

Learning from Interaction

Tarek Mokhtar*

PhD in PDBe of University, USA; Assistant Professor of Architecture and Digital Design, Ain Shams University, Cairo, Egypt

In a post-PC world, while humans are in need for easier and more understandable environments – a need for simplicity, smart/intelligent architectural elements, e.g., robotic chairs, automatic doors, interactive workstations, interactive faucets, etc. – are getting more complex every day. If complexity is unavoidable and simplicity is needed, can we tame complexity to be easy to use and understand intelligent architectural elements? By better understanding humans and their interactions with these intelligent objects we will be able to understand and learn how intelligent architectural elements should be designed. Don Norman [1], Jon Kolko [2], and Adam Greenfield [3] agreed on the need for “managing complexity” so as to satisfy current complex human needs.

Unavoidable Complexity

Nowadays, designers should learn how users interact with the intelligent architectural elements around them so as to decrease errors and complications. For example, a simple automatic sliding door not only needs to be able to recognize the presence of a person in front of it, but should also be able to recognize the users’ scale (short or tall), the user-object distance (close or far), emergency cases (e.g., in case of fire), user-object speed (slow or fast), and the amount of time to wait before closing. This simple intelligent sliding door is unavoidably a “complex” system. Adding to this, the unexpected experiences the user is facing with the door, e.g., have you ever changed your mind and decided to go back to your car once Walmart’s entrance door has opened? That is a situation where designers should be aware of it by observing the human-door interaction so as to consider and manage the time to wait before closing the door and also to decide where to put the sensors. The owner of this building should be able to disable/shutdown the automatic feature of the door to secure the building whenever needed. Thus, by knowing the daily scenarios and experiences of the users with such intelligent element (e.g., the door), users will have a better interactional experience with the intelligent architectural elements in our built environment – Learning from Interaction.

Learning from Interaction

In the information era, architects who are designing smart objects need to follow a non-linear conceptual framework to better develop a usable and understandable architectural elements. The suggested

framework consists of four steps: (a) Designer-Element Interaction: in which the designer “learn by doing”, i.e., he/she gets in contact with the intelligent element and use it so as to learn the basic properties and functionality of the element; (b) Designer-User Interaction-I: in which the designer “learn by observing” users interacting with the intelligent architectural element; (c) Designer-User Interaction-II: in which the designer asks the user to speak aloud what he/she will do when interacting with the object; finally, (d) User-Object Interaction: in which the designer leaves a feedback sheet for users to comment on their experiences with the object and freely express their reactions (a low bandwidth relation). The suggested framework is an iterative design process, i.e., the designer will have low-fidelity prototypes to get feedback on each of these four phases and whenever he/she receives a feedback a new prototype evolves. The steps are repeated until the designer receives a satisfactory feedback.

Automation and Intelligence

However, Jaron Lanier [4] warned designers and developers not to accept the automation of technology because it kills creativity and interaction – from a social perspective, and Jeff Hawkins [5] believes that we suffer from the lack of knowledge about real intelligence, intelligent/automated technologies are getting everywhere. Both arguments are valuable and should be considered when designing intelligent objects. Architects should not neglect the social and interactional challenges that occur from undeveloped programming for these objects – a need to use advanced programming will help; use the latest developments in IT for our designs; and, continuously learn from user interaction – the user should always be the center of the design process.

References

1. Norman DCE(2010) Living with Complexity, MIT Press.
2. Kolko J (2010) Thoughts on Interaction Design, Morgan Kaufmann.
3. Greenfield A (2006) Everywhere: The Dawning Age of Ubiquitous Computing, New Riders.
4. Lanier J (2010) You Are Not a Gadget, P.U.U.
5. Hawkins J, Blakeslee S (2004) On Intelligence, Owl Books.

*Corresponding author: Tarek Mokhtar, Ph.D. in PDBe of Clemson University, USA; Assistant Professor of Architecture and Digital Design, Ain Shams University, Cairo, Egypt, E-mail: tmokhta@clemson.edu, E-mail: tmokhta@g.clemson.edu

Received June 26, 2012; Accepted June 28, 2012; Published June 30, 2012

Citation: Mokhtar T (2012) Learning from Interaction. J Archit Eng Tech 1:e103. doi:10.4172/2168-9717.1000e103

Copyright: © 2012 Mokhtar T. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.