CAR-TR-521
CS-TR-2557

November 1990

Touchscreen Toggle Switches: Push or Slide? Design issues and usability study

Catherine Plaisant[1] & Daniel Wallace[2]

Human-Computer Interaction Laboratory
[1]Center for Automation Research
[2]Department of Psychology
University of Maryland
College Park, MD 20742

Abstract:

This article describes six different touchscreen based toggle switches allowing the control of two state (ON/OFF) devices. The user interfaces, ranging from button type toggles to sliding toggles are described and compared. A usability test with 15 subjects was conducted. Error rates, user preferences and subjective satisfaction ratings were collected. Results indicate that all the toggles described here can be used with low error rates. The sliding toggles were rated harder to use and were least preferred. Individual differences in personal preference were shown to be very large. It was also observed that users spontaneously or after one trial use a sliding motion to activate a control showing a sliding affordance.

Touchscreen Toggle Switches: Push or Slide?
Design issues and usability study

Catherine Plaisant[1] & Daniel Wallace[2]

Human-Computer Interaction Laboratory
[1]Center for Automation Research
[2]Department of Psychology

University of Maryland
College Park, MD 20742

Abstract:

This article describes six different touchscreen based toggle switches allowing the control of two state (ON/OFF) devices. The user interfaces, ranging from button type toggles to sliding toggles are described and compared. A usability test with 15 subjects was conducted. Error rates, user preferences and subjective satisfaction ratings were collected. Results indicate that all the toggles described here can be used with low error rates. The sliding toggles were rated harder to use and were least preferred. Individual differences in personal preference were shown to be very large. It was also observed that users spontaneously or after one trial use a sliding motion to activate a control showing a sliding affordance.

1. Introduction

The present study explores various designs of toggle switches to allow novice or occasional users to control two state (ON/OFF) devices in a touchscreen environment.

Toggle switches can be very confusing. Examples can be found in everyday life. For instance, in one of the authors' house the outdoor light switch is merely a pushbutton, giving no indication of its state so that it is necessary to lean by the window to see if the light is ON or OFF. Therefore we see that switches should provide a means of demonstrating the current state of the device. A second issue identified is the difficulty of showing the user how to change the state of a device with the toggle switch.

Computers allow designers to design many new types of "soft" toggle switches by providing an easy way to create and modify the appearance and behavior (or look and feel) of controls as they move into a two dimensional world. It is no longer necessary to select a control from a catalog, but unfortunately the lessons from traditional control design [CH72][KA82] are often ignored. This additional freedom brought a new wave of inadequate toggle designs (Figure 1).

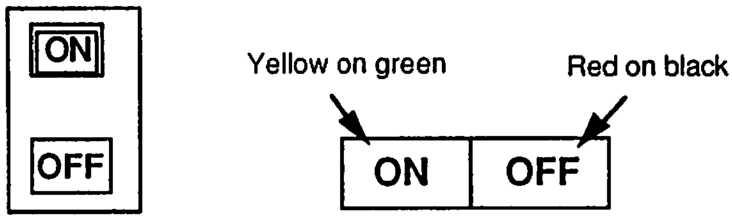


Figure 1 - Examples of ambiguous toggles we found in real systems.

The classic wall toggle switch has two possible states (typically ON and OFF) and an obvious means to change that state. It usually utilizes a short "handle" to throw the switch from OFF to ON or vice versa. The fact that the switch is represented in three dimensions provides a wealth of benefits that are difficult to obtain from a two-dimensional representation. Therefore, the transition from the 3-D world to a 2-D computer screen must take into consideration the psychological and physiological impact of such a transition. Steps must be taken to minimize or solve the problems that arise from moving technology into the computer screen.

Examples can be found in the literature describing the difficulties encountered by computer users when toggle switches were evaluated [VA85] [WA88]. Toggles are by nature simple and the toggles used in our computer systems should be usable without any hesitation by any newcomer to the environment.

This project had a highly practical orientation since this work was conducted in collaboration with an advanced technology company (Custom Command Systems Inc.). Custom Command designs and installs home automation systems [TI89, SM88] which integrate entertainment, security, lighting, climate control, communication and other devices in affluent homes and offices. The control of these systems is afforded through a touchscreen interface. Our goal was to select a usability tested/error free toggle for Custom Command to use in home control systems and to better understand the problems involved in toggle design for a touchscreen environment. In addition to providing a natural method of interaction, touchscreens are easy to learn, efficient, have no moving parts, require less work space than other input devices, and evoke a high degree of user satisfaction [KA86] [SE90].

2- TOGGLE REQUIREMENTS AND ISSUES

Based upon what we already know about the successful wall switch, it is clear that a toggle needs to perform the following functions:

1. It must indicate the current state of the device in a completely unambiguous manner (e.g.: On, Off; Open, Close; Armed, Disarmed; etc.)

2. It should suggest that the user can change the state of the device and how to do so.

3. It must acknowledge the actions of the user (appropriate feedback)

The most common problem encountered is the confusion between state indication and possible action label: does the label ON indicates the state of the device or does it indicates the resulting state when the toggle is activated?

Another common problem comes from the difficulty of deciding what to do to change the state of the device. The design needs to signal the appropriate action. For example Valk showed that users were confused by a design showing a slider, but only touches on the end of the slider were permitted by the system and sliding was not possible [VA85].

Despite the expected simplicity of a two-state device control many attributes can be identified. Chapanis recommend the use of pushbuttons or toggle switches[1] as controls for two-state devices [CH72]. Kantowitz and Sorkin list 7 human-factors considerations for such controls [KA82]. Three of them in particular can be adapted to software toggles.

1. Physical parameters: In the case of a "soft" toggle they are linked to the type of metaphor used: button to press, slider to slide, lever to push, rocker to swing, etc. Chapanis recommends that the control/display direction of movement relationship satisfy natural relationships (e.g. move up to open a blind), existing practice (e.g. up for ON), and consistency.

2. Coding and labeling: Labels, size, color, brightness, and backlighting effects can be used to indicate states.

3. Feedback: In the case of soft buttons users need to rely on visual and audio feedback extensively since tactile feedback is not available.

In this study we decided to design a small number of toggles and evaluate them by measuring the error rates and user satisfaction. Another approach to the study of toggles would be to design an array of toggles constructed by varying each attribute of toggle design and comparing them in a systematic manner. The rather large number of possible attributes pushed us to choose the first approach. If careful consideration is given to the issues identified above, a small set of usable may be obtained. Reducing hesitation and the need for trial and error exploration should increase the subjective satisfaction of users, improve response times and reduce error rates.

[1] Chapanis separates pushbuttons(having only one field) from toggle switches (having 2 or 3 fields e.g. like a lever toggle or rocker toggle). Nevertheless we use the word toggle globally for both since it seems to be the name used by most user interface designers.

3 - USABILITY TEST

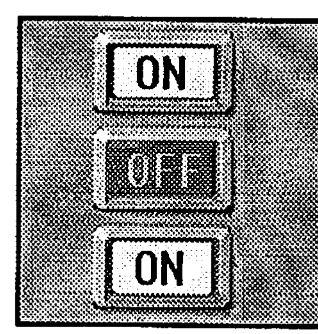
Participants: Fifteen undergraduate students volunteered to participate in the study. Three had used a touchscreen before on an occasional basis.

Apparatus: The Microtouch touchscreen used returns a continuous flow of coordinates with a 1024x1024 resolution. This allows the dragging of objects, the identification of sliding motions and the use of a lift-off strategy for selection. This strategy reduces the error rates and allows the selection of small targets (as small as one pixel [SE89]). The color, graphical screen displays are implemented under MS-DOS in the high resolution VGA mode (640x480 pixels). The touchscreen and all the home devices are controlled by a NEC386 personal computer.

Stimuli (Toggles): A requirement imposed by our particular application was to design toggles allowing lists of devices or options to be presented on the screen. This limited us to horizontal toggles to increase the number of possible toggles and labels per page. There were six different designs chosen for this study.

The first toggle is a single field pushbutton. The next three are push toggles activated by touches, while the last two toggles require a sliding motion to change state.

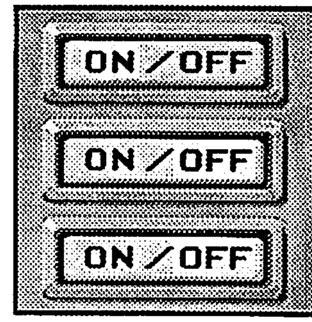
One-button toggle (Pushbutton) : The state is shown by label and color only. When the device is ON the pushbutton shows a label ON in black letters on a bright buttercup yellow background (as if lighted). When users touch the button the background color darkens and the state changes to OFF when the finger is released. The OFF label is in gray letters on dark gray background. When the device is switched to OFF a low pitch click is produced (respectively a high pitch click when turned ON).



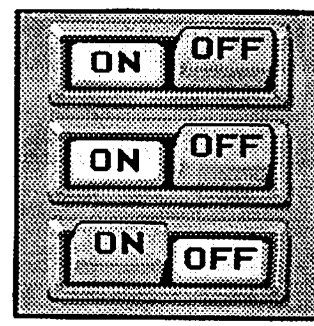
The main problem with pushbuttons is that their identification as a toggle (and not a simple indicator) might not be obvious for all users: they may not realize that they can change the state by touching it. Once recognized as a toggle the pushbutton has the advantage of being graphically simple and uncluttered and its size can be reduced if necessary.

Words toggle: If the device is ON and users touch the OFF side of the toggle, OFF is highlighted and when the finger is released the state changed to OFF. Once the toggle is touched users can go back and forth between ON and OFF by moving their fingers from one side to the other to correct an error if they need to. If the device is ON and users touch ON nothing happens.

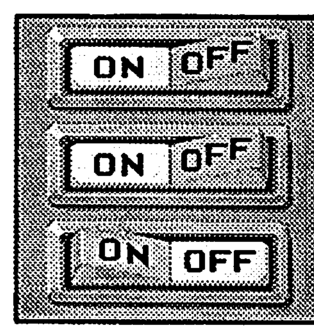
In this word toggle the state is shown by label and color only. When the device is ON the label is on a bright green background, when OFF the label OFF is on a bright background. No audio feedback is given (this design tries to simulate the current implementation used in Custom Command's system, in a style consistent with the other toggles).



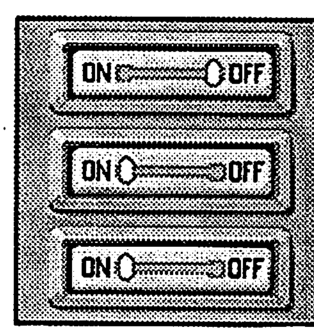
Two-buttons toggle: The visual effect of moving buttons is used in addition to labels and colors. When the device is ON the ON side is depressed and the ON label is on a buttercup yellow background while the OFF label is gray like the button itself. When users press the OFF side the OFF side of the toggle is depressed under their finger and the ON side goes up. A high pitch click is heard when the device is turned ON (respectively a low pitch click for OFF).



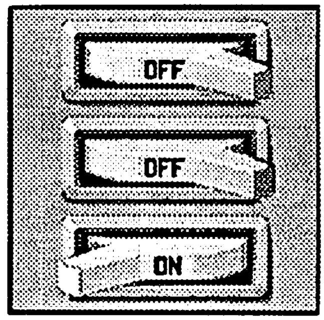
Rocker toggle: This toggle only differs from the two-buttons by its graphical appearance. The rocker toggle suggests a rocking movement from one side to the other.



Slider toggle: In this toggle a sliding/dragging movement is required to change the position of the yellow pointer from one side of the toggle to the other. A simple three step animation shows the movement of the pointer along the slide. If the device is ON the pointer is on the ON side. Users can then grab the pointer and slide it to the other side. If the finger is released before reaching the other side the pointer springs back to its previous position. A click is heard when the state changes (high pitch for ON, low pitch for OFF).



Lever toggle: Same "behavior" as the slider. Only the graphical appearance is different. In this image of a realistic lever switch only the label of the current state appears. It relies on the 3D effect to indicate that this is a toggle and to suggest what action is necessary to change the state.



Our prediction was that all six toggles would allow users to identify the state of the device with relatively few errors. We expected the sliding toggles to be harder to use but more attractive that the push-toggles.

Procedure: Each subject used every toggle (in a counter-balanced order) without instruction or demonstration (see figure 2). They were asked to perform a series of representative tasks in a brief scenario, and were then asked to fill out an adapted version of the Questionnaire for User Interaction Satisfaction (QUIS) [CH88]. After using all toggles, subjects were given a forced choice preference rating for each possible pairing of toggles. The entire session lasted about 30 minutes.

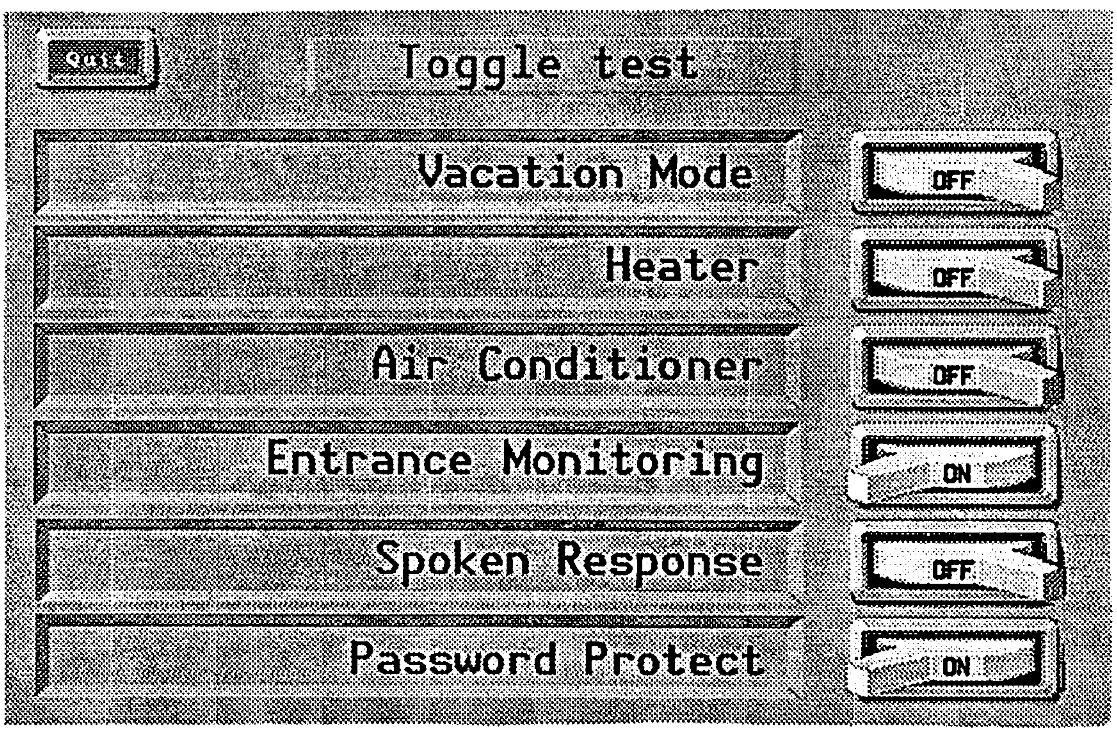


Figure 2: A Sample Screen of the Lever Toggle Prototype

Results: A fairly consistent rank ordering of preference between the toggles was found. The order of preference (from higher to lower) was typically: one-button, rocker, two-button, words, slider, lever.

Forced-choice preference ratings: The number of times a particular toggle was rated to be preferred over another was used as an aggregate measure of total preference. A significant effect of total preference was found ($p<.05$) (Figure 3). The one-button and rocker toggles were each preferred over the slider and lever ($p<0.05$).

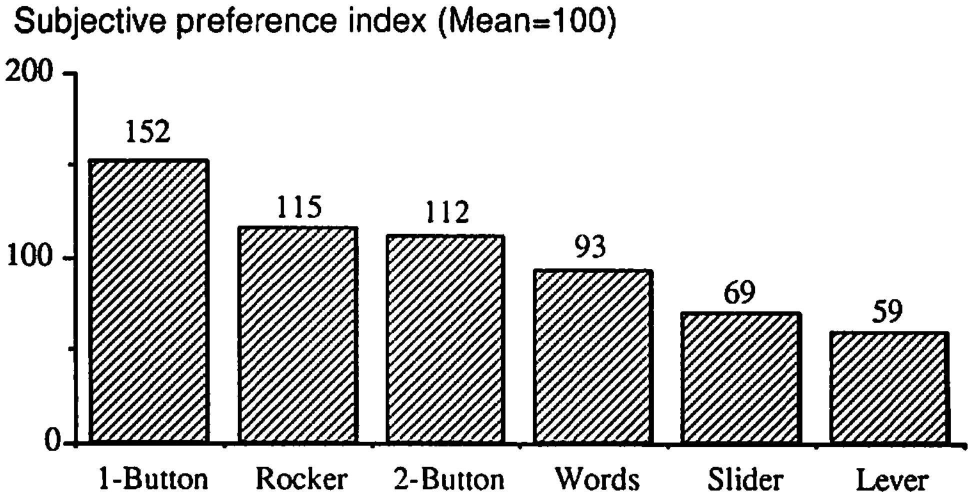


Figure 3 - Preference rating

Overall satisfaction: Responses to the QUIS inventory were summed for each toggle representation, and the overall ratings obtained were analyzed using ANOVA. An overall significant effect was again obtained ($p<.05$). The same rank ordering was obtained as with the forced choice preference ratings.

Ease of determination of the current state: No errors were made when subjects were asked to determine the current state for each of the prototype toggles. This suggests that is that is possible to design error free toggle switches. When QUIS questions relating to this factor were analyzed no significant differences were found. Nevertheless for all toggles except the 1-button toggle, there was at least one low rating (<5) in the question "I am unsure(1) / confident(9) about whether a device is ON or OFF" .

Ease of manipulation: We observed that several subjects had difficulties manipulating the slider and lever toggle. This was confirmed by the ordering of the means of ratings of ease of manipulation (a combination of 5 QUIS questions). A similar ordering of toggles was obtained (Figure 4), with an overall significant effect ($p<.05$). The one-button toggle was rated significantly easier to use that the slider and lever.

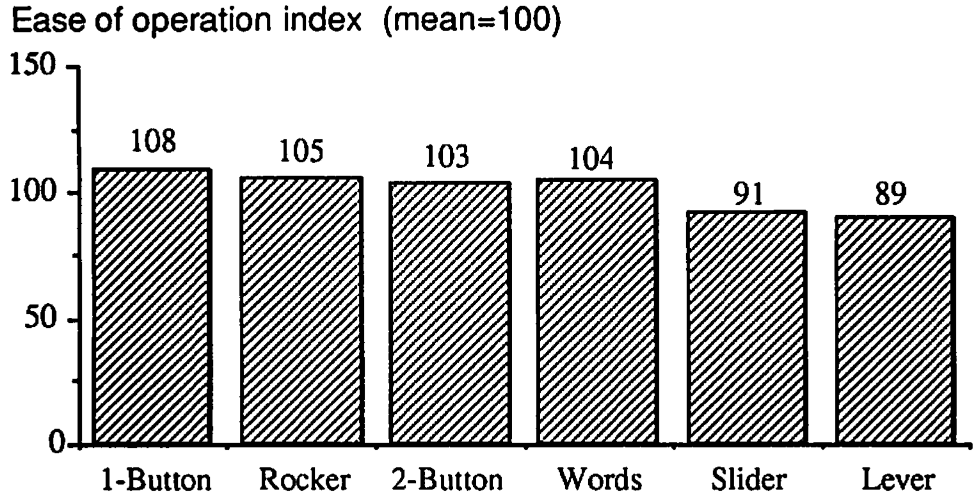


Figure 4 - Ease of operation index

No significant differences were found on the ratings about the use of feedback and the attractiveness of the toggle.

Discussion: Although few statistically significant differences were found at the a priori 0.05 level the mean differences were relatively sizable. The reason for this lack of statistical power is attributed to the relatively small number of subjects and the large individual differences in preference.

Our conclusion was that the pushbutton toggle was appreciated for its simplicity, and was thus rated consistently higher. Of course subjects knew that they were dealing with devices to turn ON and OFF and therefore the dangers of a pushbutton being taken for a status indicator were eliminated. This concern should be remembered when using pushbuttons in an interface for one-time users.

The rocker toggle was among the highly preferred toggles probably because the required movement on the touchscreen to switch back and forth between ON and OFF is quite compatible with the action on a real rocker switch. The graphic is also simpler than the two-button toggle and more attractive than the word toggle.

The toggles that are pushed seemed to be preferred over the toggles that slide. A possible explanation is that sliding is a more complex task than simply touching. Another thing we noticed is that sliders are more difficult to implement than buttons. The usability test brought to light some small flaws in the design of our two sliding toggles (e.g. because of its strong perspective the lever toggle was often touched too low - this bias can be corrected; the slider pointer should be larger, and the lever or pointer should highlight when touched to signify that the user now has control over it).

This study was a opportunity to observe spontaneous reactions to touchscreen sliders. Even if several subjects first attempted to touch the extremities of the toggle before trying to slide the lever or pointer. We observed that all subjects (spontaneously or after one trial) used sliding motions successfully to manipulate the lever or slider toggle. It is interesting to note that among the subjects that did not use sliding right away were the 3 subjects that had already used touchscreens. One of them specified in his comments that he first did not expect the touchscreen to recognize the motion. Even many user interface experts would probably not attempt a slide, since it is often found in the literature that touchscreens only recognize first touches.

Even if sliders where not preferred, the fact that users used them correctly is encouraging since many other controls can be designed using sliding motions. Sliders can be used to adjust the speed of a machine, the volume of musical instrument or the temperature of room. Another advantage of the sliding movement is that it is less likely to be done inadvertently (at the right location in the right direction) therefore making the toggle more secure. This advantage can be pushed further and controls can be designed to be very secure by requiring more complex gestures (e.g. a U or W shape slider can be used for respectively a 2 or 3 setting control).

The evaluation of the toggles showed some important differences in personal preferences. Every toggle had at least one unconditional fan. Only the one-button and rocker received all positive or neutral comments. Therefore if one toggle had to be recommended as a potential "always acceptable toggle" (the vanilla ice-cream of toggles), the rocker implementation is probably the best bet (see the potential dangers of the 1-button described earlier).

Conclusions: It is possible to design a variety of unambiguous toggles resulting in high subjective satisfaction. A set of valid implementations of toggles has been described and tested, only the slider and lever toggle are less preferred and seem harder to use. Users will spontaneously use sliding movements to manipulate a toggle displaying a sliding affordance.

In light of the fact that we found large differences in personal preference, we find it difficult to recommend a "perfect" toggle representation. because of this, we feel that these strong personal differences make toggles excellent candidates for personal customization through a control panel selection.

Even if all the toggles presented were used without difficulty by the subjects of the usability test, designers should not conclude that any toggle will be easy to use. It is the combination of toggle type, graphic representation, adequate use of color coding, audio feedback, labeling, correction of touch biases and many other parameters that makes an effective toggle.

Possible future directions of research include the study of toggle usage under stress and speed constraints, for example an airplane pilot has to select and manipulate the appropriate toggle quickly and accurately. Because several types of toggle can be designed successfully it is also of interest to study combinations of different types of toggles that would ameliorate the grouping problem (which switches controls which function?) [NO88] . Combinations might be used to provide distinctiveness as in an automobile where headlight, wiper, heater and radio controls are purposefully different.

Acknowledgements: We would like to thanks Ben Shneiderman and all of the members of the Human-Computer Interaction Laboratory for their useful suggestions, and the subjects for their assistance. We appreciate the support of the Maryland Industrial Partnerships program and Custom Command Systems Inc. in providing funding for this research.

References:

[CH72] Chapanis and Kinkade, (1972). Design of controls, chapter 8 in *Human Engineering Guide to Equipment Design*, Sponsored by Joint Army-Navy-Air Force Steering Committee, John Wiley & Sons.

[CH88] Chin, J., Diehl, V., Norman, K. , (1988). Development of an instrument measuring user satisfaction of the human-computer interface. *Proc. of the Conference on Human Factors in Computing Systems.* ACM, New York, 213-218.

[KA82] Kantowitz and Sorkin, (1982). Chapter 10: Controls and tools, in *Human factor Understanding people-systems relationships*, John Wiley & Sons.

[KA86] Karat, J., McDonald, J., and Anderson, M., (1986). A comparison of selection techniques: touch panel, mouse, keyboard, *International Journal of Man-Machine Studies, 1, 73*-92.

[NO88] Norman, D. (1988), *The Psychology of Everyday Things*, Basic Books, New York, 1988

[SE89] Sears, A., Shneiderman, B., (1989). High-precision touchscreens: Design strategies and comparisons with a mouse, University of Maryland Technical Report, CS-TR-2268. *To appear in the International Journal of Man-Machine Studies.*

[SE90] Sears, A., Plaisant, C., Shneiderman, B., A new era of touchscreen applications, University of Maryland Technical Report, CS-TR-2487. To appear in *Advances in Human-Computer Interaction*, Vol.3, Hartson R. and Hix D., Ed.

[SM88] Smith, R.L., (1988). *Smart House - The coming revolution in housing*, GP Publishing, Inc., Columbia, Maryland, USA.

[TI89] Boosting your home's IQ, (1989). *Time*, January 23, 1989

[VA85] Valk A. M. , (1985). An experiment to study touchscreen " button" design, *Proceeding of the Human Factors Society*, 29, 127-131.

[WA88] Wallace, D., Norman, K., Plaisant, C., (1988). The American Voice and Robotics "Guardian" system: A case study and user interface usability evaluation. Technical report CS-TR-2113, University of Maryland, College Park, MD 20742.