Aesthetics Matter: Leveraging Design Heuristics to Synthesize Visually Satisfying Handheld Interfaces

Yeonsoo Yang
Stanford University
353 Serra Mall Stanford, CA 94305
USA
Toshiba Corporation
1, Komukai-Toshiba-cho Saiwaiku,
Kawasaki 212-8582 Japan
yyang1@cs.stanford.edu
yeonsoo.yang@toshiba.co.jp

Scott R Klemmer
Stanford University
353 Serra Mall Stanford, CA 94305
USA
srk@cs.stanford.edu

CHI 2009, April 4 – 9, 2009, Boston, MA, USA
ACM 978-1-60558-246-7/09/04.

Abstract
We present a tool for automatically generating UI layouts for handheld devices based on design principles. This tool introduces a gestalt approach to visual interface design rather, complementing prior work on user cost minimization. We aim to increase user satisfaction using this approach. The tool automatically generates size and position of widgets drawn from the UI design heuristics of simplicity, structuring, and proportion. *Simplicity* refers to excluding non-core functionality; *structuring* to contextual grouping, and *proportion* to best-practice geometric ratios of width, height, and spacing. Layouts are generated from device constraints and simple XML containing UI component hierarchy. These layouts can be directly manipulated using a GUI editor.

Keywords
GUI layout, automatic interface generation, design tool.

ACM Classification Keywords
H5.2. Graphical User Interfaces, Screen design: User Interfaces.

Introduction

A fundamental challenge of handheld interaction design is the heterogeneity of screen sizes and physical input transducers. Because screens are small, interface elements are competing for scarce resources. This creates a tension: successful design requires careful planning –one would like to be able to precisely specify a layout – yet screen-size diversity makes this intractable. Designers could manually enumerate all options, which may be feasible once. But the overhead of changing many interface instances for each small functionality change (or new device introduction) would significantly impede interface revision, which is critical. This paper introduces a technique that mediates this tension by automatically generating user interfaces from high-level design principles.

To date, automatic interface generation tools have focused on manual and perceptual performance [13, 10], often at the expense of aesthetic considerations. This paper addresses the aesthetic shortcomings of this prior work by applying best-practice design heuristics. We have manifested these ideas in Aesther design tool. Aesther's library includes three principles: simplicity, structuring, and proportion.

Related Work

Research on automated user interface generation began in the 1980s. Early model-based approaches faced limitations on the application types that could be generated and poor usability [4]. We limit the UI generation problem into focusing on design-time layout result pursuing automation from a simple UI component hierarchy instead of generating run-time functional UI code from a complicated description for any particular software architecture.

The SUPPLE system introduced an optimization-based approach to automatically generating interfaces for heterogeneous devices [13]. While this research significantly advanced the state of the art, its exclusive reliance on manual and perceptual models for cost minimization often produces complicated, asymmetric layout for handheld devices.

Personal Universal Controller generates PocketPC GUIs from specifications of appliance functionality using a rule-based approach [10]. Its design principle is consistency among different appliances so performance is higher than the original controller that shipped with that appliance. It takes *insufficient space problem solving approach* to fix layout which makes it unpleasing.

Principles of UI Generation

We were inspired by Mullet and Sano's conviction that "Good design defuses the tension between functional and aesthetic goals" [12]. The influence of their writing on this work was also more direct: the Aesther prototype generates interface layouts using three of the book's visual interface design principles. These principles were in turn inspired by the Gestalt theory of perception. Gestalt theory is often referred to when speaking about the principles of visual interface design. Elements of Gestalt theory such as balance/symmetry, simplicity, and proportion in web page design improve user satisfaction [7].

Simplicity

We use this principle to exclude non-core functionality based on XML UI descriptions and to assign the same size to the same type of widgets. According to Fisher and Smith-Gratto (1998–99) similar objects will be

counted as the same group and structuring can be used to draw a viewer's attention [7].

Structuring
Viewers will mentally organize closer elements into a coherent object, because they assume that closely spaced elements are related and those further apart are unrelated (Fulks 1997, Fultz 1999) [7].We use this principle to position GUI components and the spacing between them based on UI hierarchies.

Proportion
Proportion is the metric that guides the choice of scales in a contrast relationship [12]. We use this principle to put margin space around the whole UI.

Tool Usage

GUI design is performed iteratively in several stages. The GUI layout generation we present here is not specific to a particular software infrastructure, but provides all the information necessary to produce UI layout design. The Aesther framework can adapt to any kind of software infrastructure although it currently only supports mainstream XML-based infrastructures. The primary functionality is as follows.

Choose target device
We assume that users have both target devices and an application in mind before beginning the design process. Users select the target device type from the menu. That selection produces information about device constraints, including display size and input method.

Provide UI description
Good GUI design should reflect information architecture and content. Based on information design, a user might gain an idea of the UI component hierarchy and core-functionality. Users describe this information with XML. In the text editing space, users list component types in order, followed by their labels. The parent and child relationship reveals the hierarchy of UI components. Users who have used HTML will easily adapt to Aesther.

Add additional devices
Users can adapt their designs to different devices. By selecting successive target devices, users can continue the GUI design process. This feature provides consistency between layouts for different target devices.

Export layout
GUI layout results can be finalized by exporting to a particular format. Aesther currently only allows export to MXLM format.

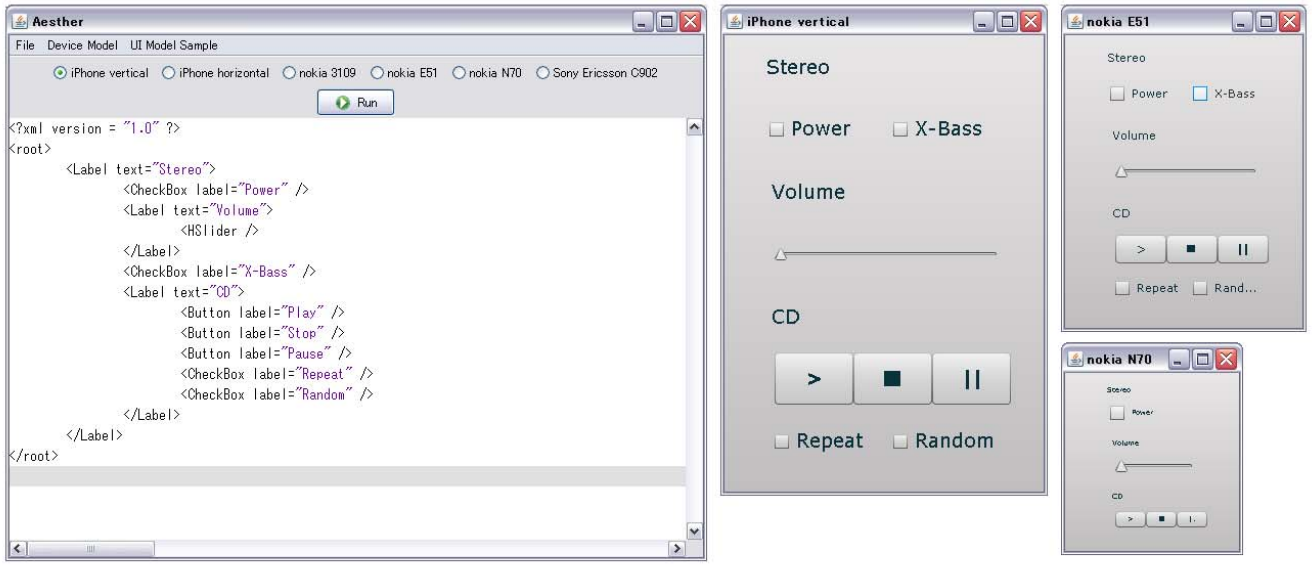

Figure 1. UI description in Aesther and its generation

Implementation

Aesther is written in Java to maximize portability. Aesther lets user describe UI models in XML. We chose a simple UI description scheme for *low threshold* of use instead of requiring users to write a UI concept model. To parse XML we used the Java W3C DOM API. To render GUI layout results we used the VersaEdge Software JFlashPlayer framework and the Adobe mxmlc compiler [1, 11]. We tested our tool in Windows Vista on a Pentium D at 3GHz, with 2GB of RAM.

Evaluation

There are two aspects to our evaluation: effectiveness of tool usage and satisfaction levels with UI layouts. As a first step we evaluated Aesther generated UI layouts by performing a screen shot comparison with an equivalent UI generated by SUPPLE. In user tests we used Amazon's Mechanical Turk [2]. If special care is taken Mechanical Turk tasks can reduce the time and monetary cost of user studies with reasonable results [3]. Questions were drawn from the QUIS [15]. Out of six questions on a 9-scale, 3 questions are for overall reaction and 3 questions are for specific reactions to the consistent positioning of messages on screen, the ease with which characters can be read, and the clear organization of components. Figure 2 shows screen shots of the images that were used in the task.

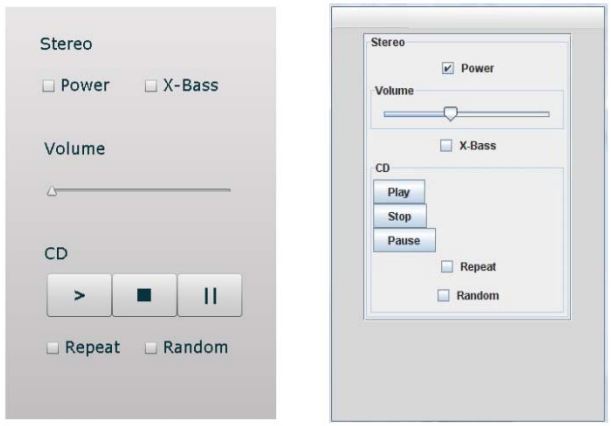

Figure 2. Screenshot used in comparison task. Left side is result of Aesther and right side is result of SUPPLE.

We filtered 40 responses down to 30 by deleting responses that took less than 60 seconds working time and including unanswered questions. As shown in Figure 3 both layouts scores were very close. In four out of six questions however, Aesther's were slightly higher. Using a T test we found out that the number of users was not large enough. The P values of the 6 questions were 0.38, 0.22, 0.16, 0.02, 0.19 and 0.04, respectively. Preferences were divided into 3 groups: 12 subjects preferred the Aesther's layout to SUPPLE's, 11 subjects preferred SUPPLE's to Aesther's, 7 subjects had no significant preference; the difference was less than 1 (Figure 4). This study made us wonder whether UI layout is a personal preference and cannot be generalized.

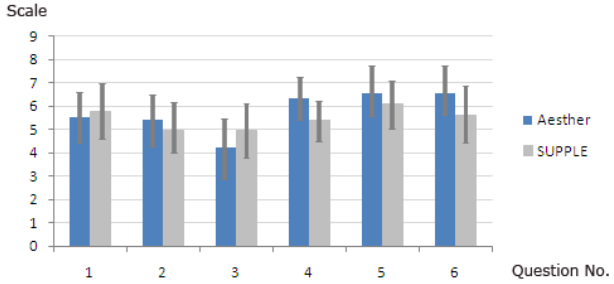

Figure 3. Aesther received slightly higher satisfaction scores.

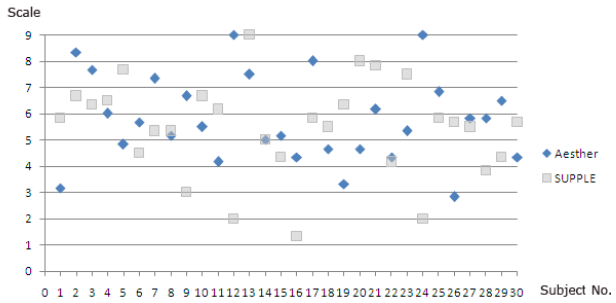

Figure 4. Personal preference is revealed in average of six questions.

Conclusion and Future Work

Based on the assumption that aesthetics affects usability we developed the tool, Aesther, for the handheld device application domain. Aesther automatically generates a UI layout using aesthetic principles as a metric rather than task performance. We estimated user satisfaction levels as a first step by performing a screen shot comparison with SUPPLE. Compared to SUPPLE, which only focused on task performance, Aesther received slightly higher scores.

In the study to keep the same component number and type as SUPPLE we were unable to leverage all of the aesthetic principles we have. To see how aesthetics affect usability a different user test treatment might be needed within Aesther. We plan to prepare several types of application examples which gradually introduce aesthetic principles. This study will produce proven UI layout principles for the handheld application domain. UI layouts should consider not only visual design effects but also task performance. We do not think this is a contradiction and a combination of these metrics might be a further possible research direction.

References

[1] Adobe Flex 3. http://www.adobqe.com/products/flex/.

[2] Amazon Mechanical Turk. https://www.mturk.com/mturk/welcome

[3] Aniket Kittur, Ed H. Chi, Bongwon Suh, Crowdsourcing user studies with Mechanical Turk. In Proc. CHI 2008, ACM Press (2008), 453-456.

[4] Brad Myers, Scott E. Hudson, Randy Pausch. Past, Present, and Future of User Interface Software Tools. *TOCHI Vol.7*, ACM Press (2000), 3-28.

[5] Cameron Moll. *Mobile Web Design*. Cameron Moll Publisher, Salt Lake City, UT, USA, 2007.

[6] D.C.L Ngo, L.S. Teo and J.G. Byrne. Formalizing guidelines for the design of screen layouts. *Display 21*, Elsevier Science Publishers (2000), 3-15.

[7] Dempsey Chang, Laurence Dooley and Juhani E. Tuovinen. Gestalt Theory in Visual Screen Design – A New Look at an Old Subject. *In Proc. CRPIT 2002*, Australian Computer Society, Inc. Press.

[8] James Fogarty, Scott E. Hudson. GADGET: a toolkit for optimization-based approaches to interface and display generation. *In Proc. UIST 2003*, ACM Press (2003), 125-134.

[9] James Lin, James A. Landay. Employing Patterns and Layers for Early-Stage Design and Prototyping of Cross-Device User Interfaces. *In Proc. CHI 2008,* ACM Press (2008), 1313-1322.

[10] Jeffrey Nichols, Duen Horng Chau, Brad A. Myers. Demonstrating the viability of Automatically Generated User Interfaces. *In Proc. CHI 2007*, ACM Press (2007), 1283-1292.

[11] JFlashPlayer. http://www.jpackages.com/jflashplayer/

[12] Kevin Mullet, Darrell Sano. *Designing Visual Interfaces*. SunSoft Press, Mountain View, CA, USA, 1995.

[13] Krzysztof Gajos, Daniel S. Weld. SUPPLE: Automatically Generating User Interface. In Proc.IUI 2004, ACM Press (2004), 93-100.

[14] Scott Jenson. *The Simplicity Shift*. Cambridge University Press. Cambridge CB2 2RU, UK, 2002.

[15] Questionnaire for User Interaction Satisfaction (QUIS). http://lap.umd.edu/quis/