Human–Computer Interaction

Fundamentals and Practice

Gerard Jounghyun Kim

CRC Press
Taylor & Francis Group

Human–Computer Interaction

Fundamentals and Practice

Human–Computer Interaction

Fundamentals and Practice

Gerard Jounghyun Kim

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20141208

International Standard Book Number-13: 978-1-4822-3390-2 (eBook - PDF)

Contents

Preface

Human–computer interaction (HCI) is becoming ever more important in interactive software. Such software has long been evaluated in terms of the availability and breadth of its functions and its algorithmic efficiency. While such a developer's perspective is still somewhat valid, it has become difficult to differentiate among similar software components from such an aspect given the amazing computing performance of today's hardware and the spread of algorithmic knowledge and systems development know-how. Thus software quality is increasingly judged from the users' external point of view in terms of their expectations, satisfaction, and experience. This external view or user experience may be defined in many ways, but it is most obvious that it has quite a lot to do with how the software users interact with it and, hence, its design. HCI will become even more critical as everything around us becomes digital and unknowingly embedded with interactive computing services that make our everyday lives more exciting, efficient, and convenient.

Therefore, software (at least software that is highly interactive and targeted for a high number of users) must now be developed with HCI as one of its higher priorities. However, at the undergraduate level, it is still often the case that HCI is not given the attention it deserves in the education of future software developers. Most entry-level HCI textbooks are structured around high-level concepts and guidelines

and are not directly tied to the software development process. Some of these books may offer design patterns, but students at the undergraduate level might still find it puzzling as to how HCI fits in with their basic software development knowledge. In fact, most of the HCI concepts and guidelines are fairly commonsense or very easy to comprehend. (After all, how difficult would it be to make one understand that users are important?) But it is in the practice and within the context of actual development that one has to make the difficult choices to produce highly usable interactive software.

Following this line of thinking, this book was designed around the overall development cycle for an interactive software product. It starts with the required basic HCI knowledge, which is kept as compact as possible by including only the basic essentials (Chapters 1–3). The intention is to convey the spirit of HCI rather than a long list of compiled knowledge. The book then moves into the application of this knowledge by iteratively forming the HCI requirements and modeling the interaction process (Chapter 4), designing the interface (Chapter 4), implementing the resulting design (Chapters 5–7), and finally evaluating the implemented product (Chapter 8). The book is targeted mainly at undergraduate students of computer science and information technology (IT), but it is easy enough to be taken up by readers in other fields. Some knowledge of computers and programming would be desirable, but it is not absolutely necessary. (Those not interested in the detailed aspects of implementation can skip some of Chapters 5–7.)

The core content of the book is based on the introductory undergraduate HCI course (advanced junior or senior level) that I have taught since 2006 at Korea University. The following table shows how one might structure a similar course using this book (or pace oneself for self-teaching).

	Lecture
Weeks 1–2	Chapters 1–2: Introduction, HCI principles, and guidelines
Weeks 3–5	Chapter 3: Cognitive science, GOMS, human factors
	Homework 1: • Application of HCI principles/guidelines • GOMS exercise
Weeks 6–8	Chapter 4: HCI design

	Homework 2: • Project proposal (Part 1): Functional and UI requirements, user analysis, etc. • Design of the app (Part 2): Interaction model, scenario, storyboards, basic interface design, and wire-framing • Short presentation
Week 9	Midterm exam (Chapters 1–4)
Weeks 10–11	Chapters 5–7: Implementation issues
	Homework 3: • First implementation of project (using the MVC model) • Presentation (MVC structure) and working demo 1
Weeks 12–13	Chapter 8: Evaluation
Weeks 14–15	Chapter 9: Future of HCI
	Homework 4: • Self-heuristic evaluation for the project • Carry out and receive peer review for other projects and one's own project • Redesigning/reimplementation of the project app • Presentation of "before" and "after" and working demo 2
Week 15/16	Final exam

The PowerPoint lecture slides and the source code for the example application used in this book ("No Sheets 1.0," also downloadable through Google Play) are available through the publisher's resource website (see http://www.crcpress.com/product/isbn/9781482233896). I sincerely hope that the book will help readers to develop and acquire an HCI mindset as an important step to becoming a capable IT professional in the field.

The completion of this book was possible only with the greatest help and understanding from many people. My first thanks go to my graduate students at the Digital Experience Laboratory at Korea University (Youngsun, Youngwon, Changhyun, Jong-gil, Sang-yong, Jae-dong, Myong-hee, and Euijae). They helped me with proofreading, drawing figures, formatting, and many other tasks in the midst of research, projects, classes, and all the other things that make up the life of a graduate student. My dear colleagues in the HCI community have also given me much valuable feedback regarding the content and structure of the book. In particular, I thank Prof. Jee-in Kim, Dr. Gun Lee, Prof. Woontak Woo, Prof. Jinwoo Kim, Prof. Jongwon Lee, Prof. Jong-il Park, Prof. Seokhee Jeon, Prof. Si-Jung Kim, Dr. Ungyeon Yang, Prof. Junho Kim, Prof. Chang-Guen Song, Prof. Jin-seok Seo, Prof. Sookjin Kim, Prof. Junho Choi, and Prof. Mincheol Hwang. I am very grateful for the support of the

KRF-funded Engineering Center of Kwangwoon University (head: Professor Eunsoo Kim). CRC Press has been very patient and prompt with assistance for all my writing problems, not to mention seeing the value in publishing this book. Finally, I thank my wife Sooah, my parents, and my children (Andrew and Ellen) for their understanding and just for being there!

About the Author

Gerard "Gerry" Jounghyun Kim earned his bachelor's in electrical and computer engineering at Carnegie Mellon University and his master's and PhD at the University of Southern California. He started his academic career at POSTECH in 1996 after a short post at the U.S. National Institute of Standards and Technology as a National Research Council postdoctoral fellow. In 2006, he moved to Korea University. Since 1996, he has conducted research in the field of HCI, including virtual and mixed reality, mobile interaction, and multimodal interaction. Dr. Kim has written more than 100 articles in international and domestic journals and conferences, and he is the author of *Designing Virtual Reality Systems* (Springer, 2005).

1 Introduction

1.1 What HCI Is and Why It Is Important

Human–computer interaction (HCI) is a cross-disciplinary area (e.g., engineering, psychology, ergonomics, design) that deals with the theory, design, implementation, and evaluation of the ways that humans use and interact with computing devices. *Interaction* is a concept to be distinguished from another similar term, *interface*. Roughly speaking, interaction refers to an abstract model by which humans interact with the computing device for a given task, and an interface is a choice of technical realization (hardware or software) of such a given interaction model. Thus, the letter *I* in HCI refers to both interaction and interface, encompassing the abstract model and the technological methodology (Figure 1.1).

HCI has become much more important in recent years as computers (and embedded devices) have become commonplace in almost all facets of our lives. Aside from merely making the necessary computational functionalities available, the early focus of HCI has been in how to design interaction and implement interfaces for high usability. The term *high usability* means that the resulting interfaces are easy to

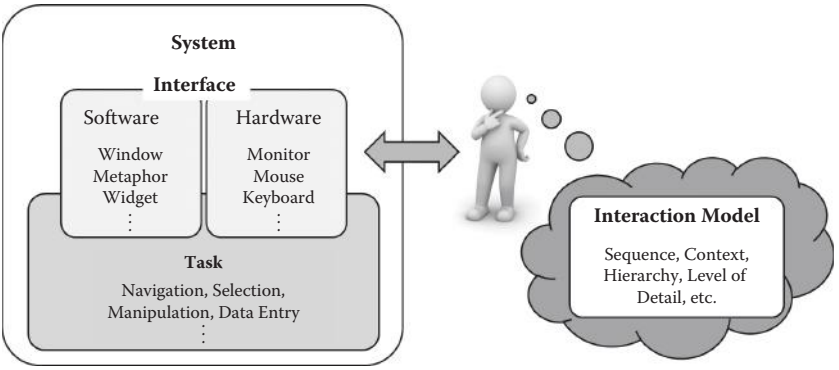

Figure 1.1 The distinguishing concepts of interaction (model) and interface.

use, efficient for the task, ensure safety, and lead to a correct completion of the task. Usable and efficient interaction with the computing device in turn translates to higher productivity.

The simple aesthetic appeal of interfaces (while satisfying the need for usability) is now a critical added requirement for commercial success as well. The family of distinctly designed Apple® products is a good example. Apple products are attractive and have created a multitude of faithful followers even though their functionality may be virtually equal to their competitors. In this context, the concept of *user experience* (UX) has lately become a buzzword, a notion that not only encompasses the functional completeness, high usability, and aesthetic appeal of the interactive artifact, but also its seamless integration into one's lifestyle or even creating a new one around it (Figure 1.2).

A less acknowledged fact is how HCI has had a huge impact in the history of computing and changed our daily lives. It was probably the invention (or rediscovery) of the mouse that was the linchpin in the personal

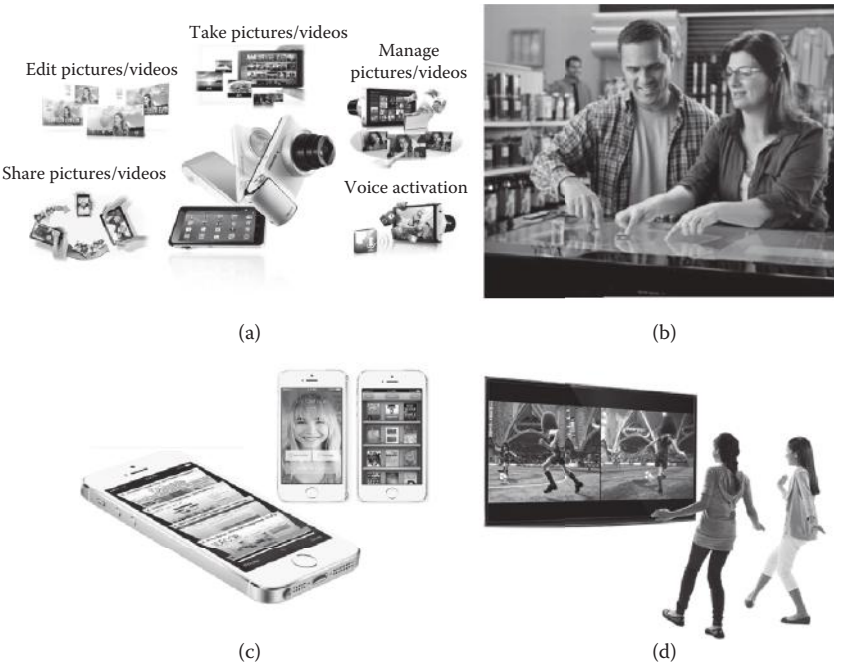

Figure 1.2 Goals of human–computer interaction (HCI): (a) functional completeness (Apple iPhone 5s, http://www.apple.com/iphone-5s), (b) high usability (Microsoft® Pixelsense, http://blogs.msdn.com/b/pixelsense), (c) aesthetic appeal (Apple iPhone 5s), and (d) compelling user experience (UX) (Microsoft Kinect, http://www.xbox.com/ko-KR/Kinect).

Figure 1.3 The evolution of interfaces in the course of the history of computing (i.e., terminal and keyboard, graphic user interface and mouse, and handheld and touch-based interface). (Courtesy of Cox, J., https://www.flickr.com/photos/15587432@N02/3281139507, Melbourne, FL.)

computer revolution, making the operation of a computer intuitive and much easier than the previous system of keyboard commands. The spreadsheet interface made business computing a huge success. The Internet phenomenon could not have happened without the web-browser interface. Smartphones, with their touch-oriented interfaces, have nearly replaced the previous generation of feature phones. Body-based and action-oriented interfaces are now introducing new ways to play and enjoy computer games. HCI still continues to redefine how we view, absorb, exchange, create, and manipulate information to our advantage (Figure 1.3).

1.2 Principles of HCI

Despite its importance, good HCI design is generally difficult, mainly because it is a multiobjective task that involves simultaneous consideration of many things, such as the types of users, characteristics of the tasks, capabilities and cost of the devices, lack of objective or exact quantitative evaluation measures, and changing technologies, to name just a few. A considerable knowledge in many different fields is required. Over the relatively young history of HCI, researchers and developers in the field have accumulated and established basic principles for good HCI design in hopes of achieving some of the main objectives (as a whole) that were laid out in the previous section. These HCI principles are general, fundamental, and commonsensical, applicable to almost any HCI design situation. Here, we provide a short review of the main HCI principles.

1.2.1 "Know Thy User"

The foremost creed in HCI is to devise interaction and interfaces around the target users. This overall concept was well captured by the phrase,

"Know thy user," coined by Hansen [1] in 1971, even though the so-called user-centered design approach has become a buzzword only in recent years. This principle simply states that the interaction and interface should cater to the needs and capabilities of the target user of the system in design. However, as easy as this sounds, it is more often the case that the HCI designers and implementers proceed without a full understanding of the user, for example, by just guessing and pretending to know and be able to predict how the representative user might respond to one's design. Ideally, comprehensive information (e.g., age, gender, education level, social status, computing experience, cultural background) about the representative target user should be collected and analyzed to determine their probable preferences, tendencies, capabilities (physical and mental), and skill levels. Such information can be used to properly model interaction and pick the right interface solution for the target users.

Consider a situation where a developer is working to change an interface, supposedly to achieve higher usability. However, we might need to remember that while young adults are extremely adept at and open to adopting new interfaces, older generations are much less so. Here is another example. Males are generally known to be better than females in terms of spatial ability and, as such, one might consider such a fact in employing three-dimensional (3-D) user interfaces. However, other studies point to females majoring in engineering and science to possess an equivalent level of spatial ability as their male counterparts [2]. So sometimes, conventional wisdom alone may not be sufficient to warrant proper interface design. These examples illustrate that there are a great many aspects that need to be considered in this regard. If a direct field study is not feasible, an experienced and humble HCI designer will at least try to leverage the vast knowledge available from cognitive psychology, ergonomics, and anthropomorphic data to assess the capabilities and characteristics of the target user group. Figure 1.4 shows examples of user-centered designs of web pages for kids and the elderly.

A related (or perhaps even opposing) notion to the user-centered design is the concept of "universal usability," which roughly promotes "humane" interfaces that cater to a wide (rather than a specific) range of users, i.e., across age groups, skill levels, cultural backgrounds, and disability levels. Such a notion has become almost required in our advanced multicultural societies. However, as wonderful as it sounds, it is generally very difficult to achieve this with a single interface.