of 2.38 groups were created per participant, with a goal of two groups per participant. We believe that the majority of users getting the desired outcome first time is important. It indicates that the participants’ personal assumptions/expectations about how such a system should function, matched its functionality, both in terms of adapting their view of the problem to match the system and translating that knowledge to the system.

Participants reported that the system progressed in a similar logical order to how they would have described the scenario to another person, as well as how they thought through the problem mentally. One participant noted that it was easier to work through the scenario using the system instead of mentally, with another noting “I can’t think of a more suitable way”. One participant highlighted the flexibility offered by the system workflow as a benefit.

Participants reported in the questionnaire that introducing objects was both intuitive and easy to perform, with all participants responding favourably to intuitiveness and 19 responding favourably regarding ease of use. For creating groups, 19 participants gave a favourable rating regarding intuitiveness, with 16 giving favourable ratings regarding ease of group creation. For defining interaction rules, 18 participants gave favourable results for both intuitiveness and ease of creation. All participants gave favourable feedback saying that the system progressed in a similar way to how they thought the problem through both in their head, and followed how they would have explained it to another person. The most problematic part of the study was the tracking systems, especially for skeleton tracking and gesture recognition, which varied greatly between users. Whilst the touch interaction was an improvement, accidental touches still occurred for all participants. Almost all participants directly addressed the tracking issues in their feedback. As such, participants did not feel the system to be overly responsive. Despite not being the focus of this work, we believe these problems will be addressed in the future. In spite of the tracking problems, users still found the system enjoyable to operate.

The vast majority of users found the guidance provided by the system to be almost ideal, with 20 participants giving favourable responses. The virtual keyboard had a nearly equal number of participants that liked and disliked it. Most users found the selected gestures both intuitive and appropriate for the tasks. Participants especially liked introducing and defining objects, as well as grouping objects and text-to-speech voice prompts by the system.

One interesting observation was how participants customized the examples, by naming the town, enemy and defensive army different names (e.g. NATO). This implied users were making cognitive connections between the objects and the context of the demonstration. Overall, the results of the study demonstrated a favourable result for this form of ad-hoc interaction, even for participants without a technical background. All participants were able to complete the tasks after watching only a single interaction being defined. Results from the questionnaire reflect a strong approval for both TAM and SAR-TAM.

8 Applications of Generalized Tangible User Interfaces

As mentioned early in this paper, given UIs must be generalized since the designer does not know who the end user is or how it will be used, a major application of systems such as this would be to enable extensibility by

<table>
<thead>
<tr>
<th></th>
<th>Mean Interactions</th>
<th>Mean Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task 1</td>
<td>1.00</td>
<td>0.19</td>
</tr>
<tr>
<td>Task 2</td>
<td>1.29</td>
<td>0.33</td>
</tr>
<tr>
<td>Task 3</td>
<td>1.46</td>
<td>2.38</td>
</tr>
</tbody>
</table>

Table 2: Average attempts per task
(presumably) novice end-users. Their use as a means of externalizing internal thoughts into tangible artefacts leans towards their use in groupware to support computer-supported cooperative work and group collaboration. Such functionality would enable individuals to quickly externalize their internal views of how a system works, which instantly transfers to a collaborative interaction medium. As such, TAM has applications to tangible systems where end users are not the original developers. Additional functionality can be defined by the end user to address customization or new functions. In addition, core functionality currently programmed by the original system designers, could also be at least partially replaced and defined using the system.

Example systems for the application of these types of systems include real time planning (tangible war games table using live data from the field), education aids and any scenario where users could benefit from externalizing the problem into tangible problem solving.

9 Future Work

This paper describes our new TAM architecture and our first implementation in the SAR-TAM application. Arguably, the power of this system comes from the ability to map between properties/variables in a tangible manner, something we are yet to fully explore. We are investigating how to edit these mappings once created, which is an obvious limitation of the existing system. This would enable users to not only develop basic tangible systems, but when combined with virtual mappings, allow the user to develop complex applications. Example applications include support for education and problem solving as well as support for such small sub-systems including the previously highlighted multi-lingual virtual keyboards, tangible phone books, etc.

Key areas of future work include supporting:
- mapping for an object where it is both the input and the output (embodied TUIs),
- non-boolean events and temporal input and output,
- creation of physical-virtual tools (Marner et al., 2009),
- prerequisites for actions/rules (‘daisy chaining’),
- undo/redo as well as saving/restoring saved configurations in a tangible realm, and
- debugging existing objects, rule sets and mapping configurations.

Following the results of the user study, we will also be exploring alternative control methods to supplement/replace the existing pose-based control system.

10 Conclusion

TAM enables the development of and interaction with novel TUIs with no background development. Through providing an abstracted set of interactions, novice users can program rules-based interactions, utilizing both individual objects and group-based interactions, offering type definition and substitution. Our system allows users to quickly externalize mental systems, as well as define novel TUIs. The user study results show TAM as both an effective and flexible means for allowing technical and non-technical users to create tangible systems ad-hoc. Future development will enable a wider range of interactions, which paired with a more advanced mapping system for programming logic, will allow for a richer set of interactions.

11 Acknowledgements

The authors would like to thank Thuong Hoang and Markus Broecker for proofreading the paper and the reviewers for their feedback and ideas.

12 References


moving depth camera. Proc. of the 24th annual ACM symposium on User interface software and technology. Santa Barbara, California, USA: ACM.


vsInk – Integrating Digital Ink with Program Code in Visual Studio

Craig J. Sutherland, Beryl Plimmer
Department of Computer Science
University of Auckland
cj.sutherland@auckland.ac.nz, beryl@cs.auckland.ac.nz

Abstract
We present vsInk, a plug-in that affords digital ink annotation in the Visual Studio code editor. Annotations can be added in the same window as the editor and automatically reflow when the underlying code changes. The plug-in uses recognisers built using machine learning to improve the accuracy of the annotation’s anchor. The user evaluation shows that the core functionality is sound.

Keywords: Digital ink, code annotation, Visual Studio.

1 Introduction
This paper presents a technical, usable solution for adding digital ink annotation capacity to the code editor of an Integrated Development Environments (IDE). We support a transparent ink canvas on the code editor window; the user can add, move and delete digital ink. The annotations are anchored to a line of the underlying code and maintain their relative position to this line as the window scrolls and lines above are added or removed. The application also handles collapsible regions within the editor and provides spatial and temporal visualizations to support navigation.

Using a pen is a natural and easy way to annotate documents. One of the main reasons why people prefer paper documents to online documents is the ability to annotate easily using just a pen (O’Hara and Sellen, 1997). This form of annotation does not interrupt the reading process and allows the reader the freedom to annotate as they prefer. More recent research by Morris, Brush and Meyers (2007) and Tashman and Edwards (2011) found that pen-based computers allowed readers the same easy ability to annotate as paper and even overcome some of the limitations of paper (e.g. re-finding existing annotations and lack of space for annotations).

One form of document that is often reviewed is program code (Priest and Plimmer, 2006). While program code is a form of text document it differs significantly from other documents. Code is predominately non-linear – it is broken up into methods and classes – and is often split across multiple files. It can be printed out and annotated but as the program size increases it becomes more difficult to follow the flow of program logic. To help developers read and understand code they usually use tools like IDEs.

An IDE is a complex environment that provides many tools for working with program code. Tools include editors with intelligent tooltips and instant syntax feedback, powerful debuggers for tracing program flow and examining state and different visualisers for showing how the code fits together. Text comments interspersed with the code are widely used for documentation and notes. For example, a developer may add a TODO comment to mark something that needs to be done.

One feature that is lacking is the ability to use digital ink for annotations. Ink annotations are a different level of visualization. They are spatially linked to specific parts of the document but they are quickly and easily discernible from the underlying document (O’Hara and Sellen, 1997).

Previous prototypes for digital ink annotation in IDEs (Chen and Plimmer, 2007, Priest and Plimmer, 2006) failed to support a canvas on the active code window because of the lack of appropriate extension points in the APIs of the IDEs (Chang et al., 2008). These earlier prototypes cloned the code into a separated annotation window resulting in an awkward user experience.

We have built a plug-in to Visual Studio 2010, called vsInk that allows people to use digital ink to annotate code in the active code editor. The requirements for vsInk were garnered from the literature on IDE annotation and general document annotation. In order to achieve the desired functionality a custom adornment layer is added to the code window. vsInk applies digital ink recognition techniques to anchor and group the annotations to the under lying code lines. We ran a two-phase user evaluation to ensure that the basic usability was sound and identify where more development is required.

2 Related Work
The literature reports three attempts to integrate digital annotations in IDEs. Priest and Plimmer (2006) attempted to modify Visual Studio 2005. With their plug-in, called RichCodeAnnotator (RCA), they tried but failed to add annotations directly to the code editor. Instead RCA used a custom window that allowed users to annotate over a copy of the code. The plug-in grouped strokes together into annotations based on simple spatial and temporal rules. The first stroke of an annotation is called the linker and is used to generate the anchor for the whole annotation. RCA allowed for two types of linkers – circles and lines. Simple heuristic rules were used to determine the linker type. When code lines are added or removed above the anchor the whole annotation is moved. The next IDE to be modified was Eclipse with CodeAnnotator (Chen and Plimmer, 2007). This plug-in was based on the experiences with RCA and used the same approach (e.g. grouping strokes and using a linker).
One issue with both plug-ins was integrating the annotation surface directly into the code editor (Chang et al., 2008). Neither Visual Studio 2005 nor Eclipse provided sufficient extensibility hooks to allow third parties to directly integrate with the editor. To get around this issue both RCA and CodeAnnotator used a separate window for annotating. The code in the annotation window is read-only although it is automatically refreshed when the code is modified.

Another common issue is how to group together individual strokes to compose annotations. Both RCA and CodeAnnotator used simple rules for determining whether a stroke belonged to an existing annotation. These rules were based partially on previous work with digital annotations (Golovchinsky and Denoue, 2002). Both plug-ins used two simple rules – a stroke was added to an existing annotation if it was added within two seconds of the last stroke or it was within a certain distance of other strokes. The area assigned to the annotation is indicated visually with a box that expands automatically as the annotation grows. There is the implication that they were only semi-effective as both plug-ins made adjustments to the grouping process. RCA allowed users to manually select an existing annotation to force strokes to join. CodeAnnotator changed the rules defining whether a stroke was close to an existing stroke. The final rule involved different distances for each side. Neither paper reports the accuracy of their grouping strategy.

Another common issue with annotations is how to anchor them to the underlying text. This is a common issue with annotations in general, not just for program code. Using an x-y co-ordinate for the anchor works fine for static documents but as soon as the document can be modified the x-y co-ordinate becomes meaningless (Brush et al., 2001, Golovchinsky and Denoue, 2002, Bargeron and Moscovich, 2003). Previous attempts at solving this issue have typically included context from the underlying document to generate the anchor point. For example, XLibris (Golovchinsky and Denoue, 2002) uses the underlying text while u-Annotate (Chatti et al., 2006) and iAnnotate (Plimmer et al., 2010) both use HTML DOM elements. The approach used by RCA and CodeAnnotator is to select the closest line of code to the annotation. They do not mention how this line is tracked as the code changes.

One issue with annotations in an IDE is how to navigate between them. This is particularly problematic for program code because of the non-linear flow of the documents. Navigation was partially addressed in CodeAnnotator by the addition of an outline window (Chen and Plimmer, 2007). The navigation window displayed a thumbnail of each annotation in the document. Selecting an annotation automatically scrolls to the annotation. The main limitation of this is it assumes that the user is only interested in navigating annotations in the same order they are in the document. Given the non-linear nature of code users are likely to add comments as they trace through the code. Indeed other annotations systems such as Dynomite (Wilcox et al., 1997) and XLibris (Schilit et al., 1998) provide timeline views that organise annotations in the order they were added.

The final IDE plug-in reported in the literature is CodeGraffiti (Lichtschlag and Borchers, 2010). CodeGraffiti was designed as a pair programming tool that extends the Xcode IDE. One person would use CodeGraffiti in Xcode and a second person can view and add annotations via a remote session (e.g. on an iPad or a second computer). There are few details provided about CodeGraffiti’s functionality but it appears that it works by anchoring annotations to lines of code. It does not mention whether annotations are allowed directly in the editor, how strokes are grouped or any navigation support.

One issue that has not been mentioned in any study is how annotations should behave when the underlying code is hidden. Most IDEs allow users to define collapsible regions within files which can be collapsed and expanded as desired. Brush, et al. (2001) investigated how people expected annotations to behave when the underlying context was changed or deleted but this assumes permanent changes to the document not temporary changes like collapsing a region.

In summary, the current literature describes a number of issues in adding annotations to IDEs. Limitations in the IDE extensibility models have prevented past attempts from integrating ink directly into the code editor window. Other issues include how to group together single strokes into annotations, how to calculate an anchor point for repositioning annotations and how to navigate through existing annotations. One area that has not been investigated at all is handling collapsible regions within code.

3 Requirements

From the literature review five requirements were identified for vsInk. First, annotations need to be directly integrated within the code editor. Second, strokes need to be automatically grouped together into annotations. Third, annotations need to be anchored to the underlying code in a way that allows them to be consistently repositioned after any modification to the code. Fourth, support is needed for collapsible regions. And fifth, it should be easy for users to navigate through annotations.

3.1 Direct Editor Integration

Users should be able to directly add annotations within the code editor. As mentioned above previous attempts required the user to annotate code in a separate read-only window (Chang et al., 2008) which has the potential to cause confusion. Adding an annotation should be as simple as turning on ink mode and drawing. None of the existing editor functionality should be lost (e.g. the user should still be able to modify the code, debug, etc.).

3.2 Grouping Strokes into Annotations

As users add strokes they should be grouped together in a way that appears natural. The rules used in RCA and CodeAnnotator (Chen and Plimmer, 2007, Priest and Plimmer, 2006) can be used as a starting point but these may need to be expanded. As an example, when a user is writing text they typically expect all the letters in a word to be grouped together in a single annotation. In contrast annotations on consecutive lines may not belong together.
3.3 Anchoring and Repositioning Annotations

When the code editor is scrolled or the code is modified, the annotations should stay positioned relative to the associated code. To handle this introduces two requirements. First some way of identifying an anchor point is needed. In Visual Studio 2010, anchoring and repositioning annotations was handled by adding adornments in the editor. The user can toggle these regions by clicking on an indicator in the margin of the editor (see Figure 1).

![Figure 1: Examples of a collapsible region in Visual Studio 2010. Top view – the region is expanded. Bottom view – the same region when collapsed.](image)

Both navigation elements should allow the user to navigate to a selected annotation. When a user selects an annotation the editor should automatically scroll to the location in the code and any collapsed regions expanded so the annotation is visible.

4 Implementation

vsInk has been implemented using C#, WPF and the .Net 4 framework. It uses the Visual Studio 2010 SDK for integration with Visual Studio 2010. It consists of a single package that can be installed directly into the IDE. While it has been designed to be used on a Tablet PC it can be used with a mouse on any Windows PC. Figure 2 shows the main elements in the user interface.

This section describes the five major features of vsInk: editor integration, grouping annotations, anchoring annotations, annotation adornments and navigation.

4.1 Editor Integration

In Visual Studio 2010 it is possible to extend the editor by adding adornments in the editor. A Visual Studio adornment is a graphic element that is displayed over the code in the code editor. A plug-in extends the editor by defining an adornment layer and adding adornments to it. This adornment layer is added to the editor. Visual Studio offers three types of adornment layers – text-relative (associated with the text), view-relative (associated with the viewport) and owner-controlled (Microsoft). vsInk works by adding a new owner-controlled adornment layer. This layer contains an ink canvas that covers the entire viewpoint.

Initially we tried to use the text-relative and viewport-relative layers but both of these resulted in problems. The text-relative layer approach failed because it required each annotation to be associated with a location in the text. vsInk requires a single ink canvas to cover the entire viewport (rather than an adornment per annotation) so that free form inking is supported anywhere in the document.

The viewport-relative layer initially seemed more promising as it allowed the annotations to scroll in sync with the code. However there were a number of scenarios where the scrolling broke (e.g. when moving to the top or bottom of a long file). These appeared to be caused by the way Visual Studio regenerates the viewport on the fly. Various attempts to fix these issues failed, so the viewport-relative approach was abandoned.

Using an owner-controlled adornment layer gives vsInk full control over how the elements are displayed – Visual Studio does not attempt to do any positioning. This flexibility does come at a cost: vsInk now needs to position all the adornments itself. The ink canvas in vsInk is the only UI element that is added directly to the adornment layer – all other UI elements are added to the ink canvas. The ink canvas is positioned so it covers the entire viewpoint – from the top-left corner to the bottom-right. The actual viewport in Visual Studio is more than just the viewable screen area; it also includes some lines above or below the viewable space and is as wide as the widest line.

The annotation anchor requires two items: the line number ($Line_n$) and a line offset ($Offset_{line}$). The editor in
Visual Studio is broken up into a number of lines (see Figure 3). When a new annotation is started the closest line to the linker anchor point is selected (see Figure 4) – this is Line\textsubscript{e}. Internally Line\textsubscript{e} is recorded as a Visual Studio tracking point. Storing Line\textsubscript{e} as a tracking point enables vsInk to use Visual Studio’s automatic line tracking to handle any changes to the code.

When the user scrolls through a document Visual Studio fires an event notifying any listeners that the viewport has changed (this change can be either repositioning or regeneration). vsInk listens for this event and updates every annotation on the canvas when it is received. First, each element is checked whether its Line\textsubscript{e} is visible. If Line\textsubscript{e} is not visible then the annotation is hidden. If Line\textsubscript{e} is visible it is checked to see if it is inside a collapsed region, again the annotation is hidden if this check is true. If both of these checks pass the annotation is displayed. Finally a translation transform is applied to each annotation to move it into the correct position.

The actual positioning of each annotation requires a number of steps. First the line number of the first line in the viewport (Line\textsubscript{Viewport}) is subtracted from Line\textsubscript{e} to give the offset line number (Line\textsubscript{offset}). Line\textsubscript{offset} is multiplied by the line height (Line\textsubscript{height}) to get the line position (Position\textsubscript{line}) in the adornment layer. The viewport offset (Offset\textsubscript{Viewport}) is subtracted from Position\textsubscript{line} to get the viewport-relative position (Position\textsubscript{Viewport}) (see Figure 6). Finally Offset\textsubscript{line} is added to Position\textsubscript{Viewport} to get the actual position (Position\textsubscript{actual}) (see Figure 5).
Position is used to translate the annotation into its correct position on the canvas.

Collapsible region support is added by listening to the relevant events in the editor. The region manager in Visual Studio has three events for handling changes to regions (RegionsChanged, RegionsCollapsed and RegionsExpanded). When any of these events are received vsInk performs an update of the ink canvas (the same as when the viewport changes). Since a collapsed region check is already performed in the update no further changes were needed to support collapsible regions.

Changes to the code are handled in a similar way. vsInk listens to the events that fire whenever the underlying code is changed and performs a canvas update. Because a tracking position is used to record the closest line the line number is automatically updated when lines are added or removed. The final step is to detect whenever a line with an annotation anchor has been deleted. In this case the entire annotation is deleted as well.

The annotation ink and associated elements are serialised to a binary format and saved automatically every time the associated code file is saved. The strokes are stored using Microsoft’s Ink Serialise Format (ISF). When a new document window is opened vsInk checks to see if there is an associated ink file. If the file exists the existing annotations are deserialised and added to the canvas, otherwise a new blank canvas is started.

4.2 Grouping Strokes into Annotations
Grouping annotations is performed by using simple rules. The initial version of vsInk used a boundary check to group strokes with annotations. A boundary region for each annotation is calculated by getting the bounding box for the annotation and adding 30 pixels to each side (see Figure 7). vsInk tests a new stroke against all annotations in a file to see if the new stroke intersects any existing boundary region. If a stroke intersects a boundary region for multiple annotations it is added to the first annotation found. If the stroke does not intersect any existing annotations it starts a new annotation.

Usability testing (see below) showed that this was not accurate enough. The two main conditions under which this failed were when the user started a new annotation too close to an existing annotation or the user was trying to add a new stroke to an existing annotation but was too far away. In addition when multiple annotations were found the new stroke was often added to the wrong annotation!

Three changes were made to address these issues. First, the boundary was decreased to 20 pixels. Second, a timing check was added – if a new stroke was added within 0.5 seconds of the last stroke it was added to the same annotation. The literature reports two numbers for temporal grouping – 0.5 seconds (Golovchinsky and Denoue, 2002) and 2 seconds (Priest and Plimmer, 2006). Both were trialled and 0.5 seconds was found to be more accurate for grouping.

The final change was for selecting which annotation when multiple possible annotations were found. The annotation chosen is the annotation that has the closest middle point to the starting point of the stroke. Euclidean distances are used to calculate the closest middle point.

4.3 Anchoring Annotations
vsInk adopts the concept of a linking stroke for generating the anchor from Priest and Plimmer (2006). In vsInk the linking stroke is the first stroke of a new annotation. The actual anchor point is calculated based on
Both RCA and CodeAnnotator used some simple heuristics for determining the type of the linking stroke which could be a line or a circle (Chen and Plimmer, 2007, Priest and Plimmer, 2006) but this was too limiting, especially as new linker types are needed.

To overcome this we used Rata.Gesture (Chang et al., 2012) to recognise the stroke type. Rata.Gesture is a tool that was developed at the University of Auckland for generating ink recognisers. Rata works by extracting 115 features for each stroke and then training a model to classify strokes. This model is then used in the recogniser for classifying strokes.

To generate the recogniser for vslnk an informal user survey was performed to see what the most common types of linking strokes would be. This produced the list of strokes in Table 1. Ten users were then asked to provide ten examples of each stroke, giving a total of 600 strokes to use in training. These strokes were manually labelled and Rata used to generate the recogniser.

When a new annotation is started the recogniser is used to classify the type of linker. Each linker type has a specific anchor location (see Table 1) – this location is used to find the $Line_3$ for anchoring.

### Table 1: Recognised linker types. The red cross indicates the location of the anchor.

<table>
<thead>
<tr>
<th>Linker Type</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line – horizontal</td>
<td><img src="image1" alt="Example Image" /></td>
</tr>
<tr>
<td>Line – vertical</td>
<td><img src="image2" alt="Example Image" /></td>
</tr>
<tr>
<td>Line – diagonal</td>
<td><img src="image3" alt="Example Image" /></td>
</tr>
<tr>
<td>Circle</td>
<td><img src="image4" alt="Example Image" /></td>
</tr>
<tr>
<td>Brace</td>
<td><img src="image5" alt="Example Image" /></td>
</tr>
<tr>
<td>Arrow</td>
<td><img src="image6" alt="Example Image" /></td>
</tr>
</tbody>
</table>

4.4 Annotation Adornments

Each annotation can have a number of associated adornments. These adornments are slightly different from the Visual Studio adornments in two ways: they are associated with an annotation rather than an adornment layer and their visibility is controlled by vslnk. There are two default adornments in vslnk: the boundary region indicator and the anchor indicator. In addition vslnk allows for third parties to write their own custom adornments. An example of a custom adornment is provided in the project; it displays the user name of the person who added the annotation.

When an annotation is added a factory class for each adornment is called to generate the adornments for the new annotation. This process is called for both loading annotations (e.g. when a document is opened) and for a user adding a new annotation. Each adornment is then added to a sub-layer of the ink canvas. The sub-layer is needed to prevent the adornments from being selected and directly modified by the user. Custom adornments can be added to vslnk by adding a new factory class.

Adornments are positioned using a similar process to ink strokes. If an annotation is hidden during a canvas update all the associated adornments are hidden as well. If the annotation is visible then each adornment for the annotation is called to update its location. Adornments typically update their position using the details from the annotation (e.g. the bounding box or similar).

4.5 Navigating Annotations

There are two parts to navigation – collapsed region support and a navigation outline. Collapsed region support adds an icon to a sub-layer of the ink canvas whenever a collapsed region contains annotations. The addition or deletion of the icon is performed during the canvas update process, which is triggered whenever a region is changed. This ensures the icon is always up-to-date and only displayed when there are annotations in a collapsed region.

Clicking on the icon automatically expands as many collapsed regions as needed and scrolls the editor so the annotation is in view. In addition the annotation is “flashed” to show the user where the annotation is. The actual “flash” is implemented by the different adornments – the default implementation is to change the border size for the boundary region. In addition when the user hovers the pen (or mouse) over the icon a thumbnail is displayed of the entire annotation (see Figure 8).

The navigation outline is implemented as a separate tool window within Visual Studio. This gives the user full control over where the window is positioned. The window contains a scrollable list showing a thumbnail of each annotation within the document (Figure 9). Each
annotation is scaled to between 25% and 100% of the original size – this is to try and fit as much of the annotation as possible in the thumbnail without making it unrecognisable.

The user can switch between position and timeline views of the annotations. This is achieved by changing the sort order of the annotations within the list. The position view uses the line number as the sort and the timeline view uses the time the annotation was first added.

Finally the navigation view can be used to navigate to the actual annotation by clicking on an annotation in the window. This works in a similar way to the collapsed region icon. It includes the automatic scrolling and region expansion and the annotation “flash”.

![Image](70x361 to 181x611)

Figure 9: The ink navigation tool window.

5 Evaluation

To assess the usability of vsInk a task-based usability study was carried out. Subjects were asked to perform two code review tasks in Visual Studio 2010 and to annotate any issues found. Usability information was collected via researcher observation, questionnaires and informal interviews. This section describes the methodology of the study and then the results.

5.1 Methodology

There were eight participants in the study (6 male, 2 female). Four were computer science graduate students, three full-time developers and one a computer science lecturer. All had some experience with Visual Studio with most participants saying they use it frequently. Participants were evenly split between those who had used pen-based computing before and those who hadn’t. All but one of the participants had prior experience reviewing program code. Two rounds of testing were performed – after the first round of testing the major flaws identified were fixed and then the second round of testing was performed.

Each study started with a pre-test questionnaire to gauge the participant’s previous experience with the tools and tasks. The researcher then showed vsInk to the participant and explained how it worked. The participant was then allowed three minutes to familiarize themselves with vsInk. For the first task the participant was given a set of simple C# code guidelines (eight in total) and a small application consisting of four C# code files. They were asked to review the code and annotate where they thought the code did not conform to the guidelines. As the task was to evaluate the annotation experience the participant was only given eight minutes to review the code (although they were allowed to finish earlier if desired). After the review was finished an automatic process updated the code and the participant was asked to re-review the code to see if all the issues had been fixed. The researcher observed the participant and noted down any usability issues. In addition a questionnaire was filled in after each review task. After the tasks the researcher and participant went through all the annotations and identified whether the annotation had been correctly repositioned after the update. Finally there was an informal, semi-structured interview. The main purpose of the interview was to find out what each participant liked and disliked about vsInk.

5.2 Results

After the first four subjects the results were reviewed and a number of issues were identified. Before the second round of testing changes were made in an attempt to fix these issues. The main issue found was strokes were being incorrectly added to existing annotations. During the tests the opposite (strokes not being added to annotations correctly) occurred rarely. Therefore the three changes mentioned (see 4.2 above) were made to the grouping process.

The other refinements to vsInk were as follows. Some of the participants complained that the lines were too small or the ink too fat. To fix this the ink thickness was reduced and the line size increased slightly. Another common complaint was the adornments obscured the code. This was fixed by making all adornments semi-transparent and removing non-necessary ones (e.g. the name of the annotator). Participants also mentioned the ink navigator distorted the annotations too much so the amount of distortion was limited to between 20% and 100% of the original size. Observations suggested that the navigation features were not obvious. When a participant selected an annotation in the navigator they did not know which annotation it matched on the document (especially when there were several similar annotations). To fix this the flash was added to identify the selected annotation.

In addition to the issues mentioned above, there were other issues noted that were not fixed due to time constraints. These included: tall annotations disappearing when the anchor point was out of the viewport, cut/paste not including the annotations, and annotations not being included in the undo history.

After the modifications the second set of participants tested vsInk. We found most of the modifications had the desired effect and vsInk was easier to use. However there were still issues with the grouping of strokes into annotations. Using time to group strokes sometimes caused
strokes to be added incorrectly, especially when the participant moved quickly down the code file. The boundary region still caused problems when trying to start a new annotation when the code lines of interest were close together.

Together the researcher and participants identified a total of 194 annotations. Each participant added a mean of 28 annotations—a t-test found no evidence for any difference between the two rounds of testing (p-value > 0.05). Of the 194 annotations 57 (29%) were incorrectly repositioned after the code update. While this dropped from 36% in the first round to 26% in the second round a t-test found no evidence of this being statistically significant (p-value > 0.05). The majority of the incorrectly positioned annotations (52 out of 57) were as a result of grouping errors.

The questionnaire asked the participants to rate vsInk on a number of features (see Figure 10). The questionnaire consisted of a number of statements using a 5-point Likert scale. The subjects were asked whether they agreed with the statement (score = 1) or disagreed (score = 5).

The majority of the participants agreed that the exercise was enjoyable, using vsInk helped complete the task and that annotating code was easy. There appeared to be a slight drop in agreement for all three statements in the second task. To test this Mann-Whitney U-Tests were performed but there was no evidence of any difference between the two tasks for any of the statements (p-value > 0.05). Finally the majority of the participants agreed that it was easy to find annotations.

During the informal interviews most subjects stated they liked how vsInk provided the freedom to do any annotations they wanted. There was also general agreement that ink annotations stood out better than inline text comments and were much faster to find. However some subjects found that the annotations obstructed the underlying code, making it harder to read. While most of the participants understood why vsInk grouped the strokes together they thought the grouping routine was too inaccurate. In addition to improving the grouping the some suggested improvements were being able to selectively hide annotations (maybe by colour), having some form of zoom for the code under the pen and displaying the code in the navigator window.

6 Discussion and Future Work

With vsInk we have integrated annotation ink into the Visual Studio code editor. The annotations sit over the code and all the existing functionality of the editor is still retained. This is possible because Visual Studio now exposes extension points to allow modifying the actual code editor. However this is still a non-trivial task due to the way the editor works.

The other main technical challenge for vsInk was how to correctly position the annotation relative to the code. Initially the challenge was to integrate with the Visual Studio editor so the annotation always behaved as expected. The first attempts relied on the functionality in Visual Studio doing most of the work. This proved fruitless and we changed to having vsInk do most of the work with the positioning. Once the positioning was working correctly the next challenges were with usability.

The usability study identified two major challenges with combining annotations and code. The first challenge is how to group strokes together into annotations, which if incorrect, in turn causes problems with the repositioning. The second challenge was tall annotations would disappear un-expectedly.

Previous research has shown that grouping strokes is not easy (e.g. Shilman et al., 2003, Shilman and Viola, 2004, Wang et al., 2006). Part of the challenge is the huge
variety of different types of annotations, both from the same person and between people (Marshall, 1997). Trying to find a simple set of rules that can handle this variety is always going to result in failures.

Annotations in vsInk are built as the strokes are added or removed – strokes are only added to a single annotation. This could potentially be one reason why the grouping is inaccurate – people do not always add strokes to an annotation in order. Another limitation is vsInk does not use any contextual information. The only contextual information used in deciding to group strokes is the location of the other annotations. However program code itself is a rich source of contextual information that can potentially be used to enhance grouping. For example, when a person is underlining they tend to stay reasonably close to the bottom of the text. If they then start a new underline on the next line of code it is most likely to be a new annotation, not an extension of an existing one. The same applies for higher levels of code structure – e.g. a stroke in a method is more likely to belong to an annotation within the same method than outside.

The other main usability issue that was not resolved is tall annotations tended to disappear. This is caused by vsInk using a single anchor point for each annotation. While this is acceptable for short annotations it fails as annotations increase in height. Since Line is not visible vsInk hid the entire annotation. While this did not happen very often (only 4 annotations out of the 194 had this problem) it does happen in a specific scenario – using arrows to indicate code should be moved to another location. One approach mentioned in previous research is to break a tall annotation into shorter segments (Golovchinsky and Denoue, 2002) with each segment having its own anchor. This approach was considered but a trial implementation uncovered a flaw with the approach. Since the underlying code can still be edited it was possible to add or delete lines within a tall annotation which broke the anchoring. A decision was made to focus on the grouping instead and this functionality was removed.

One of the interesting suggestions during the user study was to include some way of zooming the interface during inking. This approach has been attempted before with DIZI (Agrawala and Shilman, 2005). DIZI provided a pop-up zoom region that automatically moved as the user annotated. When the user finished annotating the annotation would be scaled back to match the size of the original. A user study found the zooming was most useful when there was limited whitespace. This may be useful for annotating code, especially for dense “blocks” of code.

Some possibilities for future work are improving how strokes are grouped together into annotations, handling tall annotations, adding zooming, and implementing missing “common” functionality. Missing “common” functionality includes cut/paste and the undo history. Some fixes for the grouping issue include using contextual information in the rules, using data mining for generating the grouper and using a single-pass grouper. Possible solutions for handling tall annotations include segmenting annotations and adding a mechanism for handling line insertion and deletions. There is also a need for additional user studies around how people would actually use annotations in a code editor.

7 Conclusions
This paper presents vsInk, a tool for annotating program code within Visual Studio using digital ink. This is the first tool that fully integrates digital ink into a code editor in an IDE. It is also the first tool to provide support for annotating within collapsible code regions. The usability study showed that overall vsInk is easy and enjoyable to use. There are two significant issues uncovered during the user study that are yet to be addressed – how to group strokes into annotations and tall annotations disappearing unexpectedly.

Because of the functional deficiencies in earlier prototypes there has been little work on assessing the value of annotating program code. We look forward to exploring real-world user experiences with vsInk.

8 References


computer-human interaction special interest group (CHISIG) of Australia on Computer-
human interaction: design, activities, artifacts and environments - OZCHI ’07. New York, New
York, USA: ACM Press.

symposium on User interface software and technology - UIST ’02. New York, New York,
USA: ACM Press.

Lichtschlag, L. and Borchers, J. (2010): CodeGraffiti: Communication by Sketching for Pair
Programming. Adjunct proceedings of the 23nd annual ACM symposium on User interface


Digital Display Surfaces for Active Reading Tasks. Workshop on Horizontal Interactive

New York, New York, USA: ACM Press.

multi-user ink annotation in web browsers. AUIC ’10 Proceedings of the Eleventh
Australasian Conference on User Interface. Darlinghurst, Australia: Australian Computer
Society, Inc.

ACM SIGCHI New Zealand chapter’s international conference on Computer-human
interaction design centered HCI - CHINZ ’06. New York, New York, USA: ACM Press.

free form digital ink annotations. Proceedings of the SIGCHI conference on Human factors in


Eurographics Workshop on Sketch-Based Interfaces and Modeling (SBIM). Grenoble, France.

York, USA: ACM Press.

Documents. Sketch Based Interfaces and Modeling - SBIM. Eurographics.

Wilcox, L. D., Schilit, B. N. and Sawhney, N. (1997): Dynomite: a dynamically organized ink and
audio notebook. Proceedings of the SIGCHI conference on Human factors in computing
Supporting Informed Decision-Making under Uncertainty and Risk through Interactive Visualisation

Mohammad Daradkeh, Clare Churcher, Alan McKinnon

PO Box 84 Lincoln University
Lincoln 7647 Canterbury, New Zealand

{Mohammad.Daradkeh, Clare.Churcher, Alan.McKinnon}@lincoln.ac.nz

Abstract
Informed decisions are based on the availability of information and the ability of decision-makers to manipulate this information. More often than not, the decision-relevant information is subject to uncertainty arising from different sources. Consequently, decisions involve an undeniable amount of risk. An effective visualisation tool to support informed decision-making must enable users to not only distil information, but also explore the uncertainty and risk involved in their decisions. In this paper, we present VisIDM, an information visualisation tool to support informed decision-making (IDM) under uncertainty and risk. It aims to portray information about the decision problem and facilitate its analysis and exploration at different levels of detail. It also aims to facilitate the integration of uncertainty and risk into the decision-making process and allow users to experiment with multiple “what-if” scenarios. We evaluate the utility of VisIDM through a qualitative user study. The results provide valuable insights into the benefits and drawbacks of VisIDM for assisting people to make informed decisions and raising their awareness of uncertainty and risk involved in their decisions.

Keywords: Information visualisation, Interaction design, Informed decision-making, Uncertainty, Risk.

1 Introduction
Decision-making is a central activity of human beings as situations that require making decisions constantly arise in almost all endeavours of their lives. All decisions, whether personal, business, or professional, are likely to bring about some future benefits to someone or something and involve choices. Some decisions such as which company’s shares to buy, involve making a choice among multiple alternatives while others such as whether or not to invest in a new product are more “yes/no” decisions. Whatever the type of decision, the information available is considered a key element in the decision-making process as it provides the basis for making informed and reasoned decisions.

Ubiquitous in realistic situations, the information on which decisions are based is often subject to uncertainty arising from different sources. Typical sources include the lack of knowledge of true values of decision variables/parameters and future possibilities and outcomes. For example, the decision about whether to invest in a new product depends on the uncertain market conditions (e.g. whether the demand will go up or down). The possible outcomes of the decision (e.g. making profit or loss) are also dependent on how much the demand goes up or down and its interaction with other variables (e.g. the price of the product). In this situation, the decision-maker usually evaluates the possible outcomes and their associated likelihood under different scenarios, and bases his or her decisions on this evaluation. Such decisions are inherently risky as the best alternative will generally involve some chance of undesirable outcomes.

Ignoring uncertainty and its associated risk may simplify the decision-making process, but it does not result in making informed decisions. Thus, the uncertainty should be explicitly considered from the beginning of the decision-making process as an integral part of the information on which decisions are based. However, the integration of uncertainty into the decision-making process poses significant cognitive challenges. It brings additional complexity and confusion to the task of decision-making which is already complicated. One example of such confusion occurs when comparing or ranking multiple alternatives, each with a range of possible outcomes. Moreover, the process of integrating uncertainty into the decision-making process is a highly technical subject, and often not transparent or easy to grasp by decision-makers who lack the necessary numerical skills.

Information visualisation can play an important part in assisting people to make informed decisions under uncertainty and risk. It provides an effective means for depicting information in ways that make it amenable to analysis and exploration. It also can facilitate the integration of uncertainty into the decision-making process and raise the awareness of decision-makers about its effect. Moreover, it can enhance the ability of decision-makers to process and comprehend information, thereby making more informed decisions (Tegarden, 1999; Zhu & Chen, 2008).

In this paper, we present an information visualisation tool, called VisIDM, for assisting people to make informed decisions under uncertainty and risk. The intention of VisIDM is to portray information about the key elements of the decision problem and facilitate their analysis and exploration at different levels of detail. It is also intended to facilitate the integration of uncertainty
and risk into the decision-making process and allow users to experiment with multiple “what-if” scenarios.

The remainder of this paper is organised as follows. Section 2 discusses some related work in the area of information visualisation to support decision-making. Section 3 discusses the requirements and considerations underpinning the design of VisIDM. Section 4 describes the main components of VisIDM and demonstrates its practical use through an application example of a financial decision-making problem. Section 5 briefly describes a qualitative user study conducted to evaluate the usefulness of VisIDM. In this section, a summary of the results is presented while details of the results are reported and discussed elsewhere (Daradkeh, 2012). Finally, Section 6 concludes the paper and outlines some perspectives for future work.

2 Related Work

Several information visualisation tools that claim to be helpful in decision-making have been developed in many different areas. For example, the TreeMap (Asahi et al., 1995), a visualisation tool for hierarchical data spaces, has been applied to support decision-making based on the Analytical Hierarchy Process (AHP) developed by Saaty (1980). AHP is a multi-criteria decision-making approach that decomposes the decision problem into a hierarchical structure with three main levels: the decision space, the criteria of evaluation, and the available alternatives. The decision space is represented by the entire area (the base rectangle) of the TreeMap. For each evaluation criterion, the screen area is sliced (either horizontally or vertically) to create smaller rectangles with areas proportional to their relative importance or weight. Each criterion is then diced into sub-criteria recursively, with the direction of the slicing switched 90 degrees for each level. The most interesting feature of the TreeMap is that adjusting weights for criteria is possible by resizing the areas of the rectangles. The total score for each alternative is automatically calculated based on the AHP and presented as a horizontal bar.

Dust & Magnet (Yi et al., 2005) has been applied to support the multi-attribute decision-making based on the weighted additive (WADD) decision rule (Keeney et al., 1999). Using the WADD rule, each alternative is given a total score based on multiplying the value of each attribute with its relative importance (subjective weight or probability) and summing these weighted attribute values. The alternative with the “best” score is chosen as the optimal solution. Using Dust & Magnet, the attributes are represented as black squares and work as magnets, whereas the alternatives are represented as black dots and work as dust particles. The Dust & Magnet metaphor is an intuitive representation of the weighted additive (WADD) decision rule. In addition, it is engaging and easy to understand because it involves animated interaction (Yi, 2008).

Another visualisation tool that has been designed to support decision-making based on the weighted additive decision rule (WADD) is ValueCharts+ (Bautista & Carenini, 2006). It displays the decision alternatives and evaluation attributes in a tabular paradigm, where each row represents an alternative and each column represents an attribute. It uses horizontal bars to represent the weighted value of a particular attribute (i.e. its value multiplied by its relative weight). These bars are then accumulated and presented in a separate display in the form of horizontal stacked bars, representing the total score of each alternative.

Decision Map and Decision Table (Yi, 2008) are two multivariate visualisation tools that have been developed based on ValueCharts+. These two tools were developed to complement each other in supporting a decision-making problem related to selecting a nursing home based on a set of attributes. The Decision Map is inspired by HomeFinder (Williamson & Shneiderman, 1992) and uses a web-based interactive map similar to Google Maps1. It provides geographic information related to the alternatives (i.e. nursing homes). Conversely, the Decision Table displays the information in a tabular form with rows representing the available alternatives and columns representing their attributes. Similar to ValueCharts+, it uses horizontal bars to represent the weighted values of attributes.

Despite the availability of several information visualisation tools to support decision-making, the uncertainty and risk have often been neglected or treated in a superficial way. Most of the information visualisation tools are designed and applied based on the assumption that the information available to decision-makers is deterministic and free of uncertainty. Thus, each decision alternative leads to a specific, known outcome and there is no risk involved in decision-making. Such precise knowledge, however, is rarely available in practice. Most real-world decision problems typically involve uncertainty and risk which if not considered could result in infeasible and less informed decisions.

Owing to the nature of decision-making under uncertainty and risk, information visualisation to support decision-making faces special challenges such as dealing with uncertainty and its integration into the decision-making process. Focusing on this area of research, the next section discusses the information requirements and considerations that need to be addressed when designing information visualisation tools to support informed decision-making under uncertainty and risk.

3 Requirements and Design Considerations

3.1 Information Requirements

Decision-making under uncertainty and risk is usually described as a process of choosing between alternatives, each of which can result in many possible outcomes. These outcomes reflect the uncertain and stochastic nature of decision input variables and their propagation through models and criteria used in the decision-making process. Typically, not all possible outcomes are equally desirable to the decision-maker. Consequently, risk accompanies decisions because there is a chance that the decision made can lead to an undesirable rather than a desirable outcome. From this description, there are four basic elements of the decision problem under uncertainty and risk. These are: 1) the set of alternatives from which a

1 http://maps.google.com
preferred alternative is chosen; 2) the input data and their associated uncertainties; 3) the range of possible outcomes associated with each alternative and their probabilities; and 4) the risk of obtaining undesirable outcomes involved in each alternative. All these elements should be taken into consideration when designing information visualisation tools to support informed decision-making. This is because in the presence of uncertainty and risk, decision-makers usually base their decisions not only on the possible outcomes but also on the uncertainty and risk each alternative entails.

3.2 Analysis and Exploration of Alternatives at Different Levels of Detail

In addition to the aforementioned information, decision-makers need to be able to explore and compare alternatives at different levels of detail. The presence of uncertainty in the values of input variables implies that there are many possible realisations (or values) for each input variable. This gives rise to the presence of many possible scenarios, where each scenario represents a possible combination of all values of input variables, one for each variable (Marco et al., 2008). In this situation, the visualisation tool should allow the generation of all possible scenarios. This requires facilities for enabling decision-makers to provide their own estimates of the values for each uncertain variable and its distribution. In addition, it requires computational facilities for propagating all uncertainties through models and criteria used in decision-making. Once all uncertainties are propagated through the models, the visualisation tool should then provide decision-makers with a complete picture of all generated scenarios and the distribution of uncertainties and risks anticipated to exist in these scenarios. At the same time, it should allow decision-makers to interact with the decision model to allow experimentation with different possible “what-if” scenarios and exploration of the outcomes and risks associated with alternatives under these scenarios. The ability to analyse “what-if” scenarios is a key requirement for developing understanding about the implications of uncertainty, which in turn leads to making more informed and justifiable decisions (French, 2003).

3.3 Integration of Uncertainty and Risk into the Decision-Making Process

If uncertainty is integrated into the decision-making process, the criteria used to assess the performance of decision alternatives should reflect this. It’s widely recognised that, in the presence of uncertainty, the risk of obtaining undesirable outcomes is a frequently used criterion for exposing the effect of uncertainty and evaluating the decision alternatives (Maier et al., 2008). This is because the risk of obtaining undesirable outcomes offers a clear way to make sense of uncertainty and address it explicitly in the decision-making process (Keeney et al., 1999).

Our approach to making uncertainty an integral part of decision-making is to view the whole process as one of determining the risk associated with the decision. This approach is shown in Figure 1 where decision-makers specify the risk criterion to be used and also the uncertainty for each input variable. For example, in the case of considering an investment decision problem, the two components of the risk might be the probability of making a loss and the amount of money that could be lost as a consequence of making a decision. The decision-maker is then interested in both the risk that the investment will make a loss, and how that risk is affected by his or her knowledge of the uncertainties in the variables relating to this particular investment.

![Figure 1: The proposed approach for incorporating input uncertainty into the decision-making process.](image)

4 Description of VisIDM

Based on the requirements and considerations discussed above, we have designed VisIDM which consists of two main parts: Decision Bars and Risk Explorer as shown in Figure 2. The left side of Figure 2 shows the Decision Bars which provide overview information on the available alternatives, their range of possible outcomes, and the overall risk of undesirable outcomes associated with each alternative. The right side of Figure 2 shows Risk Explorer which provides decision-makers with a detailed view of the alternatives and allows them to explore the uncertainty and risk associated with these alternatives at different levels of detail.

In the following sections, we describe the components of VisIDM in more detail and demonstrate its practical use through an application example of a financial decision-making problem.

4.1 Application Example: Financial Decision Support

The example problem to be explored and visualised is a decision-making scenario of choosing an investment based on uncertain information. Some examples of such a scenario include the decision on whether or not to buy a property for investment and rental income, or a decision to select from among a set of projects available for investments. In making such decisions, decision-makers usually specify evaluation criteria (e.g. a potential profit and an acceptable risk of making a loss associated with the investment). The decision-makers also define the key variables that influence the evaluation criteria and their possible values (e.g. the income from the investment and its running cost). Then, they use a financial model to predict and evaluate the profitability of the investment under multiple scenarios and base their decisions on this evaluation (Tziralis et al., 2009).
To predict and analyse the profitability of an investment, a financial model for investment decision-making called Net Present Value (NPV) is commonly used (Magni, 2009; Tziralis et al., 2009). The NPV model is emphasised in many textbooks as a theoretically and practically sound decision model (e.g. Copeland & Weston, 1983; Koller et al., 2005). It represents the difference between the present value of all cash inflows (profits) and cash outflows (costs) over the life of the investment, all discounted at a particular rate of return (Magni, 2009). The purpose of NPV is basically to estimate the extent to which the profits of an investment exceed its costs. A positive NPV indicates that the investment is profitable, while a negative NPV indicates that the investment is making a loss. A basic version of calculating NPV is given by Equation 1:

$$NPV = C_0 + \sum_{t=1}^{n} \frac{(CI_t - CO_t)}{(1 + r)^t}$$

Where

- $C_0$ is the initial investment.
- $n$ is the total time of the investment.
- $r$ is the discount rate (the rate of return that could be earned on the investment).
- $CI_t$ is the cash inflow at time $t$.
- $CO_t$ is the cash outflow at time $t$.

As shown in Equation 1, in its basic form, the NPV model consists of five input variables. In practice, each of these variables is subject to uncertainty because the information available on their values is usually based on predictions, and fluctuations may occur in the future. Consequently, the investment decision can lead to many possible outcomes (i.e. different values of NPV). Since not all possible outcomes are equally desirable to the decision-maker, the investment decision involves a degree of risk. The risk is present because there is a chance that the investment decision can lead to an undesirable rather than a desirable outcome.

### 4.2 Decision Bars

As shown in Figure 3 from top to bottom, the Decision Bars interface consists of three panels: Outcome, Risk and Likelihood Bars.

The Outcome Bars shown in the top panel of Figure 3 present the decision alternatives, each of which is visualised by a bar with a different colour. The length of the bar represents the range of possible outcomes associated with the corresponding alternative. The black part of each bar represents the mean value of possible outcomes. The dashed blue line along each bar represents the probability distribution of possible outcomes.

The Outcome Bars enable the user to identify the worst and best possible outcomes for each alternative. For example, in the top panel of Figure 3, the decision-maker can identify that alternative 5 has the largest potential gain and also the largest potential loss. The Outcome Bars also help in distinguishing the proportion of desirable (or positive) outcomes from undesirable (or negative) outcomes for each alternative. For example, the Outcome Bars in Figure 3 show that more than half of the NPVs of alternative 1 may result in making a loss (NPV < 0), whereas most of the NPVs for alternative 4 result in making a profit (NPV > 0). The probability distribution of possible outcomes (the dashed blue line) enables the user to identify the relative likelihood of occurrence of
possible outcomes. For example, the dashed blue line of alternative 4 is skewed to the top showing that the higher outcomes are more likely.

The Risk Bars shown in the middle panel of Figure 3 provide information on the overall risk of obtaining undesirable outcomes (in this case, the probability of obtaining negative NPVs). The risk associated with each alternative is shown as a vertical bar. The height of the bar represents the degree of risk (i.e. the probability of undesirable outcomes). The higher the bar, the higher the risk of obtaining undesirable outcomes. For example, the middle panel in Figure 3 shows that among all possible outcomes of alternative 4 about 5% will result in a loss compared to about 13% in alternative 2.

The Likelihood Bars provide information on the likelihood of a particular alternative having the highest outcome. In other words, these bars show the percentage of outcomes of a particular alternative that are better than all outcomes of other alternatives. The higher the bar, the higher the percentage. For example, the bottom panel of Figure 3 shows that about 40% of the outcomes (NPVs) of alternative 5 are higher than all outcomes (NPVs) of other alternatives.

4.3 Risk Explorer

Risk Explorer, shown in Figure 4, adds to the other parts of VisIDM a visualisation tool for exploring and analysing the uncertainty and risk associated with available alternatives at different levels of detail. It allows the user to specify the range of values for each input variable through the corresponding text boxes. Then, it portrays the distribution of risk (i.e. the probability of undesirable outcomes) in a uniform grid layout. The grid also displays the range of possible values of each input variable divided into a number of divisions (cells in the grid).

Risk Explorer uses colour to convey the risk of undesirable outcomes. The colour of each cell in the grid conveys the degree of risk (i.e. the probability of undesirable outcomes) associated with the alternative based on the variable’s value shown in the cell. Yellow means no risk (i.e. the probability of obtaining undesirable outcomes = 0). Dark orange represents the highest risk (i.e. the probability of obtaining undesirable outcomes = 1). The risk of undesirable outcomes is calculated based on fixing the value in the cell and taking every possible value of all other variables and calculating what proportion of these combinations will result in undesirable outcomes. The numerical values of the risk of undesirable outcomes can also be retrieved by hovering over the cells. For example, the popup window in Figure 4 shows that if the discount rate is 10% then if we consider all other possible combinations of values for the other input variables about 78% (probability 0.778) will result in an undesirable outcome of a loss.

Risk Explorer also displays the range of possible outcomes resulting from the uncertainties in the input variables as horizontal red/green bars (see Figure 4). The range of possible outcomes is calculated by allowing all input variables to vary within their ranges of values and calculating all possible combinations of these values. The horizontal red/green bar informs the user about the maximum and minimum potential outcomes under all possible scenarios (i.e. all possible combinations of the variables values). In addition, by observing the red part of the bar, the user can identify the proportion of undesirable outcomes (e.g. the negative NPVs that will make a loss as in the example shown in Figure 4). Conversely, he/she can identify the proportion of desirable outcomes (e.g. the positive NPVs that will make a profit) by observing the green part of the bar.

As shown in Figure 4, Risk Explorer displays the information in a uniform grid which facilitates the presentation of the uncertainty and associated risk of undesirable outcomes in an organised way. It makes it easy to see and follow the change in the risk degrees across the cells, which in turn facilitates the recognition of trends and relationships between the uncertain values of input variables and the risk of undesirable outcomes. Furthermore, all input variables are bounded by known maximum and minimum values and all possible values in between are discretised into a finite number of divisions. Therefore, they can be mapped onto equal-sized cells. In this way the decision-maker can run through or compare several scenarios with various values and easily determine the risk level at various degree of uncertainty. Colour was chosen for the purpose of presenting risk of undesirable outcomes because it is widely used for risk visualisation and communication. In addition, it is an important visual attention guide that can highlight levels of risk (Boström et al., 2008).

4.3.1 Providing an Overview of the Uncertainty and Risk of Undesirable Outcomes

Risk Explorer provides an overview of all possible scenarios (i.e. possible values of input variables) and the risk of undesirable outcomes associated with the decision alternative under these scenarios. By observing the colour variation across the grid cells, the decision-maker can quickly and easily get an overview of the risk of undesirable outcomes and its distribution. The decision-maker can use this overview to compare alternatives in terms of the risk involved in each alternative before focusing on a specific set of scenarios. For example, as shown in Figure 5, when comparing alternatives 1 and 2, the decision-maker can recognise that the risk of making a loss associated with alternative 1 is much higher than that associated with alternative 2; the colour of many cells in the grid of alternative 1 is much darker than that of alternative 2. The same overview information can also be obtained from the Decision Bars interface (see Figure 3). However, Risk Explorer provides an explanation of the factors that form the risk of undesirable outcomes associated with the decision alternatives.
4.3.2 Analysis and Comparison of Multiple Alternatives at Several Levels of Detail

Risk Explorer allows the user to focus on particular scenarios (i.e. specific values of input variables) and compare alternatives under these scenarios. To focus on a specific scenario, the decision-maker needs to fix the values of input variables that represent the scenario. This can be done by clicking on the cell containing a specific value of one of the input variables. This will open up a new grid showing the new range of risk of undesirable outcomes with this value fixed. Values of other input variables in the new grid can also be fixed. For example, Figure 6 shows an example of exploring and analysing alternatives 2 and 5 under specific scenarios based on fixing the two input variables initial investment at $35000 and discount rate at 10%. As shown in Figure 6, the first fixed value of $35000 in the top grid is highlighted and a new grid is shown for each alternative. The new grid shows the risk values for the other three input variables. The risk values are calculated by fixing the values in the highlighted cells and taking every possible value of the other variables and calculating what proportion of these combinations will result in undesirable outcomes. This process is then repeated by fixing the discount rate to 10% in the second grid. In addition to the resulting grid, a new red/green bar is shown to the right of the grid for each alternative. The red/green bar shows the range of possible outcomes resulting from fixing the variables’ values in the highlighted cells while varying the other variables within their ranges of values.

Based on the resulting grids and red/green bars, the decision-maker can evaluate and compare alternatives in terms of the risk of undesirable outcomes and the range of possible outcomes under different scenarios. For example, the new grids and red/green bars in Figure 6 show that if the two input variables initial investment and discount rate are fixed at $35000 and 10% respectively, then about (27%) of NPVs of alternative 2 will result in a loss compared to about 20% for alternative 5 (see the popup windows shown in Figure 6). Conversely, according to the red/green bars, the maximum loss and profit potential associated with alternative 5 (-$16046, $40816 respectively) are greater than those associated with alternative 2 (-$8464, $21862 respectively).
5 User Study

We conducted a qualitative user study to explore how VisIDM was used by participants and what features supported their exploration and perception of information. Twelve postgraduate students (2 females and 10 males) from different departments in the Faculty of Commerce at Lincoln University were recruited. The number of participants was not predetermined before the initiation of the study, but rather was determined by reaching a saturation point (Patton, 2005). Recruitment ceased when the information being collected became repetitive across participants and further information and analysis no longer yielded new variations.

5.1 Setup and Procedure

The study was setup in a lab-based environment. A case study of an investment decision-making problem under uncertainty and risk that was relevant to the knowledge and experience of the participants was utilised in this study. The decision problem consisted of five investment alternatives. The data was prepared so that each investment alternative had a different risk/profit profile. Because all alternatives involved the investment of dollars, the Net present Value (NPV) model was used for evaluating and comparing the profitability of alternatives (refer to Section 4.1 for a description of NPV model). We put the participants in the situation of making decisions taking into account the uncertainty and risk associated with each alternative.

The procedure used in this study was as follows: the participants were given a brief introduction to VisIDM and the study procedure. Then, they were given a set of practice tasks to familiarise themselves with VisIDM. After completing the practice tasks, the participants were given a scenario for decision-making consisting of a set of investment alternatives. Then, they were asked some open-ended questions where they had to make decisions taking into consideration the uncertainty and risk associated with each alternative. We designed the questions to be of an open-ended nature because we were not intending to quantitatively record the performance of our participants, but rather have them exercise all parts of VisIDM and get their feedback on its utility.

The following open-ended questions were given to the study participants:

- What do you think are the best two alternatives? (Ranking problem)
- From among your best two alternatives, which alternative do you prefer the most? (Choice problem)
These questions were designed to be consistent with the ultimate objectives of decision-making. Generally, decision-makers are interested in either choosing one alternative (a choice problem) or obtaining an order of preferences of the alternatives (a ranking problem) (Nobre et al., 1999). To achieve these ultimate objectives, the participants had to utilise different types of information provided by VisIDM and perform several tasks.

While they solved the open-ended questions, the participants were instructed to follow a think-aloud protocol. Data was collected using observations and content analysis of participants’ written responses and answers of open-ended questions. Each session lasted from approximately 90 to 120 minutes.

5.2 Results and Discussion

The results of the study provide valuable insights into the usefulness of each feature of VisIDM for informed decision-making under uncertainty and risk. They allow us to shed light on how the participants utilised the given interactions and visual representations of information to arrive at their final decisions. They also allow us to explore how VisIDM affected their perception and interpretation of the uncertainty and risk information.

5.2.1 Decision-Making Processes

The results show that the participants were able to perform several tasks to arrive at their final decisions. Examining these tasks, we note that the participants adopted different strategies for decision-making. For example to decide on whether one alternative is better than another, some participants compared them first based on the maximum NPV, which was interpreted as the maximum profit potential. Then, they further compared them based on the minimum NPV, which was interpreted as the maximum loss potential. At this point, they stopped searching for further cues and made their decisions based on the maximum and minimum NPV values. Other participants preferred to continue searching the visualisation interfaces for other information (e.g., proportions of positive and negative NPVs) and made comparisons based on this information. This result supports the proposition that people rarely appraise and use all the information conveyed by these bars. For example, one participant commented: “Initially I thought that the likelihood bars would be helpful, but they didn’t add much to the previous information. Also, I found them confusing.” Another participant commented: “The Likelihood Bars adds more information but it can be misleading and it’s difficult to utilise information of the likelihood bars.” The Likelihood Bars could be eliminated from future versions of VisIDM and replaced by something easier to understand and use. For example, it could be a useful idea to replace the Likelihood Bars by bars that present information about the probability of obtaining desirable outcomes. This would allow VisIDM to provide more balanced presentation of potential risks and benefits of available alternatives, thus allowing decision-makers to make better informed decisions.

Risk Explorer was used by all participants to get an overview of the risk associated with alternatives through colour coding. Prior to focusing on specific scenarios, all participants made comparisons between alternatives in terms of the risk of making a loss based on an overview of all possible scenarios. They also used the horizontal red/green bars to compare alternatives in terms of their profit and loss potential.

Risk Explorer was also used to analyse and compare the uncertainty and risk associated with alternatives under particular set of scenarios. Some participants made comparisons between alternatives in terms of the risk of making a loss and profit potential under similar-value scenarios (e.g., similar amount of initial investment). To do so, they identified and fixed similar or nearly similar values of one or more variables. Then, they explored and analysed the resulting risk of making a loss and range of outcomes (i.e. range of possible NPV values) of alternatives based on the selected scenarios. Other participants made comparisons between alternatives in terms of the risk of making a loss and profit potential under similar-case scenarios (e.g., worst-case or best-case scenarios). For example, one participant made a comparison between alternatives under pessimistic
(worst) and optimistic (best) estimates of cash inflow. Other participants used different variables (e.g. one participant made a comparison between alternatives under worst and best initial investment). Some participants also made comparisons between alternatives under worst and best cases of more than one variable. For example, one participant made a comparison of alternatives in terms of the risk of making a loss and profit potential based on fixing the cash inflow at the minimum value and discount rate at the maximum value.

The use of colour gradations to convey risk magnitudes enabled participants to compare alternatives when they have different risk profiles; i.e. when the difference between the risk of making a loss with one alternative and the risk of making a loss with another was clear and can be distinguished. This suggests that the use of colour to represent the risk (in this case, the probability of making a loss) can be useful for attracting and holding people’s attention. However, in many scenarios, the participants were not able to compare alternatives in terms of the risk of making a loss by observing the colour variation across the cells; particularly, when the scenarios had similar risk profiles. In such cases, the participants relied on the red/green bars to identify the risk of making a loss. In particular, the participants used the maximum potential loss (i.e. minimum NPV), and the proportion of negative NPV values (the red part of the resulting bars) to form their impressions about the risk, regardless of probability.

### 5.2.2 Risk Perception and Interpretation

The results show that the participants have problems in understanding and interpreting the uncertainty and risk information. In particular, they have a tendency to ignore the importance of probability information and rely, in large part, on the values of undesirable outcomes to form their impression about the risk.

Using the Outcome Bars interface, most participants did not use the probability distribution to evaluate the risk of undesirable outcomes associated with each alternative. Rather, they focused their attention on the minimum possible NPV, which represents the maximum potential loss. Consequently, they perceived the alternative with higher potential loss as more threatening than that with lower potential loss, regardless of probability. The same issue of risk perception was also observed when the participants used Risk Explorer. Some made use of the red/green bars, which show the range of possible outcomes to evaluate the risk of making a loss. Others evaluated the risk by observing the colour variation across the cells of the grids. Interestingly, the majority of participants did not try to retrieve numerical values of the risk (i.e. the probability of making a loss), although they clearly understood how to do so in the practice phase of this study.

The literature on risk perception and decision-making suggests several possible explanations for the observed issue of risk perception; i.e. ignoring the importance of probability and relying on the outcomes to form the impression about the risk. Some of these possible explanations seem consistent with the observed risk perceptions of participants in this study. In the case of the Outcome Bars interface, it seems that the way information pertaining to the risk was presented led to the outcomes being made more prominent and easier to identify than their probabilities. Consequently, the participants focused their attention on the outcomes rather than their probabilities. This explanation seems consistent with previous research suggesting that prominent information is more likely to draw attention, be given more consideration, and have a stronger effect on risk-related behaviour than less prominent information (Stone et al., 2003). A second possible explanation for the observed issue of risk perception could be related to the attitude of the participants towards the risk. The majority of participants showed a preference for minimising the loss rather than maximising the profit. This might lead them to overestimate the risk involved in the alternatives with high potential loss. This bias in estimating the risk has been previously reported in the graphics perception literature, suggesting that people are poor at estimating “objective risk” (Stone et al., 2003). They have a tendency to perceive the low probability/high consequence outcomes as more risky than high probability/low consequence outcomes (Schwartz & Hasnain, 2002).

### 6 Conclusions and Future Work

This paper presents an information visualisation tool to support informed decision-making under uncertainty and risk called VisIDM. It consists of two main parts: the Decision Bars and Risk Explorer. Decision Bars provide overview information of the decision problem and available alternatives through three panels: Outcome, Risk and Likelihood Bars. Using these bars, decision-makers can compare and then choose preferred alternatives before focusing on particular alternatives for detailed analysis and exploration. On the other hand, Risk Explorer provides decision-makers with a multivariate representation of uncertainty and risk associated with the decision alternatives. Using Risk Explorer, decision-makers can interactively analyse and explore the available alternatives at different levels of detail.

To explore the benefits and drawbacks of each feature of VisIDM, we have conducted a qualitative user study. The results suggest that VisIDM can be a useful tool for assisting people to make informed decisions under uncertainty and risk. It provides people with a variety of decision-relevant information and assists them in performing several tasks to arrive at their final decisions. It also can make people aware of the uncertainty and risk involved in their decisions.

Participants’ feedback confirmed that further research is needed to improve the design of VisIDM, so that it provides decision-makers with a better understanding of uncertainties and risks associated with decision-making. Some participants found it difficult to make use of probability distribution information. Hence, it could be improved so that it provides the probability information in a clearer and more informative format. Some alternative formats for portraying the probability information are available in the literature on risk visualisation. For example, cumulative distribution functions, histograms, and box plots can show different
types of information that people usually seek for decision-making purposes (Gresh et al., 2011). It would be useful to explore whether these formats can provide probability information in a more intuitive way. Perhaps, though, there is a need to develop much more innovative approaches for conveying probability information.

More evaluation studies are also needed to provide more evidence of the usefulness of VisIDM to support informed decision-making under uncertainty and risk. These studies should be expanded beyond hypothetical decision-making scenarios and lab-based environment to real world settings. They should also be expanded to include different measures and factors related to informed decision-making such as measures of beliefs, attitudes, perception of risk, and knowledge (Bekker et al., 1999).

Acknowledgements

We would like to acknowledge all participants without whom the study would not have been completed.

7 References


Metadata Manipulation Interface Design

Stijn Dekeyser  Richard Watson

Department of Mathematics and Computing
University of Southern Queensland
Toowoomba, Australia
{dekeyser,rwatson}@usq.edu.au

Abstract

Management of the increasingly large collections of files and other electronic artifacts held on desktop as well as enterprise systems is becoming more difficult. Organisation and searching using extensive metadata is an emerging solution, but is predicated upon the development of appropriate interfaces for metadata management. In this paper we seek to advance the state of the art by proposing a set of design principles for metadata interfaces. We do this by first defining the abstract operations required, then reviewing the functionality and interfaces of current applications with respect to these operations, before extending the observed best practice to create a generic set of guidelines. We also present a novel direct manipulation interface for higher level metadata manipulation that addresses shortcomings observed in the sampled software.

1 Introduction

Computer users of all kinds are storing an ever increasing number of files (Agrawal et al. 2007). The usage ranges from the straightforward personal storage of generic media files to the specialised storage of outcomes of scientific observations or simulations and includes diverse and increasingly mandated archival storage of corporate and government agency documents.

While the increasing aggregate size of stored files presents significant challenges in storing the bit-streams (Rosenthal 2010), there are other important and complex issues related to the growing number of files, most prominently the attendant problem of (a) organising and (b) locating individual files within a file store. The traditional hierarchical file system is no longer able to support either the kinds or organisation or the search strategies that users need (Seltzer & Murphy 2009). Alternate, post-hierarchical file system architectures have been proposed (e.g. Ames et al. 2006, Dekeyser et al. 2008, Gifford et al. 1991, Princeleau & Ridoux 2003, Rizzo 2004, Seltzer & Murphy 2009) whose logical organisation is based on a rich collection of file metadata rather than the familiar nested directory structure.

This problem—how to organise and find growing numbers of electronic artifacts—extends beyond the desktop file system. A huge number of files are now stored in cloud-based systems, and non-file objects such as email have very similar characteristics (increasing quantity, need to organise and locate) to files.

We believe that metadata-based systems hold the key to designing better ways of managing our burgeoning collections of electronic things. Informally, metadata is a collection of attributes and corresponding values that is associated with a file or object. While all systems that manipulate objects will create some metadata such as creation time, and others can extract metadata such as keywords from the objects themselves, we focus here on metadata that the user can create or modify.

We will utilize the term user-centric metadata to refer to values provided by users and associated with predefined named attributes. In other words, the structure of such metadata (also known as schema) is considered to be fixed while its instance may be modified by users. User-centric metadata is a subset of the richer user-defined metadata where the user may define new attributes as well as the associated values.

Motivation

The post-hierarchical file systems cited earlier rely on the use of metadata to organise and search large collections of files. If we assume that a file system has a complete set of information for all user-centric metadata, it is straightforward to argue that organising and searching files become much simpler tasks. Unfortunately, the assumption is problematic. Indeed, it has been claimed (Soules & Ganger 2003) that users are unlikely to supply metadata and that automatic collection of metadata values is a better alternative. While admitting that file location systems based on automatically collected metadata (Freeman & Gelernter 1996, Halpem et al. 2011, Soules & Ganger 2005) are indeed valuable, we hold that working with user-centric metadata is still important and in many cases indispensable. We offer four arguments to support our case:

1. Some files simply do not contain the objective data that is necessary for them to be included in some collection deemed meaningful by the user, and hence an automatic process cannot hope to extract it.

An example is of a scanned image of a building construction plan that is part of a legal case. The particulars of the legal case are not present in any part of the file, its bitmap content, or the file system; it must be provided by a person tasked with documenting the case.

This argument is especially valid in the context of organisations that archive information for public retrieval; much of the required metadata will have to be manually collected at some point.
2. Some kinds of metadata are inherently subjective rather than objective; since the values for such attributes depend purely on the user, there is no software process that can obtain them. An obvious example is the ‘rating’ tag that is associated to music or image files. More generally (Sease & McDonald 2011), individual users catalogue (i.e. attach metadata to) files in idiosyncratic ways that suit their own organisational and retrieval strategies.

3. Searches based on automatically extracted metadata (such as document keywords or a user’s contextual access history) may well locate a single file or range of similar files, but only a well-organised set of manually assigned metadata is likely to return logically-related collections of files. The argument here is that an automatic system would attempt to cluster files according to extracted metadata; however, the number of metadata attributes is relatively large, and values for many would be missing for various files. Clustering is ineffective when the multidimensional space is sparsely populated, so this approach is unlikely to be able to retrieve collections without user input.

Consider as an example a freelance software engineer who works on several long-running projects concurrently. A time-based search in a ‘flat’ document store is likely to return files that belong to more than one project. If the freelancer only works from home, other searches based on automatically extracted contextual metadata (e.g. location, or audio being played while working (Hailpern et al. 2011)) are unlikely to be able to cluster files exactly around projects. Again, the user will need to supply the project details—not as directory names, but as metadata attribute values.

4. Finally, the simple fact that a large number of applications exist that allow users to modify metadata (together, perhaps, with the perceived popularity of those applications that manage to do it well) is proof of a need for such systems.

Given our assertion of the importance of user-centric metadata, and recognising that users may be reluctant to commit effort to metadata creation, we arrive at the central issue addressed in this paper: how to increase the likelihood that users will supply metadata? Our thesis is that (1) there is a clear need to develop powerful and intuitive interfaces for actively empowering users to capture metadata, and (2) very few such interfaces currently exist.

**Organisation** In this paper we will first propose a framework (Section 2) including definitions and then in Section 3 proceed to assess a set of software titles (representing the state-of-the-art in the area of metadata manipulation interface design) with respect to the framework. Having identified a hiatus in the capabilities of assessed software, we then add to the state-of-the-art by introducing a tightly-focused prototype in Section 4. Both the software assessment and the prototype then lead to a number of guides or principles in Section 5 for the design of user interfaces for updating metadata.

**Scope** This paper deals with interface design issues for systems that allow users to create and modify metadata. The related issue of query interfaces will not be considered in detail. Most of the systems examined are desktop applications that manage files on a local file system. We also consider different target objects: email and cloud-based file systems. Although web-based systems are examined briefly we note that, even with the advent of AJAX-ified interfaces, these are frequently less rich in terms of interface design. Mobile apps have not been considered, as touch interfaces are arguably not (yet) optimized to manipulate sets of objects, and screen size limitations are a significant limiting factor.

**Contributions** The contributions made through this paper include: (1) a proposed framework for assessing metadata manipulation interfaces; (2) assessment of a number of relevant software titles; (3) the presentation of a partial prototype that addresses identified shortcomings; and (4) the enumeration of a concrete set of guiding principles for UI design in this context.

**2 Framework**

**2.1 Metadata**

What is metadata? Intuitively, metadata is “data about data”. While this description is acceptable in many contexts, this paper is concerned with metadata manipulation, so a more precise definition is necessary. Before offering such a definition, and because the term ‘metadata’ has many interpretations, we briefly explore the kinds of metadata exhibited in current systems so that we can establish a context for the following discussion.

Metadata can be classified on at least three coordinates.

1. Where is it stored? Possible locations are within the object, within the file system, or in some third location such as a database.

2. Who manages it? This could be the user (perhaps a privileged user like an archivist) or the computer system. System created metadata is often read-only (file size), but sometimes user-writable (ID3 image metadata).

3. Descriptive or representational? Most metadata is descriptive, and pertains to a single file: a file size, creation date, file type, etc. Representational metadata (Giaretta 2011) describes the format or structure of a file’s data (e.g. the JPEG image standard). It is “data about the containers of data” (maybe it could be called meta-metadata); many objects share a single piece of representational metadata.

This paper addresses user-centric metadata manipulation. Using the classifications above, the metadata manipulated will be user-modifiable, descriptive, and may reside anywhere (file system, with content, or separate file).

We posit the following definition for the kind of metadata used in this paper, and include details about its logical structure.

**Definition:** User-centric metadata is a set of (attribute, value) pairs that is associated with an object. The metadata values are user-modifiable.

We define the type of metadata in Figure 1. The value type V can be a simple (non-structured) type, a collection of values of the simple types (referred to as multi-valued attributes), or a collection of (attribute,
The type \( T \) represents application-defined enumerated types. The inclusion of enumerations gives more control over the values that can be stored. Just as programming languages that require names to be declared before use can largely prevent errors due to typographical mistakes, the use of predefined metadata values rather than unconstrained character strings can mitigate the proliferation of value synonyms and mistyped values. (Values of an enumerated type would typically be used to populate the values of a GUI selection widget.)

Note that this scheme can model tags, value-less attributes commonly used in web and social media applications, by using a multi-valued attribute named (say) \( \text{tag} \).

### Expressive Power
Most of the software that we assess in this paper, and most of the post-hierarchical file systems that have been proposed in the literature, are limited to metadata attributes of simple types. There are, however, a select few that support complex types. In particular WinFS (Rizzo 2004), LiFS Ames et al. (2006), and MDFS (Dekeyser et al. 2008) all support the notion of relationships between (file) objects.

Relationships, as defined in the Entity-Relationship Model, are important to represent associations between objects. The three file systems mentioned above are built around the argument that creating explicit associations between files adds significantly to the usefulness of metadata-based file systems.

One-to-Many relationships can be supported through our definition of metadata, as multi-valued attributes are supported, provided each object has a unique identifier. Relationship attributes can be described by using the recursive \( T \) type given above to construct a record for each link between objects.

Many-to-Many relationships between two sets of objects can then be simulated by implementing the One-to-Many relationship in both directions. A naive implementation of this simulation would be prone to inconsistent and redundant data; however, the definition given above is independent of implementation details.

### Logical data model
To be able to present a language and operations to update metadata for a set of objects, we propose a simple logical model for a metadata store, based on the definition of user-centric metadata given above.

**Definition:** A metadata store is a relation \( R \subseteq V_1 \times \ldots \times V_n \) where \( V_i \) is the set of all possible values for an attribute \( a_i \) of type \( V_i \) as defined in Figure 1.

Given the nature of the type system presented in Section 2.1, it is clear that \( R \) is a nested relation as defined in the Nested Relational Model (Gyssens & van Gucht 1988).

The following statement updates the complex Address attribute:

\[
\text{CHANGE SET Addresses.No:=7 WHERE 'Seattle' IN Addresses.City}
\]

The examples illustrate that the update language must contain the necessary constructs to deal with nested relations and complex types. Compared to the

---

**Figure 1:** Metadata type
fundamental system-level update operation, it also adds the ability to modify multiple attributes for multiple objects in one high-level call. But ultimately statements in this high-end update language can be readily translated into a sequence of the fundamental system-level one-file-one-attribute-replace-value updates.

### 2.4 GUI operations

Given the power of the update language specified above, the challenge is to describe a set of GUI operations that users can perform to carry out a **CHANGE statement**. There are at least the following two requirements for these operations.

- **Efficiency.** For a given **CHANGE** statement the length of the sequence of GUI operations (keystrokes or mouse operations) should be minimized. This is a real challenge, though work on gathering metadata automatically through awareness of the user's context may point a way forward, as is presenting suggestions for values for metadata based on the current instance (the recognition vs. recall principle).

- **Power.** The need for advanced metadata management is predicated upon the growing number of files/objects, which means that any interface must, wherever possible, be able to manipulate metadata for many files and/or attributes with single operation. Traditional GUI systems have often been criticised for their inability to perform the kind of complex or repetitive operations possible with CLI update languages or command scripts (Gentner & Nielsen 1996); we consider it a mandatory requirement that "bulk" operations be supported through a GUI interface.

Powerful, efficient interfaces allow users to define and accomplish a complex task quickly. Speed is a key factor in user acceptance and we argue that users in general will create or maintain metadata only if they can do it rapidly and with little effort.

Given these challenges, what kinds of operations on metadata should these interfaces support? The answer depends on the complexity of the metadata values, as defined by three alternatives for type V in Figure 1, and the quantity of files and attributes addressed by the operation.

Figure 2 depicts the range of operations on metadata that an application may provide. An application would provide operations that correspond to one or more of the cells in the table. Power increases from top left to bottom right. The vertical axis represents the richness of the values that can be stored (each row corresponds to a value type alternative in Figure 1) and the horizontal axis depicts the scale of an operation—how many attributes a single operation can affect.

The utility of applications that maintain metadata can be rated in part by examining the complexity of the operations that they support. We will use the operation grid of Figure 2 as one basis for evaluating software in Section 3.

User interfaces do more than provide a scaled-up version of the API and even of the CLI update language. A particular application may add functionality or constraints to improve usability. For example, if only the logical delete and add attribute value operations were implemented, then the user would be required to first delete the value then retype the entire text (add the value) when a misspelling appeared in a string-valued attribute. Instead a typical interface would present a character-at-a-time editing interface to the underlying API. A typical constraint may be that a list of values represent a set, prohibiting duplicate values. Users may also be constrained in choice of value, picking from a predetermined set of appropriate values.

More examples of user interface functionality will be seen in Section 3 where we examine some example metadata manipulation interfaces.

Note that in database parlance, we have so far only described *instance* data manipulation operations. An advanced metadata management system would also allow *schema* operations, such as defining new attributes, or new enumerated data types. In the introduction we referred to this as user-defined metadata. We believe that these schema operations are necessary to build a truly capable system; such operations are orthogonal to the instance operations that are the focus of this paper.

### 3 Evaluating interfaces

In this section we report the results of a critical evaluation of a number of applications that display and manipulate metadata. Based on this evaluation, we propose guidelines for developers of advanced interfaces.

#### 3.1 Applications

Thirteen desktop and three web applications were selected. Some applications are designed to manage metadata for specific file formats (image, video, audio, bibliography) while others are not format specific. Without claiming to be exhaustive, we have chosen a set of applications that we believe to be among the best representatives of commercial and freeware software across the range of domains.

Table 1 lists the applications selected by application domain and by platform. Some applications were available on both Windows and Linux platforms (Pi-casa, Tables, Clementine); the table shows the version tested. All programs were tested on the authors’ computers except Adobe Bridge–we relied mainly on Adobe instructional material to evaluate this product.

#### 3.2 Support for types

All except the simple tagging systems (Tabbles and the web applications) support single-valued at-
Table 1: Applications

<table>
<thead>
<tr>
<th>Type</th>
<th>Application</th>
<th>Ver</th>
<th>Code</th>
<th>Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image</td>
<td>ACDSee</td>
<td>12</td>
<td>AC</td>
<td>Windows</td>
</tr>
<tr>
<td></td>
<td>Picasa</td>
<td>3.9</td>
<td>Pic</td>
<td>Windows</td>
</tr>
<tr>
<td></td>
<td>Adobe Bridge</td>
<td>Br</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td></td>
<td>iTag</td>
<td>476</td>
<td>Tag</td>
<td>Windows</td>
</tr>
<tr>
<td></td>
<td>Shotwell</td>
<td>0.6.1</td>
<td>Sw</td>
<td>Linux</td>
</tr>
<tr>
<td></td>
<td>flickr.com</td>
<td></td>
<td>flkr</td>
<td>Web</td>
</tr>
<tr>
<td>Video</td>
<td>Personal VideoDB</td>
<td>Vdb</td>
<td>Windows</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tabbles</td>
<td>1.1.4</td>
<td>Us</td>
<td>MacOSX</td>
</tr>
<tr>
<td></td>
<td>Flickr</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Music</td>
<td>iTunes</td>
<td>10</td>
<td>iTu</td>
<td>MacOSX</td>
</tr>
<tr>
<td></td>
<td>MP3tag</td>
<td>2.49</td>
<td>MP3</td>
<td>Windows</td>
</tr>
<tr>
<td></td>
<td>Clementine</td>
<td>1.0</td>
<td>Cl</td>
<td>Linux</td>
</tr>
<tr>
<td></td>
<td>Papers</td>
<td>2.1</td>
<td>Pap</td>
<td>MacOSX</td>
</tr>
<tr>
<td></td>
<td>gmail.com</td>
<td></td>
<td>gml</td>
<td>Web</td>
</tr>
<tr>
<td>Mail</td>
<td>Explorer</td>
<td>7</td>
<td>Exp</td>
<td>Windows</td>
</tr>
<tr>
<td></td>
<td>Tabbles</td>
<td>2.0.6</td>
<td>Tab</td>
<td>Windows</td>
</tr>
<tr>
<td></td>
<td>box.com</td>
<td></td>
<td>box</td>
<td>Web</td>
</tr>
</tbody>
</table>

tributes. These can be categorised in two groups. Some provide only a limited number of attributes (Tag, Sw, Pic, Us, iTu, Cl) while others support an extensive set (AC, Br, Vdb, Mp3, Exp). Typical examples include “rating” (numeric), “length” (time/duration), “title” (string), “track” (numeric), “last played” (date/time), “comments” (text), and “metering mode” (enumeration).

3.3 Range of operations

Figure 2 illustrates the range of operations can extend in three dimensions: how many files, how many attributes, and what type of attribute are involved in an operation. We have used this characterisation to evaluate the applications. This is a high-level view: we are only interested in knowing if an application is capable of displaying or updating metadata in any way at a particular scale.

3.3.1 Selecting files/objects

The applications we reviewed typically list either all files they manage in one grid, or use the file system’s directories to list a tree and then display files belonging to one directory in a grid. Users are then able to visually multi-select files in the grid for collective updating.

A few applications allow automatic selection of files based on the values of their metadata attributes. Media players such as Clementine have a keyword search function that may match the value of any attribute, a practice which trades precision for recall. More advanced is Windows 7 Explorer which allows users to filter files based on values for specific attributes by manipulating the attribute titles in the grid.

Hence, no application supports the full power of the where clause in the update language we presented in Section 2.3. This is unsurprising given the expressive power of the condition expression; however, there is scope for applications to use a Query-By-Example (QBE)-type approach (Zloof 1975) to increase the selection capabilities for users. We will return to this issue in Section 4.

3.3.2 Assessment

In terms of range, Adobe Bridge is clearly the most capable: it supports operations involving many files and many attributes on all kinds of data types. Almost half of the remaining 12 applications (Pic, AC, Us, Mp3, Exp) provide operations of all kinds (multiple file and multiple attribute) except on complex data types.

We are being a little loose (and also generous) in describing these as multiple attribute operations. The applications do display a complete set of attributes, but update is on a sequential, one attribute at a time, basis except for ACDSee. Simultaneous update of many attributes is discussed further in Sections 3.6 and 4. iTag, iTunes and Clementine supports single-valued data completely, but supports operations involving only one attribute per operation for multi-valued attribute operations. Conversely, Vdb supports multi-valued attributes fully, but lacks multi-file support for single-valued attributes.

Papers supports single file operations only. Shotwell operates only on a single attribute at a time. The tag-based systems (Tab, box, flkr, gml) support a single multi-valued attribute. Tabbles and gmail support multi-file/object tag operations, while box and flickr perform metadata update on a file-at-a-time basis.

3.4 Display and update semantics

The most useful operations concern collections of files. In the following we will examine the semantics of display and update operations on metadata when multiple files have been selected through the user interface. We consider single and multi-valued attributes separately. All applications except box and flickr support metadata display/update for multiple file selections.

How should an application display a single-valued attribute of a collection of files? A very common approach (Tag, AC, iTu, Cl, Mp3, Exp) is this: if the value is the same for all files then display that value, otherwise indicate in some way (often by stating “multiple values”) that a range of values is present.

Richer and more useful behaviour was observed for some applications for non-text attribute types. For date attributes that differ between files, iTag displays the range of the dates. Windows Explorer treats dates similarly; it sums file size and track length (if audio file) and it averages numerical ratings.

Update behaviour is uniform and unsurprising when a new value is supplied for the attribute it is set for all files selected. There are more design choices when displaying a multi-valued attribute of a collection of files. This is because the attribute values (a set) of two files will typically differ but may contain some common elements. A minimal approach is to use the “multiple-value” technique when attributes differ (Us, Mp3). More useful is to display the intersection (Tag, Exp) or the union (Pic, Pap) of all attribute sets. Intersection and union can both provide useful information; ACDSee gives both views.
3.5 Value management

In some systems (e.g. iTu) the value of an attribute is a simple character string (an editable sequence of characters), while others (e.g. Exp) treat values as atomic elements (enumerated types) represented by a non-editable character string.

The “editable string” approach is versatile and simple to implement but limiting. A major issue is the possible proliferation of values due to typographical errors or because (as is common) the range of existing values is unknown.

The enumerated type scheme requires a more sophisticated implementation but provides a more powerful and usable interface. Operations relating to enumerated values include:

- show all values of an attribute type (Vdb, AC, Tag, Sw, Us, Tab, box, flkr, gml)
- select (using e.g. menu) from list of values (Vdb, AC, Tag, Us, Tab, box, flkr, gml)
- explicit and implicit value creation (Vdb, AC, Tag, Sw, Us, Pap, Tab, box, flkr, gml)
- rename a value (AC, Sw, Tab, flkr, gml)
- delete a value (Pap, Tab, flkr, gml)
- create a new attribute (Us)

3.6 Advanced features

Two applications (Br, AC) support the notion of a ‘template’ that can be defined once and applied multiple times on different sets of files. The idea is to make it easy for users to apply a default set of values whenever they obtain/create a new set of files. It is no coincidence that both applications manipulate image metadata; photographers have a clear need of attaching identical copyright and other data to their photographs. Having to retype this information after each shoot is cumbersome. Of the two implementations, ACDSee is more advanced as it can interpret an expression yielding a new value per image. Importantly, in both cases the creation as well as management of templates involves additional special-purpose interfaces that are not reused from the ‘default’ update mechanism. We will return to this issue in Section 4.

Two applications (Br, Us) allow for the schema of metadata to be updated. Usher permits addition of a multi-valued attribute via the user interface while Adobe Bridge supports creation of complex structured attributes. The process, however, by which an end-user can create new attributes in Bridge is prohibitively complex; in essence an intricate XML document valid over a highly complicated XSD schema needs to be placed in a settings directory prior to program start-up. This mechanism in effect limits the functionality to professional programmers.

3.7 Discussion

While notions of “maturity” or “cleanliness” are less objective than the expressive power discussed in the previous sections, it should be noted that very few of the applications we tested had a fully professional feel to their interfaces. Perhaps unsurprisingly, the more mature solutions were typically created by the large software companies; however, this does not mean that they were most expressive. Almost to the contrary; hobbyist implementations (such as iTag) often surprised us in providing significant power in one or two of the dimensions tested. Unfortunately they also tended to be rather unwieldy through a large number of windows each seemingly able to do only one task (Clementine was a notable culprit in this aspect).

Disappointingly, some major commercial software, while quite powerful in many ways, also felt surprisingly clunky. ACDSee and Adobe Bridge were both assessed positively in terms of power (see above), but their tendency to split functionality over a large number of windows as well as confusing and at times overwhelming menu options were problematic.

The (single attribute) tag-based systems (Tabbles and the three web applications) all handled attribute value management better than the systems that supported multiple attributes. While a little surprising, it perhaps reflects the smaller design space of these systems.

Of all the software reviewed, Windows 7 Explorer left the best impression both in power and in maturity. The interface is appropriately simple (all operations happen in a single window) yet allows for updating several attributes (including multi-valued types) for a group of files of different types. Even so, in terms of interface design we list multiple items for improvement in Section 5. Finally, with respect to power, Explorer could be extended by (a) allowing use of templates (see Section 4), (b) allowing creation of attributes, (c) supporting complex types, and (d) providing an undo mechanism for metadata updates.

4 Updatable views

In Sections 3.3.1 and 3.6 we indicated (1) a lack of powerful file selection mechanisms in almost all applications, and (2) a problem with the non-generic implementation of the template notion as featured in two programs (Br, AC).

Addressing the latter first, we note that Adobe Bridge and ACDSee offer two significantly different methods for updating metadata. They share the first method with all other applications: modify attribute options in a special-purpose interface yet allow for updating several attributes (including multi-valued types) for a group of files of different types. Even so, in terms of interface design we list multiple items for improvement in Section 5. Finally, with respect to power, Explorer could be extended by (a) allowing use of templates (see Section 4), (b) allowing creation of attributes, (c) supporting complex types, and (d) providing an undo mechanism for metadata updates.

While this is a powerful and useful feature, it suffers from interface duplication and increased complexity. These are potential inhibitors for user uptake.

An important contribution that we make to the state-of-the-art as assessed in Section 3, is to recognise that the template idea can be merged with a more expressive search/filter interface and reuse existing file-browser interactions to support single operation updates of many attributes over many files.

Our proposal is best described as an extension of Windows 7 Explorer: once a user has applied a filter...
to a set of files (e.g. by indicating that the value of the 'author' attribute should be 'John'), she can drag other files from another explorer instance into the filtered list, causing the new files to acquire the 'John' value for the 'author' attribute. It is no coincidence that this is akin to a current copy-action in Explorer: in a flat file store, there are no directories to copy from and to; instead, attribute values determine the logical organisation of the file store. Hence the GUI operation is reused soundly.

When a provision is added to ‘save’ the filter action (essentially a query), we arrive at a clean alternative for templates. Saved queries become views that users can interpret as folders. This corresponds to the virtual directory (or folder) concept of semantic file systems (Gifford et al. 1991) and also the collections within the Presto system (Dourish et al. 1999). Views give users not only a powerful mechanism to look for files, but also a second, repeatable means for updating metadata.

Note that not all views would be updatable: this is tightly related with relational view updatability. In those cases, when a user attempts to drag in files, an appropriate feedback mechanism should alert the user that this action is not permitted. That is again consistent with current practice in file browsers.

4.1 Prototype

To illustrate the proposal we have made in this section, we briefly present a prototype interface that we developed in the context of metadata-based file system (Dekeyser et al. 2008). Note that the implementation did not focus on the other issues identified in Section 3; it is purely meant to demonstrate the notion of saveable updatable views as a clean alternative to templates.

The prototype was developed on top of a technology preview of Microsoft’s WinFS. The main feature is a file browser application which allows (1) the listing of objects in the file store, (2) a simplified mechanism to capture rich metadata, and (3) the creation of virtual folders (view definitions).

Figure 3 illustrates the use of virtual folders as a means to capture metadata through a drag and drop operation. The screenshots of the prototype show that initially four Photo objects were selected from the “Photos” folder and subsequently dragged into the virtual folder “Photos with Comments ‘Family Holiday’”. The second screen then depicts the content of the latter, and shows that the four objects have obtained the necessary metadata to belong in the virtual folder.

Dekeyser (2005) first proposed this novel drag and drop approach to metadata manipulation and the technique has been independently implemented (Kandel et al. 2008) in a system that allows field biologists to annotate large collections of photographs. While targeted at a particular problem rather than a generic file system, their system established through extensive user experience the viability of the concept.

The Query-by-Example interface is illustrated in Figure 4. It is possible to create a propositional calculus style query that is a set of relational expressions between attributes and values that are joined by conjunctive or disjunctive logical operators. A new query (view) is initially anonymous (“Untitled”) but can be assigned a meaningful name.

5 Design principles

In the following sections we propose a set of design principles for file metadata manipulation systems. These have been distilled from the better features, as well as being informed by the poorer features, observed in the candidate software. We have also sought to extend and unify some of the interface techniques.

These principles augment or specialise, but do not replace, existing widely recognised generic interface guidelines (e.g. Schneiderman & Plaisant 2004). The following sections enumerate the general principles. We describe how these principles can be applied to the metadata manipulation domain by formulating specific design recommendations.

We assume that a key function of the interface is to manipulate metadata for a collection of files.

5.1 Minimise work

A metadata operation should require as few interface steps as possible. This is a generic goal motivated by an understanding that users are reluctant to invest significant time in metadata maintenance. The principles in the following support this goal, as does this specific design feature.

APPLICATION: Use a single non-modal interface.

Providing complex modal interfaces to do some of the tasks described below, such as value creation or renaming, would result in a decrease in usability and reduced use of key features.

5.2 Facilitate metadata visualisation

Consider some identified collection of files. There may be many attributes present but any file may only have a subset of these attributes. Scientific metadata in particular is characterised by being high dimensional and sparse. Interfaces must display metadata in a compact but useful way to allow users to easily perceive and to subsequently manipulate it.

APPLICATION: Show the names of each file’s attributes, but identify specifically those attributes that are the common to all selected files.

We should not provide update capability for attributes that are not common to all files as this would be ambiguous–users would be unsure of the outcome of their actions. However, users may reasonably need to know the names of other non-common attributes, so that they can be accessed via a further file selection operation.

APPLICATION: Display both the intersection and union of each file’s attribute values.

This applies to both single value and multi-value attributes if single value attributes are considered to be singleton sets. For any attribute shared by a collection of files, a user may wish to know (1) what values are associated with all files (intersection), (2) if all the attribute values are the same (intersection = union), and (3) the range of values present (union). This supports users to make decisions when updating an attribute; providing maximal information reduces the possibility of further keystrokes being needed to seek information.
5.3 Provide systematic support for the manipulation of attribute values.

**APPLICATION:** Support typed attributes, and particularly user enumerations rather than a string type.

Adopting a typed metadata system, similar to the definition in Section 2.1, offers significant advantages. Typing of attributes assists in display and interpretation (e.g. sort order, non-textual displays) of values, and enables provision of appropriate aggregation functions for each type. It also facilitates input validation, and other non-UI features such as specialised storage index construction. Typing does not necessarily require a cumbersome “declare an attribute” modal window as types can be inferred from user actions and a sophisticated interface could provide hints about expected types.

**APPLICATION:** Provide an operation to change the representation of a value.

Values may need to be renamed to better reflect meaning. Value renaming is a global operation that can affect attribute values of many files. Normally renaming to an existing name would cause an error, but it is useful to identify value **merge** as a special case of rename. This is a shorthand for “set attribute value to **new** for all files with attribute value **old**” followed by deletion of the “old” value.

**APPLICATION:** Provide an operation to delete a value from an enumerated attribute type.

If the value is currently associated with a file attribute then confirmation should be sought before proceeding.

5.4 Provide powerful update mechanisms

Here are two proposals for metadata update interfaces for collections of files. The first scheme updates a single attribute, and the second applies updates to many attributes in a single operation.

**APPLICATION:** Update an attribute based on value selection.

We propose the following unifying approach to updating attributes. This is described in terms of attribute sets but, as already noted, if single valued attributes are modelled by singleton sets, the operations below are similarly applicable.

- Select if possible from existing values; if necessary create a new value before selection.
- Update operations assume the existence of three lists of attribute values for a given attribute
  1. The intersection of values for selected files
  2. The union of values for all files
3. The union of values for selected files (this could be displayed as an annotated subset of the “all files” union)
   - Removal of one or more items from list 1 (intersection) results in _deletion_ of those values from the attribute of all selected files
   - Selecting one or more items from list 2 (universal union) results in _addition_ of those values to the attribute of all selected files.
   - A shortcut could be provided for a special case of the addition operation where the values in the selected file union (list 3) are added to the attribute. This operation is can be informally described as “share all values of an attribute among the selected files”.

**APPLICATION:** Reuse the file search interface for views-as-templates.

As described in Section 4 we propose that applications include a QBE-like search/filter and allow resulting views to be saved. In addition, if the view is updatable, it should be possible for it to be used as a template: by dragging files into the view, they acquire the metadata that is needed for them to be members of the view. This principle has the advantage of overloading the traditional file-browser drag-to-copy action with an intuitive update operation.

6 Conclusions

We restate our claim that effective management of user-centric metadata through appropriate and powerful interfaces is vital to the maintenance and everyday use of burgeoning file systems and other electronic repositories.

We have observed and assessed a variety of approaches exhibited by various software in a range of application domains.

All short of implementing uniform generic and powerful metadata operations though some provide pointers for a way forward.

There is a paucity of exemplars of higher-level metadata manipulations, those that can operate on many attributes of many files in a single operation, and their interfaces are byzantine. We describe the prototype of an elegant and novel direct manipulation interface to achieve such higher-level operations.

Our proposed principles and associated application guidelines generalise and extend current best practice and so can be used to guide the creation of the next generation of metadata interface systems.

Metadata based storage systems are not a new idea. But thus far no major advances in interface design have emerged and become widely adopted. Why is this? Why is this problem so hard? Here are a few observations that attempt to answer these questions. Firstly, this is a difficult problem that likely needs an innovative solution rather than simple application of existing techniques. Further, any new approach(s) would require extensive user testing (formal or informal) in order to refine the solution. This is a significant issue: independent developers and researchers typically do not have sufficient resources to carry out such evaluation. On the other hand, commercial vendors may have the resources but are also justifiably wary of foisting new systems, however well tested, onto their customers.

Another issue is the scale of the problem. Systems such as Haystack (Karger & Quan 2004) and the shelved WinFS attempt to extend storage management well beyond file storage and email into generic object management. The dimensions of the design space thus grow very rapidly which further complicates interface design.

The motivation to develop metadata based systems will continue to strengthen. We believe techniques such as the prototype drag and drop interface presented here exemplify the kind of alternate approaches that will be required. We encourage researchers to build systems that explore new interaction or manipulation paradigms in order to advance towards a new era of storage management systems.

References


of the 23rd IEEE / 14th NASA Goddard Conference on Mass Storage Systems and Technologies'.


Understanding the Management and Need For Awareness of Temporal Information in Email

Nikash Singh, Martin Tomitsch, Mary Lou Maher
Faculty of Architecture, Design and Planning
The University of Sydney
148 City road, The University of Sydney, NSW 2006
{nikash.singh, martin.tomitsch, mary lou.maher}@sydney.edu.au

Abstract
This paper introduces research into the presence of temporal information in email that relates to time obligations, such as deadlines, events and tasks. A user study was undertaken which involved a survey, observations and interviews to understand current user strategies for temporal information management and awareness generation in email. The study also focused on current difficulties faced in temporal information organisation. The results are divided across trends identified in use of the inbox, calendar, tasks list and projects as well as general temporal information organisation difficulties. Current problematic conventions and opportunities for future integration are discussed and strong support for careful visual representation of temporal information is established.

Keywords: Time, Temporal information, email, User Interface, Information management, Awareness.

1 Introduction
User Interfaces with which users regularly interact provide ideal conditions under which to monitor critical information subject to change. Despite the gradual nature of this change, software alerts and notifications are often abrupt and inconveniently timed. One common type of critical information requiring regular monitoring is time-obligations (such as deadlines, appointments and tasks). A software environment, which presents these opportunities and yet has struggled to evolve to offer more intuitive means of representation and interaction is email. As such, in this paper we investigate prospects of time management and awareness in email.

The success and ubiquity email enjoys as a channel of communication is well recognised: From the 2.8 million emails currently sent every second [21], through to the 3.8 billion accounts anticipated by 2014 [22]. The rapid proliferation of the Internet has seen email emerge as one of the most successful information systems created. Its use, and indeed preference, as a productivity tool has also been well documented [3,4,5,18,24]. This use includes purposes that it has not specifically evolved to meet, such as Task and Time Management. Flags, appointment reminders and even the inbox itself have served as stopgap solutions to this end. Whittaker, Bellotti and Gwizdka [27] described Task Management in email as “THE critical unresolved problem for users”. Ducheneaut and Bellotti [12] described email as being “overloaded, providing inadequate support for tasks it is routinely used to accomplish”. The email interface provides an interesting set of opportunities and challenges for such integration.

Due to the time-related information email is privy to, especially in the workplace, enterprise-focused email clients (such as Microsoft Outlook and Lotus Notes) often append calendar/scheduling and task-list sub-applications to the email environment. A trend which is increasingly reflected in social webmail solutions (such as Hotmail and Gmail). This partnership is intended to allow users to benefit from the incoming information they receive in their inbox that impacts their allocation of, and ability to manage, time.

We describe information relating to time in the context of email as “Temporal Information” (hereafter TI). TI represents a time-related subset of Personal Information Management (PIM) and is closely related to, but not restricted to, Task Management (TM) and Project Management (PM). It refers to instances where explicit dates, times or spans-of-time occur in email communication. In practice, this often translates to the presence of deadlines, task times, project lifespans, availability and meeting details. This important, but easily overlooked, information becomes buried within the bodies of long messages or amidst a high-traffic inbox.

Beyond these explicit instances of TI, there are often more difficult to define implicit associations, such as the useful lifespan of an email, the optimal response time, the need to prioritise tasks by immediacy, the timely delivery of notifications or reminders. Due to the use of email as a productivity tool in enterprise environments it often contains, or implies, such time-sensitive information.

This distinction of explicit and implicit TI differs slightly to existing work [1], in that it categorises any embedded TI as explicit. It also broadens the definition of implicit TI to include connections between projects and contacts that exist only as associations in the user’s mind, and normal conversation that triggers temporal associations. This type of implicit TI includes for example, remembering that a deadline occurs on an unknown day in the second week of October, knowing that a task cannot be assigned to a colleague who has a large workload, or knowing that a meeting cannot occur in July because the project manager will be on away.
Despite not having a specific date and time associated, this knowledge is useful, but difficult to accommodate in current email applications.

In spite of this strong interdependence of communication and time, modern email interfaces still lack any substantial representation of time, other than as text in amongst text messages. Right now the time-of-delivery is the only “active” TI element in inbox interfaces: allowing sorting. Some email environments, such as Gmail, offer date-recognition [13], but use of this feature is contingent on users also maintaining a separate calendar. While generally email clients and calendar applications are separately robust programs, they are typically not well integrated. Despite their complimentary nature, users are forced to use two fairly idiosyncratic interfaces to manage the same type of information, forcing duplication and segregation of information and interrupting the flow of execution. As we elaborate in the next section, prior research has broadly acknowledged these incompatibilities between the inbox and associated sub-applications (calendar, task list, projects) and prompted further investigation into their effects. We therefore conducted a study to investigate how email users currently manage TI, whether the current integration of the inbox and accompanying sub-applications inhibits their ability to effectively manage TI, and to what extent existing email features allow users to remain informed of TI. The goal of developing this understanding is to identify ways to improve the management and awareness of TI in email. We pose that understanding TI, and to what extent existing email features allow users to remain informed of TI, is not a new or emerging problem. Despite not having a specific date and time associated, this knowledge is useful, but difficult to accommodate in current email applications.

2 Related Works

2.1 Understanding Email Use

This research borrows from earlier work in understanding and characterising email users. Whittaker and Sidner [28] made the distinction between Non-filers, Frequent-filers and Spring-cleaners that made it easier to understand user strategies in coping with email overload, equally as applicable to TI as PIM in general. Barreau and Nardi [2] identified three types of information in email: Archived, Ephemerall and Working, to which Gwizdka [14] made the further distinction of Archived information being either Prospective or Retrospective in nature. Prospective information in this context includes email warranting future action, an activity that Gwizdka suggests is inadequately supported [14]. The naming of these information types alone suggests a temporal order or logic to the timeliness or status of information within email. Identifying user characteristics helps identify how their behaviour will map into UI requirements, such as the presence of folders and labels for Frequent-filers. Past research demonstrates both a preference for the use of email over specialised Task Management tools [3,5,24] and success in the prototyping of email-integrated TM features [4,18]. The presence of a visual representation of TI may rend yet another important user-modelling characteristic.

2.2 Email Content-Analysis: Recognising TI in Email

Due to the rich dataset email inboxes create, content analysis is an area of email research that will prove valuable in identifying and prioritising TI relating to tasks, events and project deadlines. In the past, it has been used to consolidate organisational knowledge [11], facilitate life-logging [16], automate filtering of email [9,19,20] and of particular relevance, date-extraction [23]. While the focus of this research is not data-mining or inbox analysis, this research will need to make use of intelligent content analysis techniques to isolate both the explicit TI (existing in message content) and the implicit TI (existing as connections of knowledge in the user’s mind and conversation history). Understanding where content-activation is feasible will inform the interaction possibilities involving TI, underpinning critical UI design decisions.

2.3 Visualisation in a Congested User Interface

A final domain of relevant email research, and one that warrants careful delineation, is email visualisation. Cognitive theory provides support for easing understanding of a complicated conceptual schema (like time) with a more commonly experienced schema (like space) [8]. However, prior research attempts have demonstrated a critical difference between traditional visualisations (which results in a context where the final representation of information is the primary visual element in the solution, often becoming the entire user interface) and supplementary visualisations (which we identify as elements that integrate into, and support, the overall user interface). That is, supplementary visualisations are secondary elements, which complement, rather than commandeer, the interface. Gwizdka [15], Viegas et al. [25] and Yiu et al. [30] were successful in highlighting specific attributes of email in their prototypes using traditional visualisations, but (sometimes intentionally) at the cost of familiarity to the traditional inbox list environment. Conversely, Remail’s “Thread-arcs” [17] represents threaded conversation in a navigable reply-chain diagram alongside the message body. This “supplementary” style solution is the type aspired to in this study, as user’s current familiarity with the inbox-metaphor has proven a powerful and effective association worth retaining as the primary user interface [26]. In 2001, Ducheneaut and Bellotti [12] even posed the concept of email as a “habitat” which referenced worker’s familiarity with, and prolonged exposure to, the tool.

2.4 A Gulf of Communication Between Sub-Applications

The “gulf” we refer to (the persistence of which we confirm in the results) is not a new or emerging problem. It is a staid and widespread phenomenon across many modern email applications, referring to the disconnect between the inbox and calendar and has been briefly identified among other areas for investigation in prior email research. More than ten years ago, Bellotti and...
Smith [6] recognised a “compartmentalising of information ... within non-integrated applications” as an inhibitor to email-calendar PIM. Then again in 2005, Bellotti et al. [5] stated “Current mail tools compartmentalize resources along application lines rather than activity lines, with attachment folders, contacts and calendar features as separate components”.

In 2006 and then 2007, collaborations between Whittaker, Bellotti and Gwizdka [26,27], every one an established email researcher in their own right, distilled their combined experience with this problem in these words;

“We may schedule meetings and appointments using email, but email itself does not provide dedicated support for calendaring functions.” [26]

“These problems are exacerbated by the fact that most email systems have no inbuilt support for PIM besides Folders.” [27]

Despite these broad early observations, the impact of the “gulf” on TI management has not received significant focus in prior user studies. Further to this, we suggest that the isolation of the inbox from all other sub-applications in email (including the task-list, projects and contacts) poses a critical “trickle-down” concern. This is because the inbox serves as the point of arrival for almost all communication in email, thereby determining the extent to which the sub-applications can assist in TI management and awareness altogether. As such, this research places importance on understanding the “flow” of information through the email environment. The detailed investigation presented in this paper contextualises the severity of these problems of disconnection against real user needs and concerns.

3 Methodology

In order to understand the TI needs of email users, a survey, observations and interviews were conducted. During the observations, users were given hypothetical tasks, and the observations and interviews took place as combined sessions. This combination of quantitative and qualitative research methods was employed to obtain balanced results about levels of user knowledge, feature use, shortcomings and ad-hoc solutions.

3.1 Survey

110 anonymous participants took part in an online survey to gauge initial trends about the use and knowledge of PIM features in email and treatment of TI. Participants included students, creative industry professionals and information workers ranging in age from 18 through 52 (mean=27). 18 short questions focused on user habits, both electronically and on paper, in a number of different situations requiring information to be remembered. For example, participants were asked the following questions;

• What type of information is the most crucial for you to be kept aware of (in email)?
• Do you use post it notes (or other handwritten notes) to remember things?
• How often do you create appointments in your email application?
• If you had to find an important date in your inbox quickly, how would you go about finding that?

More general questions about email client choice and life-logging tools (blogs, journals, calendars etc) were used to characterise participant PIM strategies. Three questions were open, though several closed questions asked for elaboration. Links to the online survey were distributed via email lists and IP addresses were recorded to prevent duplicate responses. No remuneration was offered for survey completion.

3.2 Observations

A dozen student and workplace participants (five male, seven female), ranging in age from 21 through 50, took part in approximately half-hour observations conducted using their preferred email client. Locations for the observations varied depending on the types of users (typically in offices), but quiet and private places were selected so participants would feel comfortable answering honestly and to maintain their privacy. When not using their workplace computers, participants were provided a laptop to access their preferred Webmail applications. The first four questions required users to think-aloud as they dealt with mock-emails arriving from friends, managers or colleagues emulating real work and social tasks spanning different lengths and requiring varying degrees of effort to action. This approach was selected because it was not feasible to conduct observations targeting the unpredictable arrival of email containing very specific types of TI without violating participants’ right to privacy and in a fixed window of availability.

The remaining 15 questions were a mix of mock-email situations and hypothetical situations or questions about their current inbox and strategies for remembering messages, during which participants could be probed further about actions they had performed. The situations presented were diverse but commonplace, such as small tasks like reminding oneself to send photos to a friend, through to preparing for the start of a large new project. Situations posed during the observation session included;

• You’ve just received a new email (email arrives), you have to remember to check this email in two days time. How would you do that?
• How would you find out what deliverables you have due this week?
• A long term project has commenced this week which may last several months and involve many people, what steps do you usually take to organise yourself? If any

Emphasis was placed on what actions were taken when the emails were received (such as immediate-reply, leave-in-inbox etc) and also the action taken on the emails themselves. The observations also presented the opportunity to observe how participants structured their workspace (when possible) and the presence of post-it notes, notebooks, schedules and bulletin boards. Data from the observations was collected in the form of notes, automatic logging, and voice recordings.
3.3 Interviews
The same dozen participants who partook in the observations also answered the 19 open-ended questions from the structured interviews. The interviews were conducted immediately after the observations, in the same location but away from the computers. While the observations focused on demonstration of feature knowledge and strategies, the interviews focused more qualitatively on how participants related to email as a tool. The interviews took approximately 45 minutes each. They asked participants to reflect on their habits in dealing with TI, their impressions of different email applications and features as well as which TI needs could be better supported in email. For example, the following questions were asked of interview participants:

- Do you keep a calendar (paper or digital)? If so, where? If not, why?
- Does your email application have a calendar feature? How often would you say you access/use this feature?
- Have you created appointments before? How would you describe the process of creating appointments in email?

The interviews also prompted for further explanation of self-reminding email strategies. The focus is on how participants related to email and what they struggled with differentiated the scope of the interviews from the observations, which focused on demonstration of knowledge and strategies.

4 Results
The following points were identified from trends in the surveys, which were explored further in the observations and interviews. Results are presented together to provide quantitative and qualitative support for the findings.

The results in this section cover a broad range of TI issues, consideration of which provide some indication as to the way different facets of email will be affected by TI integration attempts. To demonstrate how these recommendations would impact modern email applications, the findings are divided into the four TI-focused sub-applications: inbox, calendar, tasks and projects. The contacts/address book is omitted as, despite having TI relevance, it did not feature prominently in participant responses. Additional results pertaining more to general TI integration than to any existing sub-applications are also included.

4.1 The Inbox
4.1.1 Style of Inbox: Breadth or Depth of Feature Visibility
The surveys revealed a preference for Gmail for social accounts, with 72% of participants actively using the web application and participants demonstrating excellent knowledge of its features during observation. Although opinion was strongly divided for (P10,P11,P12) and against (P2,P3,P8) its interface during interviews. For work, even amongst Gmail users and advocates, Outlook was most commonly identified by interview participants as the best email application they had used (by seven out of twelve interview participants, with 43% of survey participants using the client). One implication derived from this distinction was in regards to feature visibility. Feature-knowledge participants displayed during observations of Gmail use originated from the fact that Gmail’s smaller, but more focused, feature-set was highly visible (e.g. search, tagging, chat and priority inbox, all accessible from the inbox). By comparison Outlook’s more robust feature set, including sophisticated scheduling, filtering and Task Management had to be found through menus, tabs, dialogue boxes and screens that remove users from the context of the inbox. When asked how they recall project emails during observations, Gmail users were usually one click away from performing a sort or search. Interestingly, one of Gmail’s few discreet features, the date-recognition pane that appears circumstantially in the message view, was not well sighted by observation participants (with three out of eleven Gmail users noticing the feature and only one having used it).

4.1.2 Finding Time in the Inbox
Survey participants were asked how they organise email that contains an important date so that they will remember it later. While responses between flagging, filing, filtering, leave-in-inbox, memorising and other were quite evenly split, the easily distinguished preferred action was search (40% of responses). This is to say that these participants take no preparatory actions in remembering dates, relying instead on opportunistic search [28]. Both the observations and interviews confirmed this preference for search when looking for dates in their email;

“I don’t have a failsafe ... I’m fairly confident I’ll be able to find it (using Gmail search)” -P7.

A more general question later in the survey asked which methods participants employ to retrieve an email message which contains a specific date, again search (with 79%) proved the dominant technique. During the observations, given the same scenario, every one of the twelve participants chose to search.

On the surface, this would seem to suggest that the only course worth exploring is how to enhance search for all information retrieval in email. But we have seen that users who do take preparatory actions (like folder creation) use these prepared means of relocation before opportunistic search [28]. Further to this, as Whittaker et al. point out [26], search does not address the TM aspect of email, or facilitate reminding. Scalability also presents concerns, reliance on search necessitates indexing and trawling over categorical sorting and organization.

It is worth investigating whether easier methods of organising TI would reduce reliance on search or simply prove more effective for TI retrieval and awareness-generation specifically. It then becomes a question of establishing the threshold at which any preparatory action is deemed acceptable, given the potential recollection benefits gained.
4.2 The Task List

4.2.1 Creating a Task: a Task Itself

When asked what type of email information participants forgot, 38% of survey participants made mention of TI issues, most regularly citing events and deadlines as the type of TI they forgot. This was followed by resources (such as links and files) at 30%. When compared to paper post-it-note users, which comprised 68% of all participants, 68% of this subgroup attributed TM information as the content of post-it-notes while only 7% attributed resources. This indicates that while both tasks and resources are frequently forgotten, participants required task information to be made more visible. During interviews, participants explained that post-it-notes were for “jotting something down quickly” (P3) and for “urgent and immediate” things (P12). Previous studies have also identified visibility as a critical factor [6,27]. Given this impression of immediacy, it is common that website URLs and file paths are both difficult and inconvenient to manually transcribe. By comparison, times, dates, names and room numbers are easy and quick. While this clearly has implications for PIM entry techniques, we note that in these specific circumstances, duplication and externalisation occurs despite the information system itself being the source of the TI. That is, users willingly rewrite this information where it will be seen, if it is convenient to do so. This can be seen as a failing of email, as information must be externalised to remain useful and clearly is not accommodated sufficiently within the inbox to facilitate recall. In order for TI implementation in email to succeed then, the interaction must be perceived to be quicker and easier than post-it-note use to provide real incentive to switch. Bellotti and Smith drew similar conclusions on paper note association, overabundance and being non-differentiable.

“I flag everything red, but then you become immune to red” –P1.

“In Outlook you can click to flag, but then (it takes) another 3 to 4 steps before you can set a date” –P3.

4.3 The Calendar

4.3.1 All-or-Nothing Calendar Use

In current email environments appointments are the only link between inbox and calendar. Our survey results hint at a dichotomous state where users either frequently, or never, create appointments. Almost half (46%) of all participants had either never (31%) created an appointment or did so rarely (15%).

Self-reported statistics from the survey (representing the total number of calendar entries for the current month) mirror this “all or nothing” distribution, with a high standard deviation of 26 resulting from responses as low as 0 (the mode), as high as 150, a median of 3 and mean value of 14. Only four participants reported having 100 or more events in their calendar.

4.3.2 Having to Check the Calendar

This divide is further apparent from results in our survey that indicate 30% of participants enter important dates from email into the calendar, while 52% found alternate means of remembering the date (like transferring-to-a-diary, transferring-to-phone, marking-as-unread, flagging, transferring-to-paper-note). The remaining 18% of participants took no action to remember the date. One interview participant commented (on calendar use):

“It makes sense to, but I just don’t. I never got around to using it” –P11.

Modern address books in email remember contacts users have been in correspondence with, without requiring manual data-entry. In this context the address book now learns intelligently from the inbox. Calendars have a similar wealth of, so far untapped, information (Ti) embedded in email messages and interaction history. Participant’s perception of disconnection between the inbox and calendar makes them separate tools. This disconnection was not just a hurdle for current users of calendars, but an inhibitor to calendar use adoption altogether. Visibility plays an important role in influencing this perception:

“If I don’t see the calendar when I sign in, it would help if I saw it on sign in or if it was more visible” –P8.

4.3.3 Calendars: a continued practice of rigidity

During observations, participants who took no action in creating calendar entries were asked to elaborate why. Their responses suggest the “information scraps” dilemma identified by Bernstein et al. [7] plays a role. Some pieces of information are perceived to be too small to warrant significant action (such as the creation of a calendar event).

“(I do) nothing... or create a calendar entry if it is important. Deadlines warrant a calendar entry.” –P4.

Alternative actions varied greatly from paper-notes, mobile-phone reminders, relying on memory, asking friends for reminders or doing it immediately. By comparison, small tasks involving resources provided more consistent responses (such as the use of bookmarking to remember websites). Browsers go to lengths to minimise data-entry effort by using information that is already available (page title and URL) while calendars generally make little attempt to leverage the existing context.

Information that cannot be accommodated by the system sometimes becomes a burden for human memory instead. During the surveys participants were asked what proportion of information (which needs to be recalled) they commit to memory, rather than record. Almost half (46%) memorised “few” things, while more than a third (36%) claimed to memorise “half” or the “majority” of items they had to remember.

Calendars, in email, have also traditionally been a destination for appointments and events. Despite being suitable for the aggregation of other types of TI, allocations of time that are both smaller (tasks, tracking) and larger (projects) are not typically present. Because of this, Calendars are not an accurate reflection of workload, only availability.
The observations made it apparent that in spite of features like flags, tasks and events, participants did not have a place in email where an aggregated view of their obligations could be made clear. Participants were asked where they went to get an impression of how busy they were that week. Five participants went to the calendar appended to their email client, but found only meetings there. Participants had to check a number of places to get a feel for where their deliverables were recorded;

“I go to the calendar to see how busy I’m going to be for meetings and the paper to-do list. Also, I keep an excel spreadsheet where I check off tasks. The Excel is online. It’s not really helping” – P1.

Flags (despite being in common use) once created, could only be recalled as a separate list of “red” emails, of which there were usually many. Again, they provided little indication of immediacy, priority or project associations. Without this context, the list of flagged emails proved quite overwhelming. Participants were usually quick to leave this view rather than interact with it further.

4.3.4 The Gulf Between Inbox and Calendar

Discussion with the workplace interview participants revealed that calendars were used because they provided the only way to organise meetings. Their use was expected in workplaces. Some professional participants had not even considered a calendar for personal use outside of work.

Most notably, projects, which stand to benefit from placement within a calendar environment due to their long lifespan, critical deadlines and need for visualisation were absent. Also omitted were tasks and other TI uses (like completion rates, deliverables, delays and postponements), despite their presence in the adjoining inbox.

The current need to laboriously transfer TI from email messages to the calendar is an overhead that has users questioning whether each instance of TI is worth the effort involved in transfer;

“Make it easier. Dates, people. It doesn’t feel streamlined” – P12.

Once transferred, the separation of the inbox and calendar creates a further concern that the TI becomes invisible. Awareness is only maintained by switching between these programs, posing a task-switching concern. For a user who has few instances of TI and no professional obligation to refer to the calendar, this constant “switching” is a burden of little benefit.

“I opened Google calendar and was impressed with how good it was, but I just keep forgetting to check it” – P11.

“Things (calendar events) go past without me noticing” – P8.

As outlined in the related works section, the “compartmentalising” [5,6] of the inbox and calendar was identified more than a decade ago and the same disconnect was reconfirmed five years ago [26]. Our current findings suggest the gulf remains problematic even today, in modern email applications for even experienced email users (for whom email access has been commonplace throughout their entire schooling or work history).

One interesting way participants coped with the “gulf” was to copy all email message content into a calendar event description field. This duplication was necessary because, in time, the connection between the event and the email that initiated it would be lost. This is further demonstration of the isolated way these complimentary features work;

“There are events with emails associated, but no way to link them. Maybe a keyword search” – P6.

“There is no relation to the original email. Linking would be handy” – P4.

4.4 Projects

4.4.1 Big Association For a Little-known Feature

It is interesting to note, from the survey results, the poor performance of the “projects” feature (3%) of Entourage in retrieving temporal information. The projects feature is a substantial effort to co-locate information and resources in email that can be categorised into projects. Yet it is seldom used or known despite a prominent location alongside the calendar and contacts shortcuts, ever present at the top-left of the Entourage interface. A similar feature called “Journals” is available in Outlook, but is much less visible. Only one of the twelve interview participants had heard of either feature. Observation participants who were asked to locate all communications and deadlines relevant to a specific current project again confirmed the division between preparatory and opportunistic retrieval strategies. Filers chose to view their project folders while non-filers chose to search. Interview participants were surprised to learn of the projects feature altogether;

“(I knew about) the task manager, yes, but not the project manager. Didn’t know it existed” – P3.

When explained what the projects feature allows, during interviews, participants expressed interest in investigating the feature. Not surprising given the strong role projects play in categorising and retrieving emails. In our survey, participants were asked what they associate emails with. “People” proved the primary association (74% of participants) with “Projects” also proving a strong association (for 50% of participants). Projects usually have critical TI associations, usually in the form of deadlines or milestones that are not yet well accommodated in email.

During observations, when participants were asked what actions they take upon the start of a new project, creation of a folder was the only suitable action within email. The calendar played no role. A later question asked participants how they could find all the important dates pertaining to one of their current projects. For participants who did not have a “roadmap” document or timeline available, the email-related solutions were grim;

“email the project manager” – P1.

“I could do a series of searches based on assumed dates” – P2.

“Manual process of looking through the emails” – P3.

“Look through the folder, pretty time consuming” – P12.
Given the strong connection between projects and TI, projects will be an important aspect of TI awareness-generation and management. Designers will need to be mindful to not re-create the “disconnection” that calendars currently face.

4.5 Temporal Information in Email

4.5.1 Satisficing in the Absence of Temporally Displaced Actions

During the observations, participants were asked to demonstrate how they handle a number of tasks that were temporally displaced, biased towards future action. In the first instance, participants were given a resource (link) via email that they needed to remember to come back to in 2 days time. Flagging and Leave-in-inbox were the most common solutions (9 participants), but checking it immediately was also a popular choice (7 participants). Participants counted on remembering to check it again at the right time, often coupled with the above techniques.

The second instance provided a similar scenario, where participants were given an email address for a contact who could not be contacted for another three days. Seven out of twelve participants chose to email immediately anyway, only one participant chose to use the “draft” feature to write an email in preparation of the correct day.

The third instance asked how participants organise to send an important email on a day they knew they would not be near a computer. Five participants opted to use a smart-phone on the correct day. Two participants organised for colleagues to send on their behalf. Other responses included flagging, sending immediately, using public computers and transferring to a paper calendar.

The final instance provided the participants with an email that would no longer be useful in two days time. The most common response (eight participants) was to leave-in-inbox.

In the above scenarios the participants were given enough information to know what the useful lifespan of the email was. Many of the decisions above suggest that despite knowing what the optimal response time/method was, participants made sub-optimal compromises due to the lack of features that could facilitate the correct course of action.

4.5.2 Reluctance of Change to a Sensitive Environment

An important insight was gained from the interviews which was not apparent in the survey or observations: while participants could identify shortcomings between their TI needs and actual feature availability and scope, they expressed reluctance in changes to the email interface. This is despite ten out of twelve participants agreeing that the integration of TI features would benefit their productivity:

“It would be nice to have a quicker way to link dates with emails” –P3.

“It definitely would be an improvement, any ability to turn an email into a task would be very good” –P4.

Further discussion led some participants to confide that they doubted integration could be done without serious changes to the current User Interface. Several participants mentioned that they had just discovered a comfortable balance with email. In recognition of this we supplement Ducheneaut and Bellotti’s “habitat” analogy [12], based on observed user hesitation, by suggesting that email is a “sensitive habitat”, almost an ecology. Suggesting that proposed changes to the interface need be mindful of the balance users have created for themselves in which they are comfortable, in an environment that they frequent for much of the working day.

So when moving forward, this “balance” is of critical importance. It determines what will be tolerated in this sensitive combination of interface elements. Solutions that present drastic change fail to capitalise on users’ current familiarity with the effective list-view inbox. While very subtle improvements, like Gmail’s date-detection feature and Entourage projects risk going unnoticed altogether.

4.5.3 But Careful Integration is Necessary

Meanwhile, participants of our interviews call attention to the missed opportunities of TI integration. When asked, “how good is your email application at letting you know of upcoming dates and times” responses included;

“It doesn’t do that, really” –P11.

“Gmail doesn’t let me know. Outlook doesn’t unless you set a reminder. I have to do it myself” –P12.

But when asked whether participants would welcome a “more integrated” calendar in the inbox, ten out of twelve participants said yes. Ten participants also agreed that having the calendar separate and out-of-view resulted in events and deadlines creeping up unexpectedly. Visibility is a critical factor in TI awareness generation.

“I think visually if it was all there, it would be easier to see what’s coming up. You have everything in front of you” –P5.

While having everything in front of you is a recipe for clutter and confusion, having the right things in front you are grounds for making more informed decisions and remembering critical information. Given the responses we have attained and prior research prioritising email content [10], we have confidence that (due to the criticality of deadlines, meetings and tasks) TI is the “right” information. The “right” representation of this information though, remains a challenge for researchers and email User Interface designers.

5 Discussion

Some Personal Information Management habits examined here have endured more than a decade. The inbox is still the to-do list for many email users, social or professional. The calendar has been available during that time yet has not emerged as the TI management tool of choice, only of obligation. Marrying this new (and evolving) communication technology with these traditional (and static) time management tools is proving inadequate for modern needs.
Despite working well separately, they are limited in their ability to work together in their existing forms. Nor should they need to, their existing forms could not have anticipated modern functionality. This conservative logic needs to give way to new thinking that recognises email as the high-traffic productive tool it is and re-imagines the accompanying calendar and other sub-applications in a way that compliments email’s strengths, but acknowledges its capacity to overwhelm. The paper metaphors that sustained these sub-applications as individual tools (a grid calendar with boxes that can be written in, a lined task list with checkboxes that can be ticked, project “folders” which accumulate resources) are proving restrictive in a digital context. They do not reflect the highly interactive nature of email or the rich interconnections of email messages and the deliverables they engender. These tools need to be revisited in order to better support, rather than hinder, TI management and awareness generation. Rather than forcing users to regularly visit four separate applications to manage their time obligations in isolation, which becomes a time obligation itself, users request solutions that minimise organisation effort and streamline real-world processes.

The integration of TI features needs to consider both temporal information and interactions. TI needs to be made available and visible at the right time, and can be complimented by temporal actions which allow users to act on emails at a time of their choosing. Being able to delay or advance the sending, deleting, sorting and sharing of email can assist in information management, while appropriate and contextual representation can assist recall. For example, if a message will no longer be required in the inbox beyond a specific date, users should not have to wait until that date to remember to delete that message. Here, preemptive action has the potential to save time and effort in inbox maintenance later (when it is less imperative). A simple option to mark-for-deletion on that date would suffice. If that date is present within the message body, it should be identifiable. Any solution though, will need to be carefully considered and have minimal impact on the familiarity and even affection users display for their preferred email application. A critical balance between useful addition and intolerable distraction awaits designers who would attempt to change this complicated ecosystem.

6 Conclusion

Our fieldwork, coupled with the findings from research in this field, leads us to conclude that there are opportunities to improve handling of TI in email which may better facilitate the flow of TI through the inbox and into the other sub-applications, and importantly, back again. Current interactions possibilities are not modelled after modern workflows or observed user behaviour, but instead are driven by paper methodologies that pre-date and ignore email use strategies and do not scale well with the sheer volume of incoming information which creates unique challenges for the email environment. In particular, the supporting sub-applications that come bundled with email are resulting in an out-of-sight out-of-mind condition rather than providing truly integrated task, event and project management support. Successful integration is contingent on designing easier information management techniques and facilitating recall by leveraging existing data, presented in context, and in time to be useful.

One critical area for improvement, identified by participants as a problem in the inbox, calendar and task list, is the lack of visibility of TI. Without visibility of obligations, awareness is difficult to generate. A further risk to awareness generation comes in the form of over-reliance on opportunistic search. Without preparatory effort made by the user in information organisation, the system will grow inefficient in identifying connections and making suggestions in anticipation of obligations. The visual representation of these obligations also suffers similarly. The onus shifts back to the user in having to remember the right search queries and doing so in time to meet a deadline. This is not meant as a recommendation to dissuade development of opportunistic retrieval methods, but as encouragement for the design of more persuasive preparatory methods. At the heart of this persuasion lies a difficult integration attempt which must convince users that the email environment is a single cohesive information management tool, the organisational benefits of which outweigh the perceived effort involved.

7 References


12. Ducheneaut N. and Bellotti, V. (2001): E-mail as habitat: an exploration of embedded personal information management. interactions 8, 5, 30-38.
An Online Social-Networking Enabled Telehealth System for Seniors
– A Case Study

Jaspreet Singh Dhillon, Burghard C. Wünsche, Christof Lutteroth
Department of Computer Science
University of Auckland
Private Bag 92019, Auckland, New Zealand
jran055@aucklanduni.ac.nz, {burkhard, lutteroth}@cs.auckland.ac.nz

Abstract
The past decade has seen healthcare costs rising faster than government expenditure in most developed countries. Various telehealth solutions have been proposed to make healthcare services more efficient and cost-effective. However, existing telehealth systems are focused on treating diseases instead of preventing them, suffer from high initial costs, lack extensibility, and do not address the social and psychological needs of patients. To address these shortcomings, we have employed a user-centred approach and leveraged Web 2.0 technologies to develop Healthcare4Life (HC4L), an online telehealth system targeted at seniors. In this paper, we report the results of a 6-week user study involving 43 seniors aged 60 and above. The results indicate that seniors welcome the opportunity of using online tools for managing their health, and that they are able to use such tools effectively. Functionalities should be tailored towards individual needs (health conditions). Users have strong opinions about the type of information they would like to submit and share. Social networking functionalities are desired, but should have a clear purpose such as social games or exchanging information, rather than broadcasting emotions and opinions. The study suggests that the system positively changes the attitude of users towards their health management, i.e. users realise that their health is not controlled by health professionals, but that they have the power to positively affect their well-being.

Keywords: Telehealth, senior citizens, perceived ease-of-use, behavioural change, Web 2.0.

1 Introduction
Home telehealth systems enable health professionals to remotely perform clinical, educational or administrative tasks. The arguably most common application is the management of chronic diseases by remote monitoring. This application has been demonstrated to be able to achieve cost savings (Wade et al., 2010), and has been a focus of commercial development. Currently available commercial solutions concentrate on managing diseases rather than preventing them, and are typically standalone systems with limited functionality (Singh et al., 2010).

They suffer from vendor lock-in, do not encourage patients to take preventive actions, and do not take into account patients’ social and psychological needs.

In previous research, we argued that in order to significantly reduce healthcare cost, patient-centric systems are needed that empower patients. Users, especially seniors, should be able to manage their health independently instead of being passive recipients of treatments provided by doctors. Based on this, we presented a novel framework for a telehealth system, which is easily accessible, affordable and extendable by third-party developers (Singh et al., 2010; Dhillon et al., 2011b).

Recent research demonstrates that web-based delivery of healthcare interventions has become feasible (Lai et al., 2009). An Internet demographics trend from the Pew Research Center reports that more than 50% of seniors are online today (Zickuhr and Madden, 2012). Searching for health-related information is the third-most popular online activity for seniors, after email and online search in general (Zickuhr, 2010). In addition, Internet use by seniors helps to reduce the likelihood of depression (Cotton et al., 2012).

Web 2.0 technologies have the potential to develop sophisticated and effective health applications that could improve health outcomes and complement healthcare delivery (Dhillon et al., 2011a). For instance, PatientsLikeMe.com, a popular website with more than 150,000 registered patients and more than 1000 medical conditions, provides access to valuable medical information aggregated from a large number of patients experiencing similar diseases. According to Wicks et al. (2010), there is a range of benefits from sharing health data online including the potential of improving “disease self-management”.

Most patient-focused social health networks offer a basic level of service, emotional support and information sharing, for a variety of medical conditions (Swan, 2009). However, most of these applications are expensive, do not offer a comprehensive suite of functionalities, target mostly younger health consumers, and do not replace traditional telehealth platforms (Dhillon et al., 2011a). A recent review of web-based tools for health management highlights that there is a lack of evidence about the effectiveness, usefulness and sustainability of such tools (Yu et al., 2012).

To address the aforementioned shortcomings, we have developed a novel web-based telehealth system, Healthcare4Life (HC4L), by involving seniors, its target users, from the outset (Dhillon et al., 2011b). Our focus is on seniors in general, which includes both people with...
and without health problems. It is anticipated that the system will be useful to healthy individuals to maintain their health, while patients are assisted with monitoring and controlling their disease and with rehabilitation. A formative evaluation of a functional prototype of HC4L via a multi-method approach confirmed that seniors were satisfied with its usability, but further functionalities promoting exercises and supporting weight management were expected (Dhillon et al., 2012a). Results and feedback received from participants of the study were used to improve the final version of the system.

In this paper, we present a summative evaluation of an improved version of HC4L with a larger number of users. The goals of this study were to test the feasibility and acceptability of a web-based health management system with seniors. The secondary objectives were to assess the user satisfaction, effectiveness of the system, its content and user interface.

The rest of the paper is organised as follows. Section 2 provides a brief overview of HC4L. Section 3 presents the methodology used in the evaluation of the system. Section 4 presents the results which are discussed in Section 5. Finally, we conclude the paper in Section 6.

2 Overview of HC4L (Healthcare4Life)

2.1 Functionalities

HC4L is an extendable ubiquitous patient-centric system that combines the power of social networking with telehealth functionalities to enable patients, especially seniors, to manage their health independently from home (Singh et al., 2010). User requirements for the system were elicited from a group of seniors, details of which are presented in Dhillon et al. (2011b). The system was developed using Google’s OpenSocial technology and the Drupal CMS (Dhillon et al., 2012b).

Similar to Facebook, the system has an open architecture that enables third-party providers to add new content and functionalities. It envisages hosting a variety of health-related applications which will be useful for health monitoring, education, rehabilitation and social support. Developers can design and deploy applications for these categories by using the OpenSocial standard, for example in the form of serious games, interactive web pages and expert systems.

HC4L encourages positive lifestyle changes by letting seniors manage their own healthcare goals. Patients are able to locate other patients suffering from similar diseases – enabling them to share experiences, motivate each other, and engage in health-related activities (e.g. exercises) via the health applications available in the system. The applications can be rated by the users thereby allowing the developers to get feedback. This is a crucial feature which allows users to get an indication of the quality and effectiveness of an application.

An important type of application is visualisations providing feedback and insight into health parameters. A growing body of evidence supports the illness cognition and behaviour processes delineated by the Common-Sense Model of self-regulation (Cameron and Leventhal, 2003; Hagger and Orbell, 2003). Visual representations allow patients to develop a sense of coherence or understanding of one’s condition, and motivating adherence to treatment (Cameron and Chan 2008; Fischer et al. 2011).

Currently, we have developed and hosted several health monitoring applications, including a weight, vital signs and exercise tracker that records the data entered by the patients and gives visual feedback in the form of graphs and bar charts. We have also developed a social memory game that allows users to test their memory by finding matching pairs of cards. For motivation and feedback, all applications contribute to a general weekly score, which is presented to the user as an overall performance percentage.

At this stage, clinicians or healthcare experts are not included in the study. The idea is to empower consumers to manage their own care. However, the users are advised to contact their healthcare providers if unusual patterns in the monitored health indicators are detected.

2.2 User Interface Design

The user interface design process of HC4L contains two parts: design of the container (the system itself) and of the OpenSocial-based applications (health apps). The main design objectives were ease of use (easy to find

Figure 1: Health Apps page in HC4L.
Table 1: Main Functionalities of HC4L

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activities</td>
<td>To share information about one’s activities with the HC4L applications, view and comment on the activities of HC4L friends (allowing users to motivate friends with positive comments).</td>
</tr>
<tr>
<td>Health Apps</td>
<td>To access health applications added by third-party developers. Patients can add applications from the applications directory and remove them from their profile.</td>
</tr>
<tr>
<td>Profile</td>
<td>To enable patients to create an online health profile, which will enable other patients of similar interest or disease to locate them in the system. It also presents a summary of recent health applications used by the user.</td>
</tr>
<tr>
<td>Mail</td>
<td>To send mails to friends and other members in the HC4L network.</td>
</tr>
<tr>
<td>Friends</td>
<td>To access friends’ profile page, find and add new friends, and invite others to join HC4L.</td>
</tr>
<tr>
<td>Settings</td>
<td>To change password and profile privacy settings, and to delete the user account.</td>
</tr>
</tbody>
</table>

domain Healthcare4Life.com. A 6-week live user evaluation of the HC4L system was carried out from June to August 2012.

Participants were recruited by posting advertisements in senior community centres, clubs and retirement homes in New Zealand. Participants were expected to be aged 60 and above. Prior knowledge or experience with computers was not required. We also contacted several senior community centres such as SeniorNet to advertise...
the study to their members. In order to avoid distortion of results due to prior experience (McLellan et al., 2012), participants of the formative evaluation of the system were not involved in the study.

The study began with a one-hour session comprising a system demo and basic explanations of how to use the system, which was offered on several days at the senior community centres. The objective was to provide an overview of HC4L, the user study, and of what was expected from the participants, and to create user accounts to access HC4L. A printed user guide containing step-by-step instructions to use basic features of HC4L was provided. Details of the user study and a softcopy of the user guide were made accessible via the HC4L homepage.

Participants were logged for later analysis. Reminders to use HC4L were provided via email once every week. Details of the user study and a softcopy of the user guide were made accessible via the HC4L homepage.

Survey No. | Assessment Milestone | Content of Questionnaire | Completed (n)
---|---|---|---
1 | Initial Meeting | Demographics, MHLC | 43
2 | End of Week 3 | MHLC, IMI, SUS | 24
3 | End of Week 6 | Additional Likert scale and open-ended items | 21

Table 2: Content of questionnaire

MHLC = Multidimensional Health Locus of Control
IMI = Intrinsic Motivation Inventory
SUS = System Usability Scale

Table 3: Subscales of MHLC and respective items (adapted from Wallston et al. (1978))

Internal
1. If I take care of myself, I can avoid illness.
2. If I take the right actions, I can stay healthy.
3. The main thing which affects my health is what I do myself.

Powerful Others
1. Having regular contact with my doctor is the best way for me to avoid illness.
2. Whenever I don’t feel well, I should consult a medically trained professional.
3. Health professionals control my health.

Chance
1. No matter what I do, if I am going to get sick, I will get sick.
2. My good health is largely a matter of good fortune.
3. If it’s meant to be, I will stay healthy.

Following previous studies (Bennett et al., 1995; Baghaei et al., 2011), a shortened version of the scale was used, where 9 items (3 items for each subscale) were chosen from the original MHLC with 6 response choices, ranging from strongly disagree (1) to strongly agree (6) (see Table 3). The score of each MHLC subscale was calculated by adding the score contributions for each of the 3 items on the subscale. Each subscale is treated as an independent factor - the composite MHLC score provides no meaning. Summed scores for each subscale range from 3 to 18 with higher scores indicating higher agreement that internal factors or external factors (“chance”, “powerful others”) determine health. In order to detect attitudinal changes, participants had to complete the MHLC scale twice: before the evaluation and at the end of the 3rd week of the study. It was anticipated that the short duration of the study would not be sufficient to gauge behavioral change of seniors towards their health management. Therefore, we have examined the results as a signal of possible future behavioral change (Torning and Oinas-Kukkonen, 2009).

The Intrinsic Motivation Inventory (IMI) is a measurement tool developed to determine an individual’s levels of intrinsic motivation for a target activity (Ryan, 1982). The scale was adapted to evaluate participants’ subjective experience in their interaction with HC4L. In particular, the scale was employed to assess interest/enjoyment, perceived competence, effort, value/usefulness, and felt pressure/tension while using the system. Several versions of the scale are available for use. The complete version comprises 7 subscales with 45 items, scored on a Likert-scale from strongly disagree (1) to strongly agree (7). We used a shortened version using 15 items (3 items for each of the 5 pre-selected subscales), which were randomly distributed in the (Wallston et al., 1978). This scale was employed to investigate whether HC4L can positively affect the users’ attitude towards managing their health, i.e. to make them realise that health is not just controlled by external forces. The scale comprises three subscales: “internal”, “powerful others” and “chance” and has 18 items (6 items for each subscale).
questionnaire (see Table 4). Items of the IMI scale as cited by McAuley et al. (1989) can be modified slightly to fit specific activities without affecting its reliability or validity. Therefore, an item such as “I would describe this activity as very interesting” was changed to “I would describe the system as very interesting”. To score IMI, firstly, the contribution score for items ending with an ‘R’ is subtracted from 8, the result is used as the item score. Then, the subscale scores (i.e. the results) are calculated by averaging across the items of the respective subscale.

<table>
<thead>
<tr>
<th>Interest/Enjoyment</th>
<th>1. I enjoyed using the system very much.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2. I thought the system was boring. (R)</td>
</tr>
<tr>
<td></td>
<td>3. I would describe the system as very interesting.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Perceived Competence</th>
<th>1. I think I am pretty good at using the system.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2. After working with the system for a while, I felt pretty competent.</td>
</tr>
<tr>
<td></td>
<td>3. I couldn’t do very well with the system. (R)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Effort/Importance</th>
<th>1. I put a lot of effort into learning how to use the system.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2. It was important to me to learn how to use the system well.</td>
</tr>
<tr>
<td></td>
<td>3. I didn’t put much energy into using the system. (R)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Pressure/Tension</th>
<th>1. I did not feel nervous at all while using the system. (R)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2. I felt very tense while using the system.</td>
</tr>
<tr>
<td></td>
<td>3. I was anxious while interacting with the system.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value/Usefulness</th>
<th>1. I think that the system is useful for managing my health from home.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2. I think it is important to use the system because it can help me to become more involved with my healthcare.</td>
</tr>
<tr>
<td></td>
<td>3. I would be willing to use the system again because it has some value to me.</td>
</tr>
</tbody>
</table>

Table 4: Subscales of IMI and respective items (adapted from IMI (2012))

User satisfaction with the system was measured using the System Usability Scale (SUS). This is a simple scale comprising 10 items rated on a 5-point Likert scale from strongly disagree (1) to strongly agree (5) that provides a global view of usability (Brooke, 1996). Table 5 lists the 10 questions of SUS. Participants’ responses to the statements are calculated as a single score, ranging from 0 to 100, with a higher score indicating a better usability (Bangor et al., 2009).

![Figure 4: Participant retention rate](Image 329x97 to 538x246)

Although SUS was originally designed to provide a general usability score (unidimensional) of the system being studied, recent research by Lewis and Sauro (2009) showed that it can also provide three more specific measures: overall system satisfaction, usability and learnability.

We have included additional Likert-type statements in the final survey, which were analysed quantitatively (see Table 9). These questions were not decided upon before the evaluation, but were formulated during the study based on the feedback we received from the participants. The objectives were to obtain participants’ feedback and confirmation on specific concerns related to their experience and future use of HC4L. Several open-ended questions were also added to allow participant to express their opinions about certain aspects of the system.

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I thought that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

Table 5: The 10 items of SUS (from Brooke (1996))

4 Results

4.1 Socio-demographic Characteristics

The initial sample consisted of 43 seniors aged 60 to 85 (mean age 70, SD = 17.68). Most of the participants were female (62.79%) and European (81.40%). Only 37.21% were living alone, with the rest living with either their spouse/partner or children. The majority of the participants were active computer users (88.37%) using a computer almost every day. Less than half of them (44.19%) used social networking websites such as Facebook. Only 32.56% used self-care tools (e.g. blood pressure cuff, glucometer or health websites). Most of the participants (65.12%) had heard about telehealth.

4.2 System Usage Data

Over the 6 weeks, HC4L was accessed 181 times, by 43 participants. The average number of logins per person was 4.21 with SD 4.96 and median 2. It was a challenge to obtain commitment from seniors to engage in the user study over 6 weeks. Although the study began with a larger sample, the user retention rate dropped over time (see Figure 4). This is in fact a common issue in live user studies (Baghaei et al., 2011). Fifteen participants...
(34.88%) logged in only once. However, a few participants continued to use the system after the 6th week. It is interesting to note that the participant with the highest frequency of usage (25 logins) had very little experience with computers, and was very keen to learn how to use the system well.

Figure 5 depicts the overall usage of the 6 main functionalities provided in the system. The Health Apps feature was most popular (35%) among the participants. The Facebook-like comment page termed Activities was the second-most commonly used feature (22%). This was followed by the Friends page (17%). The Settings page was the least-used functionality (4%). Along with the overall usage of the main functionalities, Figure 5 shows the popularity of specific health applications available in the system. The Vital Tracker was the most frequently used application (29%), followed by the Exercise Tracker (22%), and the Weight Tracker (22%). The Calorie Calculator was least used by the participants (8%).

![Figure 5: Participants’ activities in HC4L](image)

4.3 Change in Attitude

Table 6 reports the mean change scores for those participants who completed both the initial and interim MHLC questionnaires. Change scores for each MHLC subscale were calculated by subtracting baseline scores from follow-up scores.

The findings show that there were some improvements on all the three subscales. Participants responses for “powerful others”, which denotes health is controlled by others such as doctors, reduced significantly by -.29. This suggests that the use of HC4L can reduce participants’ reliance on others, such as health professionals.

<table>
<thead>
<tr>
<th>Subscale</th>
<th>M</th>
<th>SD</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal</td>
<td>.04</td>
<td>1.04</td>
<td>-4 to 2</td>
</tr>
<tr>
<td>Powerful others</td>
<td>-.29</td>
<td>1.27</td>
<td>-10 to 6</td>
</tr>
<tr>
<td>Chance</td>
<td>-.10</td>
<td>1.23</td>
<td>-6 to 5</td>
</tr>
</tbody>
</table>

Table 6: Change in MHLC subscales (n = 23)

4.4 Motivation

Table 7 presents the mean values and standard deviations of the five pre-selected subscales of the IMI (subscale range 1 to 7). It also illustrates the scores of two different age groups of seniors.

Excluding the pressure/tension scale, the results show mid scores in the range 4.11 - 4.40. The results imply that the participants were fairly interested in the system, were adequately competent, made a reasonable effort in using the system, and felt that the system has some value or utility for them. The pressure/tension subscale obtained a low score indicating that the participants did not experience stress while using the system. There are significant differences between age groups for the scores for perceived competence and value/usefulness. Seniors of age range 60-69 consider themselves more competent and find the system more valuable than older seniors.

<table>
<thead>
<tr>
<th>Subscale</th>
<th>All (n = 24)</th>
<th>Age 60-69 (n = 12)</th>
<th>Age 70-85 (n = 12)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interest/Enjoyment</td>
<td>4.40 ± 1.68</td>
<td>4.42 ± 1.73</td>
<td>4.39 ± 1.70</td>
</tr>
<tr>
<td>Perceived Competence</td>
<td>4.39 ± 1.78</td>
<td>4.89 ± 1.52</td>
<td>3.89 ± 1.94</td>
</tr>
<tr>
<td>Effort/Importance</td>
<td>4.11 ± 1.58</td>
<td>4.11 ± 1.57</td>
<td>4.11 ± 1.56</td>
</tr>
<tr>
<td>Pressure/Tension</td>
<td>2.61 ± 1.56</td>
<td>2.67 ± 1.45</td>
<td>2.56 ± 1.69</td>
</tr>
<tr>
<td>Value/Usefulness</td>
<td>4.25 ± 1.81</td>
<td>4.53 ± 1.83</td>
<td>3.97 ± 1.75</td>
</tr>
</tbody>
</table>

Table 7: Subscale findings of the IMI (M ± SD)

4.5 User Satisfaction and Acceptability

Participants rated the usability of the system positively. Twenty-four users completed the SUS scale with scores ranging between 35 and 100, with a median of 65. The average SUS score is 68.33, with only two participants rating it below 50% (not acceptable). The adjective rating of the mean SUS score is ‘OK’, which indicates it is an acceptable system (Bangor et al., 2009).

Participants’ open-ended responses were useful to gain insight into their perception of HC4L. The most frequent positive and negative comments are listed in Table 8.

Table 9 presents the participants' mean responses for additional items included in the final survey of the study, with 6 response choices ranging from strongly disagree (1) to strongly agree (6).

<table>
<thead>
<tr>
<th>Positive Responses</th>
<th>Frequency (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I like the idea of it.</td>
<td>26</td>
</tr>
<tr>
<td>It is easy to use.</td>
<td>23</td>
</tr>
<tr>
<td>The health applications are a great help to keep track of one’s health.</td>
<td>16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Negative Responses</th>
<th>Frequency (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sorting out calories values for foods seems a lot of trouble (Calorie Calculator).</td>
<td>21</td>
</tr>
<tr>
<td>I’m not so keen on the social</td>
<td>18</td>
</tr>
<tr>
<td>Facebook-like aspects of the system.</td>
<td>15</td>
</tr>
<tr>
<td>Limited applications.</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 8: Most common positive and negative comments about HC4L

5 Discussion

The summative evaluation reveals that HC4L is straightforward to use and has potential in empowering seniors to take charge of their health. The system is well accepted by the participants although there were some concerns revolving around the limited content (i.e. health applications) and social features provided in the system.
Results show that participants were keen about the general concept of HC4L that addresses the patients instead of clinicians, and encourages them to play a more active role in their healthcare. To our knowledge, this is the first study that assesses the value of a web-based telehealth system, which does not involve clinicians in the intervention. The majority of the sample (80%) acknowledged that the system allows them to be more aware of their health. One participant commented: “It makes you stop and think about what you are doing and helps to moderate behaviour.”

The participants appreciated the intention of enabling them to access a wide variety of health applications via a single interface. Most of them (72%) agree that such functionality can reduce the need for them to visit different websites for managing their health. One of the participants expressed: “I like the ability to monitor and check your weight, vitals and what exercise you had been doing on a daily basis.” Although the system had only a few health monitoring applications, they were well received by the participants, with the Vital Tracker and Exercise Tracker being the most popular (see Figure 5).

An important lesson learned is that hosted applications must be carefully designed with seniors in mind. For example, the Calorie Calculator, a free iGoogle gadget added from LabPixies.com, was least liked and used by the participants. Issues reported include: “the extreme tediously of the application”, “the foods are mostly American”, and “it is not clear where to enter the data”. This also illustrates that cultural and location-dependent issues can affect acceptance of applications. Other applications, which were specifically developed for HC4L, were regarded as interesting and useful. Most reported shortcomings can be easily corrected. For instance, the Multiplayer Memory Game, shown in figure 6, was found to be more enjoyable than the commonly found single player memory games, but the participants were not able to play it often because no other participant was online at the same time. We also had participants which commented that they prefer to play the game by themselves. One participant expressed: “I would like to be able to do memory games without having to play with someone I don’t know.”

Since HC4L was made accessible online for the study, participants expected it to be a fully functional and complete system, as demonstrated in the comment: “It is a good idea that needs smoothing out, because it has very limited programs at this stage.” The study indicates that there is a need for a wide variety of health applications tailored to the individual needs of the patients. At this stage, only 33% of the initial user group agreed to continue using the system. However, 72% of the participants stated they would be happy to continue using HC4L, if it contained more applications relevant to their needs. This indicates that seniors are ready to manage their own care via a web system provided that there are suitable health-related applications for them to use. The limited content and customisation of the system is also likely to be a reason for the reduced retention rate of the participants (as depicted in Figure 4). Users can become bored and discouraged to look after their health if they are not supported with health applications to address their needs. This highlights the advantage of having a Facebook-like interface allowing submission of third-party content, but also demonstrates the need for a large and active user community supporting the system.

Seniors usually rely on their clinicians to monitor their health (Dhillon et al., 2011a). Therefore, the elevation of selfcare solutions such as HC4L, which do not involve clinicians, might result in adverse effects on a patient’s motivation to use such systems.

<table>
<thead>
<tr>
<th>No.</th>
<th>Statement</th>
<th>n</th>
<th>M</th>
<th>SD</th>
<th>% Agree*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HC4L encourages me to be better aware of my health.</td>
<td>15</td>
<td>4.27</td>
<td>1.44</td>
<td>80</td>
</tr>
<tr>
<td>2</td>
<td>The charts/graphs presented in HC4L helped me to understand my health progress better.</td>
<td>15</td>
<td>3.93</td>
<td>1.28</td>
<td>80</td>
</tr>
<tr>
<td>3</td>
<td>I would use HC4L if there were more applications.</td>
<td>18</td>
<td>4.17</td>
<td>1.47</td>
<td>72</td>
</tr>
<tr>
<td>4</td>
<td>A system like HC4L that provide access to a variety of health applications will reduce the need to use different websites for managing health.</td>
<td>18</td>
<td>3.89</td>
<td>1.78</td>
<td>72</td>
</tr>
<tr>
<td>5</td>
<td>HC4L has the potential to positively impact my life.</td>
<td>17</td>
<td>3.82</td>
<td>1.67</td>
<td>65</td>
</tr>
<tr>
<td>6</td>
<td>HC4L has the potential to help seniors to deal with social isolation.</td>
<td>18</td>
<td>3.94</td>
<td>1.35</td>
<td>61</td>
</tr>
<tr>
<td>7</td>
<td>I would rather manage my health by myself, without anybody’s involvement in HC4L.</td>
<td>18</td>
<td>3.56</td>
<td>1.69</td>
<td>56</td>
</tr>
<tr>
<td>8</td>
<td>HC4L simplifies health monitoring tasks that I found cumbersome to do before.</td>
<td>16</td>
<td>3.06</td>
<td>1.57</td>
<td>56</td>
</tr>
<tr>
<td>9</td>
<td>HC4L allows me to get in touch with other patients with a similar disease or health problem.</td>
<td>15</td>
<td>3.6</td>
<td>1.45</td>
<td>53</td>
</tr>
<tr>
<td>10</td>
<td>The social features of HC4L (e.g. making friends, sharing activity updates with each other, playing social games, etc) motivated me to use the system.</td>
<td>15</td>
<td>2.6</td>
<td>1.45</td>
<td>33</td>
</tr>
<tr>
<td>11</td>
<td>Involvement of friends helped me to better manage my health through HC4L.</td>
<td>13</td>
<td>2.54</td>
<td>1.76</td>
<td>31</td>
</tr>
</tbody>
</table>

*Percent Agree (%) = Strongly Agree, Moderately Agree & Slightly Agree responses combined
Results of the intrinsic motivation scales show that participants rated their subjective experience with HC4L as satisfactory. Younger seniors (age 60 to 69), on the whole, yielded higher scores than the older seniors (age 70 and above), i.e. younger seniors are more motivated to leverage the system for their health. Overall, seniors were moderately motivated to use the system for managing their health despite the absense of clinicians. The SUS score also confirms that HC4L usability is satisfactory. Although a better score, 75, was obtained during the formative evaluation of the system (Dhillon et al., 2012a), there is a vast difference between the sample size and duration of the study. Moreover, the current mean SUS score is above 68, which Sauro (2011) determined as average of 500 evaluation studies.

There was some indication that the attitude of the user matters more in self-care solutions than the features provided in the system. For example, an interesting comment by one participant was: “For elderly people to improve their quality of life as they age, a positive attitude is essential for wellbeing. Interaction with others in similar circumstances goes a long way in achieving this.”

The results of the MHLC scale, especially in the “powerful others” subscale, were encouraging and suggest that HC4L has the potential to positively affect users’ attitude that their health is not controlled by external forces such as health professionals. This is likely to be the effect of engaging the participants to monitor their health progress, e.g. via the Vital Tracker and Exercise Tracker.

Although a few participants reported being unable to track their blood pressure due to the lack of the necessary equipment, the system enabled them to realise that some minor tasks usually done by health professionals, can be performed by the patient. In fact, HC4L allows users to collect more health related data than a doctor would usually do. For instance, patients can track the amount of exercise they perform within a week and make effective use of the visual feedback provided via charts and graphs (see Figure 2) to ensure they have done enough to improve or maintain their health. It was interesting to note that the majority of the participants (80%) endorsed that the charts/graphs presented in HC4L enabled them to understand their health progress better. Overall, systems like HC4L, which are not meant to replace doctors, can allow patients to realise that they have the power to positively affect their well-being. We anticipate that with more useful applications and a larger pool of users, the system would result in an even larger change of patients’ perspective towards managing their health. One participant commented “I hope this programme will become more useful as time goes on and more people use it. I can visualise this in the future.”

In the present study the social aspects of HC4L were not positively endorsed by the participants. The majority of the participants were not keen to use Facebook-like social features. This finding is consistent with the outcome of the formative evaluation of the system (Dhillon et al., 2012a). The Facebook-like comment feature was retained since the formative study, but with a clear purpose - to enable patients to encourage each other in managing their own health. The main objective of the commenting feature was changed from mere sharing of messages to a place where patients could motivate each other for taking charge of their health via the applications provided in the system. Several other features were incorporated, such as the ability to automatically share health-related activity information (e.g. exercise tracking) with all friends in the system. Apart from writing positive comments, a thumb-up button was also provided, which could possibly give a visual encouragement to the patients.

However, user feedback on these features was mixed. Most of the participants (67%) feel that the social features did not motivate them to continue using the system, and 69% of them found the involvement of friends was not beneficial to their health. Four active participants of the study expressed disappointment that their friend requests were not responded to. One of them also shared that she started off with the study enthusiastically, but received only one friend response which caused the motivation to disappear. Most of the participants were not comfortable to accept strangers as “friends” in the system. This could be due to privacy issues as a few participants made similar comments relating to their hesitation to share personal information with others. A typical comment was: “I would not share my medical details with someone I don’t know.” Figure 7 summarises with whom the participants would share their activities/information in the system.

![Figure 6: Multiplayer Memory Game](image-url)
A few participants commented that it is important for them to know someone well enough (e.g. what their goals are) before they could accept them in their friends list. One participant expressed: "I find the use of the word 'friends' for people I don't know and will never meet very inappropriate and off-putting. Also it's really important to learn more about the people in your circle so that you care enough about them and their goals to be able to offer support. Just giving them the thumbs-up because they say they've updated something seemed a bit pointless when you don't have any idea of the significance of the update to them, nor any data to respond to.” While the comment sounds negative, it suggests that the participant wants to find new friends and get to know them more (i.e. to care about them and be cared about). This indicates that the social networking functionalities of HC4L are desired, but not in the form we might know from Facebook and similar sites.

The system could be especially valuable to people who are lonely, as 61% of the participants agreed that the system has the potential to help seniors to deal with social isolation. Nevertheless, it is necessary to revise the social component in a way which fosters building of personal relationships (possibly using a video conferencing facility), and which overcomes concerns of privacy. The interviewed seniors seemed to be very careful in their selection of friends. This observation contrasts with younger users of social media sites, which are more open towards accepting friends and sharing personal information (Gross and Acquisti, 2005). Other ways of providing social support to patients in the system need to be explored. For example, it might be helpful to have subgroups for users with different health conditions, like done in the website PatientsLikeMe.com (Wicks et al., 2010), since this gives users a sense of commonality and belonging.

6 Limitations

We recognize limitations of the study and avenues for future research. Most participants had experience with computers, and results for users unfamiliar with computers may differ. The relatively small size of the sample did not allow us to determine whether the system is more useful for some subgroups than others (e.g. particular health issues, psychological or emotional conditions).

7 Conclusion

A web-based telehealth system targeted at seniors, which is extendable by third-parties and has social aspects, was developed and evaluated. A summative evaluation of the system was conducted with seniors over 6 weeks. Results indicate that the idea of using the web to manage health is well-accepted by seniors, but there should be a range of health applications which are tailored towards individual needs (health conditions). Social networking functionalities are desired, but not in the “open” form we might know from Facebook and similar social media sites. Our results suggest that web-based telehealth systems have the potential to positively change the attitude of users towards their health management, i.e. users realise that their health is not controlled by health professionals, but that they have the power to affect their own well-being positively.

8 Acknowledgements

We would like to thank the participants of this study for their kind support, patience and valuable feedback. We acknowledge WellingtonICT, SeniorNet Eden-Roskill and SeniorNet HBC for advertising the study and for allowing us to use their premises to conduct the introductory sessions. We also thank Nilufar Baghaei for her input in conducting the study.

9 References


Validating Constraint Driven Design Techniques in Spatial Augmented Reality

Andrew Irlitti and Stewart Von Itzstein
School of Computer and Information Science
University of South Australia
Mawson Lakes Boulevard, Mawson Lakes, South Australia, 5095
andrew.irlitti@mymail.unisa.edu.au  stewart.vonitzstein@unisa.edu.au

Abstract
We describe new techniques to allow constraint driven design using spatial augmented reality (SAR), using projectors to animate a physical prop. The goal is to bring the designer into the visual working space, interacting directly with a dynamic design, allowing for intuitive interactions, while gaining access to affordance through the use of physical objects. We address the current industrial design process, expressing our intended area of improvement with the use of SAR. To corroborate our hypothesis, we have created a prototype system, which we have called SARventor. Within this paper, we describe the constraint theory we have applied, the interaction techniques devised to help illustrate our ideas and goals, and finally the combination of all input and output tasks provided by SARventor.

To validate the new techniques, an evaluation of the prototype system was conducted. The results of this evaluation indicated promises for a system allowing a dynamic design solution within SAR. Design experts see potential in leveraging SAR to assist in the collaborative process during industrial design sessions, offering a high fidelity, transparent application, presenting an enhanced insight into critical design decisions to the projects stakeholders. Through the rich availability of affordance in SAR, designers and stakeholders have the opportunity to see-first-hand the effects of the proposed design while considering both the ergonomic and safety requirements.

Keywords: Industrial Design Process, Spatial Augmented Reality, Tangible User Interface.

1 Introduction
The Industrial Design Process, Traditionally, the industrial design process involves six fundamental steps, providing guidance and verification for performing a successful product design (Pugh 1990). These steps guide the design process from the initial user needs stage, assist in the completion of a product design specification (PDS), and onwards to both the conceptual and detail designing of the product. When the product is at an acceptable stage, which meets all the requirements set out within the PDS, the process follows on into manufacturing and finally sales.

This incremental development process encourages total design, a systematic activity taking into account all elements of the design process, giving a step-by-step guide to evaluating and producing an artefact from its initial concept through to its production. Total design involves the people, processes, products and the organisation, keeping everyone accountable and involved during the design process (Hollins & Pugh 1990). Each stage of the process builds upon knowledge gained from previous phases, adhering to a structured development approach, allowing for repetition and improvements through the processes. This methodology of total design shows what work is necessary at a particular point within the stream of development, allowing a better-managed practice to exist. During each stage, stakeholders within the project hold meetings reviewing the goals and progress of the project. These meetings allow a measurement of success by comparing the current work with the initial system’s goals and specifications. These reviews allow for early detection of design flaws and an opportunity for stakeholders to raise concerns based on outside influences.

Throughout the development cycle of the product, the PDS is used as the main control for the project. As the product moves through each phase, the PDS evolves with new information, and changes to the original specifications. This process has been widely employed and a proven effective practice for producing designs, however the opportunity for collaborative, interactive designs are not available until the later stages of the
Augmented Reality, is an extension to our reality, adding supplementary information and functionality, typically through the use of computer graphics (Azuma 1997). Generally AR is confined to projections through either headsets or hand-held screens. Spatial Augmented Reality (SAR) is a specialisation of AR where the visualisations are projected onto neutral coloured objects by computer driven projectors (Raskar, R. et al. 2001). This variant in design compared to the more traditional means for visualisation allows the viewer to become better integrated with the task at hand, and less concerned with the viewing medium. Due to SAR’s rich form of affordance, special consideration is needed to allow for the effective utilisation of interaction techniques within the setting.

Tangible User Interfaces (TUI) are concerned with designing an avenue for interaction with augmented material by providing digital information with a physical form (Billinghurst, Kato & Poupyrev 2008; Ishii & Ullmer 1997). The goal is to provide the users of the system physical input tools, which are both tangible and graspable, to interact effectively and subconsciously with the system as if it were ubiquitous in nature. Previous efforts have been conducted in combining TUI and SAR technologies to benefit the outcome of the research (Bandyopadhyay, Raskar & Fuchs 2001; Marner, Thomas & Sandor 2009; Raskar, Ramesh & Low 2001).

Collaborative Tools, Previous work has been investigated into utilising SAR as an interactive visualisation tool for use during the design phase of the design process (Akaoka, Ginn & Vertegaal 2010; Porter et al. 2010). Currently, this research consists of the arrangement of pre-determined graphics within the SAR environment. SAR itself is not currently involved in the actual design process, only being utilised as a viewing medium, with all prototyping being done at a computer workstation before being transferred into the SAR environment. Interactions with these systems were predefined, only allowing a constant but repeatable interaction to take place.

Current detail design offers either flexibility or affordance, but not both at the same time. Flexibility is offered using a computer aided drawing package, allowing changes to texturing, detailing, and layout through the click of a button. Being computer generated however the model only exists within a two dimensional space on a monitor. Affordance is available through the creation of a physical model, which can be detailed up to suit, however this avenue does not offer for quick modification of textures and layouts. Using SAR as the visualisation tool offers affordance, but thanks to the visual properties being produced by a computer, flexibility is also available to alter and amend the design in real-time.

The means for interaction with the virtual world through physical objects is accomplished through a TUI. Tangible, physical objects (sometimes referred to as phicons), give users the avenue for interaction with augmented elements.

TUI have been used by both Marner et al. (2009) and Bandyopadhyay et al. (2001) to allow for the dynamic drawing of content within a SAR environment. Both examples allowed live content creation through the use of tangible tools. These examples present an opportunity for further investigation into SAR being used as more than solely a visualisation tool.

Contributions
- We provide new techniques for introducing constraint based design primitives within a SAR environment.
- We present a tangible toolkit for allowing the designer to perform both selection and manipulation of projected content, the addition of shape primitives and constraints between primitives within a SAR environment.
- We provide a validation of our work using SAR as an architectural / industrial design tool. This validation has allowed us to highlight key areas where SAR offers valuable areas of utilisation within the current industrial design process.

2 Related Work

Augmented reality has grown to involve many varying areas of study, all actively pursuing advancements within the field. Industrial Augmented Reality (IAR), a particular area concerned with the improvement of industrial processes attempts to provide a more informative, flexible and efficient solution for the user by augmenting information into the process (Fite-Georgel et al. 2011). This inclusion intends to enhance the user’s ability to produce quality and efficient work while minimising the disruption on already implemented procedures. Early Boeing demonstrated this benefit by creating an AR tool to supplement their installation of wire harnesses in aeroplanes (Caudell & Mizell 1992; Mizell 2001).

Prior to the aid of augmented instructions, workers would need to verify the correct wiring layout from schematic diagrams before performing their work. The diagram would only provide the start and end points for the wire, with the worker required to work out their own route across the board. This would then result with inconsistent installations of wiring across varying aircraft, dependent on which engineer performed the wiring. By introducing AR instructions to the workers process, uniform routing would be performed by all engineers, standardising the wiring across all aircraft while also allowing the isolation of the particular wire being routed, removing cases where non-optimal routing were performed. This improved the current installation process while also having an effect on the repairing process, contributing to an easier framework to be followed if any issues arose down the track (Caudell & Mizell 1992; Mizell 2001). Further work has continued this approach into augmenting instructions into the worker’s work process (Feiner, MacIntyre & Seligmann 1993; Henderson & Feiner 2007). The ARVIKA project has
also demonstrated a strong focus towards the development, production and service of complex technical products through its AR driven applications (Friedrich 2002; Wohlgemuth & Triebfurst 2000)

Examples of current research in industrial AR show a general direction towards providing an information rich solution for workers carrying out their respective roles. With any emerging technology from an industrial perspective, the solutions need to be financially beneficial, scalable and reproducible in order to become a part of daily processes (Fite-Georgel 2011).

Current IAR uses either video or optical see-through HMD’s or video screen based displays. Another area of AR which shows promise to industrial application is spatial augmented reality

Raskar et al. (1998) highlighted the opportunity of superimposing projections, with correct depth and geometry, of a distant meeting room onto the surrounding walls, where people within each room would experience the two meeting rooms as one large collaborative space. This idea spawned the current field of SAR; using projectors to augment information onto surrounding material removing the intrusiveness of a hand-held or head-worn display.

Currently, prototyping within SAR consists of the arrangement of pre-determined graphics or free-hand modelling (Bandyopadhyay, Raskar & Fuchs 2001; Marner, Thomas & Sandor 2009; Raskar, R. et al. 2001). Prototyping is generally done on a computer screen and then transferred into the SAR environment as a method of visualising the final design. The actual design phase exists outside of the SAR environment.

The goal of tangible bits (Billinghurst, Kato & Poupyrev 2008; Ishii 2008; Ishii & Ullmer 1997) is to bridge the gap between the digital world and the world that we exist in, allowing the use of everyday objects as a means to interact with the virtual world. Tangible user interfaces attempt to make use of the rich affordance of physical objects, using their graspable form as a means for interaction with digital content. The human interaction with physical forms is a highly intuitive interface, removing the need for displays to see-through, buttons to press and sliders to move (Raskar, Ramesh & Low 2001).

The metadesk allowed the control of virtual content through the use of phicons (physical icons) within the physical space (Ullmer & Ishii 1997). Ishii also developed the Urban Planning Workbench (URP) to further emphasise the concepts behind a TUI. In this scenario, phicons were used to depict buildings, with projections showing the effects of weather, shadowing, wind, and traffic congestion from the phicon positions (Ishii 2008).

Combining the use of TUIs within a SAR environment has given the idea of using the two tools together as a means for aiding the design process, allowing designers to partake within a collaborative prototyping environment (Akaoka, Ginn & Vertegaal 2010; Porter et al. 2010).

Constraint driven modelling is a key area in industrial design, conforming a design to uphold required dimensional and neighbouring constraints (Lin, Gossard & Light 1981). Constraints allow “rules” to be introduced into a design so that the fixed parts of the design are locked. This allows the user to be creative in the areas they can be without violating design requirements. Applying the key ideas of a constraint driven model user interface into a SAR environment would provide a designer with an enhanced experience.

Three dimensional co-ordinate constraints should allow both implicit and explicit constraints to be viewed within the design. Lin, Gossard & Light (1981) devised a matrix solution to uphold these neighbouring constraints within a three dimensional space. By introducing relationships between constraints, a designer would be able to follow a set of ‘rules’ which upheld the design requirements.

The city planning system (Kato et al. 2003) devised an AR table-top design, using head-mounted displays, to view the results of placement of architectural components. The designer could place components across the site, rotating, scaling and moving them about until an acceptable solution was found. The design session however did not cater for any relationships between elements positioned within the system. Planning systems should implement constraints to disallow elements from being constructed in close proximity to certain landmarks or other placed components.

Spatial augmented reality systems have been produced where designs could be projected onto their physical sized prototypes (Akaoka, Ginn & Vertegaal 2010; Porter et al. 2010). These demonstrations showed the power of SAR as a viewing medium. These designs however constrained the user to only pre-determined interactions with the content.

Free-hand modelling has been demonstrated as a means for interaction with SAR material (Bandyopadhyay, Raskar & Fuchs 2001). Physical, tangible tools have given the user the ability to paint onto the representation (Marner, Thomas & Sandor 2009). The use of a stencil to mask parts of the model gives the user a higher accuracy with achieving desired results. The placement of the stencil however is still free-hand, and does not rely on any constraints for use.

Previous research has shown promise in merging SAR with the design process (Porter et al. 2010). By introducing a physical design which incorporates interactive content, quick modifications can be applied to gain further understanding and knowledge to potential design choices. Our investigations extend this research by bringing the design aspects into the SAR environment, bringing with it opportunities for designers to make amendments during the visualisations. SAR is a powerful visualisation tool which allows collaboration with affordance. The goal of this research is to show that SAR can also improve the design process by using an intuitive interaction metaphor to allow SAR to become the design setting itself.

3 Implementing our SAR based approach

The prototype has been designed to exploit the potential behind utilising SAR as a design tool during an industrial
The system has been designed in a modular fashion, allowing blocks of constraints to be applied to expand the functionality. The construction is essentially divided into three stages:

**Constraint Logic:** This incorporates our application of constraints. Our constraints utilise vector floats, allowing the use of vector maths for calculating varying geometric measurements within our design scene. This stage considers the implementation of each constraint, and how the combination of their effects will be handled by the system.

**Scene Logic:** This area involves the ideas presented in Shader Lamps (Raskar, R. et al. 2001), involving the creation and application of textures, colours, and physical models into the scene. The scene logic also applies the calibrations between projectors, physical models and tracking software, aligning all coordinate systems.

**Tangible Interaction:** This stage involves the design and implementation of a tangible user interface, to allow an intuitive avenue for designers to access both the dynamic selection and alteration of projected content. The interactions can only be fine-tuned after the scene logic has been prepared and configured, requiring a calibration between tracked and projector coordinate systems.

### 3.1 Constraint Logic

The following section will provide an outline of our proposed constraints, and the vector maths involved to produce our results. Due to the complexity of producing an exhaustive computer aided drawing functioning system, some limitations were decided upon to better serve our exploration of the proposed amendments to the design process. We limited the inclusion of four constraints; distance, angle, parallel, and grid, with each constraint being limited in application to a set number of objects.

As discussed earlier, all of our constraint logic is based on vector geometry. This design choice allows access to an abundant amount of vector theory for calculating distances, rotations, relationships in three dimensions. We utilise a three element vector for both location and directional properties of all projected content.

Information formulas allow the designer to gain an understanding of the current layout of the scene. The distance knowledge tool would allow the designer to learn the physical distance between two elements within the scene. Likewise, through the use of the angle knowledge tool, the designer would be granted the knowledge of the internal angle between three elements.

Change formulas utilise user input to alter the physical layout of the scene. To alter the distance between two elements, the designer would first select two elements, apply the distance constraint, and enter the required distance. This sequence of events would result in the objects moving the required distance apart.

Distance is an important facet to a designer, as it describes the positional relationships between elements in a scene. The role of distance can be used to determine the space between two objects, the length of a line, or the distance between a line and object. To determine the distance between two points, the two known positions are plugged into equation 1. The returned value is a float describing the distance between the two points.

\[
|xyz| = \sqrt{(\Delta x)^2 + (\Delta y)^2 + (\Delta z)^2} \tag{1}
\]

To illustrate our design solution, we also have allowed the changing of distances between elements. We have implemented this in an ‘as is’ basis. We initially calculate the current trajectory between elements A and B as seen in equation 2.

\[
\overrightarrow{AB} = B - A \tag{2}
\]

This direction is then used within equation 3, along with the designer’s input distance, to provide a new point the required distance away from point A.

\[
B(\text{origin}) = A(\text{origin}) + \text{distance} \times \frac{\overrightarrow{AB}}{\overrightarrow{AB}} \tag{3}
\]

The parallel constraint is described in Figure 2. The first row in the hierarchical table allows parallelisation on an arbitrary axis, rather than constrained to either X, Y or Z planes. The parallel tool will always conform the

<table>
<thead>
<tr>
<th>1st plane to check</th>
<th>Arbitrary Plane</th>
</tr>
</thead>
<tbody>
<tr>
<td>If parallel in the arbitrary plane, check the closest aligned axis</td>
<td>XY, YZ, XZ</td>
</tr>
<tr>
<td>If already parallel in one of the axis, check the next closest aligned axis</td>
<td>YZ, XZ, XY</td>
</tr>
</tbody>
</table>

**Figure 2: Parallel constraint logic flow**
projection objects exist on the face of the projection prop using the up vector, before checking further planes for parallelisation. After each application, the tool will apply the next closest axis for parallelisation; however this will not be completed if the resulting constraint moves both objects to the same point in space.

To learn the inner angle between two lines, or three objects, a dot product can be applied to the two known vectors. For the case of objects, vectors b and c are the direction vectors (equation 2) between shapes AB & AC. By plugging these two vectors into equation 5, the returned value is given in radians, requiring a conversion to degrees for use by the designer (radians $\times \frac{180}{\pi}$).

$$\vec{b} \cdot \vec{c} = |\vec{b}| |\vec{c}| \cos(\theta)$$

$$\vec{u} \vec{p} = \overrightarrow{AB} \times \overrightarrow{AC}$$

$$\vec{c} = R \times \vec{b}$$

To provide the same functionality to the designer as provided with the distance constraint, a change constraint is also provided for use. Our implementation uses a rotation matrix to allow for the change of an inner angle between projection elements. The matrix, as seen in (4) is 3x3 and uses values determined by the input user angle. The chosen angle is converted to radians (from degrees) and is used to produce both $c$ (cosine*angle) & $s$ (sine*angle).

The value $t$ is the remainder of $c$ subtracted from 1, while $up$ represents the up vector of the plane of objects (calculated through the cross product of vectors $b$ & $c$ (equation 6)). This matrix is produced and then multiplied against vector $b$, to produce a new vector $c$ (equation 7).

### 3.2 Scene Logic

The implementation of our prototype system incorporates the use of 2 x NEC NP510WG projectors to provide a complete coverage of all prop surfaces. Our tangible input tools are tracked in the working area by 6x OptiTrack Flex:V100R2 cameras. The TrackingTools software generates 6DOF coordinates for each tracked object, and sends the information across the network within VRPN packets. The SARventor system then converts each received message on an object basis into projector coordinates and applies the required alterations. Our prototype is run in OpenGL within the WCL SAR Visualisation Studio, making use of a single server containing 4 x nVidia Quadro FX3800 graphics cards. Our physical prop is created in a digital form and loaded into the SARventor framework. The digital vertices of the model are used to assist in complying constraint logic with the physical representation of the model.

### 3.3 Tangible Interaction

With the motivation of this research aiming to create a design solution, which will allow designers to adaptively create a visual design within a SAR environment, our tangible user interface has been constructed with an aim to allow free-hand movement whilst also allowing the rigid application of constraints into the scene. The TUI will remove the need for a keyboard as the primary source of interaction, while still allowing further development and additions to take place.

During the development of the interface, a number of requirements were deemed necessary to achieve our goals. The requirements are based around the interactions within each varying mode, also considering the numerical data entry for constraint values. Most desktop applications are touted as having an easy to use and intuitively designed interface for user interaction. This key area of design will ultimately impact largely on the users’ experience of the system. Over the years, a considerable amount of research has been conducted into investigating various approaches of user interaction within three dimensional spaces (Bowman et al. 2005; Forsberg et al. 1997; Ishii & Ullmer 1997). Although within a three dimensional space, the visual experience can vary considerably, the underlying principles of an interface are present regardless of the medium. For this reason, it is essential to the goals of the system that an interaction paradigm be selected which both presents a suitable application to the design while meeting performance measurements.
By creating a three dimensional user interface, inherent characteristics can be utilised to better provide a collaborative experience. The solution uses the mental model from many fundamental 'hands-on' trade occupations, using a toolbox approach for all the interface tools. Within a designer's toolbox, individual jobs require particular tools. By removing a single point of contact, multiple interaction points between the user and the system can exist concurrently. Just as a carpenter would carry a hammer, nail and ruler within their toolbox, we foresee a designer having tools which offer the same variety of needs. By considering the most important facets of a design solution, we categorised our tangible markers into three varying functions; Mode markers, Action markers, and Helper markers.

Mode markers are the most abundant tool. We use the term mode to describe this group of markers because of their singular functionality. Each different function is digitally assigned to an individual mode marker (Figure 4). Mode markers are executed by stamping them onto the physical prop. Each successful stamping would result in the execution of the digital representation of the marker. Action markers are our primary interaction marker within the setting, controlling the selection and manipulation of content within the scene (Figure 4). Helper markers are essentially action markers, but provide additional functionality when combined into the setting with another action marker. This would allow rotation or scaling to be done without needing to grab a special tool, as required by our mode markers.

Our tangible markers are displayed in Figure 6. Each marker has optical balls attached to allow tracking by our tracking system. Our action marker is designed like a stylus, drawing upon known mental models from touch screen devices. An example of our action marker in use can be seen in Figure 5, where a user can select and manipulate projected content through a point and drag technique. Our mode markers are designed like a stamp, with a larger handle for easier grasping, again drawing upon a known mental model. Each mode marker is physically designed to assist the user in visually recognising their digital representations. Our distance mode tool has the letter ‘D’ moulded to its base, while the circle mode tool is shaped like a circle.

4 Expert Analysis
A qualitative review process was conducted with three experts within the area of architecture and industrial design who were not involved in the research. Each reviewer has over ten years experience within his or her individual areas of expertise. The review was conducted using an interrupted demonstration of our SARventor prototype system. The process that was adopted was to provide a theoretical background of the work, before moving onto an interactive demonstration of the system. Interruptions and questions were encouraged throughout the process to ensure that first impressions were recorded.

Three examples were demonstrated to encourage a broad understanding of the applicability of our system, and its potential uses in the industrial design process.

Texturing:
The first example provided an illustration to the power that SAR offers though its visual and physical affordance. Changes to both colour and textures can be quickly altered to allow fast and effective visualisations of potential colour schemes. Each colour and texture was demonstrated to the experts, giving a baseline understanding of the visual properties that SAR offers.

Tangible User Interface:
The second example explained our TUI, demonstrating both selection and manipulation with our action marker, illustrating our proposed interaction techniques. This example also exemplified our proposed interaction with mode markers, allowing quick and easy addition to primitive shapes onto the model. Each use case was
demonstrated, giving the expert reviewers an understanding of our proposed interaction metaphors.

Constraints:
Our final example made use of both previous examples to show an example for applying geometrical constraints between projected elements on our physical prop. This allowed a simple scenario of a very simple design begin to grow through the placement and constraining of projected content.

The demonstrations were provided to give the reviewers a wide understanding of our implementation. The information provided was practical in nature, with an emphasis on the resulting discussion on how particular elements would be affected by particular scenarios. Essentially, the prototype was used as a springboard to further discussion on more advanced elements, and the resulting repercussions from chosen design characteristics.

At the conclusion of our review session, a discussion was held on our proposed amendments to the current Industrial design process. During this time, the reviewers were given free use to experience our prototype. A particular emphasis was given on the areas where we foresaw an improvement through the use of SAR, allowing greater affordance to designers through the use of a physical prop. Our proposed improvements were relayed in questions, encouraging a detailed response from each reviewer’s opinions, allowing for an opportunity to discuss areas which were not considered by the researchers.

4.1 Results
Reviews show that our SAR based approach does offer an opportunity to be leveraged into the current design process. The initial use for the prototype being aimed around the designer was seen as potentially viable, however the use of a computer was seen as an integral part of our reviewers work, with the ability to manage designs and models. Allowing models and parts to be dragged across into the design was seen as an important feature to help streamline the use of SAR into their current workflow. They felt that there was no need to completely remove the computer from their workflow, and were happy to conduct some of their work on a computer workstation.

The idea of being able to visually organise a design with the affordance of SAR was intriguing to the reviewers however the lack of a visible toolkit was noted as a problem. The issue with a menu based approach in a 3D environment requires it to exist either on the tabletop or on the floor around the design medium. It also raises further issues with orientation, and whom it should orientate towards.

The biggest limitation with the proposed SAR based approach was its inherent property of being a surface based medium. Both the user interface and the design space were inherently attached to a surface, disallowing the true manipulation of volume. This was seen as an issue with certain design cases, and areas of use within the design process.

It was agreed however that SAR itself is not a tool to replace current applications within the industrial design process, but to complement them instead. By looking beyond the designer as the sole user of our SAR based approach, further opportunities arise.

Allowing dynamic alteration of content during meetings, and allowing the annotations of reasoning for the changes would allow SAR to become much more than just a tool for designers to utilise, it would become a marketing medium to be used throughout the entire industrial process. Our constraint based approach would allow structured changes to be applied during these meetings, further providing an opportunity for structured reasoning for changes.

The key areas noted during the expert review of our SAR based approach are as follows:

- The prototype system was seen as a viable design solution, capable of being of benefit to designers during the conceptual and detail design stages of the process, with some further additions to assist in the logging of information, change and reasoning.
- The designers felt that the prototype could work as a designer’s tool; however its inability to work with volume limited its uses.
- A strong case towards being used as a collaborative tool for use in feedback sessions between designers and stakeholders.
- Being used as a collaborative tool would assist in a healthier, more creatively immersive design by the enhanced view received from feedback sessions.
- With the added ability to manipulate content on the projection model, stakeholders would be able to make minor design changes during the feedback session, and allow for an updating of content based on these decisions.
- SAR has a valuable role in the industrial design process. An emphasis was seen on the communication stages, Market/User Needs & Sales as a strong area for application. The use of SAR within the other areas of the process would also assist in producing a more complete design, with each stage benefiting from its use in various ways.
- These communication stages have a high stakeholder involvement, and can benefit from the high fidelity SAR offers. Users will have a more realistic mental representation of the final product as they have seen a prototype that has the same form as the final product rather than a small rendered image on a screen.

Due to the successful response from our review, the following areas for focus would be to investigate further expansion of the system to include the ability to log and adapt changes during design and visual feedback sessions and produce quantitative testing of our interaction techniques. From the review process, it was found that SAR has an effective use within the current industrial design process. The affordance SAR offers and the transparency that a TUI gives its users demonstrates an approach that can have a greater success in driving superior levels of feedback between designers and stakeholders. With the added ability to annotate and track
changes within our SAR setting, we would have a tool which would provide an indispensible opportunity for designers to take advantage of during the duration of the process.

**Market/User Needs**

Information gathering is an important facet of this stage of the process. Through the use of SAR, visual designs can be presented to stakeholders with improved feedback and responses. This is achieved through controlled choice, allowing users to interact transparently with the system, while gauging their reception of each choice. Being a collaborative medium, SAR offers the opportunity to mix subjects from different backgrounds during the one session, offering a further in-depth analysis of the proposed designs. This was unanimously seen as a valuable role for SAR.

**Product Design Specification**

Using SAR throughout other stages of the development process, an enhanced understanding of the design scenario can be realised resulting in a much more information rich Product Design Specification (PDS). With SAR being a digital format, an opportunity arises for the PDS to be updated as changes and amendments are made. Social constraints can be applied to a PDS, and team reviews can be conducted agreeing or rejecting the proposed changes. Timestamps for particular changes can be automatically recorded, as well as the participants involved. Automatic updating of the PDS would help to minimise the risk of human errors.

**Conceptual / Detail Design**

Contributing to our approach, the reviewers see the strength of SAR also being used as a tool for feedback during these stages. Incorporating experts, focus groups and key users, further improvements can be quantified through these feedback sessions. Individuals would be able to interact and move around the model, getting an appreciation for the intended impact of design decisions. The ability to apply changes to the model would further assist in the updating of changes during these feedback sessions, with proposed changes being logged within the PDS.

**Manufacturing**

SAR can be used to quantify the accuracy of manufactured goods compared to the proposed digital model. SAR also offers the ability to have an animated and functional model before being sent off for manufacturing. This allows for the checking of interactions to be performed before a prototype is produced. This also offers the final opportunity for feedback from stakeholders before the financial outlay of a working prototype.

**Sales**

Sales was seen in the same light as Market/User Needs and unanimously seen by the reviewers as a valuable role for SAR. With the finished product, SAR offers an interactive medium to demonstrate its benefits. For kitchen appliances, SAR can replace particular products within the same space, saving on the space requirements of bricks and mortar stores. Customers are able to select the product which interests them, then alternate the surface textures from the predefined selection. This would provide a much more complete and satisfying shopping experience for buyers, gaining a better understanding on the products that interested them. Building a house requires decisions to be made on wall colours, bricks, taps, handles, counter tops, cupboard designs. This is all done from pictorial books and demo products glued to a wall. The owner is required to use their imagination to realise the ramifications of their decisions. Using a mock room, SAR allows owners to see their ideas come alive and allow a better understanding of their choices.

This application of SAR also offers an opportunity for Market/User needs to be included within a Sales application. Including conceptual ideas within the above mentioned sales approaches would allow market research to be conducted on the intended market users, during their sales decisions, allowing for more informative responses.

**SAR Conceptual Rapid Design**

Our proposed area of improvement in the industrial process was seen as a strong influence with the inclusion of SAR. Allowing for the rapid experimenting of various physical and digital designs could be very accommodating during the conceptual phase of the design process. The opportunity to learn of factors not considered until later in the process, while having an opportunity to gain feedback from stakeholders over multiple models would improve the quality of understanding during focus groups. Being of a digital nature, SAR’s inclusion would allow the re-use of previous designs, colour schemes, and layouts, further improving knowledge during this idea driven phase of the process.

SAR’s transparency allows a project’s stakeholders, people who the designers are required to communicate with, the opportunity to offer their feedback and opinions in a much more complete fashion. By offering a prototype which has affordance and interaction, people will be more willing to offer a personal opinion, instead of blindly accepting what is being shown from not actually understanding what is being presented to them. It also allows stakeholders to utilise the space and role-play the use of the product more effectively, assisting in alerting designers to any mistakes within their design.

Through the use of SAR, the high fidelity functionality of the model encourages a higher degree of interaction from the stakeholders, ensuing with a greater assessment of the design. One of the compelling features of the SAR based design is that stakeholders (designers and customers) are literally able to walk around in the design.

### Conclusions and Future Work

This paper has presented new techniques for supporting constraint based product design within a SAR environment. A TUI was produced which gives the designer a toolbox of tangible items to allow a structured approach for amending a design through interaction with the physical prop. Geometric constraints have been designed for use within our prototype system to allow the validation of our proposed amendments to the industrial
design process. Designers are able to add projected elements onto the physical prop, dynamically alter their position and apply structured constraints between fellow projections.

The prototype was evaluated by professional designers through a qualitative expert review. Initial results show promises to SAR becoming incorporated into the current industrial design process. Our SAR Conceptual Rapid Design phase offers designers an early window of opportunity for experimenting with potential designs offering affordance and interaction between themselves and stakeholders. This is seen as an integral part of improving the communication process. They felt that the initial thought of designing the prototype for use solely by designers, limited its potential. By offering collaborative measures, including annotations and the logging of changes, it would help the tool to become a more applicable solution for industrial application. This would provide an ability to apply social constraints to the session, offering higher security, accuracy, and accountability during collaborative sessions.

Future work would consider these collaborative measures, with an emphasis on providing a communication medium between stakeholder and designer throughout all stages of the design process.

6 References


Pugh, S 1990, Total design: integrated methods for successful product engineering, Addison-Wesley Wokingham, UK.


conference on Computer graphics and interactive techniques.


Music Education using Augmented Reality with a Head Mounted Display

Jonathan Chow 1 Haoyang Feng 1 Robert Amor 2 Burkhard C. Wünsche 3

1 Department of Software Engineering University of Auckland, New Zealand
Email: jcho205@aucklanduni.ac.nz, hfeng020@aucklanduni.ac.nz
2 Software Engineering Research Group, Department of Computer Science University of Auckland, New Zealand
Email: trebor@cs.auckland.ac.nz
3 Graphics Group, Department of Computer Science University of Auckland, New Zealand
Email: Burkhard@cs.auckland.ac.nz

Abstract

Traditional music education places a large emphasis on individual practice. Studies have shown that individual practice is frequently not very productive due to limited feedback and students lacking interest and motivation. In this paper we explore the use of augmented reality to create an immersive experience to improve the efficiency of learning of beginner piano students. The objective is to stimulate development in notation literacy and to create motivation through presenting as a game the task that was perceived as a chore. This is done by identifying successful concepts from existing systems and merging them into a new system designed to be used with a head mounted display. The student is able to visually monitor their practice and have fun while doing so. An informal user study indicates that the system initially puts some pressure on users, but that participants find it helpful and believe that it improves learning.

Keywords: music education, augmented reality, cognitive overlap, human-computer interaction

1 Introduction

Music is an important part of virtually every culture and society. Musical traditions have been taught and passed down through generations. Traditionally, Western culture has placed a large emphasis on music education. For example, the New Zealand Curriculum (New Zealand Ministry of Education 2007) defines music as a “fundamental form of expression” and states that music along with all other forms of art help stimulate creativity.

Traditional music education focuses on individual practice assisted by an instructor. Due to time and financial constraints most students only have one lesson per week (Percival et al. 2007). For beginner students, this lesson usually lasts half an hour, and the majority of time spent with the instrument is without any supervision from an instructor. Sanchez et al. (1990) note that during these unsupervised practice times students may play wrong notes, wrong rhythms, or simply forget the instructor’s comments from previous lessons. These issues all hinder the learning process and “provide a source of frustration to both teachers and students”. Sanchez also notes that “much of the joy that should accompany the discovery of music dissipates during practice time”.

Duckworth (1965) reports that a lack of motivation and interest has been a common problem through history. Problems such as neglecting to teach critical skills such as sight-reading (i.e., playing directly from a written score without prior practice) were already a major concern as early as 1901.

The use of multimedia to enhance practice has been explored previously. Percival et al. (2007) present guidelines for making individual practice more beneficial and note that “by ‘wrapping’ a boring task ... in the guise of a nostalgic computer game, the task becomes much more fun”. The authors give several examples of games that achieve this goal, including some of their own work. They note that due to the subjective nature of the quality of music, computer-aided tools are more suitable for technical exercises where quality and performance can be objectively measured.

Most computer supported music education tools use a traditional display to convey information to the user. Augmented Reality (AR) can be used to create a more direct interaction between the student and the system. Azuma (1997) describes augmented reality as creating an environment in which the user “sees the real world, with virtual objects superimposed upon it”. The author goes further to explain that “virtual objects display information that the user cannot directly detect with his own sense” and that “the information conveyed by the virtual objects helps a user perform real-world tasks ... a tool to make a task easier for a human to perform”. Azuma (1997) presents an overview of a broad range of disciplines that have used augmented reality, such as medical training, military aircraft navigation and entertainment. The review suggests that AR has been successfully used in a wide variety of educational applications. The main advantage of AR is that a perceptual and cognitive overlap can be created between a physical object (e.g., instrument) and instructions on how to use it.

A head mounted display can be used to combine real and virtual objects and in order to achieve an immersive experience. Two types of devices exist: optical see-through and video see-through. An optical see-through device allows the user to physically see the real world while projecting semi-transparent virtual objects on the display, while a video see-through device uses cameras to capture an image of the real
world which is processed with virtual objects and the entire image is displayed on an opaque display. An optical see-through device is preferable in real-time applications due to its lower latency and facilitation of a more direct interaction with real world objects.

This work is an attempt to overcome some of the deficiencies in the traditional music education model by using augmented reality to create a perceptual and cognitive overlap between instrument and instructions, and hence improve the end users’ learning experience and motivation.

Section 2 reviews previous work using visualisations and VR/AR representations for music education. Section 3 presents a requirement analysis, which is used to motivate the design and implementation of our solution presented in sections 4 and 5, respectively. We summarise the results of an informal user study in section 6 and conclude our research in section 7. Section 8 gives an outlook on future work.

2 Related Work

A review of the literature revealed a number of interesting systems for computer-based music education. Systems for piano teaching include Piano Tutor (Dannenberg et al. 1990), pianoFORTE (Smoliar et al. 1995), the AR Piano Tutor (Barakonyi & Schmalstieg 2005), and Piano AR (Huang 2011). Several applications for teaching other instruments have been developed (Cakmakci et al. 2003, Motokawa & Saito 2006). We will review the Digital Violin Tutor (Yin et al. 2005) in more detail due to its interesting use of VR and visualisation concepts for creating a cognitive overlap between hand/finger motions and the resulting notes.

The Piano Tutor was developed by Dannenberg et al. (1990) in collaboration with two music teachers. The application uses a standard MIDI interface to connect a piano (electronic or otherwise) to the computer in order to obtain the performance data. MIDI was chosen because it transfers a wealth of performance related information including the velocity at which a note is played (which can be used to gauge dynamics) and even information about how pedals are used. An expert system was developed to provide feedback on the user’s performance and notes are displayed on a computer screen placed in front of the user. User performance is primarily graded according to accuracy in pitch, timing and dynamics. Instead of presenting any errors directly to the user, the expert system determines the most significant errors and guides the user through mistakes one by one.

Smoliar et al. (1995) developed pianoFORTE, which focuses on teaching the interpretation of music rather than the basic skills. The authors note that music “is neither the notes on a printed page nor the motor skills required for the proper technical execution”. Rather, because music is an art form, there is an emotional aspect that computers cannot teach or analyse. The system introduces more advanced analysis functionalities, such as the accuracy of articulation and synchronisation of chords. Articulation describes how individual notes are to be played. For example, staccato indicates a note that is separate from neighbouring notes while legato indicates notes that are smoothly transitioned between with no silence between them. Synchronisation refers to whether notes in a chord are played simultaneously and whether notes of equal length are played evenly. These characteristics form the basis of advanced musical performance abilities. In terms of utilised technologies, pianoFORTE uses a similar hardware set-up as Piano Tutor.

The AR Piano Tutor by Barakonyi & Schmalstieg (2005) is based on a “fishtank” AR setup (PC+monitor+webcam), where the physical MIDI keyboard is tracked with the help of a single optical marker. This puts limitations on the permissible size of the keyboard, since for large pianos the user’s view might not contain the marker. The application uses a MIDI interface to capture the order and the timing of the piano key presses. The AR interface gives instant visual feedback over the real keyboard, e.g., the note corresponding to a pressed key or wrongly pressed or missed keys. Vice versa, the keys corresponding to a chord can be highlighted before playing the chord, and as such creating a mental connection between sounds and keys.

A more recent system presented by Huang (2011) focuses on improving the hardware set-up of an AR piano teaching system, by employing fast and accurate markerless tracking. The main innovation with regard to the visual interface is use of virtual fingers, represented by simple cylinders, to indicate the hand position and keys to be played.

Because MIDI was created for use with equipment with a rather flexible form of input (such as pianos, synthesesers and computers), a purely analogue instrument such as the violin cannot use MIDI to interface with a computer. The Digital Violin Tutor (Yin et al. 2005) contains a “transcriber” module capable of converting the analogue music signal to individual notes. Feedback is generated by comparing the student’s transcribed performance to either a score or the teacher’s transcribed performance. The software provides an extensive array of visualisations: An animation of the fingerboard shows a student how to position their fingers to produce the desired notes, and a 3D animated character is provided to stimulate interest and motivation in students.

3 Requirements Analysis

An interview with an experienced music teacher (Shacklock 2011) revealed that one of the major difficulties beginner students have is translating a note from the written score to the physical key on the keyboard. Dirkse (2009) notes that this functional skill is taught by the teacher. None of the previously reviewed systems addresses this problem. Furthermore, with the exception of the Digital Violin Tutor, none of the reviewed systems addresses the problem of lacking student interest and motivation. This issue is especially relevant for children who are introduced to music education through school curricula or parental desires, rather than by their own desire. Our research focuses hence on these two aspects.

Augmented Reality has been identified as a suitable technology for the above goals, due to its ability to create a perceptual and cognitive overlap between instrument (keys), instructions (notes), and music (sound). The association of visuals with physical keys enables users to rapidly play certain tunes, and hence has the potential to improve the learning experience and increase motivation. In order to design suitable visual representations and learning tasks, more specific design requirements must be obtained.

3.1 Target Audience

Similar to the Piano Tutor, we target beginner students, with the goal of teaching notation literacy and basic skills. This is arguably the largest user group, and is likely to benefit most from an affordable and fun-to-use system.
3.2 Instrument choice

From the various available music interfaces, the MIDI interface is most suitable for our research. It provides rich, accurate digital information which can be used directly by the computer, without the signal processing required for analogue input. MIDI is also a well-established industry standard. In order to avoid any analog sound processing we choose a keyboard as an instrument. In contrast to the work by Barakonyi & Schmalstieg (2005), we do not put any size restrictions on the keyboard and camera view, i.e., our system should work even if the users sees only part of the keyboard.

3.3 Feedback

The system should provide feedback about basic skills to the user, i.e., notation literacy, pitch, timing and dynamics. The feedback should be displayed in an easily understandable way, such that improvements are immediately visible. This can be achieved by visually indicating the key each note corresponds to. One way to achieve this is by lighting up keys using a superimposed image, as done in the Augmented Piano Tutor by Barakonyi & Schmalstieg (2005).

3.4 Motivation and Interest

It is important that the system fosters motivation and interest, as this will increase practice time and hence, most likely, learning outcomes. One popular way to achieve this is by using game concepts. Percival et al. (2007) cite several successful educational games in areas other than music. They note that the game itself does not have to be extremely sophisticated; merely by presenting a seemingly laborious task as a game gives the user extra motivation to persevere. Additional concepts from the gaming field could be adapted, such as virtual “badges” and “trophies” to reward achievements (Kapp 2012).

4 Design

4.1 Physical Setup

Based on the requirements, the physical setup comprises one electronic keyboard, one head mounted display with camera, and one computer for processing. The user wears the head mounted display and sits in front of the keyboard. The keyboard connects to the computer using a MIDI interface. The head mounted display connects to the computer using a USB interface. The head mounted display we use for this project is a Trivisio ARvision-3D HMD1. These are video see-through displays in that the displays are not optically transparent. The video captured by the cameras in front of the device must be projected onto the display to create the augmented reality effect. The keyboard we use for this project is a generic electronic keyboard with MIDI out. Figure 1 illustrates the interactions between these hardware components.

4.2 User Interface

As explained in the requirements analysis, the representation of notes in the system must visually indicate which key each written note corresponds to. We drew inspiration from music and rhythm games and Karaoke videos, where text and music are synchronised using visual cues. In our system each note is represented as a line above the corresponding key, where the length of the line represents the duration of the note. The notes approach the keys in the AR view in a steady tempo. When the note reaches the keyboard, the corresponding key should be pressed. Similarly, when the end of the note reaches the keyboard, the key should be released. This is drawn on a virtual overlay that goes above the keyboard in the augmented reality view as illustrated in Figure 2.

At the same time, the music score is displayed above the approaching notes. In order to help improving notation literacy, a rudimentary score following algorithm follows the notes on the written score as each note is played. The music score and virtual notes are loaded from a stored MIDI file. This MIDI file becomes the reference model for determining the quality of the user’s performance. This means that users must have an electronic version of the piece they want to practice, either by downloading one of the many MIDI music templates available on the Internet, or by recording an instructor playing the piece. A MIDI file contains timings for each note, which are strictly enforced by our system. This is in direct contrast to Piano Tutor, which adjusts the music’s tempo to suit the user. We decided to force timings, since maintaining a steady tempo despite making mistakes is a skill that musicians need (Dirkse 2009). However, the user has the option to manually adjust the tempo of a piece to suit their ability. This makes an unfamiliar piece of music easier to follow, since there is more time to read the notes. Slow practice is a common technique for improving the fluency of a piece of music (Nielsen 2001). This feature encourages the user to follow these time-tested processes towards mastery of a piece of music.

We added a Note Learning Mode, which pauses
each note as it arrives and waits for the user to play the key before continuing to the next note. This takes away any pressure the user has of reading ahead and preparing for future notes. By allowing the user to step through the notes one by one, the user gets used to the hand and finger motions, slowly building the dexterity required to play the notes at proper speed.

4.3 Augmented Reality Interface

Creating the augmented reality interface requires four steps:

1. Capture an image of what the user can see.
2. Analyse the camera image for objects of interest (Feature detection).
3. Superimpose virtual objects on the image (Registration).
4. Display the composite image to the user.

Steps 2 and 3 are the most complex steps and are explained in more detail.

4.3.1 Feature Detection

The feature detection step can be performed by directly analysing the camera image using computer vision techniques. An alternative solution is to use fiduciary markers and to define features within a coordinate system defined by the markers. Feature detection using markers is easier to implement and usually more stable, but often less precise and requires some user effort for setting up the system (placing markers, calibration). In our application a markerless solution is particularly problematic, since the camera view only shows a section of the keyboard, which makes it impossible to distinguish between keys in different octaves. A unique identification of keys would either require global information (e.g., from the background) or initialisation using a unique position (e.g., the boundary of the keyboard) followed by continuous tracking. We hence chose a marker-based solution based on the ARToolkit software. The software uses markers with a big black border, which can be easily identified in the camera view and hence can be scaled to a sufficiently small size. NyARToolkit is capable of detecting the position and orientation (otherwise known as the pose) of each marker and returns a homogeneous 3D transformation matrix required to translate and rotate an object in 3D space so that it is directly on top of the detected marker. Because this matrix is a standard mathematical notation, it can be used directly in OpenTK.

4.3.2 Registration

A critical component of the AR interface is to place the visualisations of notes accurately over the correct physical keys of the keyboard in the camera view. While detecting the pose of a single marker is simple using the ARToolkit, placing the virtual overlay is more difficult. The first problem comes from the fact that the user is positioned very close to the keyboard when playing, and hence usually only sees a section of the keyboard. Hence multiple markers must be laid out along the length of the keyboard such that no matter where the user looks there will still be markers visible to the camera.

We decided to use identical markers for this purpose, since our experiments showed that detecting differing markers at the same time significantly reduced performance. This was slightly surprising and might have to do with problems with the utilised libraries, the hardware, or the set-up (e.g., insufficient size of the markers).

![Figure 3: Marker configuration with unique relative distances between every pair of markers.](image)

Figure 3: Marker configuration with unique relative distances between every pair of markers.

We overcame this problem by using identical markers and devising a pattern for the markers such that the relative distance between any two markers is unique. Figure 3 illustrates this set-up. If two markers are visible in the camera view, then the distance between them (in units) can be computed from the size of the markers, and be used to identify the pair.

Figure 4 shows an example. Marker 3 is at position (0, 3) and marker 4 is at position (1, 0). If the camera can only see the area within the orange rectangle, the algorithm will calculate the positions of the markers and determine that they are 1 unit apart horizontally and 3 units apart vertically. The only markers in the figure that satisfy this constraint are markers 3 and 4. Since we know that the user is positioned in front of the keyboard, there are no ambiguities due to orientation, and the origin of the marker coordinate system can be computed and used to position the virtual overlay onto the keyboard in the camera view.

![Figure 4: Example of deducing the origin based on a limited camera view.](image)

Figure 4: Example of deducing the origin based on a limited camera view.

A further problem encountered for the registration step was jittering and shaking of the overlay, due to noise and numerical errors in the markers’ pose calculation. This not only makes it difficult to associate note visualisations with physical keys, but it is also very irritating to the user. We found that this problem was sufficiently reduced by taking a moving average of the transformation matrix for positioning the overlay into the camera view. The optimal kernel size of this moving averages filter depends on the camera quality. A larger kernel size results in a more stable overlay, but reduces response time when the user changes the view direction. Higher quality cameras with higher frame rates achieved stable registration even for small filter kernel sizes.

Figure 5 shows the augmented reality view of the keyboard.

![Figure 5: Augmented reality view of the keyboard.](image)
4.4 Performance Analysis and Feedback
The MIDI interface is used to obtain the user’s performance for analysis. MIDI is an event-based format; each time a key is pressed or released, a digital signal containing information about the way the note was played is sent to the computer. This information includes the note that was played and the velocity at which the note was played. A high velocity indicates a loud sound, while a low velocity indicates a soft sound. The time at which the note was played can be inferred from when the event was received. MIDI also supports information about other keyboard functionalities, such as pedals or a synthesiser’s knobs, but this information was outside this project’s scope.

The user’s performance must be compared against some reference model in order to assess it. Since MIDI is capable of storing such detailed information, we decided to use recorded MIDI files of the music pieces as reference models. This allows evaluating the user’s note accuracy and rhythm accuracy. Other information, such as dynamics or articulation, can be added, but as explained previously, were considered too advanced for beginners.

Feedback is important as it allows the user to learn from mistakes and to set goals for future practice. Real-time feedback on note playing accuracy is provided by colour coding the note visualisations in the AR view as illustrated in figure 6. Colour is the most appropriate visual attribute for representing this information, since colours are perceived preattentively (Healey & Enns 2012), colours such as red and green have an intuitive meaning, colours do not use extra screen space (as opposed to size and shape), and colour changes are less distracting than changes of other visual attributes (such as shape).

At the end of a performance additional feedback is given summarising how many notes were hit and missed (see figure 7). The summary allows users to monitor improvements, to compare themselves to other students or expected standards, and to set goals for subsequent practices.

5 Implementation
The application was written in C#. Although many graphics-related libraries are written in C++, the advantages of having a rich standard library, garbage collection, and simplified interface design were considered important for rapid prototyping. For capturing images from the camera, we used a .NET wrapper for OpenCV called Emgu CV. For detection and tracking of virtual markers, we used NyARToolkit, a port of ARToolkit. For displaying and drawing graphics, we used OpenTK, a .NET wrapper for OpenGL. For interfacing with the MIDI device, we used midi-dotnet, a .NET wrapper for the Windows API exposing MIDI functionality.

6 Results
6.1 User Study
A preliminary evaluation of the system was performed using an informal user study with seven participants. All users were students with a wide range of piano playing skill levels, ranging from no experience at all to many years of experience. Users were asked to learn a piece using the system. Open-ended questions were asked of each subject about likes and dislikes, how beneficial they believe the system is and an overall rating of the system.

Four participants (57%) liked the representation of the notes in the AR view, while two (29%) criticised that it was difficult to look at the written notation and concentrate on the virtual notes at the same time. Three users (43%) admitted that they did not look at the written notation at all. Six users (86%) said that keeping up with the approaching notes was very intimidating and the pressure from trying to find the following notes caused them to miss even more notes.

The feedback system, especially the summary at the end, was found to be very helpful. The display of quantitative results allowed users to set goals for improvement. In addition, the game-like nature of the system resulted in participants competing with each other on achieving higher summary feedback scores. All participants believed that the system would be helpful for starting to learn playing piano, and all participants enjoyed using the system.
6.2 Discussion

The goal of this research was to design a system for improving piano students’ notation literacy and their motivation and interest. The results of the preliminary user study are very encouraging, and both experienced and inexperienced piano players enjoyed the use of the system and did not want to stop using it. Having a competitive element has proved advantageous, and for home use the integration of multiplayer capabilities and online hosting of results should be considered.

The display of music notations proved distracting. Many users just wanted to play the piano based on the indicated keys in the AR view, rather than learning to read notes (notation literacy). A possible explanation is that most participants of the study were not piano students, i.e., had no motivation to learn reading of notes, but were keen to learn playing the instrument. More testing is required to investigate these observations in more detail. We also want to explore alternative AR visualisations and user interfaces, especially for combining written notation with the virtual notes.

The responses about the forced timings creating pressure led to the development of the tempo adjustment feature and the note learning mode described earlier. Both of these modes slow down the rate at which notes have to be played, giving the user much more time to decide what to do next.

Evaluating our application with regards to game psychology uncovers the following shortcomings (Caillois 2001, Hejdenberg 2005):

- **Game width**: a game should address multiple basic human needs such as self-esteem, cognitive needs, self-actualisation, and transcendence (the need to help others). Game width could be improved by giving more feedback during piano practice, such as praise, encouragement, and corrections; having social interactions (practicing in pairs or in a group); and by increasing the level of difficulty (adding time limits, obstacles, random events).

- **Imitation**: a game should enable the player to constantly learn. This could be achieved by ranking music pieces by difficulty and by increasing requirements or using different types of visual hints.

- **Emotional impact**: common ways to achieve an improved emotional impact are visual and sound effects and rewards (high score lists, virtual badges).

7 Conclusion

Our preliminary results indicate that the proposed application is useful to budding musicians. As a game, it breeds interest in music and the learning of an instrument. As an educational system it motivates users to practice and improve. With the exception of improving notation literacy, the requirements have been met. We have demonstrated that real-time augmented reality using head mounted displays is a viable way to convey instrument playing skills to a user. Head mounted displays are becoming increasingly available and affordable to consumers, and proposed devices such as Google’s “Project Glass” (Manjoo 2012) demonstrate that such equipment might soon be as common as mobile phones.

8 Future Work

Necessary future developments include performance analysis, such as incorporating dynamics and articulation. This could eventually be integrated into an expert system. A more comprehensive feedback summary would benefit users by narrowing down specific areas for improvement. The score following system can also be improved. Research into techniques for improving the efficiency of learning notation literacy would be beneficial to the music community since this problem has existed for a long time.

A formal user study needs to be performed to determine the usability and effectiveness of the system for piano education. Of particular interest is the effect of wearing AR goggles, and how the effectiveness of the application compares with human piano tutors and computerised teaching tools using traditional displays.

References


Caillois, R. (2001), Man, Play and Games, University of Illinois Press.


Kapp, K. M. (2012), The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education, Pfeiffer.


A Tale of Two Studies

Judy Bowen¹
Steve Reeves²
Andrea Schweer³

(1, 2) Department of Computer Science
The University of Waikato,
Hamilton, New Zealand
Email: jbowen@cs.waikato.ac.nz, stever@cs.waikato.ac.nz

³ ITS Information Systems
The University of Waikato,
Hamilton, New Zealand
Email: schweer@waikato.ac.nz

Abstract

Running user evaluation studies is a useful way of getting feedback on partially or fully implemented software systems. Unlike hypothesis-based testing (where specific design decisions can be tested or comparisons made between design choices) the aim is to find as many problems (both usability and functional) as possible prior to implementation or release. It is particularly useful in small-scale development projects that may lack the resources and expertise for other types of usability testing. Developing a user-study that successfully and efficiently performs this task is not always straightforward however. It may not be obvious how to decide what the participants should be asked to do in order to explore as many parts of the system’s interface as possible. In addition, ad hoc approaches to such study development may mean the testing is not easily repeatable on subsequent implementations or updates, and also that particular areas of the software may not be evaluated at all. In this paper we describe two (very different) approaches to designing an evaluation study for the same piece of software and discuss both the approaches taken, the differing results found and our comments on both of these.

Keywords: Usability studies, evaluation, UI Design, formal models

1 Introduction

There have been many investigations into the effectiveness of different types of usability testing and evaluation techniques, see for example (Nielsen & Landauer 1993) and (Doubleday et al. 1997) as well as research into the most effective ways of running the various types of studies (numbers of participants, expertise of testers, time and cost considerations etc.) (Nielsen 1994), (Lewis 2006). Our interest, however, is in a particular type of usability study, that of user evaluations. We are interested in how such studies are developed, e.g. what is the basis for the activities performed by the participants? In particular, given an implementation (or partial implementation) to test, is there a difference between the sort of study the developer of the system under test might produce and that of an impartial person, and if so do they produce different results? It is well known by the software engineering community that functional and behavioural testing is best performed by someone other than the software’s developer. Often this can be achieved because there is a structured mechanism in place for devising tests, for example using model-based testing (Utting & Legeard 2006) or by having initial specifications that can be understood by experienced test developers (Bowen & Hinchey 1995), or at least by writing the tests before any code is written (as in test-driven or test-first development (Beck 2003)). For user evaluation of software, and in particular the user interface of software, we do not have the sort of structured mechanisms for developing evaluation studies (or models upon which to base them) as we do for functional testing. Developing such studies relies on having a good enough knowledge of the software to devise user tasks that will effectively test the software, which for smaller scale development often means the software’s developer. Given that we know it is not a good idea for functional testing to be carried out by the software’s developer we suggest it may also be true that running, and more importantly, developing, user evaluations should not be done by the developer for the same reasons. This then presents the problem of how someone other than the software’s developer can plan such a study without the (necessary) knowledge about the system they are testing.

In this paper we present an investigation into the differences between two evaluation studies developed using different approaches. The first was developed in the ‘usual way’ (which we discuss and define further in the body of the paper) by the developer of the software-under-test. The second was developed based on formal models of the software-under-test and its user interface (UI) by an independent practitioner with very little knowledge of the software-under-test prior to the modelling stage. We discuss the different outcomes of the two studies and share observations on differences and similarities between the studies and the results.

We begin by describing the software used as the basis for both evaluation studies. We then describe the process of deriving and running the first study along with the results. This is followed by a description of the basis and process of deriving the second study as well as the results of this. We then present a comparison of the two studies and their results, and finish with our conclusions.
The Digital Parrot Software

The Digital Parrot (Schweer & Hinze 2007), (Schweer et al. 2009) is a software system intended to augment its user’s memory of events of their own life. It has been developed as a research prototype to study how people go about recalling memories from an augmented memory system.

The Digital Parrot is a repository of memories. Memories are encoded as subject-predicate-object triples and are displayed in the system’s main view in one of two ways: a graph view and a list view.
shown in Figures 1 and 2. Both views visualise the triple structure of the underlying memory information and let the user navigate along connections between memory items. The type of main view is chosen on program start-up and cannot be modified while the program is running.

The user interface includes four different navigators that can influence the information shown in the main view either by highlighting certain information items or by hiding certain information items. These navigators are the timeline navigator (for temporal navigation; shown in Figure 4), the map navigator (for navigation based on geospatial location), textual search and the trail navigator (for navigation based on information items’ types and connections; shown in Figure 3).

3 The First Study

At the time of the first study, the first development phase had ended and the Digital Parrot was feature-complete. Before using the software in a long-term user study (not described in this paper), we wanted to conduct a user evaluation of the software. Insights gained from the usability study would be used to form recommendations for changing parts of the Digital Parrot’s user interface in a second development phase.

3.1 Goals

The first study had two main goals. The first was to detect any serious usability flaws in the Digital Parrot’s user interface before using it in a long-term user study (not described in this paper), we wanted to conduct a user evaluation of the software. Insights gained from the usability study would be used to form recommendations for changing parts of the Digital Parrot’s user interface in a second development phase.

3.2 Planning the Study

The study was run as a between-groups design, with half the participants assigned to each main view type (graph vs list view). We designed the study as a task-based study so that it would be easier to compare findings between participants. We chose a set of four tasks that we thought would cover all of the Digital Parrot’s essential functionality. These tasks are as follows:

1. To whom did [the researcher] talk about scuba diving? Write their name(s) into the space below.

2. Which conferences did [the researcher] attend in Auckland? Write the conference name(s) into the space below.

3. At which conference(s) did [the researcher] speak to someone about Python during the poster session? Write the conference name(s) into the space below.

4. In which place was the NZ CHI conference in 2007? Write the place name into the space below.

The tasks were chosen in such a way that most participants would not be able to guess an answer. We chose tasks that were not too straightforward to solve; we expected that a combination of at least two of the Digital Parrot’s navigators would have to be used for each task. Since the Digital Parrot is intended to help users remember events of their own lives, all tasks were phrased as questions about the researcher’s experiences recorded in the system. The questions mimic questions that one may plausibly find oneself trying to answer about one’s own past.

To cut down on time required by the participants, we split the participants into two groups of equal size. Each group’s participants were exposed to only one of the two main view types. Tasks were the same for participants in both groups.

In addition to the tasks, the study included an established usability metric, the Systems Usability Scale (SUS) (Brooke 1996). We modified the questions according to the suggestions by Bangor et al. (Bangor et al. 2009). We further changed the questions by replacing “the system” with “the Digital Parrot”. The intention behind including this metric was to get an indication of the severity of any discovered usability issues.

3.3 Participants and Procedure

The study had ten participants. All participants were members of the Computer Science Department at the University of Waikato; six were PhD students and four were members of academic staff. Two participants were female, eight were male. The ages ranged from 24 to 53 years (median 38, IQR 15 years). Participants were recruited via e-mails sent to departmental mailing lists and via personal contacts. Participants were not paid or otherwise rewarded for taking part in the usability test.

In keeping with University regulations on performing studies with human participants, ethical consent to run the study was applied for, and gained. Each participant in the study would receive a copy of their rights as a participant (including their right to withdraw from the study) and sign a consent form. After the researcher obtained the participant’s consent, they were provided with a workbook. The workbook gave a quick introduction to the purpose of the usability test and a brief overview of the system’s features. Once the participant had read the first page, the researcher started the Digital Parrot and briefly demonstrated the four navigators (see Section 2). The participant was then asked to use the Digital Parrot to perform the four tasks stated in the workbook.

Each participant was asked to think aloud while using the system. The researcher took notes. After the tasks were completed, the participant was asked to fill in the SUS questionnaire about their experience with the Digital Parrot and to answer some questions about their background. The researcher would then ask a few questions to follow up on observations made while the participant was working on the tasks.
3.4 Expectations

We did not have any particular expectations related to the first goal of this study, that of detecting potential usability problems within the Digital Parrot.

We did, however, have some expectations related to the second goal. The study was designed and conducted by the main researcher of the Digital Parrot project, who is also the main software developer. Thus, the study was designed and conducted by someone intimately familiar with the Digital Parrot’s user interface, with all underlying concepts and also with the test data. For each of the tasks, we were aware of at least one way to solve the task with one or more of the Digital Parrot’s navigators. We expected that the participants in the study would discover at least one of these ways and use it to complete the task successfully.

3.5 Results

The median SUS score of the Digital Parrot as determined in the usability test is 65 (min = 30, max = 92.5, IQR = 35), below the cut-off point for an acceptable SUS score (which is 70). The overall score of 65 corresponds to a rating between “ok” and “good” on Bangor et al.’s adjective scale (Bangor et al. 2009). The median SUS score in the graph condition alone is 80 (min = 42.5, max = 92.5, IQR = 40), which indicates an acceptable user experience and corresponds to a rating between “good” and “excellent” on the adjective scale. The median SUS score in the list condition is 57.5 (min = 30, max = 77.5, IQR = 42.5). The difference in SUS score is not statistically significant but does reflect our observations that the users in the list condition found the system harder to use than those in the graph condition.

Based on our observations of the participants in this study, we identified nine major issues with the Digital Parrot’s user interface as well as several smaller problems. In terms of our goals for the study we found that there were usability issues, some we would consider major, which would need to be addressed before conducting a long-term user study. In addition there were some issues with the use of navigators. The details of our findings are as follows:

1. Graph: The initial view is too cluttered.
2. Graph: The nature of relationships is invisible.
3. List: The statement structure does not become clear.
4. List: Text search does not appear to do anything.
5. Having navigators in separate windows is confusing.
6. The map is not very useful.
7. Users miss a list of search results.
8. The search options could be improved.
9. The trail navigator is too hard to use.

Some of these issues had obvious solutions based on our observations during the study and on participants’ comments. Other issues, however, were less easily resolved.

We formulated seven recommendations to change the Digital Parrot’s user interface:

1. Improve the trail navigator’s user interface.
2. Improve the map navigator and switch to a different map provider.
3. De-clutter the initial graph view.
4. Enable edge labels on demand.
5. Highlight statement structure more strongly in list.
6. Change window options for navigators.
7. Improve the search navigator.

As can be seen, these recommendations vary greatly in scope; some directly propose a solution while others require further work to be done to find a solution.

4 The Second Study

4.1 Goals

The goals of our second study were to emulate the intentions of the first, that is try to find any usability or functional problems with the current version of the Digital Parrot software. In addition, the intention was to develop the study with no prior knowledge of the software or of the first study (up to the point where we could no longer proceed without some of this information) by using abstract tests derived from formal models of the software and its user interface (UI) as the basis for planning the tasks of the study. We were interested to discover if such abstract tests could be used to derive an evaluation study in this way, and if so how would the results differ from those of the initial study (if at all).

4.2 Planning the Study

The first step was to obtain a copy of the Digital Parrot software and reverse-engineer it into UI models and a system specification. In general, our assumption is that such models are developed during the design phase and prior to implementation rather than by reverse-engineering existing systems. That is, a user-centred design (UCD) approach is taken to plan and develop the UI prior to implementation and these are used as the basis for the models. For the purposes of this work, however, the software had been implemented already and as such a specification and designs did not exist, hence the requirement to perform reverse-engineering. This was done manually, although tools for reverse-engineering software in this manner do exist (for example GUI Ripper[9]), but not for the models we planned to use.

We spent several days interacting with the software and examining screen shots in order to begin to understand how it worked. We wanted to produce as much of the model as possible before consulting with the software designer to 'fill in the gaps' where our understanding was incomplete. Of course, such a detailed examination of the software was itself a form of evaluation, as by interacting with the software comprehensively enough to gain the understanding required and modelling, we formed our own opinions about how usable, or how complex, parts of the system were. However, in general where models and tests are derived prior to implementation this would not be the case. The first study had already taken place but the only information we required about this study was the number and type of participants used (so that we could use participants from the same demographic group) and the basis that the study was conducted as a between-groups study with both graph and list versions of the software being tested. We had no preconceived ideas of how our study might be structured at this point, the idea being that once the models were completed and the abstract tests derived we would try and find some structured way of using these to guide the development of our study.

![Figure 5: Find Dialogue](image)

4.3 The Models

We began the modelling process by examining screen-shots of the Digital Parrot. This enabled us to identify the widgets used in the various windows and dialogues of the system that provided the outline for the first set of models. We used presentation models and presentation interaction models (PIMs) from the work described in (Bowen & Reeves 2008) as they provide a way of formally describing UI designs and UIs with a defined process for generating abstract tests from the models (Bowen & Reeves 2009). Presentation models describe each dialogue or window of a software system in terms of its component widgets, and each widget is described as a tuple consisting of a name, a widget category and a set of the behaviours exhibited by that widget. Behaviours are separated into S-behaviours, which relate to system functionality (i.e. behaviours that change the state of the underlying system) and I-behaviours that relate to interface functionality (i.e. behaviours relating to navigation or appearance of the UI).

Once we had discovered the structure of the UI and created the initial model we then spent time using the software and discovering what each of the identified widgets did in order to identify the behaviours to add to the model. For some parts of the system this was relatively easy, but occasionally we were unable to determine the behaviour by interaction alone. For example, the screenshot in figure 5 shows the “Find” dialogue from the Digital Parrot, from which we developed the following presentation model:

```
FindWindow is
 (SSStringEntry, Entry, ())(HighlightButton, ActionControl, (S_HighlightItem))
 (PMinIcon, ActionControl, (I_PMinToggle))
 (PMaxIcon, ActionControl, (I_PMaxToggle))
 (FXIcon, ActionControl, (I_Main))
 (HSCKey, ActionControl, (S_HighlightItem))
 (TSKey, ActionControl, (?))
```

We were unable to determine what the behaviour of the shortcut key option Alt-T was and so marked the model with a $\mu$ as a placeholder. Once the presentation models were complete we moved on to the second set of models, the PIMs, which describe the navigation of the interface. Each presentation model is represented by a state in the PIM and transitions between states are labelled with I-behaviours (the UI navigational behaviours) exhibited by presentation models. PIMs are described using the $\mu$Charts language (Reeve 2005), which enables each part of the system to be modelled within a single, sequential $\mu$chart that can then be composed together or embedded in states of other models to build the complete model of the entire system. Figure 6 shows one of the PIMs representing part of the navigation of the “Find” dialogue and “Main” window.

In the simplest case, a system with five different windows would be described by a PIM with five states (each state representing the presentation model for one of the windows). However, this assumes that each of the windows is modal and does not interact with
any of the other windows. In the Digital Parrot system none of the dialogues are modal, in addition each of the windows can be minimised but continues to interact with other parts of the system while in its minimised state. This led to a complex PIM consisting of over 100 states. The complexity of the model and of the modelling process (which at times proved both challenging and confusing) gave us some indication of how users of the system might be similarly confused when interacting with the system in its many various states. Even before we derived the abstract tests, therefore, we began to consider areas of the system we would wish to include in our evaluation study (namely how the different windows interact).

The third stage of the modelling was to produce a formal specification of the functionality of the Digital Parrot. This was done using the Z specification language (ISO 2002) and again we completed as much of the specification as was possible but left some areas incomplete where we were not confident we completely understood all of the system’s behaviour. The Z specification consists of a description of the state of the system (which describes the data for the memory items stored in the system as sets of observations on that data) and operations that change that state. For example the “SelectItems” operation is described in the specification as:

\[ \text{SelectItems} \]

\[ \Delta \text{DPSystem} \]

\[ i? : \text{Item} \]

\[ l\text{i}?: \text{Item} \]

\[ \text{AllItems}' = \text{AllItems} \]

\[ \text{SelectedItems}' = \text{SelectedItems} \cup \{l\text{i}?\} \cup \{i?\} \]

\[ \text{li}.\text{itemName} = (i?.\text{itemLink}) \]

\[ \text{VisibleItems}' = \text{VisibleItems} \]

\[ \text{HiddenItems}' = \text{HiddenItems} \]

\[ \text{TrailItems}' = \text{TrailItems} \]

The meaning of this is that the operation consists of observations on the DPS system state before and after the operation takes place (denoted by \( \Delta \text{DPSystem} \)) and there are two inputs to the operation \( l\text{i}? \) and \( i? \), which are both of type \( \text{Item} \). After the operation has occurred some observations are unchanged. Observations marked with ‘’ indicate they are after the operation, so, for example \( \text{AllItems}' = \text{AllItems} \) indicates this observation has not changed as a result of the operation. The \( \text{SelectedItems} \) observation does change however, and after the operation this set is increased to include the inputs \( l\text{i}? \) and \( i? \), which represent the new items selected.

Once we had completed as much of the modelling as was possible we met with the software’s developer to firstly ensure that the behaviour we had described was correct, and secondly to fill in the gaps in the areas where the models were incomplete. With a complete set of models and a complete specification we were then able to relate the UI behaviour to the specified functionality by creating a relation between the S-behaviours of the presentation models (which relate to functionality of the system) and the operations of the specification. This gives a formal description of the S-behaviours by showing which operations of the specification they relate to, and the specification then gives the meaning. Similarly, the meaning of the I-behaviours is given by the PIM. The relation, which we call the presentation model relation (PMR), we derived is shown below:

\[ S_\text{_HighlightItems} \rightarrow \text{SelectItems} \]

\[ S_\text{_PointerMode} \leftrightarrow \text{TogglePointerMode} \]

\[ S_\text{_CurrentTrail} \rightarrow \text{SelectCurrentTrailItems} \]

\[ S_\text{_SelectItemMenu} \leftrightarrow \text{MenuChoice} \]

\[ S_\text{ZoomInTL} \leftrightarrow \text{SelectCurrentTrailItems} \]

\[ S_\text{ZoomOutTL} \leftrightarrow \text{TimelineZoomInSubset} \]

\[ S_\text{SelectItemsByTime} \leftrightarrow \text{SelectItemsByTime} \]

\[ S_\text{FitSelectionByTime} \leftrightarrow \text{FitItemsByTime} \]

\[ S_\text{HighlightItem} \leftrightarrow \text{SelectByName} \]

\[ S_\text{AddToTrail} \leftrightarrow \text{AddToTrail} \]

\[ S_\text{ZoomInMap} \leftrightarrow \text{RestrictByLocation} \]

\[ S_\text{ZoomOutMap} \leftrightarrow \text{RestrictByLocation} \]

\[ S_\text{Histogram} \leftrightarrow \text{UpdateHistogram} \]

This completed the modelling stage and we were now ready to move on to derivation of the abstract tests that we describe next.

4.4 The Abstract Tests

Abstract tests are based on the conditions that are required to hold in order to bring about the behaviour given in the models. The tool which we use for creating the presentation models and PIMs, called PIMed (PIMed 2009) has the ability to automatically generate a set of abstract tests from the models, but for this work we derived them manually using the process described in (Bowen & Reeves 2009). Tests are given in first-order logic. The informal, intended meaning of the predicates can initially be deduced from their names, and are subsequently formalised when the tests are instantiated. For example, two of the tests that were derived from the presentation model and PIM of the “Find” dialogue and “MainandFind” are:

\[ \text{State(MainFind)} \Rightarrow \text{Visible(FXIcon)} \land \text{Active(FXIcon)} \land \text{hasBehaviour(FXIcon, I_Main)} \]

\[ \text{State(MainFind)} \Rightarrow \text{Visible(HighlightButton)} \land \text{Active(HighlightButton)} \land \text{hasBehaviour(HighlightButton, S_HighlightItem)} \]

The first defines the condition that when the system is in the MainFind state a widget called FXIcon should be visible and available for interaction (active) and when interacted with should generate the interaction behaviour called I_Main, whilst the second requires that in the same state a widget called HighlightButton is similarly visible and available for interaction and generates the system behaviour S_HighlightItem. When we come to instantiate the
test we use the PIM to determine the meaning of the I-behaviours and the Z specification (via the PMR) to determine the meaning of the S-behaviours. The set of tests also includes conditions describing what it means for the UI to be in any named state so that this can similarly be tested. The full details of this are given in (Bowen & Reeves 2009) and are beyond the scope of this paper. Similar tests are described for each of the widgets of the models, *i.e.* for every widget in the model there is at least one corresponding test. The abstract tests consider both the required functionality within the UI (S-behaviours tests) as well as the navigational possibilities (I-behaviour tests).

As there are two different versions of the Digital Parrot software, one that has a graph view for the data and the other a list view, there were two slightly different sets of models. However, there was very little difference between the two models (as both versions of the software have almost identical functionality); there were in fact three behaviours found only in the list version of the software (all of which relate to the UI rather than underlying functionality) and one UI behaviour found only in the graph version. This gave rise to four abstract tests that were unique to the respective versions.

### 4.5 Deriving The Study

With the modelling complete and a full set of abstract tests, we now began to consider how we could use these to derive an evaluation study. To structure the study we first determined that all S-behaviours should be tested by way of a specific task (to ensure that the functional behaviour could be accessed successfully by users). The relation we showed at the end of section 4.3 lists thirteen separate S-behaviours. For each of these we created an outline of a user task, for example from the test:

\[
\text{State}(\text{MainFind}) \Rightarrow \\
\text{Visible} (\text{HighlightButton}) \land \\
\text{Active} (\text{HighlightButton}) \land \\
\text{hasBehaviour} (\text{HighlightButton}, \text{S.HighlightItem})
\]

we decided that there would be a user task in the study that would require interaction with the “Find” dialogue to utilise the S.HighlightItem behaviour. To determine what this behaviour is we refer to the relation between specification and S-behaviours, so in this case we are interested in the Z operation Select-ByName. We generalised this to the task:

*Use the “Find” dialogue to highlight a specified item.*

Once we had defined all of our tasks we made these more specific by specifying actual data items. In order to try and replicate conditions of the first study we used (where possible) the same examples. The “Find” task, for example, became the following in our study:

*Use the “Find” functionality to discover which items are connected to the item called “Scuba Diving”*

In order to complete this task the user will interact with the “Find” dialogue and use the S.HighlightItem behaviour that relates to the functionality in the system, Select-ByName, which subsequently highlights the item given in the “Find” text field as well as all items connected to it.

Once tasks had been defined for all of the S-behaviours we turned our attention to the I-behaviours. It would not be possible to check all navigational possibilities due to the size of the state space for the models of this system (and this may be true for many systems under test) as this would make the study too long. What we aimed to do instead was maximize the coverage of the PIM by following as many of the I-behaviour transitions as possible and so we ordered the S-behaviour tasks in such a way as to maximise this navigation. Having the PIM as a visualisation and formalisation of the navigational possibilities enables us to clearly identify exactly how much we are testing. We also wanted to ensure that the areas of concern we had identified as we built the models were tested by the users. As such, we added tasks that relied on the interaction of functions from multiple windows. For example we had a task in our study:

*What are the items linked to the trail “NZ CS Research Student Conferences” which took place in the North Island of New Zealand?*

Our final study consisted of eighteen tasks for the graph version of the software and seventeen for the list version (the additional task relating to one of the functions specific to the graph version). After each task we asked the user to rate the ease with which they were able to complete the task and provided the opportunity for them to give any comments they had about the task if they wished. Finally we created a post-task questionnaire for the participants that was aimed at recording their subjective feelings about the tasks, software and study.

### 4.6 Participants and Procedure

As with the first study, ethical consent to run the second study was sought, and given. We recruited our participants from the same target group, and in the same way, as for the initial study with the added criterion that no one who had participated in the initial study would be eligible to participate in the second.

We used a “between groups” methodology with the ten participants being randomly assigned either the Graph view version of the software or the List view. The study was run as an observational study with notes being taken of how participants completed, or attempted to complete, each task. We also recorded how easily we perceived they completed the tasks and subsequently compared this with the participants’ perceptions. Upon completion of the tasks the participants were asked to complete the questionnaire and provide any other feedback they had regarding any aspect of the study. Each study took, on average, an hour to complete.

### 4.7 Results

The second study found five functionality bugs and twenty seven usability issues. The bugs found were:

1. Continued use of “Zoom in” on map beyond maximum ability causes a graphics error.
2. The shortcut keys ‘i’ and ‘o’ for “Zoom in” and “Zoom out” on Map don’t work.
3. Timeline loses capability to display 2009 data once it has been interacted with.
4. Data items visualised in timeline move around between months.
5. Labels on the Map sometimes move around and sit on top of each other during zoom and move operations.

We had identified two of the functionality bugs during the modelling stage (the loss of 2009 data and the non-functional shortcut keys on the map) and in a traditional development/testing scenario we would expect that we would report (and fix) these prior to the evaluation. The usability issues ranged from minor items to major issues. An example of a minor issue was identifying that the widget used to toggle the mouse mode in the graph view is very small and in an unusual location (away from all other widgets) and is easily overlooked. We consider this minor because once a user knows the widget is there it is no longer a problem. An example of a major issue was the lack of feedback from the “Find” function that meant it was impossible to tell whether or not it had returned any results in many cases.

We made 27 recommendations for changes to the software. Some of these were specific changes that could be made to directly fix usability issues found, such as the recommendation:

“Inform users if “Find” returns no results.”

Whereas others were more general and would require further consideration, for example the recommendation:

“Reconsider the use of separate dialogues for features to reduce navigational and cognitive workload.”

Other observations led us to comment on particular areas of the software and interface without directly making recommendations on how these might be addressed. For example, we observed that most participants had difficulty understanding conceptually how “Trails” worked and what their meaning was. This led to difficulties with tasks involving “Trails” but also meant that participants used “Trails” when not required to do so as they found it had a particular effect on the data that they did not fully understand, but that enabled them to visualise information more easily. This is related to the fact that the amount of data and the way it is displayed and the use of highlighting were all problematic for some of the tasks.

We also determined from our comparison of the measure of ease with which tasks were completed against the participants’ perception that for many participants even when they successfully completed a task they were not confident that they had done so. For example we would record that a participant completed a task fairly easily as they took the minimum number of steps required and were successful, but the participant would record that they completed the task with difficulty. This was also evident from the behaviour of participants as they would often double check their result to be certain it was correct. This is linked to the overall lack of feedback and lack of visibility that was reported as part of our findings.

From the results of the second study we were confident that the study produced from the formal models was able to find both specific problems (such as functionality bugs) and usability problems, as well as identify general overall problems such as lack of user confidence.

5 Comparing The Two Studies

There was an obvious difference in the tasks of the two studies. In the first study users were given four tasks requiring several steps and were able to try and achieve this in any way they saw fit, whereas in the second study they were given twenty-seven tasks which were defined very specifically. This meant that the way in which participants interacted with the software whilst carrying out these tasks was very different. In the first study users were encouraged to interact with the software in the way that seemed most natural to them in order to complete the tasks, whereas in the second study they were given much clearer constraints on how they should carry out a particular task (for example: “Use the “Find” function to...” ) to ensure they interacted with specific parts of the system. While this meant that coverage of functionality and navigation was more tightly controlled in the second study (which is beneficial in ensuring as many problems and issues as possible are found) it also meant that it did not provide a clear picture of how users would actually interact with the software outside of the study environment and as such led to the reporting of problems that were in fact non-issues (such as the difficulties users had in interpreting the amount of data for a time period from the histogram).

One of the problems with the tasks of the initial study, however, was that by allowing users to interact in any way they chose particular parts of the system were hardly interacted with at all, which meant that several issues relating to the “Timeline” that were discovered during the second study were not evident in the first study due to the lack of interaction with this functionality.

The other effect of the way the tasks were structured was the subjective satisfaction measurements of the participants. The participants of the first study were more positive about their experience using the software than those of the second study. We feel that this is partly due their interactions and the fact that the first group had a better understanding of the software and how it might be used in a real setting than the second group did. However, there is also the possibility that the participants of the first study moderated their opinions because they knew that the researcher conducting the study was also the developer of the software (which is one of the concerns we were hoping to address with our work).

6 Reflections on Process and Outcomes

One of the things we have achieved by this experiment is an understanding of how formal models might be used to develop a framework for developing user evaluations. This work shows that a study produced in such a way is as good (and in some cases better) at discovering both usability and functional problems with software. It is also clear, however, that the type of study produced does not allow for analysis of utility and learnability from the perspective of a user encouraged to interact as they choose with software.

Some of the advantages of this approach are: the ability to clearly identify the scope of the study with respect to the navigational possibilities of the software-under-test (via the PIM); a framework to identify relevant user tasks (via the abstract tests); a mechanism to support creation of oracles for inputs/outputs to tasks (via the specification). This supports our initial goal of supporting development of evaluation studies by someone other than the software developer as it provides structured information to support this. However, it also leads to an artificial approach to interacting with the software and does not take into account the ability of participants to
learn through exploration and as such may discover usability issues which are unlikely to occur in a real-world use of the software as well as decrease subjective satisfaction of participants with the software.

7 Conclusion

It seems clear that there is no ‘one size fits all’ approach to developing evaluation studies, as the underlying goals and intentions must play a part in how the tasks are structured. However, it does appear that the use of formal models in the ways shown here can provide a structure for determining what those tasks should be and suggests ways of organising them to maximise interaction. Perhaps using both methods (traditional and formally based) is the best way forward. Certainly there are benefits to be found from taking the formal approach, and for developers with no expertise in developing evaluation studies this process may prove supportive and help them by providing a framework to work within. Similarly for formal practitioners who might otherwise consider usability testing and evaluation as too informal to be useful the formal structure might persuade them to reconsider and include this important step within their work. The benefits of a more traditional approach are the ability to tailor the study for discovery as well as evaluation, something the formally devised study in its current form was not good at all. Blending the two would be a valuable way forward so that we can use the formal models as a framework to devise structured and repeatable evaluations, and then extend or develop the study with a more human-centred approach that allows for the other benefits of evaluation that would otherwise be lost.

8 Future Work

We would like to take the joint approach described to develop a larger study. This would enable us to see how effective combining the methods might be, how well the approach scales up to larger software systems and studies, and where the difficulties lie in working in this manner. We have also been looking at reverse-engineering techniques and tools which could assist when working with existing or legacy systems and this work is ongoing.

9 Acknowledgments

Thanks to all participants of both studies.

References


Making 3D Work: A Classification of Visual Depth Cues, 3D Display Technologies and Their Applications

Mostafa Mehrabi, Edward M. Peek, Burkhard C. Wuensche, Christof Lutteroth
Graphics Group, Department of Computer Science
University of Auckland
mmeh012@aucklanduni.ac.nz, epee004@auckland.ac.nz, b.wuensche@auckland.ac.nz, lutteroth@cs.auckland.ac.nz

Abstract

3D display technologies improve perception and interaction with 3D scenes, and hence can make applications more effective and efficient. This is achieved by simulating depth cues used by the human visual system for 3D perception. The type of employed depth cues and the characteristics of a 3D display technology affect its usability for different applications. In this paper we review, analyze and categorize 3D display technologies and applications, with the goal of assisting application developers in selecting and exploiting the most suitable technology.

Our first contribution is a classification of depth cues that incorporates their strengths and limitations. These factors have not been considered in previous contributions, but they are important considerations when selecting depth cues for an application. The second contribution is a classification of display technologies that highlights their advantages and disadvantages, as well as their requirements. We also provide examples of suitable applications for each technology. This information helps system developers to select an appropriate display technology for their applications.

Keywords: classification, depth cues, stereo perception, 3D display technologies, applications of 3D display technologies

1 Introduction

The first attempts for creating 3D images started in the late 1880s aided by an increasing understanding of the human visual perception system. The realization that the visual system uses a number of depth cues to perceive and distinguish the distance of objects in their environment encouraged designers to use the same principles to trick the human brain into the illusion of a 3D picture or animation (Limbchar 1968).

Moreover, the realism 3D display techniques add to images dramatically improved research, education and practice in a diverse range of fields including molecular modelling, photogrammetry, medical imaging, remote surgery, pilot training, CAD and entertainment (McAllister 1993).

This success motivated researchers to develop new 3D display techniques with improved performance and for new application fields (Planar3D 2012). This process continues as more complex and more realistic display techniques are being researched (Favalora 2005).

Different 3D display technologies are suitable for different applications depending on their characteristics and the depth cues that they simulate (McAllister 1993, Okoshi 1976). Therefore, a developer must be familiar with these techniques in order to make an informed choice about which one to use for a specific application. Characterizing 3D display techniques in terms of which applications they are suited for is not easy as the information regarding their limitations, constraints and capabilities is much dispersed.

Earlier contributions (Pimenta and Santos 2010) have categorized depth cues and 3D display technologies; however there is no information provided about the significance of each of the depth cues, and the advantages, disadvantages and constraints of display techniques are not discussed. Furthermore, no guidelines are provided about which display technology is most suitable for a specific use-case.

In this paper, we address the following two research questions:

1. What are the limitations of the depth cues of the human visual system?
2. What applications is each 3D display technology suitable for?

To answer question 1, we have analysed the seminal references in that area (McAllister 1993, Okoshi 1976), in addition to references from art and psychology, and used them to build a new classification of depth cues. To answer question 2, we have analysed the most common display technologies (Planar3D 2012, Dzignlight Studios 2012) and the common characteristics of the applications they can be used for. The result is a classification of 3D display technologies in terms of their depth cues, advantages and disadvantages, and suitable application domains.

Section 2 describes the classification of depth cues. Section 3 describes the classification of display technologies. Section 4 establishes a link between the display technologies and the applications they are appropriate for. Section 5 concludes the paper.

2 Depth Cues

Depth cues are information from which the human brain perceives the third visual dimension (i.e. depth or
distance of objects). Each display technique simulates only some of the depth cues. Thus, evaluating their usability for a specific application requires knowing the importance of depth cues with respect to that application.

Visual depth cues can be classified into two major categories: physiological and psychological depth cues. Both are described in the following, and summarized in Table 1 (McAllister 1993, Okoshi 1976).

### 2.1 Physiological Depth Cues

The process of perceiving depth via physiological depth cues can be explained using physics and mathematics. That is, it is possible to calculate the depth of objects if the values for some of the important physiological depth cues are available (e.g. using triangulation for binocular parallax values). For this reason, phys. depth cues are used for applications that simulate human 3D perception, such as in robotics to estimate the distance of obstacles (Xiong and Shafer 1993, Mather 1996).

Physiological depth cues are either binocular (i.e. information from both the eyes is needed for perceiving depth) or monocular (i.e. information from only one eye is sufficient to perceive depth). In the following we describe different phys. depth cues.

**Accommodation.** The focal lengths of the lenses of the eyes change in order to focus on objects in different distances. This depth cue is normally used in combination with convergence as it is a weak depth cue. It can only provide accurate information about the distance of objects that are close to the viewer (Howard 2012).

**Convergence** is the angle by which our eyes converge when focusing on an object. This depth cue provides accurate information about the distance of objects. However, the convergence angle gets close to zero as an object moves further away, eliminating the cue for large distances (i.e. convergence angle is asymptotic to distance) (Howard 2012).

**Binocular Parallax.** Our eyes are positioned approximately 50-60mm away from each other (Oian 1997). Thus, they see images with slightly different perspectives. Two slightly different images are fused in the brain and provide 3D perception. Every 3D display system must simulate this depth cue, as it is the most important one (McAllister 1993).

**Monocular Movement (Motion) Parallax.** Objects that are further away in the scene appear to move slower than objects that are closer. This depth cue can consequently provide kinetic depth perception which is used by the brain to estimate the time to clash/contact (TTC) (McAllister 1993, Okoshi 1976, Mikkola et al. 2010).

**Depth from Defocus.** Our brain can estimate the depth or distance of objects by the blurring in the perceived image, where objects with different amount of blurring have different depths. The depth of field of an optic (e.g. an eye lens) is the distance to an object that stays clearly and sharply focused while the objects behind it are blurred (Mather 2006). The human brain uses it together with depth from focus (accommodation) to improve the results of the latter (Mather 1996, Mikkola et al. 2010). Some artificial vision systems, e.g. in robotics, use this cue alone to calculate depth (Xiong and Shafer 1993).

### 2.2 Psychological Depth Cues

All of the psychological depth cues are monocular. In the following we briefly describe all of them (McAllister 1993, Okoshi 1976, Howard 2012, Bardel 2001).

**Retinal Image Size.** If our brain is familiar with the actual size of an object, it can estimate its distance by considering its perceived size with respect to its actual known size.

**Linear Perspective.** In a perspective projection parallel lines appear closer as they move towards the horizon and finally converge at infinity. This depth cue is one of the most frequently used ones to express depth in computer graphics renderings.

**Texture Gradient.** Details of surface textures are clearer when the surface is close, and fade as the surface moves further away. Some psychologists classify linear perspective as a type of texture gradient (Bardel 2001, Mather 2006).

**Overlapping (Occlusion).** Our brain can perceive exact information about the distance order of objects, by recognizing objects that overlap or cover others as closer, and the ones that are overlapped as farther (Gillam and Borsting 1988).

**Aerial Perspective.** Very distant objects appear hazy and faded in the atmosphere. This happens as a result of small particles of water and dust in the air (O’Shea and Blackburn 1994).

**Shadowing and Shading.** Objects that cast shadow on other objects are generally perceived to be closer (shadowing). Moreover, objects that are closer to a light source have a brighter surface compared to those which are farther (shading). However, many psychologists do not consider this as a depth cue because shadows only specify the position of an object relative to the surface the shadow is cast on, and additional, more accurate estimations of distance are needed from other depth cues (e.g. texture gradient) (Bardel 2001).

**Colour.** Different wavelengths are refracted at different angles in the human eye. Thus, objects with different colours appear at different distances. Therefore, the results obtained from this depth cue are not reliable (McAllister 1993).

### 3 3D Display Technologies

3D display techniques are typically classified into two main categories: stereoscopic and real 3D. In the following we describe the most important technologies (McAllister 1993, Okoshi 1976); a summary can be found in Table 2.

#### 3.1 Stereoscopic Display

Stereoscopic techniques are mainly based on simulating binocular parallax by providing separate images for each of the eyes. The images depict the same scene from slightly different viewpoints. Stereoscopic displays are not considered as real 3D displays as users cannot find more information about the image by moving their head.
around. In other words, motion parallax is not simulated and the look around requirement is not satisfied.

However, in some of the new techniques motion parallax is simulated by adding a head tracking system (e.g. HCP). In all of the stereoscopic displays, convergence and accommodation are disconnected as viewers observe all the images from the same image plane (i.e. planar screen). These types of images are called virtual, and not everyone is able to perceive a 3D vision from them (Media College (2010) stated that 2-3% of the population are stereo blind). Stereoscopic displays are divided into two subclasses: stereo pair and autostereoscopic displays.

3.1.1 Stereo Pair

Stereo pair displays are based on blocking each eye from seeing the image corresponding to the other eye. This is usually achieved via glasses using various technologies. In some of the classic techniques, placing the pictures close to each lens prevents the other eye from seeing it.

In more efficient techniques, the right and left images are polarized and projected onto a single screen in order to provide for more than one viewer. Viewers wear polarized glasses that separate right and left images. All polarizing glasses darken the perceived image as they only let a fraction of the emitted light pass through. Stereo pair displays can be classified into two categories: non-polarized and polarized.

Non-Polarized Displays are described below:

- Side-by-Side. In this technique users wear stereoscopes as their glasses, and stereoscopic cards are placed close to the stereoscopes’ lenses, providing a different image to each eye. Although this is an old technique, it is still used in some schools for educational purposes (ASC Scientific 2011, Prospectors 2012).

- Transparency Viewers. This technique is an enhanced version of side-by-side. The images can be illuminated from behind, and therefore provide a wider field of view. These viewers are mostly used as toys (e.g. Fishpond Ltd. 2012).

- Head Mounted Displays. Each eye receives its own image via the magnifying lenses. The head tracking system has been added to this technique to enable motion parallax. HMDs are used for many AR applications. However, one of their drawbacks is their limited field of view (Fifth Dimension Technology 2011).

Polarized (Coded) Displays. There are two different ways of projecting left and right images onto the screen. Either both of the images are projected at the same time (time parallel), or sequentially (field sequential). Passive polarized glasses are worn for time parallel projection. In contrast, in field sequential projections active shutter glasses actively assign each image to its corresponding eye by blocking the opposite eye.

A disadvantage of active glasses is that they have to be synchronized with the screen every time the viewer attempts to use the display. Moreover it is not easy to switch between screens as glasses need re-synchronization. In both parallel and sequential projection, images must be projected with at least 120 Hz frequency to avoid image flicker. Polarized displays are described as following.

Anaglyph. In this technique images are polarized by superimposing additive light settings. On the viewers’ side coloured anaglyph glasses (normally red and green) take each image to its corresponding eye by cancelling the filter colour and reconstructing the complementary colours (Southern California Earthquake Centre, n.d.).

Some people complain from headaches or nausea after wearing anaglyph glasses for long time periods (ESimple, n.d.). Moreover, if glasses do not filter colours appropriately and part of an image is observed by the opposite eye, image ghosting occurs. Anaglyph photos are widely used for entertainment, educational and scientific applications (Joke et al. 2008, 3DStereo 2012).

Fish Tank Virtual Reality. This technique increases the immersion by adding a head tracking system to stereoscopic images. For this purpose a stereo technique (Li et al. (2012) use Anaglyph) is incorporated with a head tracking system to provide a cheap approach for higher immersion.

Li et al. demonstrate that the technique is reasonably efficient in providing realistic 3D perception, as it simulates three depth cues (retinal size, binocular parallax and motion parallax). Its low cost gives it a great potential as a replacement for more expensive techniques with similar functionalities (e.g. ImmersaDesk).

Vectographic Images. This technique includes printing polarized images that are formed by iodine ink on the opposite sides of a Vectograph sheet. It can provide excellent results, but creating an image requires time consuming photographic and dye transfer. Therefore it was quickly replaced by a new method called StereoJet. Vectographic images were used in the military to estimate the depth of an enemy’s facilities and by optometrists to test the depth perception of patients (especially children) (Evans et al. n.d.).

StereoJet. In this method fully colour ed polarized images are printed on Vectograph sheets with high quality (Friedhoff et al. 2010). StereoJet images are widely used in advertisements, entertainment, government and military imaging. The advantage of this technique is that the images are high quality and the projectors do not need to polarize the images as they are already polarized before being printed (StereoJetA 2012, StereoJetB 2012).

ChromaDepth. In this technique the colours used in the image depict the depth and the glasses are double prism-based. Therefore, the glasses impose different offsets on each specific wavelength and form the stereo pair images. Small regions of composite colours might be decomposed into their base colours and create some blurring regions that are called colour fringe. ChromaDepth images are used in amusement parks, and educational cases (ChromatekB, n.d.).

The advantage of this technique is that only one image is required. However, images cannot be arbitrarily coloured as the colour carries information about depth. In some stages of designing the ChromaDepth pictures, the adjustments have to be done manually while the
animators are wearing prism based glasses, which is a demanding job (Chromatek, n.d.).

Interference Filter Technology. In this technique the glasses can be adjusted to pass only one or more specific wave lengths and reflect the rest; therefore image ghosting is avoided. The glasses do not require non-depolarizing silver screens and are more durable and accurate compared to other polarized glasses.

The main advantage of these glasses is the selective wavelength filtering. However, this technique requires trained personnel to adjust the wavelengths of colours on the projectors; which increases costs (Baillard et al. 2006, Laser Component ltd. n.d.). This technique is used for analytic chemistry, physics, life science, engineering, communication, education and space science (SCHOTT 2008).

Fake Push Display. This technique is consisted of a stereo display box that is mounted on sensors with 6 DOF to simulate moving in the virtual environment. The display technique is normally used for laboratory research (e.g. molecular modelling).

Eclipse Method (Active Shutter Glasses). This method is based on field sequential image projection. It has been used in the gaming and entertainment industry for a long time. Recently other companies have experimented incorporating this technique into their products as well (e.g. Nintendo and Samsung smart phones). Although this method is popular, it becomes expensive when more than a few viewers use it. Moreover, active shutter glasses darken the image more than other polarizing glasses (Perron and Wolf 2008).

ImmersaDesk. In this technique a big screen projects polarized images and fills the fields of view for up to four people. ImmersaDesks are designed to have the same applicability of fully immersive CAVEs in addition to offering smaller dimensions and portability. Unlike fully immersive CAVEs, ImmersaDesks do not require synchronization between the images of multiple walls. The screen is tilted to allow user interaction with the floor as well. One of the limitations of ImmersaDesk is that it can only track the position of one viewer (DeFanti et al. 1999).

Fake Space System Display (CAVE). This is normally used for studying human reaction and interaction scenarios that are expensive or impossible to implement in the real world.

CAVEs require processing and synchronizing eight images (left and right images for three walls and the floor) in a high speed. Nearly seventy institutes are currently using sixty ImmersaDesks and forty CAVEs for their researches (Academic Computing Newsletter of Pennsylvania State University 2006).

3.1.2 Autostereoscopic

Autostereoscopic images do not need glasses to be worn. These techniques are described in the following section.

Autostereograms (FreeView). In this technique left and right images are encoded into a single image that appears as a combination of random dots. The viewer has to be positioned in front of the picture and move it back and forth. The right and left images are merged in the brain using transverse (crossed) or parallel (uncrossed) viewing. However, some viewers are not able to perceive 3D images from autostereograms. Autostereograms are used for steganography and entertainment books (Tsuda et al. 2008).

Holographic Stereogram. Images are stored on a holographic film shaped as a cylinder, and provide motion parallax as a viewer can see different perspectives of the same scene when moving around the cylinder (Halle 1988).

Holographic stereograms are normally used for clinical, educational, mathematical and engineering applications and in space exploration. The method has some constraints that limit its usage. For example, if viewers step further away from Holographic stereograms with short view distances the size of the image changes or distorts (Watson 1992, Halle 1994, ZebraImaging 2012).

Parallax Barrier. In this technique left and right images are divided into slices and placed in vertical slits. The viewers have to be positioned in front of the image so that the barrier conducts right and left images to their corresponding eyes (Pollack, n.d.).

Forming the images in a cylindrical or panoramic shape can provide motion parallax as viewers are able to see different perspectives by changing their position. However, the number of images that can be provided is limited, so horizontal movement beyond a certain point will cause image flipping (McAllister 1993).

Lenticular Sheets. Lenticular sheets consist of small semi cylindrical lenses that are called lenticils and conduct each of the right and left images to their corresponding eyes. Because its mechanism is based on refraction rather than occlusion, the resulting images look brighter (LenstarLenticular 2007).

Alternating Pairs (VISIDEP). This method is based on vertical parallax. Images are exposed to the viewer with a fast rocking motion to help viewers fuse them into 3D images. This method avoids image flicker and ghosting because of vertical parallax.

VISIDEP was used in computer generated terrain models and molecular models. However, not all the viewers were able to fuse the vertical parallax images into a 3D image. This method was limited in terms of implementation speed and quality of images, thus it is not in use anymore (Hodges 1985).

3.2 Real 3D Display

In real 3D displays, all of the depth cues are simulated and viewers can find extra information about the observed object by changing their position (this type of image is called solid). Real 3D displays can be classified in three main categories: Swept Volume Display, Static Volume Displays and Holographic 3D Displays. One motivation for creating real 3D displays is to enable the direct interaction between human and computer generated graphics thanks to finger gesture tracking systems (Favalora 2005).
3.2.1 Swept Volume Displays
In this method microscopic display surfaces such as mirrors or LCD displays sweep a specific volume with a very fast speed (900 rpm or 30Hz). Software applications are used to decompose a 3D object into small slices and processors compute which slices must be projected onto the display screen considering its position in the volume.

Because of visual persistence in the human brain, and the fast rotation of the display screen, the displayed points seem consistent in the volume; therefore a 3D illusion appears in the human brain. The projected lights have to decay very fast to avoid the appearance of stretched light beams (Matteo 2001). Swept volume displays can be classified as follows:

Oscillating Planar Mirror. In this method the microscopic mirror moves backward and forward on a track perpendicular to a CRT which projects the light beams (Favalora 2005).

Varifocal Mirror. In this method a flexible mirror which is anchored on its sides is connected to a woofer. The woofer changes the focal length of the mirror with a high frequency. Therefore the light beams projected on the mirror appear at different depths.

Rotating Mirror. In this method a double helix mirror or a LCD display rotates at the rate of 600 rpm and an RGB laser plots data onto its surface (Dowing et al. 1996).

3.2.2 Static Volume Display
This is a new area of research in which some projects are focused on intangible mediums that reflect light as the result of interaction with a specific frequency of infrared beams. Other projects investigate using a set of addressable elements that are transparent on their off state and emit light on their on state (Dowing et al. 1996).

Moreover, a volume space has been proposed in which fast infrared pulses that last only for a nanosecond, appear as consistent points. Therefore the display surface does not need to sweep the volume and is static (Stevens 2011, Hambling 2006).

3.2.3 Holographic Display
In Holographic Displays or Computer Generated Holography a holographic interference pattern of an object is collected and stored. Initial systems required a physical object, but recently algorithms were developed for enabling the use of computer simulated scenes, by calculating light wavefronts through complicated mathematical processes (e.g. Fourier Transform Methods) (Slinger et al. 2005).

4 Applications of 3D Display Technologies
3D applications exploit different display techniques depending on their requirements. We found that applications can be classified into eight key categories presented below. A classification of the most common display technologies and the application domains that they are most suitable for is found in Table 3.

Geospatial Studies. 3D display techniques are utilized for exploring digital elevation models (DEM) of terrains. Applications include monitoring coast erosion, predicting river levels, visual impact studies, and civil defence simulations, e.g. preparing for possible disasters such as tsunamis or tornados. Moreover, DEMs are used by the military for simulating and planning operations, and in astronomy for studying planet surfaces.

In geospatial studies, latitude, longitude and altitude of geographical points are factors of interest. In other words, only the surface of a terrain is studied and depth is the only information required to be added to normal 2D images. For this purpose, binocular parallax is simulated using anaglyph or passive polarized imaging (Li et al. 2005, Planar3D 2012)

Discovery of Energy Resources. Oil and gas drilling operations are very expensive. Therefore, seismic sensors are used to gather information from underground seismic explosions in order to prepare subterranean maps that can identify the accurate location of resources (Planar3D 2012). Unlike geospatial studies, this type of data needs to be inspected in a volumetric approach. This is because clusters of different information are mixed and form data clouds that need to be inspected manually to distinguish different features (CTECH 2012).

The Mining Visualization System (MVS) is an example of a non-stereo visualization of subterranean maps (CTECH 2012). It allows users to rotate the 3D-visualized graphs to gain exact information about the density of different substances in each point in addition to their x, y and depth coordinates. There are new applications that try to provide precise information about oil and gas reservoirs by rendering stereo 3D maps using simulated binocular parallax (Grinstein et al. 2001).

The provided information can be displayed via passive polarized techniques to preserve the brightness and the colour of the maps. For example, Fish Tank VR is a promising technology as it allows users to look around the stereoscopic map and calculate even more accurate estimations about where exactly drillings should be conducted (Planar3D 2012, Li et al. 2012).

Molecular Studies. Understanding the complex structure of biomolecules is the first step towards predicting their behaviour and treating disease. Crystallographers need to have a precise knowledge about the location of molecular constituents in order to understand their structure and functioning.

For this reason, molecular modelling has always been an application domain for 3D display technologies, and some techniques such as VISIDEP were specifically developed for this purpose. These types of applications require motion parallax and look around feature in addition to binocular parallax to enable a thorough inspection of molecular structures (Hodges 1985).

Therefore, 3D volumetric displays are the best option for molecular studies; however the volumetric applications are not practically usable yet, and normal stereo displays such as passive polarized and parallax barrier are used instead. Fish Tank VR has a potential for replacing the current stereo methods as it provides motion parallax (Pollack n.d., Planar3D 2012).

Production Design. Obtaining a realistic view of a design is essential for fully understanding a product and facilitating communication between different stakeholders.
such as designers, developers, sales people, managers and end users. Using a suitable display technique is critical in this field, as the quality of a presentation influences the success of product development and sales (Penna 1988).

For example, for interactive scenes such as videogames and driving and flight simulations, a 3D display with smooth and continuous vision is most suitable. Thus, an active polarizing system is preferred; however for demonstrating an interior design of a house, illumination and colour contrast must appear appealing. Therefore, a display technique with passive polarization, which better preserves the resolution of the image, is more appropriate.

Furthermore, demonstrating different parts of a design separately would provide a better understanding about the final product for the stakeholders and the end users. Therefore, using display techniques that allow inspecting the designed parts from different angles (such as volumetric displays, Fish Tank VR, ImmersaDesk) before the assembly stage can benefit all stakeholders (Planar3D 2012, Wickham 1987). Also, Fish Tank VR can be used for applications that require reasonable immersion as well as extended time periods (Planar3D 2012, Wickham 1987).

Most operations take long and require wearing glasses for polarized techniques are most popular for this purpose as depth, and performing more accurate operations. Passive polarized techniques are most popular for this purpose as most operations take long and require wearing glasses for extended time periods (Planar3D 2012, Wickham 1987).

**Medical Applications.** 3D display techniques (MRI, Ultrasound and Computer Tomography) have been used by radiologists, physiotherapists and physicians for a long time in order to gain a better understanding of patients’ conditions and to provide more accurate diagnosis and interventions. In addition, minimally invasive surgery (MIS) applications widely take advantage of stereo displays. MIS reduces the risk of complications and reduces recovery time by using small incisions (keyhole surgery). In MIS miniature cameras are slid through patients’ body to let surgeons monitor the process of an operation. Recently stereo 3D displays have been exploited to provide binocular parallax for helping surgeons with better recognition of body organs and their depth, and performing more accurate operations. Passive polarized techniques are most popular for this purpose as most operations take long and require wearing glasses for extended time periods (Planar3D 2012, Wickham 1987).

**Simulation and Training.** Many scenarios are impossible or expensive to simulate in the real world. For example, training novice pilots is very risky as small mistakes can have catastrophic consequences. Fully immersive display techniques are used to simulate these scenarios as realistically as possible (McAllister 1993, Planar3D 2012).

Cheaper stereo 3D displays (such as stereoscopes, StereoJet) are used for educational purposes in schools to increase the understanding rate in students by providing comprehensive 3D charts and diagrams where only binocular parallax is required (Watson 1992, ASC Scientific 2011).

**Entertainment.** The entertainment industry is one of the biggest users of 3D displays. The employed display technologies vary depending on requirements such as quality of colour and brightness, smoothness of animation, whether polarizing glasses are to be worn (if yes, how long for?), whether the display is for more than one viewer etc. (Dzignlight Studios 2012). For example, in the gaming industry smooth and continuous animation has the first priority and brightness can be compensated. Moreover, in movies wearing glasses for long time periods and brightness of the images must be taken into consideration, and the display technology should be reasonably cheap, so that it can be provided for large number of viewers (Penna 1988, Dzignlight Studios 2012).

In amusement parks such as haunted walkthroughs the combination of colours must provide the excitement and psychological impression that the images are supposed to impose on the viewers. Therefore, ChromaDepth images are used which are mainly formed by a combination of red, green and blue colours on a black background and the glasses are reasonably cheap (ChromaDepthA, n.d.).

**Informative Displays.** 3D display techniques are also used for better and more attractive public displays. Autostereoscopes have recently become popular in this application domain, as the information can be displayed in public to a large audience in a fast, affordable, and convenient way (e.g. advertisement billboards and posters) (Chantal et al. 2010).

Parallax barriers are used in airports security systems to provide a wider field of view for the security guards (BBC News 2004). In the vehicle industry new display screens use parallax barriers or lenticular sheets to direct different images to different people in the vehicle such that GPS information is provided for the driver while other passengers can watch a movie (Land Rover 2010).

Some of the new smartphones and digital cameras use parallax barriers for their screens to attract more consumers to their brands. For the same reason new business cards, advertisement brochures and posters use 3D display techniques such as lenticular sheets or anaglyph images (LenstarLenticular 2007, Dzignlight studios 2012).

### 5 Conclusion

In this paper we presented the following contributions:

- A classification of depth cues based on a comprehensive literature review, highlighting their strengths and limitations.
- A classification of 3D display technologies, including their advantages and shortcomings.
- A discussion of 3D application domains and guidelines about what 3D display technologies are suitable for them.

The classifications provide the information that a developer needs to make an informed choice about the appropriate 3D display system for their application. Based on constraints, limitations, advantages and costs of different display technologies, we have provided guidelines about the common characteristics of applications that utilize a specific 3D display technique.

As a future work we will develop benchmark scenarios that allow us to evaluate the suitability of different 3D display systems for common application domains experimentally. This would help to address the lack of quantitative guidelines in this area.
Proceedings of the Fourteenth Australasian User Interface Conference (AUIC2013), Adelaide, Australia

<table>
<thead>
<tr>
<th>Depth Cues</th>
<th>Strength</th>
<th>Range</th>
<th>Limitations</th>
<th>Static/Animated</th>
</tr>
</thead>
</table>
| Accommodation                       | Weak (McAllister 1993) | 0-2m (McAllister 1993) | 1. Not perceivable in a planar image (Mather 2006)  
2. Only works for less than 2 meters (Mather 2006) | S & A (McAllister 1993) |
| Convergence                         | Weak (McAllister 1993) | 0-10m (McAllister 1993) | 1. Not perceivable in a planar image (Mather 2006)  
2. Only works for less than 10 meters (Mather 2006)  
3. Convergence is tightly connected with Accommodation (Mather 2006) | S & A (McAllister 1993) |
| Binocular Parallax (Stereopsis)     | Strong (Kaufman et al. 2006) | 2.5-20m (Kaufman et al. 2006) | 1. The variations beyond 1.4 meters becomes smaller (Mather 2006) | S & A (McAllister 1993) |
| Monocular Movement (Motion) parallax| Strong (Ferris 1972) | 0-∞ (Mikkola et al. 2010) | 1. Any extra movement of the viewer or the scene create powerful and independent depth cues (Mather 2006)  
2. Does not work for static objects (McAllister 1993) | A (McAllister 1993) |
| Depth from Defocus                  | Strong for computer (Xiong and Shafer 1993) Weak for human (Mikkola et al. 2010) | 0-∞ (Mather 1996) | 1. Depth of field depends on the size of pupils as well. The estimated depth may be inaccurate (Mather 2006)  
2. Human eyes cannot detect small differences in a blurry scene (Mather 2006) | S (Mather 1996) |
| Retinal Image Size                  | Strong (Howard 2012) | 0-∞ (Bardel 2001) | 1. Retinal size change for distances over 2 meter is very small (Mather 2006) | S & A (McAllister 1993) |
| Linear Perspective                  | Strong (Bardel 2001) | 0-∞ (Bardel 2001) | 1. Works good for parallel or continuous lines that are stretched towards horizon (Mather 2006) | S & A (Mather 2006) |
| Texture Gradient                    | Strong (Howard 2012) | 0-∞ (Bardel 2001) | 1. Only reliable when the scene consists of elements of the same size, volume and shape. And texture cues vary slower for a taller viewer compared to a shorter (Mather 2006) | S & A (Mather 2006) |
| Overlapping                         | Strong (Bardel 2001) | 0-∞ (Bardel 2001) | 1. Does not provide accurate information about the depth. Only ordering of the objects (McAllister 1993) | S & A (McAllister 1993) |
| Aerial Perspective                  | Weak (TAL 2009) | Only long distance (Bardel 2001) | 1. Large-distance is required (Mather 2006)  
2. Provides unreliable information as it highly depends on weather, time of the day, pollution and season (TAL 2009) | S & A (Mather 2006) |
| Shadowing And Shading               | Weak (Bardel 2001) | 0-∞ (Bardel 2001) | 1. The perception depends on illumination factors (Bardel 2001) | S & A (McAllister 1993) |
| Colour                              | Weak (McAllister 1993) | 0-∞ (McAllister 1993) | 1. Objects at the same depth with different colour are perceived with different depths.  
2. Brighter objects appear to be closer (McAllister 1993) | S & A (McAllister 1993) |

Table 1: Table of Depth Cues

<table>
<thead>
<tr>
<th>Display Technique</th>
<th>Category</th>
<th>Physical Depth Cues Exploited</th>
<th>Hardware/Software Requirements And Prices</th>
</tr>
</thead>
</table>
| Side by side images | Stereo pair Non-polarized | Binocular Parallax | 1. Stereoscope (~ US$ 40) (ASC Scientist 2011)  
2. Stereographic cards (ASC Scientist 2011) |
2. Translucent films (Fishpond ltd. 2012) |
| Head Mounted Displays | Stereo pair Non-polarized | Binocular Parallax & Motion Parallax | 1. Helmet or pair of glasses (US$100-10,000) (TechCrunch 2011)  
2. Powerful processors with HDMI interfaces (TechCrunch 2011)  
3. Software (Vizard VR Toolkit) to render stereo graphics and process head tracking data (WorldViz, 2012) |
| Anaglyph | Stereo pair Time parallel Polarized | Binocular Parallax | 1. Anaglyph glasses (less than $1.0) (Southern California Earthquake Centre, n.d.)  
2. Anaglyph photos software programs such as OpenGL, Photoshop. Z-Anaglyph (Rossett 2007) |
| Fish Tank VR | Stereo pair Time parallel Polarized | Binocular Parallax & Motion Parallax | 1. A pair of cheap passive glasses (Anaglyph) (Li et al. 2012)  
2. Head Tracking system using home webcams (~ $30) (Li et al. 2012) |
| Vectographs | Stereo pair Time parallel Polarized | Binocular Parallax | 1. Vectograph sheets in the rolls of two-thousand feet length for ~US$ 37,000 (Friedhoff et al. 2010) |
| Stereojet | Stereo pair Time parallel Polarized | Binocular Parallax | 1. Vectorgraph sheets(Friedhoff et al. 2010)  
2. Stereojet printers such as Epson 3000 inkjet with four cartridges of Cyan, Magenta, Yellow and Black. Stereojet inks are ~US$ 50 for each cartridge (StereojetA 2012) |
| ChromaDepth | Stereo pair Time parallel Polarized | Binocular Parallax | 1. Double prism-based glasses (CIDTM, ChromatekB, n.d.)  
2. ChromaDepth image design applications. Micromedia Shockwave Flash 3.0 is specific for web based ChromaDepth animations (ChromatekA, n.d.) |
| Fake Push Displays | Stereo pair Time parallel Non-polarized | Binocular Parallax & Motion Parallax | 1. A box shaped binocular mounted on sensors to simulate movement in the virtual world (depending on their degrees of freedom their prices vary from US$ 10,000 to US$ 85,000) (McAllister 1993) |
| Eclipse Method (Active Shutter System) | Stereo pair Field-sequential Polarized | Binocular Parallax | 1. Stereo sync output (Z-Screen by Stereograpics Ltd.) (McAllister 1993)  
2. Normal PCs can use an emitter to enhance their screen update frequency and a software program to convert left and right images into an appropriate format for normal displays. The price for emitter is approximately US$ 400 |


Table 2: Table of 3D Display Technologies

<table>
<thead>
<tr>
<th>Display Technique</th>
<th>Main Characteristics of The Display Technique</th>
<th>Common Characteristics of Applications Utilizing the Display Technique</th>
<th>Application Examples</th>
<th>References</th>
</tr>
</thead>
</table>
### Table 3: Table of Common 3D Display Technologies and Their Applications

<table>
<thead>
<tr>
<th>Passive Circular Polarized</th>
<th>Fully Immersive CAVE</th>
<th>Volumetric Display</th>
<th>Autostereoscopic Displays</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-Cheap</td>
<td>1-Requires wearing shutter glasses</td>
<td>1-Simulates all depth cues</td>
<td>1-Does not require wearing glasses</td>
</tr>
<tr>
<td>2-Less possibility for image cross talk as the result of tilting viewer’s head (while in Linear Polarized is highly possible)</td>
<td>2-Requires gloves for interacting with virtual environment</td>
<td>2-Provides all view perspectives of an object</td>
<td>2-Can direct different images to different positions</td>
</tr>
<tr>
<td>3-Provides continuous image without flicker</td>
<td>3-Simulates binocular parallax and motion parallax</td>
<td>3-The graphical object is a bit transparent</td>
<td>3-Reduces resolution and brightness (Parallax Barrier)</td>
</tr>
<tr>
<td>4-Requires polarized projectors</td>
<td>4-Provides head tracking system</td>
<td>4-Do not require wearing glasses</td>
<td>5-Image cross talk is highly possible</td>
</tr>
<tr>
<td>5-Cross talk possible (especially in Linear Perspective)</td>
<td>5-Fully immersive</td>
<td>2-Can compensate on image resolution</td>
<td>5-Image flipping may occur</td>
</tr>
<tr>
<td>6-Darkens the images</td>
<td>6-Expensive (in terms of graphic rendering and processing tasks as well as price)</td>
<td>3-May require providing different images for viewers at different positions</td>
<td>6-Image may occur</td>
</tr>
<tr>
<td>7-Requires non-depolarizing silver screens</td>
<td>5-Can provide panoramic view</td>
<td>4-Limited budget</td>
<td>7-Cheap</td>
</tr>
</tbody>
</table>

1-For more than one viewer | 1-Study circumstances that are impossible or expensive to implement in real world (serious games) | 1-Detailed investigation of complex structures | 1-Do not require wearing glasses |
2-Colour and illumination is important | 1-Flight simulation | 2-Provides accurate information about three dimensional structure of objects | 2-Can compensate on image resolution |
3-Require wearing glasses for long time periods | 2-Pilot training | 3-Resolution of colour is not important | 3-May require providing different images for viewers at different positions |

- Most popular for movies
- Geographical and astronomical studies
- Government/Military information and security photography
- Interior and exterior designs
- Mechanical designs
- Oil and Gas discovery
- Recent medical images
- Molecular modeling and crystallography
- Minimally invasive surgery
- Radiology
- Eye surgery and ophthalmology
- Amusement parks
- Educational applications

1-Security systems of airports
2-Display screen of vehicles (show GPS data to driver and movie to passengers)
3-Display screen of smart phones
4-Business cards
5-Pool cards
6-Decoration ornament
7-Panoramic images
8-Molecular modeling
9-Educational applications
10-Advertisements

References


s=PM-TECH


Evans, H., Buckland, G., Lefer, D., They Made America: From the Steam Engine to the Search Engine: Two centuries of innovators


Favalora, E. (2005): volumetric 3D displays and applications


FilmJournal, 2009, The best is yet to come: 3D technology continues to evolve and win audience approval, accessed on 21st August 2012, http://www.filmjournal.com/filmjournal/content_display/news-and-features/technology/e38f82b8a31928f66a5f6ae6cc0db536ec?pn=2


Grinstein, G., Trutschi, M., Cvek, U. (2001): High Dimensional Visualizations, Institute of Visualization and Perception, University of Massachusetts


Horii, A. Depth from Defocusing, Computational Vision and Active Perception Laboratory (CVAP), Royal Institute of Technology, Stockholm, Sweden,


Limbchar, J. (1968): Four Aspects of The Film

Li, I. Pek, E., Wunsche, B., Latteroth, C. (2012): Enhancing 3D Applications Using Stereoscopic 3D and Motion Parallax, Graphics Group, Department of Computer Science, University of Auckland, New Zealand


Pimenta, W., Santos, L. (2010): A Comprehensive Classification for Three-Dimensional Displays, WSCG


Pollack, J. 3D Display Technology from Stereoscopes to Autostereo Displays, VP Display Business Unit, Sharp Microelectronic of Americas.


Quan, N. (1997): Binocular Disparity and the Perception of Depth, Neuron


Southern California Earthquake Centre, 2008, The Leading Technology For Stereo Images - technology


An Investigation of Usability Issues in AJAX based Web Sites

Chris Pilgrim
Faculty of Information and Communication Technologies
Swinburne University of Technology
Hawthorn, Vic, Australia

cpilgrim@swin.edu.au

Abstract
Ajax, as one of the technological pillars of Web 2.0, has revolutionized the way that users access content and interact with each other on the Web. Unfortunately, many developers appear to be inspired by what is technologically possible through Ajax disregarding good design practice and fundamental usability theories. The key usability challenges of Ajax have been noted in the research literature with some technical solutions and design advice available on developer forums. What is unclear is how commercial Ajax developers respond to these issues. This paper presents the results of an empirical study of four commercial web sites that utilize Ajax technologies. The study investigated two usability issues in Ajax with the results contrasted in relation to the general usability principles of consistency, learnability and feedback. The results of the study found inconsistencies in how the sites managed the usability issues and demonstrated that combinations of the issues have a detrimental effect on user performance and satisfaction. The findings also suggest that developers may not be consistently responding to the available advice and guidelines. The paper concludes with several recommendations for Ajax developers to improve the usability of their Web applications.

Keywords: Ajax, usability, world-wide web.

1 Introduction
The World Wide Web has evolved in both size and uses well beyond the initial conceptions of its creators. The rapid growth in the size of the Web has driven the need for innovation in interface technologies to support users in navigating and interacting with the increasing amount of diverse and rich information. The technological innovations over the past decade have strove to provide users with a supportive interface to access information on the Web with ease including new Web 2.0 models of interaction that allow users to interact, contribute, collaborate and communicate. However, with this innovation there has been an unintended consequence of increased and additional complexity for users. The new models of interaction have in some cases required a shift in the paradigm of how users expect the Web to behave. Users have had to change their view of the Web, from that as a vehicle for viewing content, to a view where the Web becomes a platform by which applications and services are delivered (Dix & Cowen, 2007). This paradigm shift breaks one of the fundamental principles of the architecture of the World Wide Web as having the unit of the ‘page’ (Berners-Lee, 1989).

One of the underlying technologies behind this evolution is Ajax. Ajax is now regarded as one of the technological pillars of Web 2.0 (Ankolekar et al, 2007) by providing the basis on which the Web can be regarded as a ‘platform’ for the delivery of services and applications that promote “openness, community and interaction” (Millard & Ross, 2006). Whilst the world has benefited from the evolution of the size and uses of the Web, the rush to embrace innovation has resulted in many developers to overlook well-established principles of good design and usability (Nielsen, 2007).

Ajax has several usability issues that have been reported in published research. In response to these issues some developer forums have provided design guidelines and technical solutions that Ajax developers could employ to alleviate any undesirable usability effects in their Web applications. What is unclear is whether commercial Ajax developers respond to these issues. This paper presents the findings of an empirical investigation into a set of Ajax enabled commercial websites to determine how web developers are responding to these usability challenges. The first section of the paper discusses the features and benefits of Ajax technologies. Some general heuristics for usable computer systems are presented and the specific usability challenges of Ajax are discussed. The methodology and results of the study are presented with the final section presenting a discussion and several recommendations for developers.

2 AJAX – Features and Benefits
The term Ajax has been attributed to Garret (2005) who coined it as an acronym for “Asynchronous Javascript and XML”. In his essay Garrett described the five key characteristics of Ajax based applications as:

- a user interface constructed with open standards such as the Dynamic Hypertext Markup Language and Cascading Stylesheets (CSS);
- a dynamic, interactive user experience enabled by the Document Object Model (DOM);
displayed through the browser. The engine can make additional requests to interact with the data without having to communicate with the server. The reply results in a full-page refresh of the browser in order to display the outcome. Where Ajax is unique is that the request that is sent to the server is asynchronous allowing the user to continue to interact with the web page whilst the request is being processed. This sequence of events differs from the classical model of the Web where a user action such as a mouse click triggers a HTML request containing the URI of a desired resource (usually a web page) to a web server. The web server accepts the request, in some cases does some server side processing and data retrieval, and then returns a complete HTML page back to the client to be displayed through the browser.

The classical model of the Web implements a turn-taking protocol in which users must wait for the reply after they submit a request. The reply results in a full-page refresh of the browser in order to display the result of the request. The model is simple, well understood and effective. Nielsen notes “Users are in control of their own user experience and thus focus on your content” (Nielsen, 2007).

Ajax eliminates any delays caused by turn-taking by introducing an Ajax engine into the browser between the user and the Web server. Results of requests, e.g. XML data, can be loaded into the Ajax engine allowing the user to interact with the data without having to communicate with the server. The engine can make additional requests to the server for new data or interface code asynchronously without blocking the user from continuing to interact with the web page. In particular, the JavaScript routine running in the browser can update only the necessary components of pages that require updating without a full-page refresh (Zucker, 2007; Kluge et al. 2007).

Ajax addresses the limitations in the classical model of the Web in two ways:

1. Enhancing response rates through data buffering. Ajax supports predictive downloading that allows data to be requested and buffered before the user needs it. Preloading of data is based on likely user actions in the context of the current page status. For example, Google Maps will automatically preload the regions of the map adjacent to the current views enabling the user to pan the map without any pause occurring as new sections are downloaded (Zucker, 2007). Another common use of data buffering is to support dynamic filtering allowing users to interact with multiple form options without the need for continuous page refreshes.

2. Enhanced user interactivity through asynchronous communication. The capacity of Ajax to update only relevant components of Web pages provides developers with the ability to create new interaction models. For example, Gmail uses Ajax to enable a new email message to be displayed in the interface when it is received without the need for the whole page to be updated. This feature enables Gmail to appear to the user to be acting more like a desktop application than a Web interface (Zucker, 2007). In addition, Ajax enables developers to present to users a range of innovative and engaging widgets and screen components that surpass the traditional controls available through HTML such as checkboxes, radio buttons, form fields and buttons.

Data buffering and asynchronous communications facilitate innovative Web applications that can be designed to be substantially more fluid than before resulting in less interruptions to the workflow of the user (Kluge et al. 2007). Oulasvirta and Salovaara (2004) suggest that interfaces should be ‘invisible’ without interruptions that cause switching of attention from a task as these can hamper memory and higher-level thought processes involving heavy load for working memory, for example when solving novel problems. A well designed Ajax-based Web application that avoids pauses, delays and interruptions may be able to provide the optimal experience of ‘flow’ that can result in engagement, enjoyment and increased productivity (Chen et al. 2000).

3 AJAX - Challenges

3.1 Usability Principles and Disorientation

Nielsen (2005) has suggested ten general heuristics for user interface design such as ‘visibility of system status’, ‘match between the system and real world’, ‘consistency and standards’ and ‘recognition rather than recall’. Nielsen (1993) also recommends five usability attributes that include ‘learnability’, ‘efficiency of use’ and ‘memorability’. Other research has produced similar sets of principles for the design of usable computer systems such as Dix et al. (1997) who suggested key attributes including ‘learnability’, ‘flexibility’ and ‘robustness’, and Shneiderman and Plaisant (2005) who proposed heuristics
including: ‘strive for consistency’, ‘offer informative feedback’ and ‘reduce short-term memory load’.

The general principles of consistency, learnability and feedback are common themes that are relevant when considering the usability of commercial Ajax-based Web applications.

Consistency: Cognitive psychologists suggest that as we interact with the world our mind constructs mental models of how things work (Johnson-Laird, 1983). Mental models may be used to anticipate events, to reason, and to explain the world. Interfaces that align with a user’s mental model will support their predictive and explanatory abilities for understanding how to interact with the system (Norman, 1988). Conflicts between the user’s mental model of a system and the reality of how a system behaves can result in disorientation and/or cognitive overhead. We would expect that the classical page-based model with the turn-taking protocol has become entrenched as part of a user’s mental model of the Web.

Learnability: A basic principle of Human Computer Interaction (HCI) is that user interfaces should be easy to use and predictable (Shneiderman and Plaisant, 2005). This is particularly important for commercial web-sites as we know that in general, Web users are impatient, require instant gratification and will leave a site if they cannot immediately figure out how to use it (Nielsen, 2000).

Feedback: Norman’s theory of affordance (1988) tells us that an interface should provide inherent clues to what actions are possible at any moment, the results of actions and the current state of the system so that users will know what to do instinctively. The success of the enhanced interactivity enabled through Ajax relies on the designer’s ability to provide appropriate feedback on the status of the Web application at all times.

A lack of consistency, learnability and feedback can result in disorientation and cognitive overhead in the users of Web applications. Conklin (1987) described disorientation as “the tendency to lose one’s sense of location and direction in a non-linear document” and cognitive overhead as “the additional effort and concentration necessary to maintain several tasks or trails at one time”. Disorientation and cognitive overhead are issues that have been thoroughly investigated in traditional hypertext systems.

3.2 AJAX Usability

Ajax can bring many benefits to the usability of the web applications by making the user interface more interactive and responsive. However, use of Ajax techniques has some challenges for achieving and/or maintaining usability. Nielsen (2007) notes that many Ajax-enabled Web 2.0 sites are “neglecting some of the principles of good design and usability established over the last decade”.

The page-based model of the Web is well entrenched as it provides the user’s view of the information on the screen, the unit of navigation (what you get when you click), and a discrete address for each view (the URL). The user’s mental model of how the Web operates has created a strong expectation that each interaction will result in a brief delay followed by a full refresh of the entire page. The simplicity of the original page-based model of the Web contributes to its ease of use and its rapid uptake (Nielsen, 2005).

Ajax “shatters the metaphor of the web page” (Mesbah & van Deursen, 2009). With Ajax, the user’s view is determined by a sequence of navigation actions rather than a single navigation action (Nielsen, 2005). The asynchronous client-server communication in Ajax may result in surprises for users as updates to a web page may occur without any user interaction. Users may also be surprised as individual sections or components of web pages are updated without a full-page refresh or without any user interaction. New innovative controls and widgets might appear on web pages providing features or functionality not normally found on web sites and without any clues to their operation. Finally, the user may find that particular features within the browser might not respond as expected such as the back button, forward button, history list, bookmarks, saving, printing and search.

The focus of this investigation was on two particular usability issues relating to Ajax implementations: inconsistencies in the operation of the Back button and the management of updates to web pages.

3.2.1 Issue 1: Back Button

There has been a substantial amount of empirical research that has investigated the use of the browser’s ‘Back’ button and the page revisitation behaviour of users. For example, studies that used client-side logging of user actions when using the Web found that dominant navigation choices were embedded links (52%) followed by the Back button (41%) (Catledge & Pitkow, 1995), and that the Back button was used for approximately 40% of user actions (Tauscher & Greenberg, 1996).

The major paradigm challenge for Ajax technologies is the unpredictable behaviour of the Back button on the browser. Since an Ajax application resides in a single page, there is sometimes no page to return to, or no page history to navigate resulting in unexpected outcomes for users (Rosenberg, 2007). Nielsen (2005) in his article entitled ‘Ajax Sucks’ noted that “the Back feature is an absolutely essential safety net the gives uses the confidence to navigate freely in the knowledge that they can always get back to firm ground. Breaking it is a usability catastrophe”.

The lack of state information resulting from asynchronous data exchange in Ajax applications also affects the user’s ability to use the Forward button and history list with confidence. Similarly, the outcomes of bookmarking a page from an Ajax application can be inconsistent with users expecting a bookmark to return a particular page status however frequently the bookmark will only return the initial screen of the Ajax application (Kluge et al, 2007).

There are several technical solutions to overcoming the Back button issue. For example, Google Maps artificially inserts entries into the history list so that when the Back button is clicked, the application state is reverted to simulate the expected ‘back’ behaviour. However there appears to be no generally accepted solution to this issue.
3.2.2 Issue 2: Update Management

One of the most powerful features of Ajax is the ability to implement functionality that causes a particular component or section of a web page to be updated rather than a full-page refresh. These ‘part-page’ updates can be implemented to occur either asynchronously without any user action such as receipt of an email in Gmail, or in response to a user interaction such as a mouse click. There are two related usability issues that can result from part-page updates.

The first issue is linked to the user’s awareness of an update occurring. Full-page refreshes in classical page-based interactions usually result in the browser displaying a visual indicator that informs the user that processing is occurring and that a new page is loading. For example, Internet Explorer has a solid circle that spins to indicate that processing is taking place, Firefox displays small spinning circles with different colours and Google Chrome has a half-circle that spins. Ajax applications cannot utilise the standard browser-based loading indicators. The default in Ajax is that no indicator is provided which can result in usability problems with Tonkin claiming “without explicit visual clues, users are unlikely to realize that the content of a page is being modified dynamically” (Tonkin, 2006). It therefore falls to developers to implement visual clues into their Ajax to inform the user that processing is occurring and also when the update has completed. Rosenberg (2007), reporting on the redesign of Yahoo Mail noted that there are no real standards for progress indicators in Ajax. Likewise, there is no standard approach to inform the user that the update has completed. Practices appear to range from sites that simply stop the loading indicator when the Ajax processing is completed whilst others display an ‘Update Completed’ or similar message. The potential for inconsistencies in how Ajax updates are reported to users could result in user disorientation.

The second issue relates to the user’s awareness of the nature and/or location of the actual change that has occurred on the page after a part-page update. Nielsen (2007) notes “users often overlooked modest changes such as when they added something to a shopping cart and it updated only a small area in a corner of the screen”. This effect is linked to a psychological phenomena called ‘change blindness’ where humans might not notice visual changes, even when these are “large, anticipated, and repeatedly made” (Resink, 2002). This effect has also been referred to as ‘attentional gambling’ where there is some uncertainty regarding where a user’s attention should be focused (Hudson, 2007). Once again, the potential for users to overlook changes as a result of part-page updates could result in usability problems.

4 Experiment

The usability challenges of Ajax described in the previous section have been documented in the research literature with some developer forums containing various technical solutions that could be employed to alleviate any undesirable usability effects in their Ajax applications. What is unclear is how commercial Ajax developers have actually responded to these issues or if Ajax enabled web sites continue to exhibit undesirable behaviours that might result in user disorientation and cognitive overhead.

An empirical investigation into the usability of commercial Ajax-based web applications was undertaken to examine the impact of these usability issues. The specific issues to be investigated included the consistency of the operation of the Back button and whether the management of part-page updates affected the user’s experience.

5 Method

Twenty students and staff from Swinburne University of Technology (6 female and 14 male) participated. Their age groups varied from 18 to 50 years. Participants were recruited from all academic disciplines using notice board advertisements. Participants were tested individually in a specialist usability laboratory and were paid a small fee for their time. Ethics approval had been received prior to conducting the study.

A repeated-measures design was used in which participants each completed two tasks on each of four commercial web sites that employed Ajax-based web technologies. The sites selected for the study were four popular hotel booking sites that incorporated various aspects of Ajax including part-page refreshes and innovative user controls. The sites were coded as O for orbitz.com, T for tripadvisor.com, K for kayak.com and H and hotels.com. The participants were provided with written instructions describing the tasks to be performed with the order in which the sites were presented to the participants being counterbalanced. The tasks involved finding a list of suitable hotels that were available in a particular city, on a particular date, in a particular neighbourhood, with a ranking of 4 or 5 stars and containing a restaurant. The participants were instructed to find hotels in two different cities, Paris and then London. Participants were encouraged to speak aloud as they completed the tasks. The participant’s actions and comments were captured using Morae Recorder and were analysed through Morae Manager to establish search times and other patterns of use. The participants completed a System Usability Scale (Brooke, 1996) after using each hotel booking site to assess views on learnability, design and overall satisfaction when using each site.

6 Results

6.1 Task Completion Times

Task completion times were operationalised as the time taken from when the initial list of hotels were displayed after the participant had selected the desired city and dates, until the resultant list of hotels were displayed. This approach measured the time taken for the participant to apply filters to the star rating, neighbourhood and restaurant settings. The outcome of changes to each filter control caused a part-page refresh of the hotel list.

Figure 1 presents the separate task completion times for each major city with the total time being the sum of the task completion times for both city tasks.

A visual examination of Figure 1 suggests that the overall task completion times on Site K and Site O were
shorter than the times for Site H and Site T. In addition, the results show that the completion times for the initial task (Paris) for each site was greater than the time to complete the task for the second task (London). This is expected, as we know that users generally perform better once they have gained an initial familiarity with a system. This is particularly evident in Site H.

A set of one-way repeated measures ANOVAs were conducted to compare the time to complete the total booking time for both tasks for each test site as well as each booking task separately. There was a significant effect for total booking time, Wilks’ Lambda = .29, F (2, 19) = 13.78, p<.0005. A pairwise comparison found that the total booking time for Site H was significantly different from Site K (M= 39.00, p<.05) and Site O (M= 30.50, p<0.5). The analysis also found that Site T was significantly different from Site K (M= 56.25.0, p<.05) and Site O (M= 47.75, p<0.5).

There was also a significant effect for the completion time of the initial Paris task, Wilks’ Lambda = .37, F (2, 19) = 9.87, p<.001. A pairwise comparison found that the Paris task for Site H was significantly different from Site K (M= 38.80, p<.05) and Site O (M= 30.80, p<0.5). The analysis also found that Site T was significantly different from Site K (M= 32.85.0, p<.05) and Site O (M= 24.85, p<0.5).

This analysis suggests that Site K and Site O provide better support to users in completing timely bookings in comparison to Site H and Site T. This difference is particularly significant when considering only the initial task (Paris) suggesting that Site K and Site O provide a more supportive experience for users interacting with the site for the first time.

6.2 System Usability Scale

Figure 2 shows the overall results of the System Usability Scale (SUS) indicating a high level of overall satisfaction with Site K and a lower level of satisfaction for Site T. Wilks’ Lambda = .09, F (2, 9) = 24.87, p<.0005. A pairwise comparison found that the SUS scores for Site K were significantly different from Site T (M= 12.00, p<.05).

These results are consistent with the analysis of the completion times that found that Site T provided the least amount of support for users when completing bookings. There was also a significant preference for Site K that yielded the shortest task completion times.

6.3 Back Button Use

There were five steps required to complete each booking task. The steps required filters to be set for the selection of the city and dates, neighbourhood, amenities, rating and then choice of hotel. Participants generally used the browser’s Back button to return to the site’s home page after viewing the list of matching hotels. The Morae Recorder captured the participant’s actions and their verbal comments when navigating back to the home page. Each web site provided a different user experience when the Back button was clicked.

Site H: The Back button performed predictably with each click stepping back through each previous status of
the Ajax application essentially undoing each search criteria in the reverse order that they were applied. To the user, this appeared to be stepping back one ‘page’ at a time and hence performed according to the classical model of the Web.

Site K: The Back button performed unpredictably. In most instances each Back click had no effect with the majority of the participants giving up and either clicking on the site’s logo to return to the home page or re-entering the URL. P15 clicked Back 13 times finally giving up and stating “Basically the Back button in the browser does not work”. The function of the Back button did not perform as expected and confused many participants.

Site T: The Back button was somewhat predictable. Each click appeared to go back one ‘page’ eventually returning to the home page, however there was no change to the filter settings after each click that was noted by several participants. 8 participants abandoned the use of the Back button after 3 or 4 clicks and clicking on the logo instead, possibly due to the lack obvious change in the page after each click. The level of feedback on each click was clearly not consistent with the classic web model.

Site O: The Back button performed relatively predictably. The majority of the participants found that when they clicked the Back button the search results page was displayed with all search criteria removed. When they click Back a second time the home page was displayed. For 7 participants there was no function for the Back button at all. Some of the participants noted that the first click removed all the search criteria, P6 stating “it should inform the user that when you click Back everything will be cleared”, whilst P18 stated “If you went back to a previous page then you would have to remember all the criteria you put in or re-select all the criteria as it is all lost”. Some participants expressed surprise that the second click of the Back button returned immediately to the home page.

6.4 Update Management
Each hotel booking site provided different approaches to: (i) indicating the request is being processed, and (ii) indicating the request is complete with a part-page refresh of the component of the page containing the list of matching hotels.

Site H: The relevant filter control is highlighted with a yellow background and is adjacent to a small spinning circle similar to the Internet Explorer processing indicator. After the part-page refresh the vertical scroll position of the page was reset to the top of the window clearly indicating that the refresh had concluded.

Site K: A pop-up box appears stating “Updating results - Filtering by…..” noting the particular criteria, e.g. star ratings or amenities. The box disappears when the refresh of the list is complete with the page remaining at the same vertical scroll position.

Site T: A pop-up box appears stating “Updating your results…”. The box disappears when the refresh of the list is complete with the page remaining at the same vertical scroll position.

Site O: A pop-up box appears stating “Loading your results”. A rotating circle similar to the Firefox processing indicator is displayed in the box. The section of the page that is being updated fades with a white transparent overlay. The box and fade disappear when the refresh of the list is complete.

An analysis of the Morae recordings was conducted focusing on the participant’s timings and comments whilst filters were being processed. The analysis examined in particular the participant’s reactions to the processing indicator for each site and their responses when processing completed. Two usability issues emerged.

The first issue related to the visibility of the processing indicator. Site H highlighted the relevant filter control with a yellow background with a small spinning circle. This method of indicating processing was visually less obvious in comparison to the methods used in other sites and as a result it became apparent that many participants did not notice the indicator. P18 stated “Doesn’t really tell you that it has done the selection criteria” whilst P20 noted “This site appeared to be reloading however did not give a more specific indication”. The lack of an obvious processing indicator may have resulted in either the participants continuing to apply filters without waiting for the processing to be completed or pausing to try to establish the outcome of the application of the filter. This effect may have contributed to the longer completion times and the lower SUS scores for Site H in comparison to Site T.

The approach utilized by Site O involving a pop-up box with a white translucent overlay of the results section of the page. This combination provided the most obvious processing indicator. It was apparent from the analysis that the majority of the participants paused until the refresh was completed before continuing any interaction. The relatively high SUS scores may suggest that this approach is effective in providing feedback to users. This approach may have resulted in slightly longer booking times than Site K due to the enforced pause with P11 stating “At least you know it’s filtering but it’s slower”.

The second usability issue related to the participant’s awareness of the status of the system when processing had completed. A significant issue that was observed in Site T related to a particular filter control that could only be viewed on a standard window size by scrolling down. It was observed that many participants scrolled the page to the top of the screen immediately after interacting with this particular filter and therefore missed seeing the processing indicator pop-up box. As a result many participants appeared unaware that processing was occurring, e.g. P6 stated “Is it automatically filtering – I cannot see any change here” and P10 stated “I cannot tell whether it has done the filtering or not”. This effect may have contributed to the longer completion times and the lower SUS scores for Site T.

A similar issue arose for Site K in cases where the popup box appeared only very briefly due to short processing times. For example, P13 stated “A little box came up really fast and I suppose this was telling me that it was changing the results”. P11 wrote in his SUS feedback form that he did not believe that the site provided appropriate information to inform that the search had completed: “the filtering sign popped up over the screen and then it was very quick”. P13 said “A
message flashed up but I didn’t always see it” and P17 stated “There was no clear indicator that the new search results had changed. You had to be very aware of the page to notice”. Whilst many participants expressed concerns it is noted that Site K provide the highest SUS score and lowest task completion times.

6.5 Other Results

The following comments were noted regarding the design of Site K and Site O that had the lowest task completion times. P19 stated regarding Site K: “I like this, neat and clean looking – my favourite”. In relation to Site O, P4 stated that “I think this site is cool” and P11 stated “The best site, slower when loading filtered data but clear and easy to use”. Participant’s comments suggested a strong preference for the design of Site K consistent with the categorized SUS scores.

There were multiple comments regarding the ‘Rating’ filter control implemented on Site H. The control was an Ajax-enabled ‘slider’ that required the user to move markers along a vertical line to select their minimum and maximum ratings. Comments included: P17: “Star rating is little bit annoying as you have to know how to use these bars”, P5: “Change the rating control to use a select box”, P7: “The Star bar is awkward and cumbersome to use”, P8: “The star checkbox need to be improved”, and P19: “The star rating feature was very awkward”.

Many participants noted difficulties with navigating Site T, particularly locating several of the filter controls that were placed towards the top-right of the page template. Comments included: P6: “Move the location control to the left side”, P9: “The neighbourhood location was difficult to find”, P11: “the filtering options on the top of the site were confusing and awkward to find out first”, and P17: “It was a little hard to navigate, was a bit annoying and it could make some of the options a bit easier to find”. This design issue may have contributed to the longer completion times and the lower SUS scores for Site T.

7 Discussion

This study has examined three aspects of a set of Ajax enabled commercial websites to determine whether the known usability issues are apparent and if they have an effect on the usability of the sites. Table 1 shows a ‘Traffic Light’ summary of the results of the study. The two issues that were investigated are presented on the left side of the horizontal axis, i.e. the action of the back button, the effectiveness of the processing indicator along with ratings based on the general design of the site. On the right are the two performance indicators, i.e. the task completion times and the system usability scores (SUS).

The summary results suggest that Site K performed the ‘best’ with the shortest completion times and highest SUS ratings however when using this site the Back button performed unpredictably. There were also issues relating to the processing indicator where some participants were unaware that the Ajax had finished processing. These issues may have been compensated by the design of the site including good navigational support with participants expressing a clear preference for the site design in verbal comments and the design related questions in the SUS questionnaire.

<table>
<thead>
<tr>
<th>Site</th>
<th>Back Button Action</th>
<th>Processing Indicator</th>
<th>Design</th>
<th>Completion Time</th>
<th>System Usability Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>Predictable</td>
<td>Not obvious</td>
<td>Filter Control</td>
<td>129</td>
<td>70</td>
</tr>
<tr>
<td>K</td>
<td>Unpredictable</td>
<td>Brief</td>
<td>Clean</td>
<td>90</td>
<td>74</td>
</tr>
<tr>
<td>O</td>
<td>Predictable</td>
<td>Very obvious</td>
<td>Cool</td>
<td>99</td>
<td>70</td>
</tr>
</tbody>
</table>

Site O was ranked the next ‘best’ having a predictable Back button, the most obvious processing indicator and a preference for the design. The slightly longer completion times may have been a result of the enforced pauses as a result of the processing indicator.

Site H had the least obvious processing indicator and issues with the Ajax-enabled ‘slider’ control for the star rating. Together these may have contributed to the longer completion times.

Site T had issues with both the predictability of the Back button and the timing of the processing indicator with many participants who missed seeing the indicator commence or finish. Whilst these may have contributed to the long completion times and low SUS ratings, the general site layout caused some frustration with many participants having difficulty in locating several filter controls which were placed towards the top-right of the page template.

The results of the study confirm that commercial web developers are inconsistently managing known usability issues. The results also indicate that combinations of the usability issues do affect user performance and satisfaction.

Section 3 of this paper describes the general usability heuristics of consistency, learnability and feedback. These general principles are common themes that are relevant when considering the usability of commercial Ajax-based Web applications. The results of this study may be considered in relation to these themes in order to generate some recommendations for web site developers who employ Ajax technologies.

Consistency: The operation of the Back button has now become entrenched as part of the mental model of Web users. This model allows users to interact with the Web with minimal cognitive overhead as they can confidently predict the outcomes of their actions and plan and execute browsing strategies. The operation of the Back button in two out of the four sites in this study broke the classical model of the Web resulting in some participants reporting confusion and disorientation. The results of the study may indicate that good navigational support and site design can alleviate the detrimental effects of an unpredictable Back button, i.e. users may not need to backtrack as much whilst navigating. The results suggest retaining Back button functionality consistent with the classical Web model is an important usability factor in conjunction with other factors. It is recommended that Ajax developers implement technical
solutions that ensure that the Back button has predictable outcomes.

Learnability: The ability for users to quickly figure out how to use a web site is a critical success factor in user acceptance [18]. The study found several design issues in Sites H and T that resulted in longer completion times and low SUS scores, particularly for the initial task when users were first exposed to the site. The effect in Site H was particularly profound as a result of an innovative Ajax ‘slider’ control (Figure 4).

Figure 4: Slider Control in Site H

Bosworth (2005) captures this issue by stating “click on this non obvious thing to drive this other non obvious result”. Frequent users may learn how to use innovative Ajax controls with a possible improvement in performance but the negative effect on first time or novice users could outweigh perceived benefits. “Users need to be able to anticipate what a widget will do, appreciate how to interact with it, monitor changes during use and evaluate what has changed after use” (Atyeo, 2006).

Feedback: Usable systems should provide inherent clues to what actions are possible at any moment, the results of actions and the current state of the system so that users will know what to do instinctively (Norman, 1988). The four sites examined in the study implemented different approaches to indicating that a request was being processed and when the request was complete. The challenge of indicating to the user that there has been a part-page refresh of a particular component of the page was not managed well by two of the sites resulting in some user confusion and a decline in performance. Users appeared to favour Site O that provided very clear processing indicators. Ajax developers should employ standard processing indicator devices to clearly inform the user of the processing status. In addition Ajax developers should be aware of the potential for ‘change blindness’ that may be caused when a user is not aware of any change during a part-page refresh. The success of the enhanced interactivity enabled through Ajax relies on the designer’s ability to provide appropriate feedback on the status of the Web application at all times.

8 Conclusion

Ajax has several usability issues including consistency of the Back button and the management of part-page updates. These issues have been reported in the literature along with guidelines and technical solutions that that could be employed by Ajax developers to reduce undesirable usability effects in their Web applications.

This paper presents the results of an empirical study into four hotel booking sites that employ Ajax technologies in order to investigate how these sites have responded to the known usability issues. The results of the study were contrasted in relation to the general usability principles of consistency, learnability and feedback.

The study found inconsistencies in how the sites managed the known usability issues and how combinations of the issues have a detrimental effect on user performance and satisfaction. The paper makes several recommendations to Ajax developers to in relation to consistency, learnability and feedback.

9 References


Determining the Relative Benefits of Pairing Virtual Reality Displays with Applications

Edward M. Peek   Burkhard Wünsche           Christof Lutteroth

Graphics Group, Department of Computer Science
The University of Auckland
Private Bag 92019, Auckland 1142, New Zealand
epee004@aucklanduni.ac.nz      b.wuensche@auckland.ac.nz     lutteroth@cs.auckland.ac.nz

Abstract
Over the last century, virtual reality (VR) technologies (stereoscopic displays in particular) have repeatedly been advertised as the future of movies, television, and more recently, gaming and general HCI. However after each wave of commercial VR products, consumer interest in them has slowly faded away as the novelty of the experience wore off and its benefits were no longer perceived as enough to outweigh the cost and limitations. Academic research has shown that the amount of benefit a VR technology provides depends in the application it is used for and that, contrary to how these technologies are often marketed, there is currently no one-size-fits-all 3D technology. In this paper we present an evaluation framework designed to determine the quality of depth cues produced when using a 3D display technology with a specific application. We also present the results of using this framework to evaluate some common consumer VR technologies. Our framework works by evaluating the technical properties of both the display and application against a set of quality metrics. This framework can help identify the 3D display technology which provides the largest benefit for a desired application.

Keywords: virtual reality, evaluation framework, 3D displays, 3D applications.

1 Introduction
Virtual reality (VR) is the name given to the concept behind the group of technologies whose purpose is to realistically create the perception of virtual objects existing in the real world through manipulation of human senses without physical representations of the virtual objects. The virtual scene is typically either an interactive computer simulation or a recording of a physical scene. The degree to which the real world is replaced with the virtual one gives rise to a spectrum of alternative terms for VR (Milgram and Kishino 1994) illustrated in Figure 1.

While virtual reality in general deals with manipulating all the human senses (sight, smell, touch etc.) the largest portion of research into VR is related to the visual and to a lesser extent auditory components. This paper focuses solely of assessing visual VR technologies and other aspects of VR are not discussed. Computer displays are the established method of producing the visual component of virtual reality with displays that are designed to achieve a high degree of virtual presence called “3D displays” or “virtual reality displays”. Virtual presence is the extent of belief that the image of the scene presented to the user exists in real space and is largely determined by the number and quality of depth cues a display is able to recreate. Depth cues are the mechanisms by which the human brain determines the depths of objects based on the images received by each eye. These cues have been well known since the 18th century and are usually grouped as shown in Table 1.

Table 1: Groups of depth cues

<table>
<thead>
<tr>
<th>Group</th>
<th>Depth Cue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pictorial</td>
<td>Perspective, Texture, Shading, Shadows, Relative Motion, Occlusion</td>
</tr>
<tr>
<td>Physiological</td>
<td>Accommodation, Convergence</td>
</tr>
<tr>
<td>Parallax</td>
<td>Binocular, Motion</td>
</tr>
</tbody>
</table>

Figure 1: Spectrum of VR-related terms
Mostly virtual  Mixed Reality  Mostly real

Copyright © 2013, Australian Computer Society, Inc. This paper appeared at the 14th Australasian User Interface Conference (AUIC 2013), Adelaide, Australia. Conferences in Research and Practice in Information Technology (CRPIT), Vol. 139. Ross T. Smith and Burkhard Wünsche, Eds. Reproduction for academic, not-for-profit purposes permitted provided this text is included.
display technologies. Stereoscopic displays are one such technology that has seen limited success in these applications (Didyk 2011) but as of yet is the only technology to be widely used in this area.

This paper exists to help determine why stereoscopy is the only technology that has achieved widespread consumer adoption and to identify applications where alternative VR display technologies may be potentially better.

2 Related Work

Relatively little work has been previously done regarding the relative benefits of alternative display technologies for applications, especially considering the large amount of general VR research available. What has been done has typically fallen into one of the following categories:

- Enhancing specific display technologies for particular applications.
- Comparing and evaluating display technologies through user testing.
- Developing classification systems for grouping different display technologies.

A large part of the current literature relating to the relative benefits of different VR display technologies is incidental and a by-product of the larger field of developing and analysing these technologies in isolation. Most of this research involves user testing to validate a developed technology, usually with a comparison to a few control displays. Since different papers use different testing setups and tasks, it is difficult to compare a broad spectrum of display technologies using this data.

Fortunately there has been some dedicated discussion regarding the relative benefits of different display technologies for specific applications (Wen et al. 2006), the benefits of a single VR display technology over several applications (Litwiller and LaViola 2011), and also measurements of how well these VR technologies compare to actual reality (Treadgold et al. 2001). While this still falls short of giving a complete picture, it does provide validation that such an overview would be useful. The results from these papers and others confirm that users’ performance and satisfaction when interacting with the virtual scene generally improves with more sophisticated display technologies that are able to achieve a higher degree of virtual presence. Another common theme with this area is that results vary significantly depending on the exact task performed (Grossman and Balakrishnan 2006). These two points provide motivation for developing a system to predict how beneficial individual display technologies will be for specific applications.

Examples of typical classifications methods are those described by Blundell (2012) and Pimenta and Santos (2012). Blundell describes a classification system where specific display implementations are grouped according to the types of depth cues they support and the need to wear decoding glasses. The differentiating cues were determined to be binocular parallax, parallax from observer dynamics (POD) and natural accommodation & convergence (A/C). Three major display groups are formed by the level of support for binocular parallax: monocular for no parallax, stereoscopic for parallax with glasses and autostereoscopic for parallax without glasses. Monocular and stereoscopic displays are then broken down further according to their support for POD resulting in tracked and untracked variants of each. Autostereoscopic displays are however differentiated by both POD and natural A/C where so called “class I” displays only support discrete POD without natural A/C, while “class II” displays fully support both.

Pimenta and Santos (2012) use a different method for classifying VR display technologies yet end up with similar final groups. Their approach groups displays according to two criteria: the number of views supported and the depth of the display. The number of views refers to how many different images with correct parallax the display can show simultaneously with groups for two, more than two and infinite views (duoscopic, multisopic and omniscopic). Conventional displays with only one view are not encompassed by their taxonomy, but it is trivial to extend it to include them by adding a monoscopic group. The second criteria regards whether the perceived location of an image is the same as where it is emitted from. Displays which produce an apparent 3D image using a 2D surface are considered “flat” while displays that produce 3D images using a 3D volume are considered “deep”. This system results in five groups, two of which can be mapped exactly to those described by Blundell (stereoscopic and autostereoscopic class I) while the other three (multi-directional, virtual volume and volumetric) can be considered subgroups of autostereoscopic class II displays.

Despite this collective effort of investigating specific display technologies and classifying them, we are unaware of any attempt to determine which applications these groups of displays are particularly suited for. This is surprising since various results indicate (Grossman and Balakrishnan 2006) and some authors acknowledge (Blundell 2012) that as of yet there is no one-size-fits-all display technology. This leaves a gap in the literature regarding a systematic way of determining how well-suit a specific VR display technology is for a specific application. Hence this paper attempts to fill this gap by describing an evaluation framework through which the relative merits of different display technologies can be compared to the unique requirements of individual applications. This will hopefully simplify in the future the identification of applications areas which could benefit from the use of non-traditional display technologies.

3 Evaluation Framework

Our paper’s contribution to this problem is an evaluation framework with the role of determining how well suited each available display technology is for specific applications. In order to produce what we would judge a useful framework, the following properties were considered important:

- Effective: It would identify applications where using non-traditional display technologies could improve the user’s experience and task performance. It would also identify applications where the de facto 3D display technology may not actually be the best choice.
• Lightweight: It should only require easily obtainable technical information about the applications and display technologies of interest, thus making it easier to apply than performing user testing.
• Generic: It should not require information specific to each combination of application and display technology, but rather each of these independently.
• Extensible: It should be easy to extend to include additional applications, display technologies and measurement criteria.

The framework we developed examines the suitability of each pairing between a display technology and application independently to meet the goals of being generic and extensible. Suitability is defined as the quality of the depth cues produced by a specific pairing and other non-depth-based factors as a way to predict the quality of the user experience without needing to take into account subjective factors. As the perceived quality of a depth cue is normally determined by several factors, the quality of a depth cue is measured by a set of quality metrics specific to that cue. Other factors are also measured by quality metrics of the same form. Through this approach, the output of our framework for a single pairing is a vector of quality values representing the suitability of the pairing (where each quality value has been produced by a single quality metric).

As each pairing is considered independently, given a list of display technologies and applications, every possible pairing combination of them can be evaluated using this framework to generate a table of how suitable each pair is. The relative merits of two or more pairings can then be seen and contrasted by comparing the quality values of their rows in the table.

To automate the completion of this task, the evaluation framework was implemented as a short (~150 SLOC not including display, application and metric definitions) Python program that took a list of displays and applications, generated all the possible pairings, ran a supplied list of quality metrics against the pairings and outputted the results as a CSV file.

The evaluation of a pairing consists of several inputs: a display technology, an application, and a set of quality metrics. What constitutes these inputs are described in the following sections with examples listed in Appendix A.

3.1 Display Technology
In the context of our framework, a display technology refers to the mechanism a display uses to present its image to the user. Since the exact mechanisms in real-world displays vary in many details which are insignificant to us, display technologies are generalised cases allowing these differences to be ignored. It should be noted that according to this definition a display technology is not tied to the display itself, signifying that display technologies are interchangeable. This is important as the goal of this framework is to evaluate how certain user experience metrics vary when only the display technology is changed.

Individual display technologies are characterised by a set of abstract and physical properties describing the technology. What exactly these properties are is determined by what the quality metrics require in order to be computed. Examples of display properties are the produced image space, image resolution and the number of independent images shown at any single point in time.

Even with grouping many display technologies often have near-identical properties, e.g. polarised stereoscopic displays and time-interlaced stereoscopic displays. This makes it natural to organise several related display technologies as a tree structure where child nodes inherit common property values from a generalised parent. Such groupings turn out to be similar to previous classifications, including that produced by Blundell (2012) found in Table 1 of his paper. Since we were mostly interested in practical applications, and considering that parent nodes tend to represent non-existent technologies (e.g. “pure stereoscopic” displays do not exist), we only considered leaf nodes in our evaluations. Regardless of how display technologies are organised, it becomes inconsequential later as they are evaluated independently leaving such grouping useful only for organising input data.

3.2 Application
An application is any common scenario in which some display technology is used to present 3D content. An example application might then be watching a movie at home on a TV set. The technology being used for the display (the TV set in our example) is not considered part of the application as it is one of the independent variables in the evaluation.

Applications could be further generalised as a combination of a task the user is performing on the display and the context in which this is performed. In the previous example the task would be “watching a movie” while the context is “at home on a TV set”. This split arises from the fact that the same task or context may appear in many applications, e.g. a movie can be watched in a theatre, on TV, on a mobile device or on a plane, across which it is likely that requirements for a high quality user experience will change. Our framework however, was designed to ignore this detail and instead consider applications as indivisible. This was because we did not expect that accommodating for varying the task and context independently would benefit enough to outweigh the added complexity. Instead, different applications are simply created every time a recurring task or context occurs.

Like display technologies, applications have a common set of properties determined by the quality metrics used, the values of which differ between individual applications. Examples of application properties required by our metrics are typical viewing distance, number of simultaneous users and the amount of user movement relative to the display.

3.3 Quality Metrics
Quality metrics describe how well the pairing produces different aspects of the user’s experience. While these are mostly aimed towards the produced depth cues, they do not need to be, and can measure some other aspect that is affected by the choice of display technology. Examples of
metrics are: the disparity between accommodation and convergence, the brightness of the display, the range of motion the user can experience motion parallax over, the weight of headgear needed or the monetary cost of implementing the system.

To enable automated evaluation of these metrics, they are implemented as functions which take the display technology and application as arguments and return a numerical value representing the quality of that pairing according to the metric.

A distinction can be made between quality metrics that are essential for a depth cue to be correctly perceived and those that merely affect the accuracy or quality of the cue. We refer to these as hard and soft metrics respectively.

### 3.3.1 Soft Metrics

Soft quality metrics are the main component of interest for this framework. Each soft metric represents a single factor that influences how well a user perceives a specific depth cue and is represented by a single numerical value. How the metric produces these values is entirely dependent on the metric in question but is always based solely on the properties of the display technology and application it takes as inputs. There is no common process or calculations between soft metrics and the values they output can be to any scale as metrics are not intended to be compared between. This also allows soft metrics to be created completely independently simplifying the process of defining and creating them. Values produced by a soft metric are however required to be consistent within that metric allowing them to be numerically compared between different display/application pairs to determine which pair better delivers that aspect of the depth cue. By creating a vector of all the soft metrics relating to a particular depth cue, the quality of the entire depth cue can compared between pairings using ordinary vector inequalities partially avoiding the need to compare metrics individually.

What follows is a short description of our “accommodation/convergence (A/C) breakdown” soft metric, discussed as an example of how soft metrics are defined and how a well-known quality factor of VR displays is handled by our framework. A/C breakdown occurs where the accommodation produced by a display does not match the convergence and is thought to be one of the major causes of asthenopia (eye strain) in stereoscopic displays (Blundell 2012). Such displays are said to have an apparent image space, while displays that correctly produce accommodation and convergence have a physical or virtual image space. Since the sensitivity of both these cues is inversely proportional to distance, the further the display is from the viewer the less of an issue A/C breakdown is. To model this our quality metric $M_{bd}$ follows the equation:

$$M_{bd}(A, D) = \begin{cases} 
\infty & \text{if } space_D = \text{'physical'} \\
\infty & \text{if } space_D = \text{'virtual'} \\
distance_A & \text{if } space_D = \text{'apparent'} 
\end{cases}$$

Where $A$ and $D$ are the application and display respectively, $distance_A$ is the viewer distance property of the application and $space_D$ is the image space property of the display.

### 3.3.2 Hard Metrics / Requirements

Hard metrics are those that determine if a display technology is capable of producing a specific depth cue for all the users of the application in the pairing. Unlike soft metrics, hard metrics do not reflect the quality of the depth cue itself and so are not included in the output of the evaluation. Instead they are used as a check to skip over any soft metrics that would otherwise represent the quality of a cue that is in fact not present. If a hard metric does not pass a specific threshold all the soft metrics dependent on it are given a value indicating they are not present (this value is different to what they would have if they were merely of poor quality).

Examples of requirements for depth cues are: to achieve binocular parallax the display must present at least 2 independent images to the user’s eyes, to achieve motion parallax the display must present a different image according to their eye location, and so on.

As with soft metrics a hard metric does not need to pertain to a depth cue. If it does not, it indicates whether the pairing is possible according to some other logical requirement, e.g. the number of simultaneous users supported by the display technology must be greater than or equal to the typical number of users of the application.

### 4 Results

To test the effectiveness of the framework we performed an evaluation with a set of 12 general display technologies and 10 consumer oriented applications. As with the selected applications, the included display types were mostly sub $1000 NZD consumer-grade technologies with a few specialised and theoretical technologies added for the sake of comparison. 20 soft quality metrics were used to judge the pairings with restriction by 5 hard metrics. Lists of these can be found in Appendix A. For the sake of brevity we have excluded the values of display technology and application properties, as well as the inner formulae for each metric. 34 pairings of the original 120 were eliminated by the hard metrics leaving 86 suitable pairings.

A portion of the raw results table can be found in Appendix B with the values of each metric normalised so that higher values are always desirable over lower values. In this way a value of positive infinity indicates that a quality metric is flawless in that pairing, although finite values can also indicate a perfectly met metric depending on what might be considered perfect for that metric. Values are also colour coded with white being bad, green being good and grey indicating a failed hard metric for that depth cue. Since the scale of the quality metrics is arbitrary and varies between metrics, individual values are not meaningful by themselves but are useful for comparisons between pairings.

### 5 Discussion

#### 5.1 Findings

Among the interesting pairings identified, one potentially worthwhile area of investigation is head-coupled perspective on mobile devices. Our evaluation showed it to perform better among the general metrics than the stereoscopy-based alternatives. This is interesting because
several mobile devices have already been released with parallax-barrier autostereoscopic displays suggesting that mobile devices with head-coupled perspective should be a feasible option.

A to-be-expected result was that fish-tank VR ranks consistently high for the entire range of desktop applications. This makes sense as it ranks high in both binocular and motion parallax metrics while other display technologies only rank highly in one of them. Fish-tank VR does not rank well in other applications however as its single user requirement usually causes it to be eliminated by the “number of viewers” hard metric.

5.2 Validity

As a method of predicting the suitability of real-world pairings, it was important to validate our framework so that the results it produces can be considered reliable and applicable to the pairings when they are physically implemented.

With respect to the structure of the framework itself, the principal condition of it being valid is that the quality of a user’s experience in interacting with a 3D display technology can be measured at least partially by properties of the display technology, the task being performed and the context in which this happens. This is not an unreasonable claim as virtually all previous research in 3D display technologies shows measurable differences in user experience based on what technology is being used and what task the user is asked to perform (Wen et al. 2006, Litwiller and LaViola 2011, Grossman and Balakrishnan 2006). From this we can conclude that the general premise of the framework is valid.

The other area in which validity must be questioned is with regard to the quality metrics themselves. One point that must be considered is that the quality metrics chosen for evaluation must measure factors that have some noticeable effect on the quality of the user experience. This effect can be noticed either consciously or subconsciously. Factors that are consciously noticeable are simple to validate by asking users whether it is something that affects their experience. Subconscious factors are slightly more difficult to validate as users may only notice the effect of them, not the factors themselves. Fortunately quality factors in virtual reality is a well-researched area making validating subconscious quality factors an exercise in reviewing the literature (e.g. the A/C breakdown cue discussed in section 3.3.1). Since user experience is subjective by definition it must be ensured that a reasonable sample of people is used to validate quality metrics to ensure they remain representative of the population of interest.

5.3 Limitations

While the developed framework does achieve its goal of providing a lightweight method to uniformly compare 3D display technologies within the context of the applications in which they are used, certain aspects of the design cause some problems to arise when analysing the results. Most of these limitations arose from a balancing issue where increasing the simplicity of the framework counters how sophisticated the performed evaluation is.

The main area our framework falls short is in providing an intelligent reduction of the raw results. Instead it requires manual inspection to identify pairings of interest. Since the number of results generated by the framework grows quadratically with the number of displays and applications, this inspection can become labour-intensive when more than a few of these are evaluated at once.

A smaller issue found with our method was the need for single values for application and display properties. This can become an issue when realistic estimates of the value are near the threshold of a hard metric. A display/application pairing may be unnecessarily rejected because of this depending on which side of the threshold the chosen value lies. An example of this happening with our data is the rejection of the pairing of console gaming with head-coupled perspective. Since our chosen value of the typical number of users for this application was greater than the single user supported by HCP this pairing was rejected even though console games are also frequently played by a single person.

Another problem is the use of ordinal scales for quality metrics. While this makes them easy to implement, it also makes anything other than better/worse comparisons impossible without understanding the range of values produced by the metric of interest. This undermines the simplicity of the framework and the validity of its conclusions, as even if one pairing is determined to be better than another, how much better it is cannot be easily quantified.

Not having a common scale between different quality metrics also hinders comparisons between them. Being able to do this would be useful as it would allow better comparisons of pairings where both pairs have some metrics than the other pair. Such scenarios are very common with real display technologies which have many trade-offs, and only theoretical technologies are generally able to be better in every way than others.

The final major limitation is not specifically with our implemented framework, but with its design goals. Our framework is intentionally designed to only consider the technical aspects of using a 3D display technology and not so much the subjective aspects. An important subjective aspect relevant to our framework is how much each of the quality metrics actually affect the quality of the user’s experience. The reason we decided to ignore this aspect comes mostly down to the amount of effort it would take to collect the required data. Since the extent to which a quality metric affects the user experience depends on the application, user testing would need to be performed for every combination of application and quality metric to determine the magnitude of this effect. This would likely cause performing an evaluation using our framework to take more effort than ordinary user testing which would defeat its purpose of being quick and lightweight.

6 Future Work

The major area for improvement with our system is to solve the problem of reducing the raw results to something more manageable and easy to analyse. An unintegrated solution would be to find another suitable evaluation method which could then further analyse the
results of our framework. Alternatively this would also partially emerge from overcoming the other limitations that complicate comparing the quality of different pairings.

One of the simplest changes that could be made to improve our framework would be to require soft quality metrics to conform to a common scale. This could be continuous (e.g. 0 to 10) or discrete (e.g. unacceptable, poor, average, good, perfect). This would make the values returned from quality metrics much more meaningful and would partially facilitate comparisons between metrics. That is, several good or perfect metrics might be able to compensate for a poor one regardless of what the metrics are.

A further refinement of fixed scales would be to facilitate calculating weighted sums of all the quality metrics of a depth cue, and/or the entire set of quality metrics. This would again reduce the complexity of analysing the results as pairings could be compared at a higher level than individual quality metrics. The trade-off for this change would be the increased effort in finding the weights for each quality metric. As mentioned in the limitations section, accurate application-specific weights would necessitate user testing. However approximated general weights might also be accurate enough for this addition to be beneficial.

Other future work would be to avoid the previously discussed issue of needing ranges of values for application properties. A trivial solution to avoid this is splitting applications into more specific scenarios. A downside to this is that it would further exacerbate the issue of producing too much output data. A more targeted solution would be to allow a range of values for properties and have the hard metrics tests be a tri-state (pass, fail or partial pass) instead of boolean (pass or fail).

With regards to improving validity, accurately identifying what quality metrics truly affect the experience of using 3D display technologies would give added weight to the results produced by this framework. Such metrics are likely universal and would therefore be useful for other virtual reality research and not just this framework.

7 Conclusions

We have developed a lightweight framework designed to evaluate the suitability of using 3D display technologies with different applications as an alternative to user testing. The evaluation tests suitability according to a list of quality metrics that represent factors affecting the quality of the user’s experience. We successfully performed an evaluation on several consumer-oriented display technologies and applications and identified pairings of future research interest. Our framework is mostly held back by the difficulty of efficiently interpreting the results it generates.

8 References


9 Appendix A

9.1 Display Technologies

- Swept volume
- Sparse integral multiview (one view per user)
- Dense integral multiview (many views per user)
- Light-field (hypothetical display capable of producing at least a 4D light field)
- Head-coupled perspective
- Fish-tank VR
- Head-mounted display
- Tracked head-mounted display
- Anaglyph stereoscopy
- Line-interlace polarised stereoscopy
- Temporally-interlaced stereoscopy
- Parallax-barrier autostereoscopy

9.2 Applications

- Cinema
- Home theatre
- TV console gaming
- TV console motion gaming
- Mobile gaming
- Mobile videotelephony
- Information kiosk
- Desktop gaming
9.3 Requirements

- Number of viewers
- Display portability
- Variable binocular parallax produced
- Variable convergence produced
- Variable motion parallax produced

9.4 Quality Metrics

9.4.1 General

- System cost
- Cost of users
- Rendering computation cost
- More views rendered than seen
- Scene depth accuracy
- Headgear needed

9.4.2 Pictorial

- Spatial resolution

9.4.3 Motion Parallax

- Parallax unique to each user
- Amount of induced parallax
- Degrees-of-freedom/number of axis supported
- Latency
- Continuous or discrete

9.4.4 Binocular Parallax

- Amount of wobble
- Stereo inversion

9.4.5 Accommodation

- A/C breakdown

9.4.6 Convergence

No metrics for convergence, assumed constant quality if present.

10 Appendix B

<table>
<thead>
<tr>
<th>Application</th>
<th>Enhancement</th>
<th>Accommodation</th>
<th>Binocular Parallax</th>
<th>Convergence</th>
<th>General</th>
<th>Motion Parallax</th>
<th>Pictorial</th>
<th>Colour distortion</th>
<th>Brightness</th>
<th>Per-use</th>
<th>Interaction</th>
<th>Resolution</th>
<th>Refresh rate</th>
<th>Redefinition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cinema</td>
<td>Anaglyph Stereoscopic</td>
<td>1</td>
<td>1</td>
<td>10</td>
<td>1</td>
<td>0.75</td>
<td>4</td>
<td>5</td>
<td>0.01</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Console Gaming</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0.5</td>
<td>1</td>
<td>0.2</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC CAD</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0.5</td>
<td>1</td>
<td>0.2</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC Gaming</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0.5</td>
<td>1</td>
<td>0.2</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop Videotelephony</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.75</td>
<td>4</td>
<td>5</td>
<td>0.01</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Home Theatre</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.75</td>
<td>4</td>
<td>5</td>
<td>0.01</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Information Kiosk</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.75</td>
<td>4</td>
<td>5</td>
<td>0.01</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Motion Console Gaming</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.75</td>
<td>4</td>
<td>5</td>
<td>0.01</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mobile Gaming</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.75</td>
<td>4</td>
<td>5</td>
<td>0.01</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mobile Videotelephone</td>
<td>Anaglyph Stereoscopic</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.75</td>
<td>4</td>
<td>5</td>
<td>0.01</td>
<td>1</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Motion Console Gaming</td>
<td>Dense Integral Multiview</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Cinema</td>
<td>Dense Integral Multiview</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Console Gaming</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC CAD</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC Gaming</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop Videotelephony</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Home Theatre</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Information Kiosk</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mobile Gaming</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mobile Videotelephone</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Motion Console Gaming</td>
<td>Dense Integral Multiview</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC CAD</td>
<td>Fish-tank VR</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC Gaming</td>
<td>Fish-tank VR</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop Videotelephony</td>
<td>Fish-tank VR</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC CAD</td>
<td>Head-coupled Perspective</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC Gaming</td>
<td>Head-coupled Perspective</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop Videotelephony</td>
<td>Head-coupled Perspective</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mobile Gaming</td>
<td>Head-coupled Perspective</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mobile Videotelephony</td>
<td>Head-coupled Perspective</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC CAD</td>
<td>Head-mounted Display</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop PC Gaming</td>
<td>Head-mounted Display</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Desktop Videotelephony</td>
<td>Head-mounted Display</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Cinema</td>
<td>Light Field</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Console Gaming</td>
<td>Light Field</td>
<td>0.5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Desktop computer-aided-design (CAD) + Desktop videotelephony

Temporal resolution/refresh rate + Relative brightness + Colour distortion + Objects can occlude
CONTRIBUTED POSTERS
An Ethnographic Study of a High Cognitive Load Driving Environment

Robert Wellington  Stefan Marks

School of Computing and Mathematical Sciences
AUT University, 2-14 Wakefield St, Auckland 1142, New Zealand,
Email: robert.wellington@aut.ac.nz/stefan.marks

Abstract

This poster outlines Ethnographic research into the design of an environment to study a land speed record vehicle, or more generally, a vehicle posing a high cognitive load for the user. The challenges of empirical research activity in the design of unique artifacts is discussed, where we may not have the artefact available in the real context to study, nor key informants that have direct relevant experience. We also describe findings from the preliminary design studies and the study into the design of the yoke for driving steer-by-wire.

Keywords: Yoke, Steering, Ethnography, Cognitive Load

1 Introduction

The task for the research team was to create an environment to undertake research on the cockpit design of a Land Speed Record vehicle, this being inspired by the public launch of the New Zealand “Jetblack” land speed record project, and our growing awareness of numerous land speed record projects sprouting up around the globe. We have conceptualised this research project slightly more broadly as ‘undertaking research on vehicle user interaction in a high cognitive load environment’. This gives us a unique environment, in contrast to the majority of current research that concentrates on what is considered a normal cognitive load, and the focus is then on attention, fatigue, and distractions (Ho & Spence 2008). In contrast, our focus is on an innately higher risk activity with a cognitive load that is bordering on extreme.

So how do you go about researching artifacts that dont exist? Every method for undertaking empirical research is a limited representation of reality, a simplification, and for a reality that is hypothetical this potentially exacerbates these limitations. We have predominantly been using an Ethnographic process (Wellington 2011) to gather data from participants and experts about the elements we have been designing. The specific design is only the focus of the context that we are interested in. Too much emphasis has been placed on the role of Ethnography to provide sociological requirements in the design process (Dourish 2006). The objective here is also to explore the HCI theory related to this context and activities in an analytical way.

For our project, having it’s foundations in a Land Speed Record vehicle, the cohort of potential drivers has traditionally come from available and willing airforce pilots. Therefore, we interviewed actual air force pilots in the cockpits of various military aircraft, and were able to discuss the design of the control systems, as well as the potential concepts a land speed record vehicle driver would need to be aware of in controlling his or her vehicle. We used an ethnographic data collection method for gathering the knowledge of experts, where a conversational style is preferred over structured questions, and the researcher / interviewer says as little as possible to make sure that they are collecting the interviewees ‘truth’ rather than confirming their own.

Later in the research, once we had built our simulator, the research participants were informed that we were interested in any of their opinions on any aspect of the simulation or the design, we can place more significance on anything they volunteer specifically about the yoke or steering, as this then suggests it had some importance to their experience, rather than something they had to come up with since they were prompted. When participants then start discussing something they felt in the steering we then have to restrain ourselves from asking too many questions, or explaining the technical details of the device, and simply try to capture what they are saying.
2 Findings

Our initial yoke design was a consequence of the first round of Ethnographic research involving; two ‘in cockpit’ interviews with senior military pilots, and several meetings with another senior pilot, along with conversations with landspeed record vehicle designers and engineers. Since this initial data collection we have also interviewed racing car drivers and drag race drivers in the process of gathering data in the simulator. Specifically, for a land speed vehicle, the preference of military pilots interviewed, is that thrust controls are on the right, and inhibition controls are on the left, aligned with the habituated use of an accelerator and a brake pedal in a domestic vehicle.

The initial design of the yoke was also constructed as a synthesis of available yoke designs with a major consideration to minimising the use of space and giving an uninterrupted view forward, see Figure 1. The buttons for the yoke are standard components from a local electronics store, and are reticulated through the back in a groove, held in with insulation tape. An unintended benefit of the tape was to give a better indication of finger position from video footage, without giving a cue that we were focussing on hand position. The yoke is connected to a Logitech G27 steering wheel, and the other components of the steering wheel controller are employed for the accelerator and brake pedal, although these have been removed from their casing and the casing flipped, and then the pedals mounted on the underside, to give a good angle of operation in this seating position.

The significant difference between a circular wheel and the yoke – given similar configurations of limited steering angle – is that the participants are more likely to attempt to turn the circular steering wheel through a greater range of motion. We can tell from direct observation (for an example see Figure 1), that there is greater uniformity of the positioning of the minor digits in comparison to the index finger, as is shown in Figure 2. There are fewer data points for the index finger shown in this diagram, as many drivers wrapped their index finger around the back of the grip and it was not visible in video footage, or the position was not able to be predicted with any degree of confidence.

Although there were also other odd behaviours, such as holding the yoke by the ‘spoke’ held lightly between one participants index finger and thumb, these odd behaviours were often ephemeral as the participants realised the difficulty of the task and began to concentrate, and to hold the yoke by the handles. The person’s thumb then typically attained one of four positions: hovering over the primary buttons, hovering over the secondary buttons, settled between the buttons, or wrapped – opposing the finger grip.

The thumbs of each hand could be matched or in a combination of these positions, with no single position dominating through the observations. The ‘naturalness’ of these different positions is reassuring, as the degree of turbulence and vibration of a real vehicle is unknown at this stage, it is unknown whether the driver can maintain a light hold of the yoke, or whether on occasion they will need to hold on tight.

There was a noticeable difference in the position of the left and right thumbs, where the right thumb was often placed above or adjacent to the button for firing the solid rocket booster, the left thumb was often placed quite some distance away from the primary button position – which in the case of the left yoke control was for the parachute. This would suggest that the space between the buttons is insufficient to give the drivers confidence that they could rest their thumbs there, and the limited number of observations of the right thumb in this position would reinforce this observation.

After these two phases of research we are comfortable that the overall yoke design is suitable for the application, but the arrangement of the buttons, and the haptic feedback from them, can be improved, and we have data to suggest the direction this improvement should take. Furthermore, we will continue to collect observational data of the evolving simulation and be as unstructured as possible to allow for Ethnographic data to lead us into areas the users perceive as useful.

References


Experimental Study of Steer-by-Wire Ratios and Response Curves in a Simulated High Speed Vehicle

Stefan Marks  Robert Wellington

School of Computing and Mathematical Sciences
AUT University,
2-14 Wakefield St, Auckland 1142, New Zealand,
Email: stefan.marks/robert.wellington@aut.ac.nz

Abstract
In this poster, we outline a research study of the steering system for a potential land speed record vehicle.

We built a cockpit enclosure to simulate the interior space and employed a game engine to create a suitable virtual simulation and appropriate physical behaviour of the vehicle to give a realistic experience that has a suitable level of difficulty to represent the challenge of such a task. With this setup, we conducted experiments on different linear and non-linear steering response curves to find the most suitable steering configuration.

The results suggest that linear steering curves with a high steering ratio are better suited than non-linear curves, regardless of their gradient.

Keywords: Yoke, Steering, High Cognitive Load

1 Introduction
The task for the research team was to create an environment to undertake research on the cockpit design of a Land Speed Record vehicle, this being inspired by the public launch of the New Zealand “Jetblack” land speed record project, and our growing awareness of numerous land speed record projects sprouting up around the globe.

Creating this environment elevates the sensitivity of the quality of the user interaction design significantly, and will allow us to trial and evaluate many designs and gather rich data. The aim of our research in collecting this data is targeted at developing theory rather than just evaluating a set of designs or undertaking a requirements gathering activity. We do intend to develop the simulation to be as close to the physical reality as possible, as the land speed record context provides something concrete for participants driving in the simulator to imagine, and a target context for participants to relate their experiences. Making this context explicit then provides a fixed reference point to combine the variety of experiences of the participants that have ranged from games enthusiasts, pilots, drag race drivers, and engineers, to general office staff and students.

2 Steering Design
The steering of a landspeed record vehicle is very different from a standard automobile. Instead of a standard steering wheel, a yoke is used for controlling the vehicle. The rotation range of the yoke is limited to about 90 to at most 180 degrees, since the pilot constantly has to keep both hands on it. A larger motion range would result in crossing arms or uncomfortable rotation angles of arm and hand joints. In addition, the maximum range of the steering angle of the front wheels of the vehicle is very limited as well. The vehicle is designed primarily to drive a straight course without any bends. In our simulation, we found that during most runs, the front wheels were rarely rotated more than ±1 degree.

While there is a significant body of research into vehicle control via steering wheels, yokes, and joy-sticks, e.g., (McDowell et al. 2007, Hill et al. 2007) in the context of military vehicles, we were not able to find any research output in the context of high-speed land vehicles such as Jetblack.

For the experiments, we implemented a steering module with two parameters: an adjustable yoke/front wheel transfer ratio, and an adjustable response curve. The steering module expects the yoke input as a value between -1 and 1 and translates it to an intermediate value in the same range by applying a simple power function with an adjustable exponent. An exponent of 1 results in a linear curve while higher exponents (e.g., 1.5 or 2) result in the nonlinear curves shown in Figure 1. The intermediate value is then multiplied by a factor that represents the steering ratio (e.g., 1:30 or 1:60), the ratio between the yoke input angle and the
front wheel output angle. As an example, for a yoke with a range of 90 degrees of rotation (±45 degrees), a 1:60 ratio would result in the front wheels being adjusted by 1.5 degrees (±0.75 degrees) for the full 90 degree movement.

3 Methodology

We implemented a vehicle simulator with a cockpit created from plywood, a Logitech G27 force feedback wheel and pedals, a large size projection screen, and a virtual simulation environment created with the Unity3D engine (see Figure 2). For the experiments, simulation-driven force feedback on the steering wheel was disabled, leaving only a medium spring force that would return the wheel to the centre position. We also removed the original steering wheel and replaced it by a yoke as discussed in Section 2.

The results presented in this poster were collected from three participants. To avoid the influence of the initial learning curve and distortion of the data by treating the simulation more like a game, we chose participants who were either familiar with the simulator and the serious nature of the experiment, or participants who had a background in driving real high-speed vehicles, e.g., New Zealand Drag Bike racers.

In total, we collected data of 60 runs with a mixture of the following steering configurations:

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Power</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1*</td>
<td>1 (linear)</td>
<td>1:20</td>
</tr>
<tr>
<td>2</td>
<td>1 (linear)</td>
<td>1:30</td>
</tr>
<tr>
<td>3</td>
<td>1 (linear)</td>
<td>1:45</td>
</tr>
<tr>
<td>4</td>
<td>1 (linear)</td>
<td>1:60</td>
</tr>
<tr>
<td>5</td>
<td>1.5 (quadratic)</td>
<td>1:45</td>
</tr>
<tr>
<td>6</td>
<td>2 (quadratic)</td>
<td>1:45</td>
</tr>
<tr>
<td>7*</td>
<td>3 (cubic)</td>
<td>1:45</td>
</tr>
</tbody>
</table>

The configurations were randomised and changed after every run. Configurations with an asterisk were only tested once or twice to test if the participants would notice such extreme values.

Data was logged for every timestep of the physical simulation which ran at 200 times per second. As a measure for the stability of a run, we evaluated the average of the lateral velocity of the vehicle during the acceleration phase at speeds above 500km/h. Above this speed, the randomised, simulated turbulences and side wind had a major effect on the stability of the vehicle, and therefore required most steering influence from the participants.

4 Results

The results are shown in Figure 3. We found that the linear response curves with a high steering ratio like 1:45 or 1:60 lead to the least amount of lateral velocity of the vehicle at speeds above 500km/h. The worst results were achieved using the quadratic or cubic response curve, even leading to crashes due to complete control loss of the vehicle.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>50% Inner Quartile Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.537 m/s</td>
</tr>
<tr>
<td>2</td>
<td>0.572 m/s</td>
</tr>
<tr>
<td>3</td>
<td>0.396 m/s</td>
</tr>
<tr>
<td>4</td>
<td>0.394 m/s</td>
</tr>
<tr>
<td>5</td>
<td>0.522 m/s</td>
</tr>
<tr>
<td>6</td>
<td>0.588 m/s</td>
</tr>
<tr>
<td>7</td>
<td>1.5611 m/s</td>
</tr>
</tbody>
</table>

An additional factor for the rejection of quadratic or higher exponents in the response curve is the fact that only for those configurations, the vehicle got out of control on some occasions while all runs with a linear curve were successful.

We are currently collecting more data with more participants to solidify the statistical significance of the data.

References


3D Object Surface Tracking Using Partial Shape Templates Trained from a Depth Camera for Spatial Augmented Reality Environments

Kazuna Tsuboi and Yuji Oyamada and Maki Sugimoto and Hideo Saito
Graduate School of Science and Technology
Keio University
3-14-1, Hiyoshi, Kohoku-ku, Yokohama, Kanagawa,Japan 223-8522
kazunat@hvr1.ics.keio.ac.jp

Abstract
We present a 3D object tracking method using a single depth camera for Spatial Augmented Reality (SAR). The drastic change of illumination in a SAR environment makes object tracking difficult. Our method uses a depth camera to train and track the 3D physical object. The training allows marker-less tracking of the moving object under illumination changes. The tracking is a combination of feature based matching and frame sequential matching of point clouds. Our method allows users to adapt 3D objects of their choice into a dynamic SAR environment.

Keywords: Spatial Augmented Reality, Depth Camera, 3D object, tracking, real-time, point cloud

1 Introduction
Augmented Reality (AR) visually extends the real environment by overlaying virtual objects on to the real world. Spatial Augmented Reality (SAR) aims to enhance the visual perception of the real environment by projecting virtual objects directly on to the real environment with a projector. The projection allows multiple users see the fusion with the naked eye which is effective for cooperative scenes like product designs.

An AR system becomes convincing when the geometric consistency between the real environment and the virtual object is kept in real time. This is achieved by real-time determination of the relation between the camera and the real environment. Solving this problem is the most fundamental and challenging task for AR research.

This paper presents a marker-less tracking method of a moving object under SAR environment using single depth camera.

2 Related Works
The illumination of a SAR environment changes drastically by the projected light. The common approach to avoid interference of light during the tracking is using sensors unaffected by illumination as in Dynamic Shader Lamps (Bandyopadhyay 2001). However, special sensors must be prepared and attached to target objects. Instead, colour camera and computer vision techniques were used to avoid physical sensors. Audet proposed a marker-less projector camera tracking system which analyse the position of the projecting image and the actual projection (Audet 2010), but is limited to planner surfaces. A depth camera is robust to illumination change and computer vision technique is applicable to the depth image.

A depth image can be converted to point clouds. The ICP algorithm (Besl & McKay 1992) can align point clouds by assuming that the corresponding points are close together and works in real time as in Kinect Fusion (Izadi. 2011). A feature based algorithm is also used which automatically finds the corresponding points with similar feature. It enables global matching and also gives a good initial registration for ICP algorithm (Rusu 2009).

3D objects have different views when perceived from a variety of viewpoints. Prior 3D object tracking methods commonly track objects by matching 3D model of the target with the input data. Azad matched the input colour image and a set of 2D images of the object from different views generated from the 3D model (Azad 2011). However, this conversion of the 3D model takes time and resources. By using depth camera, direct matching of the model and the input is possible.

The proposed method trains the shape of the object as point clouds and tracks the object using a depth camera. The method is illumination invariant and handles the view change by direct matching of the model and the input point cloud. The tracking is in real-time by matching point clouds combining the feature based algorithm and ICP algorithm.

3 Proposed method
The proposed method tracks a moving rigid object using single depth camera under SAR environments. The setup of the system is as shown in Figure 1. The system tracks the moving object in front of the camera. The virtual object is rotated and translated based on the result of tracking and projected onto the object. The proposed method can be divided into two step, off-line training which trains several point clouds of the object from different viewpoints and on-line tracking which is a combination of feature based and frame sequential matching.
3.1 Training of Partial Shape Templates
The point clouds of the object from different views are first trained. The relations between each template are labelled and also trained with the templates. These point clouds are called partial shape templates. This relationship is used to guess the new template when the object view changes and re-initialization is necessary. The trained point clouds are meshed to polygon model for surface projection. The tracking method only uses the point cloud of the templates to track the surface of the object.

3.2 Feature Based matching
The feature used in the feature based matching is FPFH (Rusu 2009) which describes the relationship between the point and the surrounding points. By using the feature based matching, all the templates are matched to the first input point cloud of tracking. The best matching template is chosen as the template of the current view. The chosen template cloud is rotated and translated to the position of the object. This result becomes the initial position of the following Frame Sequential Matching.

3.3 Frame Sequel Matching
Frame to frame matching starts from the initialized point cloud. The shape and the position of the point clouds are similar in the consecutive frame which can be easily matched by ICP algorithm. In order to track in real-time, Generalized ICP (Segal 2009) is used. The result of ICP algorithm, which are the rotation and translation of the consecutive frame, is calculated and applied to the template. The position of the template indicates the current position of the 3D object.

3.4 Re-initialization and Selection of New Template
The appearance of the object changes as the object rotates which eventually will not match the template currently used. A new template is chosen from the trained templates by analogizing the movement and the relation of templates. The template is initialized again to the current input point cloud and frame sequential matching starts again.

4 Experiment Result
We implemented a texture projection SAR application to test our method under illumination changing environment. The texture is augmented by projecting a surface polygon made from the trained templates to the position of the tracking result. The camera and the projector are pre-calibrated. The used devices are the following:

- Depth camera: Kinect
- CPU : Intel(R) Core(TM)i7-2600 3.40GHz
- Memory : 3.49GB

The result is shown in Figure 2. We projected yellow and blue texture on the object. The texture followed the object while it moved and turned. There was a small delay on the projection. The average tracking speed was about 10fps for this object.

5 Conclusion
We presented a 3D object tracking method for SAR application using a depth camera. The proposed method has off-line training and on-line tracking. Experiment result show that 3D objects were tracked in video rate speed and under illumination changing environment. For future extension, we will implement a application which user could change the texture of the object while tracking which would be useful for designing such as products.

6 References
My Personal Trainer - An iPhone Application for Exercise Monitoring and Analysis

Christopher R. Greeff1,Joe Yang1,Bruce MacDonald1
Burkhard C. Wünsche2

1 Department of Electrical and Computer Engineering
University of Auckland, New Zealand,
Email: chizzajt@gmail.com, lyan101@aucklanduni.ac.nz, b.macdonald@auckland.ac.nz
2 Graphics Group, Department of Computer Science
University of Auckland, New Zealand,
Email: burkhard@cs.auckland.ac.nz

Abstract

The obesity epidemic facing the Western world has been a topic of numerous discussions and research projects. One major issue preventing people from becoming more active and following health care recommendations is an increasingly busy life style and the lack of motivation, training, and available supervision. While the use of personal trainers increases in popularity, they are often expensive and must be scheduled in advance. In this research we developed a smartphone application, which assists users with learning and monitoring exercises. A key feature of the application is a novel algorithm for analysing accelerometer data and automatically counting repetitive exercises. This allows users to perform exercises anywhere and anytime, while doing other activities at the same time. The recording of exercise data allows users to track their performance, monitor improvements, and compare it with their goals and the performance of other users, which increases motivation. A usability study and feedback from a public exhibition indicates that users like the concept and find it helpful for supporting their exercise regime. The counting algorithm has an acceptable accuracy for many application scenarios, but has limitations with regard to complex exercises, small number of repetitions, and poorly performed exercises.

Keywords: accelerometer, activity monitoring, signal processing, exercise performance, fitness application, human-computer interfaces

1 Introduction

The worldwide prevalence of obesity has almost doubled between 1980 and 2008 and has reached an estimated half a billion men and women over the age of 20 (World Health Organization 2012). Exercises help to fight obesity, but are often not performed due to lack of motivation (American Psychological Association 2011). Motivation can be increased by enabling users to self-monitor and record exercise performance and to set goals. For example, an evaluation of 26 studies with a total of 2767 participants found that pedometer users increased their physical activity by 26.9% (Bravata et al. 2007).

In this research we present an iPhone based “personal trainer” application, which assists users with performing exercises correctly, self-monitoring them, and evaluating performance. A key contribution is a novel algorithm for counting repetitive exercises. This helps users to easily keep track of daily exercise data, and encourages them to perform simple exercises frequently, e.g., during work breaks and recreational activities.

2 Design

2.1 Software Architecture

The application is designed with a three tier architecture. The presentation tier of the application consists of two parts: The exercise view gives feedback to the user while performing exercises, whereas the data view is used for planning and monitoring exercises and provides instructions and visualisations of recorded data. The repetition counter generates information about user performance based on the selected exercise and an analysis of accelerometer data. The database manager is responsible for saving and retrieving information (e.g., educational material and exercise performance data).

2.2 User Interface & Functionalities

Users can choose a predefined exercise or define a new exercise. Each exercise has a short name used in the monitoring application, an image and/or video explaining the exercise, and a short description of it including where the iPhone should be attached to the body in order to record repetitive motions. After an exercise is selected a counting view is shown. The user must attach the iPhone to the body part to be moved during the exercise. Some exercises require a body position, which makes it cumbersome to press the Start button directly before an exercise. We hence added a short 5 second countdown during which the user can get prepared for the exercise. A beep sound indicates to the user that the counting application is ready.

2.3 Counting Algorithm

The number of repetitions of an exercise is determined using the algorithm illustrated in figure 1. The algorithm works satisfactory for exercises with smooth consistent motions (e.g., arm curls or side raises), but problems occur for exercises prone to shaking and jerking (e.g., lunges). Even after simplification with the Douglas Peucker algorithm, the data can contain large variations and sudden jumps which will result in inaccurate counting. We reduced this problem by...
additionally requiring that the cycle time (distance between peaks and valleys) is above 0.5 seconds and moderately regular, i.e., very short and very long cycles are not counted. This restriction is acceptable since an interview with an exercise therapist confirmed that exercises are most useful when performed with a smooth moderate motion.

3 Results

3.1 Methodology

We performed a user study involving 20 participants aged 16 – 60 years. Exercise frequency ranged from 1 – 2 hours to 5 – 10 hours of exercises per week. To test the counting algorithm of our application we asked participants to perform 5 – 10 repetitions of the following exercises: arm curls, side raises, push-ups, squats, lunges and a customised exercise that users defined themselves and added to the application. We observed the participants and counted the number of repetitions, and compared it with the number of repetitions recorded by the algorithm. We then computed the measurement accuracy as percentage variation from the actual count.

3.2 User Study Results

The overall accuracy, including customised exercises, was over 80%. The exercise with the highest accuracy, averaging roughly 95%, was the arm curl and the side raise. The exercise with the lowest accuracy, averaging roughly 55%, was the lunge exercise. The push up, squat and custom exercise averaged roughly 80%. A closer examination of the results revealed that two problems occurred: The counting algorithm often had problems detecting the first and last repetition of an exercise, especially for exercises containing complex motions, or exercises which were physically difficult. As a consequence the measured count was frequently 1-2 repetitions too low. For exercises, such as push-ups, where some users achieved only 5 repetitions, this resulted in an error of 20-40%. The second problem was related to the smoothness of the performed exercise. The algorithm works best for simple motions where users can easily get into a “rhythm”. In such cases an accuracy of 98% was achieved.

Overall users were satisfied with the design and information content of the application. Users regarded the application as only moderately useful, but slightly more than half of the participants could imagine to download and use the application, if available. Subsequently we presented the application at a public display in the university. The visitor feedback was overwhelmingly positive and several visitors were keen to buy the application on the Applet app store.

4 Conclusion & Future Work

We have presented a novel iPhone application assisting users with getting physical active by providing information on simple exercises, and automatically recording exercise performance. A key contribution is a novel algorithm for analysing accelerometer data in order to detect the number of repetitions in an exercise performance.

A user study confirmed that the algorithm is satisfactorily accurate for simple smooth motions and high number of repetitions. Problems exist for complex motions, exercises with a very low number of repetitions, and exercises performed with jerky and irregular motions.

More work needs to be done to make the presented prototype useful in practice, and in particular to achieve behavioral change. The counting algorithm needs to be improved to make it work for a larger range of exercises, and to make it more stable with regard to low number of repetitions and irregular and jaggy motions. Usability and motivation could be improved by adding voice activation and feedback. A controlled long term study is necessary to measure behavioural change, such as more frequent or longer exercises when using the application.

References


Figure 1: The steps of the counting algorithm: (a) Raw accelerometer data of a user performing ten arm curls with the iPhone strapped to the lower arm. (b) Acceleration coordinate with the largest standard deviation. (c) Data after simplification with the Douglas Peucker algorithm and computing the mean value and standard deviation of its sample points. The number of repetitions is computed as the number of cycles with sections below the threshold ($\mu - \sigma$) followed by a section above the threshold ($\mu + \sigma$). The above graph represents 10 repetitions of arm curls.
Interactive vs. Static Location-based Advertisements

Moniek Raijmakers\textsuperscript{1}, Suleman Shahid\textsuperscript{1}, Omar Mubin\textsuperscript{2}
\textsuperscript{1}Tilburg University, The Netherlands
\textsuperscript{2}University of Western Sydney, Australia
\{s.shahid@uvt.nl, o.mubin@uws.edu.au\}

Abstract

Our research is focused on analysing how users perceive different mediums of advertisements on their mobile devices. Such advertisements are also called location-based advertisements (LBA’s) as they relay brand and product information to mobile phones that are in the vicinity. We investigated two different ways of presentation marketing information (static vs. interactive). Our results clearly showed that interactive (clickable advertisement with additional information) LBAs were preferred to static LBAs.

Keywords: Location-based ads, mobile commerce

1 Introduction

Location Based Advertising is a new form of marketing communication that uses location-tracking technology in mobile networks to target consumers with location-specific advertising on their mobile phones (Unni & Harmon, 2007). Because of the mobility that these devices have nowadays, advertisements can be personalized for specific consumers and sent to them based on their geographical location (Gratton, 2002).

Previous research into using LBA for marketing messages are rather scarce and have focused on technological issues, for instance the research by Ververidis and Polyzos (2002) who developed a software prototype and an information system for LBA. Other related research has focused on the success or acceptance of LBA by consumers as compared to traditional media. Heinonen and Strandvik (2003) found that consumers are open to this new form of advertising but their research showed lower responses towards LBA than towards traditional media.

One perspective that has not yet been used often in the research into LBA is the design and amount of information presented in such advertisements. The mobile advertisements that most of today’s smartphone users know consist of a one-page screen. Almost all of these LBAs are non-interactive and on one page all the information is given about the promoted item. Schrum, Lowrey and Liu (2009) explain that when consumers view a banner ad on a website, they can, depending on the level of interest, either ignore the advertisement completely, notice and view the advertisement without taking any further steps or they can click on the advertisement to access a deeper layer of information.

Research by Liu & Schrum (2009) has shown that interactivity in marketing is considered to have a positive influence on persuasion. Their research showed that in case of low task evolvement, the mere presence of interactivity served as a peripheral cue that led to more positive attitudes. In LBA, interactivity can be implemented by deepening the levels of information that is accessible. Consumers that have higher levels of interest in the advertised product can gather more information while only viewing the ad, before taking further steps towards purchasing the product. From a design perspective, making LBAs more interactive would mean making them dynamic e.g. having a more active design to accommodate buttons or links which could lead to additional information within the advertisement. Research of Liu & Schrum suggests that the presence of interactivity and deeper levels of information attracts more interest than static pages. Schrum, Lowrey and Liu (2009) indicate that personalized and customised marketing content is appreciated by consumers because they have access to more information and they can choose what is important or relevant for them. This leads to the following hypothesis:

H1: The more information LBAs contain, the more persuading they will be.

H2: Interactive LBAs are rated higher on overall liking than static Location Based Advertisements (LBAs).

2 Method

For this study, the product category of fast food (food chain: Subway®) was chosen as the domain of research. The stimuli that were used in the experiment were Location Based Advertisements for four different types of Subway sandwiches. These advertisements were designed for an iPhone. Of every sandwich, two advertisements were created: One static advertisement (one page) and one dynamic advertisement (see Figure 1).

Presenting the information in the new interactive LBAs can be done in different ways. Research by Schaffer et al. (1996) suggests that for websites and other more intricate interfaces, a fisheye style of navigation is preferred over a full zoom style (full page) of navigation. The fisheye style zooms in on a small part of the interface, leaving the context and structure visible. Translated into the LBAs in this study, the fisheye style would result in a small pop up screen, leaving the rest of the advertisement visible in the background.

In our experiment, the interactive advertisement (pop-up), unlike the static one, displayed nutritional information such as what is on the sandwich and how many calories, fat, carbohydrates and proteins it contains. In the case of the interactive advertisement, clicking on the sandwich resulted in a pop-up window with more information about that sandwich. The target group
consisted of students and young professionals between 18 and 35 years old. A total of 20 participants participated in the experiment and they were selected after a survey where their experience with smartphones, fast food products and knowledge of the brand Subway was measured.

The randomized experimental study had a within subject design: half of the participants saw the static version first and other half saw the interactive version first. In order to make the interaction with the LBAs seem as natural as possible, participants were presented with a scenario e.g. it is around lunchtime while you are out for a day of shopping. After this scenario, different LBAs were presented to them on an iPhone. A questionnaire, consisting of four variables: Professionalism/Trust, Information Level, Overall Liking and Purchase Intention, was used to measure the acceptance and appreciation of LBA’s. At the end, a semi-structured interview was conducted.

![Image](image.jpg)

**Figure 1: Two visualization techniques**

### 3 Results

The data of the experiment have been analysed by comparing the means scores of different variables using a paired sample T-test. All details of results are presented in the table 1. The results clearly show that for all categories the interactive pop-up advertisement was preferred over the static one.

![Table](table.png)

**Table 1: Mean, p and t values for four measurements**

### 4 Discussion and Conclusion

In general, interactive LBAs were considered to be more professional, trustworthy, and informative than static LBAs. The overall liking for interactive LBAs was clearly higher than the static LBAs. For the participants purchase intentions, the pop up version showed a significant and very high increase between static and interactive with a very large effect size. This also confirms the first hypothesis. During the interviews, participants also stated they were very pleased with the possibility to read more about the product if they so desired than simply see a picture, a name and a price. These results are in line with Schrum, Lowrey and Liu (2009), who have indicated that the possibility to browse through more information creates a more personalized and customized marketing content. We were also able to confirm the second hypothesis. Participants liked the navigation options for the dynamic LBAs and stated that it made the advertisement more attractive. Not only did the participants find the dynamic LBAs more likeable, they also found them to be more informative, intelligible and professional. The confirmation of this hypothesis is congruent with the research of Liu and Schrum (2009) that indicates that the mere presence of interactivity attracts more interest than the static pages.

Our research has confirmed the hypothesis that interactive LBAs are preferred over static ones and most importantly additional information in the form of a pop-up screen enhances purchase intention. In the future we would like to extend our research by investigating LBA’s for products external to the food industry e.g. clothing. We would also like to explore different visualization techniques for showing additional information. We would also like to clarify that essentially our findings could also apply to in general any mobile displays and not necessarily location based advertisements. In addition, a limitation of our study was that it was “lab-based” and not conducted in a real outdoor environment.

### 5 References


Temporal Evaluation of Aesthetics of User Interfaces as one Component of User Experience

Vogel, M.
Berlin Institute of Technology, Germany
Research training group prometei
Franklinstraße 28/29, 10587 Berlin, Sekr. FR 2-6, Germany
marlene.vogel@zmms.tu-berlin.de

Abstract
User experience (UX) is gaining more and more relevance for designing interactive systems. But the real character, drivers and influences of UX are not sufficiently described until now. There are different theoretical models trying to explain UX in more detail, but there are still essential definitions missing regarding influencing factors i.e. temporal aspects. UX is increasingly seen as a dynamic phenomenon, that can be subdivided in different phases (Pohlmeyer, 2011; Karapanos, Zimmerman, Forlizzi, & Martens, 2009, ISO 9241-210). Trying to gain more knowledge about temporal changes in UX, an experiment was conducted examining the influence of exposure on the evaluation of aesthetics as one hedonic component of UX. A pre-use situation was focused including an anticipated experience of the user and no interaction was accomplished. It could be found that a repeated mere-exposure (Zajonc, 1969) does significantly influence the evaluation of aesthetics over time.

Keywords: Mere-Exposure Effect, Dynamics of User Experience, Evaluation, Aesthetics.

1 Introduction
User Experience (UX) is a highly complex phenomenon. There are quite a few models (e.g. Hassenzahl, 2003, Mahlke & Thüring, 2007) trying to explain the character of UX by defining several components (i.e. instrumental and non-instrumental product qualities and emotional reactions) as well as influencing factors, e.g. user characteristics, system properties and task/context related aspects. But in the last few years UX is increasingly considered as a more dynamical experience that is influenced by time and memory (Karapanos et al., 2009, 2010). Pohlmeyer (2011) proposed the theoretical model ContinUE, where she defined several phases of UX. It is assumed that the user experience already takes place before a real interaction is performed (ISO 9241-210, 2010). The pre-use phase is related to an anticipated experience that is mainly based on prior experience and the attitude of the user towards the system or brand. But how is the experience influenced by time related aspects, i.e. exposure rate? An effect that can be related to time sensitive changes of evaluations of stimuli is the mere-exposure effect (MEE). The MEE was defined by Zajonc (1968) and intensively investigated in the late 60ies, 70ies and 80ies. Bornstein (1989) published a meta-analysis describing several parameters for mere-exposure experiments and related effects, but non related to the HCI context.

2 Experiment
2.1 Objective
The present experiment investigated the influence of mere-exposure on the evaluation of aesthetics of user interfaces within a pre-use situation. It was assumed that there is an influence of the exposure rate on the evaluation of aesthetics over time similar to the effects that Grush (1976) could identify in his study. The aesthetics of interfaces were manipulated twofold (high & low). We expected an increase in the aesthetical evaluation for the more beautiful target stimulus and a decrease for the evaluation of the less beautiful interface.

The manipulation of aesthetic and exposure rate were conducted as within variables. Reaction time and subjective evaluation of aesthetics were dependent variables. Additionally, previous experience with smart phones was included as control variables into the analysis.

2.2 Participants and Stimulus Material
Thirty-one people (age: M=26.9, SD=5.4, 13 female/18 male) participated voluntarily in the experiment and were rewarded with ten Euros/hour. All subjects were right-handed and had good German language skills. The interfaces shown in fig. 1 represent the two versions of aesthetic manipulation.

Fig. 1a: low aesthetic (A-) 1b: high aesthetic (A+)

In a pilot survey the yellow interface (fig. 1a) was evaluated significantly uglier (A-) than the black one (A+) (fig. 1b) on a 7-point Likert-scale. These two
interfaces were used as target stimuli which were presented thirty times each. They were randomized in 6 blocks each consisting of five A+ stimuli, five A- stimuli and seven filler items. Fillers looked analogue. They differed in colour, orientation of buttons or existence of keypad only and were presented one time each. Everything was presented on a 15” computer display.

2.3 Procedure
First subjects had to read and sign a letter of agreement after arriving at the lab. Subsequently the experiment started with a short instruction and the participants had to practice the usage of the single-item scale by evaluating two sample items (high vs. low aesthetics of non-HCI related objects). After that, the actual experimental block started. Participants had to evaluate 102 pictures of interfaces, fillers and target stimuli were randomized presented in 6 blocks. Each interface was presented until participants evaluated it by keystroke. The time was measured as reaction time. Between two stimuli a fixation cross appeared for two seconds. Finally, participants had to fill in several questionnaires: demographic data, previous experiences with smart phones, TA-EG (technical affinity) questionnaire and CVPA (Centrality of Visual Product Aesthetics) questionnaire.

3 Results
One multivariate analyses of variance with repeated measures (MANOVA) was computed. Aesthetic manipulation (high & low) of the target stimuli served as independent variable and entered the analysis as within subjects’ factors.

3.1 Subjective Aesthetics Evaluation
A significant influence of exposure rate on the evaluation of aesthetics could be detected ($F(29,725)=2.01$, $p=0.001$, $\eta^2_{PART} = 0.08$). Additionally a significant interaction of aesthetics and exposure rate ($F(29,725)=2.08$, $p=0.001$, $\eta^2_{PART} = 0.08$) (see fig. 2) could be found.

![Fig. 2 Evaluation of aesthetics for target stimuli using a single-item scale ranging from 1= ugly to 7= beautiful and an exposure rate of 30 times for each target stimulus.](image)

3.2 Reaction Time
For reaction time (in ms) a significant influence of exposure rate could be detected within a pair wise comparison between 1st and 30th exposure rate ($p< 0.001$, $SE = 401.47$). There was no significant effect of the aesthetics manipulation on the reaction time and no significant interaction occurred either.

4 Discussion
The results show an influence of mere-exposure on the evaluation of aesthetics of user interfaces over time and demonstrate the dynamical character of the UX phenomenon once more. Hence, the evaluation of aesthetic is not a static impression the user gets once. Thereby we have to be aware that the perception of hedonic aspects of UX can change within the user experience lifecycle. Therefore, it’s difficult, if we want to evaluate UX and only measure momentary experience. This experiment shows, that just by increased exposure the perception of and the attitude towards stimuli can be influenced.

5 Conclusion
It is questionable if this effect can influence the perception of other UX components i.e. usability and emotional reactions, too. Everyone who is dealing and designing for UX should be aware of the highly dynamical character of this concept and should include time aspects in their concepts of newly developed products and services. It is still questionable how long this effect will last, more research is needed.

6 References


Author Index

Amor, Robert, 73
Bowen, Judy, 81
Chow, Jonathan, 73
Churcher, Clare, 23
Daradkeh, Mohammad, 23
Dekeyser, Stijn, 33
Dhillon, Jaspaljeet Singh, 53
Feng, Haoyang, 73
Greeff, Christopher, 127
Irlitti, Andrew, 63
Lutteroth, Christof, 53, 91, 111
MacDonald, Bruce, 127
Maher, Mary Lou, 43
Marks, Stefan, 121, 123
McKinnon, Alan, 23
Mehrabi, Mostafa, 91
Mubin, Omar, 129
Oyamada, Yuji, 125
Peek, Edward M., 91, 111
Pilgrim, Chris, 101
Plimmer, Beryl, 13
Raijmakers, Moniek, 129
Reeves, Steve, 81
Saito, Hideo, 125
Schweer, Andrea, 81
Shahid, Suleman, 129
Singh, Nikash, 43
Smith, Ross T., iii
Sugimoto, Maki, 125
Sutherland, Craig J., 13
Thomas, Bruce H., 3
Tomitsch, Martin, 43
Tsuboi, Kazuna, 125
Vogel, Marlene, 131
Von Itzstein, Stewart, 3, 63
Wünsche, Burkhard C., iii, 53, 73, 91, 111, 127
Walsh, James A., 3
Watson, Richard, 33
Wellington, Robert, 121, 123
Yang, Joe, 127
Volume 113 - Computer Science 2011
Contains the proceedings of the Thirty-Fourth Australasian Computer Science Conference (ACSC 2011), Perth, Australia, 17-20 January 2011.

Volume 114 - Computing Education 2011
Contains the proceedings of the Thirteenth Australasian Computing Education Conference (ACE 2011), Perth, Australia, 17-20 January 2011.

Volume 115 - Database Technologies 2011
Contains the proceedings of the Twenty-Second Australasian Database Conference (ADC 2011), Perth, Australia, 17-20 January 2011.

Volume 116 - Information Security 2011
Edited by Colin Boyd, Queensland University of Technology, Australia and Josef Pieprzyk, Macquarie University, Australia. January 2011. 978-1-920682-96-5.
Contains the proceedings of the Ninth Australasian Information Security Conference (AISC 2011), Perth, Australia, 17-20 January 2011.

Volume 117 - User Interfaces 2011
Contains the proceedings of the Twelfth Australasian User Interface Conference (AUIC2011), Perth, Australia, 17-20 January 2011.

Volume 118 - Parallel and Distributed Computing 2011
Contains the proceedings of the Ninth Australasian Symposium on Parallel and Distributed Computing (AusPDC 2011), Perth, Australia, 17-20 January 2011.

Volume 119 - Theory of Computing 2011

Volume 120 - Health Informatics and Knowledge Management 2011
Edited by Keryn Butler-Henderson, Curtin University, Australia and Tony Sahama, Queensland University of Technology, Australia. January 2011. 978-1-921770-00-3.
Contains the proceedings of the Fifth Australasian Workshop on Health Informatics and Knowledge Management (HIKM 2011), Perth, Australia, 17-20 January 2011.

Volume 121 - Data Mining and Analytics 2011
Edited by Peter Vamplew, University of Ballarat, Australia, Andrew Stranieri, University of Ballarat, Australia, Kok-Leong Ong, Deakin University, Australia, Peter Christen, Australian National University, and Paul J. Kennedy, University of Technology, Sydney, Australia. December 2011. 978-1-921770-02-9.
Contains the proceedings of the Ninth Australasian Data Mining Conference (AusDM'11), Ballarat, Australia, 1-2 December 2011.

Volume 122 - Computer Science 2012
Edited by Mark Reynolds, The University of Western Australia, Australia and Bruce Thomas, University of South Australia. January 2012. 978-1-921770-04-3.
Contains the proceedings of the Thirty-Fifth Australasian Computer Science Conference (ACSC 2012), Melbourne, Australia, 30 January – 3 February 2012.

Volume 123 - Computing Education 2012
Edited by Michael de Raadt, Mooloolaba Pty Ltd and Angela Carbone, Monash University, Australia. January 2012. 978-1-921770-05-0.
Contains the proceedings of the Fourteenth Australasian Computing Education Conference (ACE 2012), Melbourne, Australia, 30 January – 3 February 2012.

Volume 124 - Database Technologies 2012
Contains the proceedings of the Twenty-Third Australasian Database Conference (ADC 2012), Melbourne, Australia, 30 January – 3 February 2012.

Volume 125 - Information Security 2012
Contains the proceedings of the Tenth Australasian Information Security Conference (AISC 2012), Melbourne, Australia, 30 January – 3 February 2012.

Volume 126 - User Interfaces 2012
Edited by Haifeng Shen, Flinders University, Australia and Ross T. Smith, University of South Australia. January 2012. 978-1-921770-07-4.
Contains the proceedings of the Thirteenth Australasian User Interface Conference (AUIC2012), Melbourne, Australia, 30 January – 3 February 2012.

Volume 127 - Parallel and Distributed Computing 2012
Edited by Jianjun Chen, University of Technology, Sydney and Rajiv Ranjan, CSIRO ICT Centre, Australia. January 2012. 978-1-921770-08-1.
Contains the proceedings of the Tenth Australasian Symposium on Parallel and Distributed Computing (AusPDC 2012), Melbourne, Australia, 30 January – 3 February 2012.

Volume 128 - Theory of Computing 2012
Contains the proceedings of the Eighteenth Computing: The Australasian Theory Symposium (CATS 2012), Melbourne, Australia, 30 January – 3 February 2012.

Volume 129 - Health Informatics and Knowledge Management 2012
Contains the proceedings of the Fifth Australasian Workshop on Health Informatics and Knowledge Management (HIKM 2012), Melbourne, Australia, 30 January – 3 February 2012.

Volume 130 - Conceptual Modelling 2012
Contains the proceedings of the Eighth Asia-Pacific Conference on Conceptual Modelling (APCCM 2012), Melbourne, Australia, 31 January – 3 February 2012.

Volume 134 - Data Mining and Analytics 2012
Edited by Yanchang Zhao, Department of Immigration and Citizenship, Australia, Juyong Li, University of Southern Australia, Australia and Peter Christen, Australian National University, Australia. December 2012. 978-1-921770-14-2.
Contains the proceedings of the Tenth Australasian Data Mining Conference (AusDM'12), Sydney, Australia, 5-7 December 2012.