

Age-specific usability issues of software interfaces

Simone Wirtz, Eva-Maria Jakobs, Martina Ziefle

Simone Wirtz, Human Technology Centre, RWTH Aachen University
Eva-Maria Jakobs, Textlinguistics/ Technical Communication, RWTH Aachen University
Martina Ziefle, Human Technology Centre, RWTH Aachen University

The aim of the present study was to identify shortcomings of electronic interface designs for older users. Beyond general ergonomic factors, the study focused primarily on visual and language aspects of interface design. By means of a multi-method approach, combining cognitive walkthrough procedures with a usability test, age-specific problems and requirements of older computer users were examined. In the cognitive walkthrough shortcomings of a sample application were analyzed. In the usability test the navigation performance of older users (aged 55+) was contrasted to a young adult group, to identify usability problems, which are age-exclusive (interaction difficulties only for older users) and age-specific (general problems that are more pronounced in the older group).

BACKGROUND

In many western societies the group of older adults is one of the most growing user groups of information and communication technologies. The rate of computer usage among people aged 55+ is still low compared to younger age groups, but is continuously increasing: In Germany, for example, 31% of users older than 55 years reported to use a computer regularly in 2004. Until 2007, this rate grew up to 41% equaling an increase rate of 10%. In the same period, increase rates in other age groups did were considerably lower (age group 10-24 growth of 2%; age group 25-54 growth of 7%) (destatis, 2008). Furthermore, the usage of new information and communication technologies, like computer technologies, becomes more and more important and necessary in the everyday life of older adults (e.g. using ticket machines, doing one's taxes).

Age-specific usability demands for software interfaces are examined from different disciplinary and methodological perspectives, such as psychology, computer science, economic studies and engineering (e.g. Czaja & Lee, 2003; Schneider, Schreiber, Wilkes, Grandt, & Schlick, 2008; Zajicek, 2001; Ziefle & Bay, 2005; 2006; 2008). Depending on the research perspective, different issues are focussed on, ranging from performance and acceptance issues over software features, to technical parameters. Characteristically, outcomes remain within the respective research community, mostly not satisfactorily communicated across disciplines. In addition, hardly any usability study explicitly focus on language issues, even though the successful interaction of humans with technical devices depends to a large extent on the quality of the device communication. Language issues and technical communication is an integral part of electronic interfaces.

The design of software interfaces is interdisciplinary by nature, but there is a lack of multidisciplinary approaches modeling the relationship between influencing factors as well as the relationship and interplay of interface elements (design, language, operational elements, structure etc.). This lack is not comprehensible and not acceptable facing the increasing number of old and frail computer users on the one hand and the solid (disciplinary) knowledge concerning the aging impact for

computer usage on the other hand. Findings from recent studies indicate that a huge set of different influencing factors has to be considered (Jakobs, Lehnen, & Ziefle, 2008).

The study at hand combines approaches and findings from different disciplines to examine the usability of software interfaces especially for older adults, to identify relevant language aspects of software interfaces, and to check the effect of selected influencing factors. Important influencing factors are e.g. biological or socio-cultural factors.

Biological factors

Ageing is accompanied by a number of physical processes, which are relevant for the successful interacting of older adults with technical systems.

Sensorial changes. All sense organs decrease with increasing age. Among the visual functions, visual acuity decreases color perception and contrast sensitivity decrease while the sensitivity and susceptibility to glare increases (e.g. Kline & Scialfa, 1997; Oetjen & Ziefle, 2007; Schlick, Ziefle, Park & Luczak, 2008). Concerning the design of software interfaces, changes in the visual system should be taken into consideration for layout decisions, i.e. the choice of colours, contrast, font, type size, size and design of objects such as buttons.

Furthermore, the ability to hear often decreases in older age. Older people have difficulties in perceiving soft tones at high frequencies and when distracting background sounds occur. Acoustic impairments are relevant for audio elements of software applications such as alerts, voice output, and acoustic feedback. Tactile changes mostly occur at the age of 70. The ability to distinguish tactile stimuli can diminish and as a consequence movements that demand fine motor skills may cause problems. Using touch screens therefore can be very difficult for older people.

Physical changes have an influence on e.g. mobility, (muscular) strength and fine motor skills (e.g. Vercryssen, 1997). Fine motor skills can be affected by shiver, immovability, and decreasing sense of touch in hands and fingers. Many older people have difficulties to operate with input devices such as mouse or touch pad. Small operating elements at close quarters

can only be handled with great difficulties. Functional deficits in this vein are therefore relevant for the design of operating elements (size, type and spaces between elements).

Cognitive changes affect various aspects such as intelligence, information processing, and memory. Within cognitive abilities, fluid components are distinguished from crystallized intelligence components. Fluid components include reasoning abilities, abstract thinking and problem solving processes. Fluid intelligence is assumed to be independently of learning processes. In contrast, crystallized intelligence components refer to accumulated (domain)knowledge, expertise and experience, including socially mediated, culturally determined cognitive strategies and skills, such as speaking, reading, writing, as well as professional skills. Fluid intelligence tends to be negatively affected by age, whereas crystallized intelligence remains or even improves (e.g. Baltes, 1991). Crystallized abilities are strongly linked with socio-biographic (e.g. education and educational opportunities) and social factors (technical socialization). Therefore biological factors cannot be regarded isolated. Different factors, which have an influence on computer usage can be closely related and interact.

Moreover, information processing can be affected in old age. The amount of information that can be processed at the same time decreases and more time for information processing is needed (Park & Schwarz, 1999). Acquiring, linking and memorizing new information are hindered. Searching and activating memorized information are affected. These changes should be considered for the development of menus and structures as well as for the complexity and volume of content.

Working memory declines as well as spatial memory in older age. Older people have difficulties to localize objects and to remember non-verbal elements and tend to have poorer memory for spatial tasks (Hawthorn 2000). These aspects could be relevant for the design of labels, headings, permanence of important information and objects, the structure and visualization of content and functions in software interfaces.

These biological factors are samples for changes in older age. Age-related functional deficits and restrictions may vary inter- and intraindividually, but, when they occur, they affect the computer performance, and thus, have to be considered to guarantee usable interfaces for older adults.

Socio-cultural factors

From a socio-cultural perspective it is important to examine which kind of technologies dominated during the adolescence of a generation. Based on Mannheim's generation theory (1964), generations have to be understood as groups of cohorts that differ due to social change. The groups can develop a collective consciousness because of drastic social occurrences (Sackmann & Weymann 1994). The term "generation" refers to contemporary-historic generations, who developed a generational consciousness because of shared experiences with public discourses, objects, situations, and events – such as technology innovation. The invention of technical devices, their diffusion in everyday life and the public discourse form the generational consciousness and the understanding of technology. The experiences with technology that members of a technology generation have during their youth and adolescence form their attitudes towards and their way of handling technology. The technology they grew up with becomes a refer-

ence point and object of comparison. That's why technology generations vary e.g. in their competence using technology (Sackmann & Weymann, 1994). This can be explained with the concept of technology grammar (Rudinger, 1996), which means knowledge about principles of technology usage as a (cognitive) set of operation principles. Sackmann & Weymann (1994) differentiate between four technology generations that correspond with waves of technology innovation. Older adults (age >55) belong to the "generation of increasing technization of households" (born between 1949-1963) or to one of the former generations ("generation of household revolution" born 1939-48; "pre-technical generation" born before 1939). They grew up with home technologies (television, radio etc.) and devices with mainly mechanical features. Operating with mechanical devices is radically different from operating with electronic devices. Typical for using machines are e.g. pressing buttons with perceptible resistance, switches that have to be turned over or sounds that indicate an action in progress. In contrast, characteristically for electronic devices is the multi-functionality and as a consequence several functions per key as well as fully determined programme sequences that are not perceptible for the user (Rudinger & Jansen, 2005). Lacks in older adults technology grammar for electronic devices can lead to operating errors, e.g. sometimes they use inadequate operation principles. Because of the differences between the technology grammar for mechanical devices and for electronic devices it is hard to reason by analogy.

In order to get a deeper understanding in which ways the technical performance of older adults is affected, an empirical study was undertaken, in which older adults (age >55) and younger (control group; age 23-29) were examined regarding basic difficulties when using a software programme. The study primary focused on visual aspects and language issues. The aim of the present study was to identify pitfalls and shortcomings older users experience and to clarify whether usability problems are age-exclusive (causing interaction difficulties only for older users) or age-specific (problems that cause serious interaction difficulties preferably for older people), and if so, of what kind they are.

SAMPLE APPLICATION AND METHOD

The quality of a software programme named 'Touch Speak Designer' was evaluated with a multi-method approach. The software is part of the system Touch Speak. Touch Speak is a computer based portable communication assistant for people who suffer from a speech disorder, especially developed for aphasia patients. Aphasia often occurs in older age as a consequence of a stroke. Patients can use the communication aid to complete sentences with words/phrases they cannot pronounce or they can even substitute verbal communication with help of the device (depending on the severity of their disorder).

The system consists of two components: a small screen device (communication aid) and the software programme to configure the communication aid and to create and manage the vocabulary. The software contains a database with about 10.000 so called speech-elements already available. Normally the vocabulary and the device are configured by the attending speech therapist or by family members. Approximately two-thirds of the Touch Speak users are older than 60. The probability that family members that have to deal with the

software are also older ones is relatively high. Thus the intuitive usability could be tested.

The software interface contains a toolbar and a menu bar and is divided into two main areas: the database area (left) and the area for the individual configuration of the vocabulary (right) (see figure 1). Two additional areas (preview and simulator) can be activated.

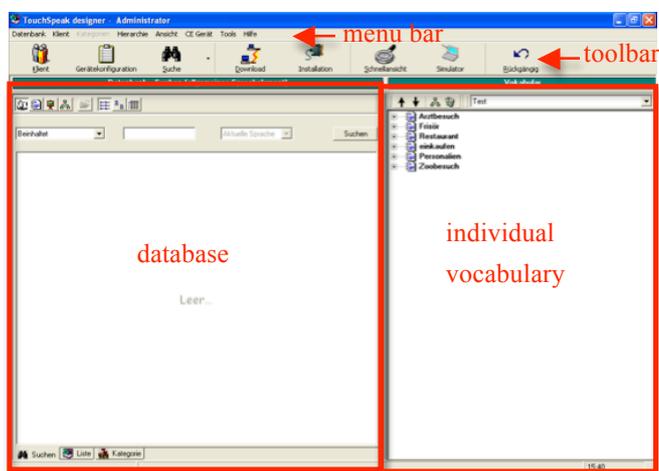


Figure 1: Constituents of the sample applications interface

In a first step, the software was analyzed in a multiple cognitive walkthrough to identify barriers and presumable usability problems (expert evaluation). All available features were examined taking the perspective of a typical user (old aged family member that has to configure the communication aid). The analysis based on general principles of web- and interface design (Norman, 1988; Shneiderman, 2004; Wagner, 2002) and specific ones for older adults (e.g. Coyne & Nielsen, 2002) as well as principles for communication and interaction quality (e.g. Grice, 1975). Furthermore, age-related functional deficits and changes were considered as well as social factors (e.g. technical socialization; Sackmann & Weymann 1994). The result of the expert evaluation was a classification of problems, which could lead to massive interaction difficulties for older adults. Twenty different types of usability problems were identified and classified into five categories: Consistency, language issues, feedback, layout and structure.

In a second step, older and younger participants were confronted with the software programme in a usability test to verify the results of the expert evaluation. Altogether, 14 volunteers with different professional backgrounds participated (5 female, 9 male). Five of them were computer and/or software experts; nine were classified as novices. This composition of the sample should allow the identification of gender-specific findings as well as the differentiation of age-specific problems from those that were caused by lack of experience in working with computers and various software programmes.

Participants had to solve six prototypical tasks, which were embedded into a comprehensive scenario. The tasks were: (1) identification of icons in the toolbox (naming), (2) orientation (identification of the areas functions), (3) finding a speech element in the database, (4) finding a specific function, (5) creating a new speech element and (6) trying the simulation. Every task had a time limit and varied in complexity and severity: the participants started and finished with easy tasks; task 3 and 5 were more complex (several steps to be taken) and

task 4 was in advance classified as rather difficult because the function that had to be found was hidden in a context menu.

While solving the tasks the participants were instructed to speak out loud what they were thinking, doing, and why they were doing it (think aloud method). Their comments and non-verbal behavior were recorded with video camera, screen camera, and microphone. In a retrospective interview that followed immediately, they were questioned about the tasks, the usability of the software interface, and the test situation. All collected data were transcribed, combined, and evaluated.

The analysis focused on three aspects: effectiveness (could the tasks be solved within the time limit), time on task (how long did they need to solve the tasks), and mainly qualitative problems within the usage.

QUANTITATIVE RESULTS

Results show that the older users needed considerably more time to complete the tasks and were less effective than the younger ones. The more complex a task is the more distinct are age differences. A comparison of task 1 (most easy task) and 5 (most difficult task) shows the differences: the younger participants needed about 2 min on average to finish with task 1, the older ones 3,4 minutes (see figure 2).

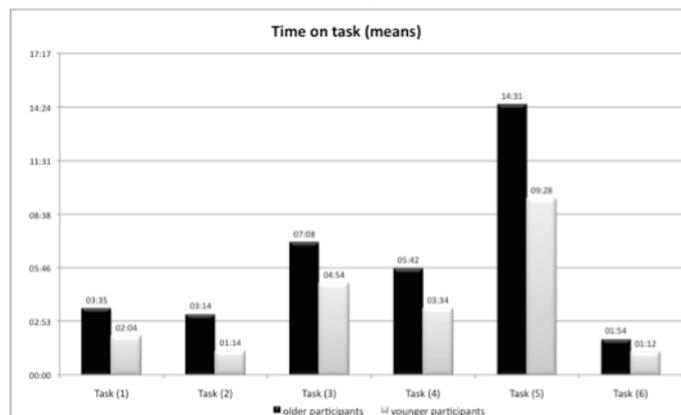


Figure 2: Time on task (in min.) – older vs. younger participants

The time difference for task 5 clearly exceeds (younger: 9,3 min; older: 14,3 min). The fastest participant (one of the younger group) needed 16 min less than the slowest (one of the older group). These results correspond with previous findings (e.g. Baltes et al., 1995; Ziefle & Bay, 2005), according to which task complexity led to an asymmetrically stronger performance decrement in the older compared to the younger group (Baltes, Lindenberger, & Staudinger, 1995).

Beside the time, the success rate was analyzed. Every sub-task could be solved correctly (rated with 2 points), could be partly solved (1pt) or not solved (0pt). A subtask was counted as partly solved when the procedure was correct but realized at a wrong time or place, e.g. when a speech element was inserted into the wrong category or a task was solved later on in the testing. In this analysis, task 1 and 2 were left out because the orientation and identification of icons (naming) could not be evaluated as solved correct or wrong. Over all tasks, a total score of 20 pt equals a 100% success rate) was to be reached. While the younger adults reached a success rate between 80-90 % the older users, in contrast, showed a considerably lower effectiveness (between 25-30 %). Only one of the older group, a computer expert, was able to solve 85% of the tasks.

The latter result may give a hint for an enormous potential of compensating abilities (high expertise in combination with a high variability within cognitive processing) as well in the older group (Morrow et al., 1999).

QUALITATIVE RESULTS

In the following, qualitative findings are reported. The findings do correspond to the usability problem categories, which were identified in the expert evaluation. In order to visualize the particular problem types and categories, descriptions of the usability problems are flanked by quotes from casual remarks and statements of participants in the retrospective interview. Data were anonymized. The notation is as follows: O stands for old and Y for young participant. E is the abbreviation for computer experts, N for novices. R stands for a quote from a retrospective, and I is the abbreviation for the Interviewer.

Consistency

The first category consists of consistency problems of different sources: inconsistent colouring, use of labels and headings, terminology, and operations. Consistency means the consistent design of the interface itself as well as the consistency to similar systems (under consideration of conventionalized design/patterns). Consistency is one of the most important principles of interactive systems. It is fundamental for the user to understand the programme and to build up an adequate conceptual model. A user's conceptual model of a technical system (software programme) bases on experiences with the system or similar systems (knowledge about operational routines and conventionalized patterns) and particularly on the interface (Norman, 1988). If the software interface is designed inconsistently users are confused, have difficulties to build up a correct conceptual model, cannot understand how the system works, get frustrated and, in the last resort, are not able to perform the required operation or task (Wagner, 2002). Furthermore, consistency brings cognitive relief, facilitates automation and learnability of operating processes, allows transferring acquired knowledge to unknown applications, and helps to locate relevant information (Nielsen, 1993; Shneiderman, 2004).

Inconsistent colouring. Inconsistent colouring means the usage of different colours for areas or objects with the same function. Task 5 demanded that participants create a new speech element with different components (title, message, sound, picture). To do so, a dialogue window has to be opened which contains different areas to insert and configure the components. Two of the older users could not solve this task because of the inconsistent colouring of the input fields. Usually input fields – in TouchSpeak Designer as well as in other software interfaces – are white. The input fields in the opened dialogue window for inserting the title or the message text are grey (see figure 3). This colouring irritated the users as one stated in the retrospective interview: “Above, I inserted the title because the cursor was positioned there. But beneath, this dark message window looked like it was disabled and as if you can’t insert anything. That’s [grey colour for disabled] usual. (...)” (RYE 01). This inconsistency did hinder the older participants more strongly than the younger, even though all participants stumbled on this inconsistency.

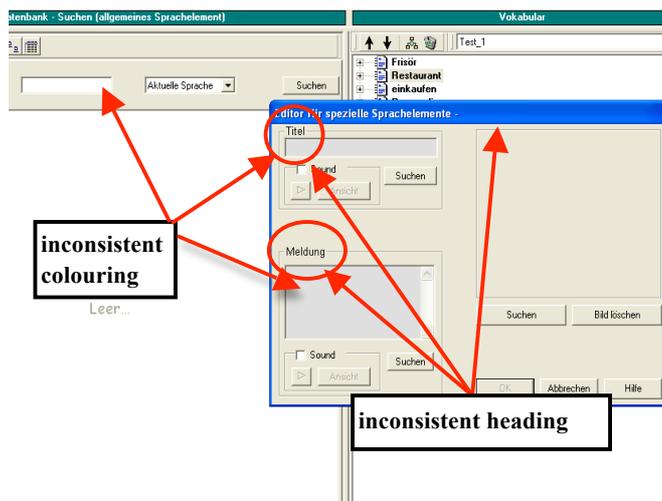


Figure 3: Inconsistency in colouring and the use of headings

Inconsistent use of labels/ headings. The use of labels and headings is inconsistent if they are not used for all areas or objects with the same or similar function without a reason.

Participants had difficulties in identifying the area to insert an appropriate picture – caused by an inconsistent layout respectively a missing heading (see figure 3). The areas for the configuration of title and message were in contrast clearly headed. Thus many participants, primarily the older ones, did not notice the picture area as such and searched for a picture everywhere but not in the dialogue window. Three out of four older users could not solve the subtask to add a picture to the new created element. Only one of the younger ones didn’t succeed. Again, we see that inconsistencies are bothersome for all users, however, they interfere especially older adults interacting with the interface.

Inconsistent operations. If an identical operation is not repeatable the same way, it is a matter of inconsistent operations. For example when users want to configure the synthesized voice output for a speech element sometimes the already inserted message text is taken over automatically – sometimes it is not and it has to be entered into a new inserting field again. This inconsistency makes the system unpredictable and leads to massive irritation. Two participants (YE 01 and OL 01) recognized this problem during the user test. As a consequence the older participant (OL 01) did not solve the task because of this problem; the younger participants (YE 01) on the contrary sees through the problem after a while and conducts the operation correctly.

Although inconsistent colouring of input fields and inconsistent labeling lead to a general irritation for users of both age groups, they have a higher impact on the older users causing severe difficulties. Thus, consistency, particularly consistent use of headings, labels and layout, is more important for older users than for younger ones. The conclusion to be drawn is, that problems of consistency tend to be age-specific.

Language issues

The category language issues includes all problems with language aspects of a software interface, e.g. naming of functions (menus, buttons), tool tips, and headings. In the expert evaluation four different types were identified: general naming problem, reference problem, polysemy, use of technical terms

and foreign words.

General naming problem. This category describes the false or vague denomination of functions or objects (Jakobs & Villinger, 1999). In the tool bar of the sample application, e.g. is a button named “download”. Download means to receive data to a local system from an external source. Contrary to this common sense, in TouchSpeak Designer it denominates the function to transfer data from the computer to the small screen device, which is in fact an upload. This general naming problem caused irritation and misinterpretation: participants used to interpret the button as “download” or “update from the Internet”. As differences between the age groups could not be detected, this type of problem is classified as not age-specific.

Reference problem. A *reference problem* occurs when there is more than one possible reference object and neither the term nor the context reveals the real reference object. In TouchSpeak Designer, e.g. there is a menu item “editor for tool bars”. However, there are several tool bars on the software interface as well as on the communication aid to which this menu item could refer. Similarly, several reference problems occurred because of an inadequate separation of functions dealing with the configuration of the communication aid and functions to operate within the software. “Installation” or “hardware configuration” for example are functions for the small screen device, but are partly interpreted as functions to configure the computer or the software, as the following statement from OE 02 shows: “Probably [a function] to configure the programme according to individual preferences of the developer or user.”

Polysemy. Polysemy in contrast means the usage of one term with different meanings and concepts to denominate different objects or functions. Using polysem terms is problematic if ambiguity cannot be resolved by the functional and linguistic context (e.g. as part of a specific menu). In the sample application the term “view” is a menu item to choose between different display options for the simulator (something like a preview), but it is also used as a button to open an editing window for the configuration of sounds that belong to a speech element. This leads to misinterpretations: Users do not unequivocally identify the button as a possibility to open an editing window; they mistake it is a preview for the inserted text, as the following sequence shows:

I: “What did you think this button [view] is supposed to be?”

OE 02: “That I can see the text.” (ROE 02)

Use of technical terms and foreign words. The use of technical terms or foreign words (Anglicism) can be problematic if the user has not enough background knowledge about the domain, respectively computers, and/or insufficient command of English. The TouchSpeak Designer interface contains some technical terms as well as Anglicism.

For example one button in the tool bar is labeled “client”, which was incomprehensible for some of the younger users. “I cannot image anything [what this could be]“ (RYE 02). All older users identified the button correctly as “user administration”. Obviously this term is more familiar to the olds. A client is a recipient of services, usually known from judiciary but also used in the medical context. Probably the older ones have more experiences with doctors and are more familiar with the medical domain and domain specific terminology. We assume that problems with domain specific terms decrease with in-

creased domain knowledge. In contrast computer specific terminology causes interaction problems only for computer novices – for younger as well as for older adults.

Summing up, results show no age differences regarding this problem category. Obviously problematic language issues lower usability for all users to the same extent. This result corresponds with findings from the literature: suboptimal names for functions have negative effects independent of the users age (Schröder & Ziefle, 2005).

Structure

This category includes general structural problems, problems of categorization.

General structural problem. A general structural problem is the use of an inadequate structural principle. According to Norman (1991) it can be distinguished between eight structural principles for interface elements, e.g. chronological order, alphabetical order, semantic order. Depending on the purpose of use, context and user different structural principles can be adequate or not (K. L Norman 1991; Wagner 2002). The tool bar in the examined software interface is structured by frequency of usage: it contains the most often used functions (according to the manual). But the order is designed according to the mental model and usage of speech therapists, not to the usage of people who use the communication aid at home. The effects of this structural problem can be observed in the test as a whole (not during a specific task). The participants mostly search for required functions in the tool bar to solve the tasks. They click on buttons like “client” and “hardware configuration” that are irrelevant for their operations. The availability in the tool bar signalizes “this function is important” and thus they are delved into distraction. Other functions like “creating a new element”, which are very important to solve the tasks – and of course to work with the software at home – are not present on the interface and can only hardly be detected (e.g. in context menus). This user specific structure impedes the usage of the software for people who use and configure the system at home. The usage of the structural principle frequency of usage is not adequate if the perspectives of different user groups diverge like in this example.

Problems of categorization and assignment. In the expert evaluation several problems of categorization and assignment were identified. Both problem types are strongly related: any assignment problem is as well a problem of categorization. A problem of categorization in this case is a wrong categorization in terms of the denomination of a group of functions. Problems of categorization occur mainly in the menu bar.

Previously one categorization problem was estimated to be severe: the function “creating a new speech element” is located in a context menu and can only be found if users open the context menu via right click in the vocabulary area (right area). This positioning violates the principle of visibility (Norman, 1988). Elements must be eye catching in order to attract users attention, beyond the demands of a good visual ergonomic design. The importance of this principle can be shown when considering participants’ difficulties in task 4. Here, the function “creating a new speech element” had to be detected. The result was clear: none of the participants could find the function. They searched nearly everywhere, consulting the online help and trying various strategies to find the function and to

solve the task. Most of the users assumed the function to be located in the database area. YE 02 explains in the retrospective interview, why he/she searched in this area: “I thought, on the left side, that there is the database and everything that already exists. I want to extend this [data set]. Probably I want to reuse it [new speech element]. And if you only create it [in the vocabulary area], it is only on the one device and not left [database].” (RYE 02). This explanation shows that the positioning of the function is not transparent and intuitively understandably, but rather causes severe interaction problems. As a consequence, users are hindered to solve the task and, in the last resort, are prevented to use the communication aid effectively. This problem is not age specific, neither older users nor younger ones could find this function. Schröder & Ziefle (2005) come to corresponding results: They tested menu structures of mobile phones with younger and older participants and found out that poor function categorization and assignment lead to interaction difficulties independent of the age.

Altogether no significant age differences were identified for the category structural problems.

Feedback

Feedback is the response (of a system) to the user. It shows which action has been conducted and which result has been achieved (Norman 1989). Feedback can be given in various forms: visual (change of an object or a status indication), acoustic (sound, alert) or textual via message or dialogue windows or in the status line of applications (Wagner, 2002). In this case problem reports, error messages and security queries belong to the feedback forms as well.

The expert evaluation led to four different types of feedback problems: wrong feedback, non-informative feedback, lack of feedback and imperceptible feedback.

A security query or verbal feedback is reasonable if a task comprises several steps in succession. Complex operations offer more potential spurious actions and identifying sources of errors is difficult. In task 5, creating a new speech element with title, message, voice output and picture, included such complex subtasks and several feedback problems, which caused severe interaction difficulties.

To activate the speech output for example five operating steps are necessary (see figure 4).

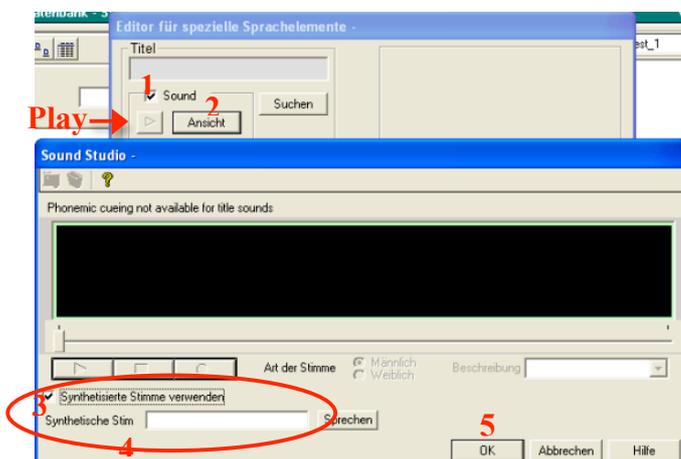


Figure 4: Operational steps configuring voice output

Because this example is rather complex, it will be explained in detail: the speech output for title and message can be edited in the dialogue window “editor for special speech elements”. Participants should configure the speech element in such a way that the message is verbalized by a synthesized voice (two voice options exist: natural voice and synthesized voice).

Therefore, a checkbox named “sound” beyond the inserting field for the message has to be activated first. By activating the checkbox, the button “view” next to it is enabled, indicated by the change of the font colour from grey to black (visual feedback). Then the user has to click the button “view”. A new dialogue window “sound studio” opens. In this dialogue window the checkbox “using synthesized voice” has to be activated. By doing so the upper area for editing existing digitalized sounds is disabled (visualized by a change of the button and type colour from black to grey) and the lower area for the voice output with a synthesized voice is enabled (visualized by the colour change of the button “speak” from black to grey). The fourth step to be conducted is entering the text into the inserting field next to the button “speak”. The action has to be finished by clicking the button “ok”. When the checkbox “using synthesized voice” has been activated and the dialogue window “sound studio” has been closed, a button to play the sound is enabled (indicated by a visual feedback: colour change from grey to blue).

In this subtask various problems interfered. Some participants did not even open the dialogue window “sound studio”. But those who did were confronted with the problem types *lack of feedback* and *imperceptible feedback*. The steps described above have to be conducted without any further information or instruction. Most of the participants who reached the dialogue window, activated the checkbox “using synthesized voice”, but then they did not recognize that they have to enter the message text into the designated inserting field. The mere existence of an inserting field is not indication enough to evoke participants’ attention and to bring them to enter text into it. An additional feedback, in form of a message or another hint is necessary. The next problem occurred immediately after, when users wanted to finish the configuration of the voice output: user closed the window via the “ok”-button, but did not get a feedback whether their action was successful or not. Irritated by this, some users opened the window again and wanted to check whether the voice output is correctly activated or not, but they did not exactly know how to find out. One participant criticizes this lack of feedback: “Here, in the “view”, I have to comment this again, there is nothing [no indication]. That is not user friendly. At least one could show a warning that this way [with this configuration] nothing is outputted.” (YE 01).

Even for those who were successful in activating the voice output with the correct message *imperceptible feedback* turned out to be problematic. Imperceptible feedback is a feedback that can hardly be seen or perceived by users because it happens outside his focus of attention and/or is too small or unremarkable. To explain this type the above mentioned example is picked up once again: the only way to ascertain that the voice output is activated is the enabled play button in the dialogue window “editor for special speech elements”, which can be seen if the “sound studio” is closed.

But the colour change of the button is nearly imperceptible for users because the time span between seeing the original

state (disabled button coloured grey before opening the “sound studio”) and the new state (enabled button coloured blue after activating the sound) is too long and the object and its change are too small. This *imperceptible feedback* problem made users insecure and they wasted a lot of time, because they were looking for a sign that the action was successful.

Within the same operation another feedback problem was detected: the play button sometimes also changes as well when no sound has been activated. This is obviously a bug. But in an interaction progress it must be evaluated as a *wrong feedback* given to the user. The effects of this problem type can not be estimated based on the user test because they are concealed by the first mentioned problem of imperceptible feedback: as participants do not notice the enabled play button as confirmation of an activated sound, they do also not recognize that it shows an available sound if there is none.

The problem type *non-informative feedback* is defined as feedback or messages that inform the user about the result of an action, but the contained information is incomplete and/or irrelevant and thus does not help the user to proceed in interaction. This problem type occurred for example in task 3. Participants had to search an existing speech element (“strawberry”) in the database. In advance, several correct approaches to search for elements were identified: the first and fastest is the search function in the database area. Furthermore, a separate dialogue window can be opened via the button “search” in the toolbar or the menu item “search” in the tool menu of the menu bar. OL 06 uses the database search, but enters the word “strawberry” unintended with leading blanks. Thus the system can not identify the query and responds with the message “TouchSpeak Designer” did not find a matching object.” This message is correct, but does not help the user to identify his typing mistake. It does not tell the user *why* no matching objects could be found. He tries various spellings (e.g. with quotation marks), but in all his attempts, he does not remove the leading blanks. OL 06 comes to the conclusion that the operation he conducted is incorrect: “I don’t know if the quotation marks belong to the entry, or if it is only ‘strawberry’ [without quotation marks that has to be entered]. But probably, I am on the wrong track.”

In this case the user is handicapped because of his limited sighting ability. He does not recognize his typing mistake and is not able to identify it on his own. The system does not support him. As the statement above shows, this usability problem does not only lead to the users’ failure in solving the task, but also to a loss of trust in his own competence. This behaviour is typical for older users (e.g. Göbel & Woo Yoo, 2005).

One of the younger participants makes the same mistake during another task, but he has no problems in detecting and correcting it. Frequently, it was observed that older users tend to make more often small mistakes, which they though do not recognize and which disturb the interaction.

Obviously, older adults need more often a correct, informative, and perceivable feedback than younger adults do – particularly they need support in trouble shooting. Younger people profit from good feedback as well. But they are rather able to locate, and eliminate mistakes on their own, and to cope with cognitively complex tasks without detailed instructions. Altogether this problem category tends to be age-specific.

Layout

The last category consists of five problem types: font problems, poor colouring and contrasts, problems of operational elements, inappropriate location, and disturbed relationship between form and function.

Font problems. This type subsumes problems of font type and size. It is recommended to use a sans-serif font type and at least 12 pt to 14 pt type size, 16 pt for headings (Zhao, 2001). Furthermore Coyne/Nielsen (2002) suggest for websites to offer a button to increase the type size for older people with visual handicaps (Coyne & Nielsen, 2002). In TouchSpeak a sans serif font type is used but headings size and button labels are too small. A possibility to increase font sizes is not available. This problem occurs in connection with a poor colouring respectively contrast.

Poor colouring and contrast. Poor colouring and contrasts are more serious for older adults due to limited sensory abilities (Kline & Scialfa, 1997). Therefore it is recommended to “maximize the contrast between foreground and background colors: use dark types on light or white backgrounds, exaggerate lightness differences between foreground and background colors, and avoid using colors of similar lightness adjacent to one another.” (Zhao, 2001). Looking at the headings in the sample application it becomes clear that they are designed suboptimally: white font on blue background, which can “cause the type to appear to ‘close’ in itself (Zhao, 2001). Also, too small font types and poor colouring (e.g. in task 2) hinder users to build up an appropriate mental model of the programme: only one (!) out of 14 participants did recognize the headings of the main areas.

Operational elements. This type summarizes problems of size, type ad labeling of operational elements. Because of functional deficits in fine motor skills and shiver older people often have difficulties in controlling the mouse (movements, clicking, dragging) and positioning the cursor pinpoint (Czaja & Lee, 2003). For this reason small operational elements should be increased, and scrollbars should be avoided as well as pull down menus. Between operational elements should be enough space so that older users do not accidentally click on the wrong element. “The larger the target and the more space between targets, the more likely it is that seniors will hit it on the first attempt” (Coyne & Nielsen, 2002). Keeping this in mind, it is obvious that the buttons in the database and in the individual vocabulary area do not meet the requirements of older users: they are undersized and too close to one another see figure 5).



Figure 5: Operational elements in the database area

Although participant OL 06 previously had changed the screen resolution (because of visual deficits he normally uses additional software to increase the screen display) he could hardly perceive the buttons. He describes his difficulties in the retrospective interview: “I have a programme and I help myself by changing to 800, 600 [dpi]. That is not as sharp as 1000. Yes, this, the labels [tooltips] are easy to see, but those [buttons] are curious.”

Furthermore, the buttons are not labeled. Only tooltips are displayed on mouse over and show their meaning. Icons and buttons should be labeled if they are ambiguous or they are unknown to some users (Horton, 1994). Irritation arose because well-known symbols (taken from everyday life) were used for programme specific elements, and, accordingly, could not be correctly interpreted as the following quote shows: “Left, I cannot imagine what this is. Second is some document. Third is television. Fourth is communication.” (OL 06)

The first four buttons are filter options for the database (see figure 5). The first icon (a book) stands for the function “show only general speech elements”. The proband cannot even identify the book probably because of its size – even less he knows what it means. The second (a document) represents “show only special speech elements”, and is interpreted as “some document”. The third (an easel) “show only pictures”, reminds the user of a television and the fourth (a stylized organigram, also used in other areas of the interface) means “show only vocabularies”. The user characterizes it as “communication”. He misinterprets the symbols despite of the explicative tooltips. Obviously they do not help him because he does not know the displayed terms. They are specific for the programme. Neither the labels nor the symbols/icons can activate correct mental concepts of the functions. In this case, the buttons should be clearly labeled with comprehensible terms to ensure that users can work effectively with the database area.

Test results show that the undersized operational elements are problematic for older users. Although missing labels (headings and labels of buttons/input fields) irritate younger users as well, they lead to severe problems for the elderly.

Inappropriate localization. This problem type includes both, absolute and relative positioning misunderstandings. Absolute localization positions of interface elements are suboptimal if users have difficulties in perceiving them because they are outside their attention focus. In western cultures the common reading direction is from left to right and from top to bottom. Accordingly, users’ attention focuses on the upper left quarter of a screen display. Other screen localization of objects provokes suboptimal information procession. In the evaluated software interface this ergonomic principle is violated (e.g. the inadequate spatial positioning of the scrollbar for the database area in list view; see figure 6).

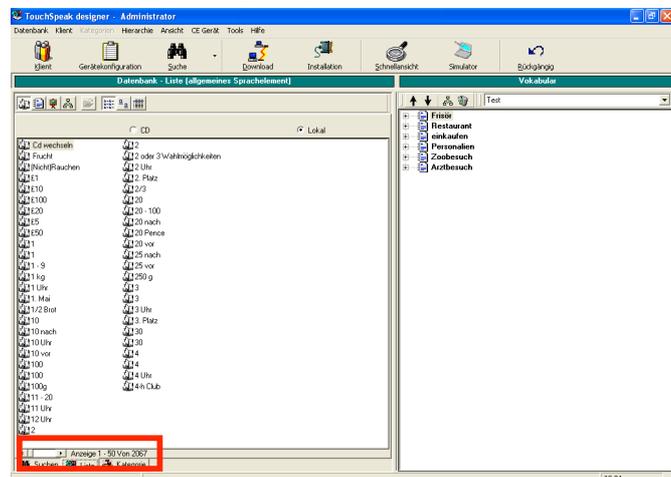


Figure 6: Inadequate positioning of undersized objects

Because of its misleading position and size many users do not recognize this operational element. Again, this shortcom-

ing mainly affects older users navigation behaviour, while younger users did recognize the scrollbar, but criticize the small size of this element. One younger user characterizes this in the retrospective interview very clearly: “Well, I don’t know but perhaps in this view here, probably these are too small. Probably if one is older it is difficult to read. The left area here, the three displaying options are very small and at first one doesn’t see it (...).” (RYE 02)

Obviously, all users recognize layout problems, but they only cause troubles for older ones. The same effect was revealed with relative localization problems. Spatial proximity of objects shows togetherness (law of proximity) (Goldstein, 1997). According to the law of proximity, objects (functions, buttons, headings, labels) that belong together should be bundled. In the search window that can be opened via toolbar or menu bar this law is violated: the label of the tab for search and the search field are separated. The distance between them is too big to indicate their togetherness and no other cues show the function of the field directly. This can be demonstrated in the next interview extract:

I: Why didn’t you recognize it [the search field]? Did you miss anything? Is there anything you required at this point?

OL 05: I actually didn’t read this “search”. I didn’t perceive it.

Coyne and Nielsen (2002) come to similar findings. In their web usability study for seniors, older adults had difficulties to identify a search field as such when it is not clearly labeled. As none of the younger participants in our study had comparable difficulties, the relative localization problem can be classified as an age specific problem.

Weak relationship between form and function. According to the design principle “form follows function” respectively “function follows form”, the form or design of things should be derived from its function and vice versa. This principle can also be applied to interface design: functions marked by a special visual appearance (form, colour) follow this principle and support an intuitive usage (Chisnell, Redish, & Lee, 2006). The same ergonomic benefit is given in conventionalized relationships, which have been established by experience and learnability effects to a standard over years of usage. Enabled input fields, for example, are usually white coloured, and have a three dimensional visual appearance. In the analyzed interface two violations of this convention have been detected: in the dialogue window “editor for special speech elements” input fields have a grey background though they are enabled. In the database area a large field with a white background was presented although it is no input field. By this colouring the conventionalized relationship between form (white) and function (input field) is disturbed. This caused irritation during the user test: in task 2, participants had to determine which functions the different areas of the software interface have. Most of them misinterpreted the database area as an editor, where new speech elements can be entered, as the following remark clarifies: “In area one, I think that it is a kind of editing field in which I can describe such an area of life and link it with commands, instructions, questions, and so on.” (OE 02)

Layout problems and ambiguities between form and function not only cause irritation, but also result in a mismatch between users’ mental model and designer’s conceptual model. Users cannot build up an appropriate mental model of the

system. This problem type impedes older users more than younger ones.

In summary the category layout problems shows a distinct tendency to be age specific. The problem types of this category mainly lead to difficulties for older users, while younger adults succeeded to compensate the ergonomic shortcomings.

DISCUSSION

In this study, we examined usability problems of older users when using specific software for aged users. Participants were asked to solve typical tasks, and were observed regarding basic difficulties and shortcomings of the interface design. In order to evaluate the extent of age-related problems, a control group of younger adults were also examined. Main research aims were the identification of problems, which are mainly age-exclusive (causing interaction difficulties only for older users) or age-specific (problems that cause serious interaction difficulties preferably for older people).

Interestingly enough, none of the identified problems turned out to be age-exclusive. The shortcomings of the interface caused irritation and interaction difficulties for both groups. This especially regards language and structural problems, which were problematic for both age groups and decrease usability independently from user characteristics. More age-specific shortcomings are caused by inconsistency, unclear feedback, or layout problems, which occurred more often for older users, causing severe troubles and perceived helplessness.

Older users especially failed whenever they had to solve complex tasks in combination with a suboptimal designed feedback. Complex tasks, comprising many steps in succession, offered more room for mistaking. Obviously, older users need more (linguistic) support (qualitative and quantitative) to identify and to correct mistakes. If there is no appropriate feedback, they cannot complete complex tasks successfully on their own. This result can be explained with sensorial and cognitive changes in older age: because of a declining perception (visual and auditory changes) in older ages it could be harder for older people to detect a visual or acoustic feedback particularly if the feedback is only a minimal change of an object or signal. For older people feedback should be designed more obvious, perceivable and permanent (Hawthorn, 2000; Wagner, 2002). In addition, linguistic messages could be conducive to the cognitive relief when complex operational sequences are divided into sub-steps and all sub-steps finish with a feedback message (security query). Thus the amount of information that has to be processed at the same time is lowered. The likelihood that mistakes are immediately detected is higher, and the user has more control over the ongoing processes.

Difficulties caused by inconsistency tend to be age-specific. This is reasonable because an inconsistent design increases the cognitive load, impedes automation and learnability of operating processes, as well as transferring acquired knowledge to unknown applications, and the location of relevant information (Nielsen, 1993; Shneiderman, 2004). Because of the cognitive changes in older age (e.g. longer information processing time, impeded information search, difficulties in solving novel problems and localizing objects) a consistent design is even more important for older than for younger users. In addition consistency is the most important precondition for under-

standing a technical system. Older users want to understand the technologies they use and therefore an inconsistent design that makes a technical system hard to understand hits older users more strongly than younger users (Jakobs, et al., 2008; Ziefle & Bay, 2004).

Focusing only on the language aspects of software interfaces it becomes clear that primarily a lack of linguistic elements (lack of function names, headings, feedback, linguistic hints or operational instructions) impedes the interaction for older users. It has to be proved if an increased usage of linguistic elements can improve software interfaces for older adults. Interestingly, older people tend to be more familiar with the medical domain and domain specific terminology whereas younger users are not. This result points out that the domain or using context and the users experiences with this domain have to be considered for the design of interfaces as well. This result could for example be relevant for the choice of metaphors, scenarios, and operational sequences of future applications.

One could argue that the problems the older adult group experienced are mainly due to the specific software which was under study. Even if this objection cannot be fully ruled out many of the findings reported here represent universal shortcomings in interface designs, which can be observed in many software applications and interface types, independently of the specific content and software environment. Based on this findings and the conclusion can be classified as transferrable to other software applications.

CONCLUSION

The conclusion to be drawn is that the visual and linguistic design of software interfaces has to be carefully considered to meet the requirements of older users. Age-friendly designs are not restricted to visibility and readability issues (font sizes, color or contrast). Still more important are the design of the interaction with the device, the design of (verbal) feedback and the usage of detailed operational instructions. Further studies will have to show if the need for detailed communication depends on the type of application, the type of device, the type of technology (e.g. information and communication technology), the domain or using context as well as the complexity of the interface. Overall, for an age-sensitive design it is indispensable to rely on the existing knowledge of different disciplines and to pursue a truly multidisciplinary approach. This multidisciplinary is the only way to fully understand the aging problem, to sensitively describe and examine aged users when using technology, and, to develop age-sensitive concepts.

REFERENCES

- Baltes, P., Lindenberger, U., & Staudinger, U. M. (1995). Die zwei Gesichter der Intelligenz im Alter [The two faces of intelligence in old age]. *Spektrum der Wissenschaft*, 10, 52-61.
- Baltes, P. B. (1991). The many faces of human aging: Toward a psychological culture of old age. *Psychological Medicine*, 21, (pp. 837-854).
- Chisnell, D. E., Redish, J. C., & Lee, A. (2006). New Heuristics for Understanding Older Adults as Web Users. *Technical Communication*, 53(1), 39-59.

- Coyne, K., & Nielsen, J. (2002). *Web Usability for Senior Citizens: Design Guidelines based on Usability Studies with People Age 65 and older*. Fremont: Nielsen Norman Group.
- Czaja, S. J., & Lee, C. C. (2003). Designing Computer Systems for Older Adults. In J. J. Jacko & A. Sears (Eds.), *The Human-Computer Interaction Handbook. Fundamentals, Evolving Technologies and Emerging Applications* (pp. 413-427). New Jersey/ London: Erlbaum
- destatis (2008). *Statistisches Jahrbuch 2008 für die Bundesrepublik Deutschland*. Wiesbaden: Statistisches Bundesamt Deutschland [Statistical Yearbook for the Federal Republic of Germany 2008]
- Göbel, M., & Woo Yoo, J. (2005). Anforderungen älterer Menschen an Moderne Technik und deren Dokumentation. In C. Schwender (Ed.), *Technische Dokumentation für Senioren* (Vol. 12, pp. 63-82). Lübeck Schmidt-Röhmhild [Older adults requirements for modern technologies and their documentation]
- Goldstein, E. B. (1997). *Psychology of perception*. Berlin: Spektrum Akademischer Verlag.
- Grice, H. P. (1975). Logic and conversation *Syntax and Semantics*, 3, 41-58.
- Hawthorn, D. (2000). Possible implications of aging for interface designers. *Interacting with Computers*, 12(5), 507-528.
- Horton, W. (1994). *The icon book: Visual symbols for computer systems and documentation*: New York: Wiley & Sons.
- Jakobs, E.-M., Lehnen, K., & Ziefle, M. (2008). *Alter und Technik*. [Age and Technology]. Aachen, Germany: Apprimus
- Jakobs, E.-M., & Villinger, C. (1999). „Das versteht kein Mensch...“. Verständliche Gestaltung von Hilfesystemen für Softwareprogramme [“Nobody understands this...”. Comprehensible design of help systems for software programmes.] In E.-M. Jakobs, D. Knorr & K.-H. Pogner (Eds.), *Textproduktion. Text, HyperText, KonText* (pp. 217-233). Frankfurt, Germany: Lang.
- Kline, D. W., & Scialfa, C. T. (1997). Sensory and Perceptual Functioning. In A. D. Fisk & W. A. Rogers (Eds.), *Human factors and the older adult*. (pp. 27-54). San Diego: Academic.
- Morrow, D., Miller, L.S., Ridolfo, H., Kokayeff, N., Chang, D., Fischer, U. and Stine-Morrow, E. (2004). Expertise and aging in pilot decision-making task. In Proceedings of the Human Factors and Ergonomics Society 48th Annual Meeting, Santa Monica, CA. (pp. 228-232).
- Nielsen, J. (1993). *Usability engineering*: Morgan Kaufmann.
- Norman, D. (1988). *The psychology of everyday things*: Basic books New York.
- Norman, K. L. (1991). *The psychology of menu selection: designing cognitive control of the human/computer interface*. Norwood: Ablex Publishing Corporation.
- Oetjen, S. & Ziefle, M. (2007). The Effects of LCD's Anisotropy on the Visual Performance of Users of Different Ages. *Human Factors*, 49(4), 619-627
- Park, D.C. & Schwarz, N. (1999). Cognitive Aging. Philadelphia: Buchanan.
- Röhr-Sendlmeier, U. (1993). Kompetenz im Alter: Der Beitrag der Technik. [Competence in the every day life: the contribution of technology]. *Paedagogische Rundschau*, 47(1), 69-83.
- Rudinger, G. (1996). Alter und Technik. [Age and Technology] *Zeitschrift für Gerontologie und Geriatrie*, 29, 246-256.
- Rudinger, G., & Jansen, E. (2005). Technik, Neue Medien und Verkehr [Technology, new media and traffic]. In S.-H. Filipp & U. M. Staudinger (Eds.), *Entwicklungspsychologie des mittleren und höheren Erwachsenenalters*. Göttingen, Germany: Hogrefe.
- Sackmann, R., & Weymann, A. (1994). *Die Technisierung des Alltags. Generationen und technische Innovationen* [The technization of the every day life. Generations and technical innovations]. Frankfurt, Germany: Campus.
- Schlick, C.; Ziefle, M.; Park, M. & Luczak, H. (2007). Visual displays. In J. Jacko and A. Sears (eds.). *The Human Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*. Chapter 11 (pp. 201-228). Mahwah, New Jersey: LEA.
- Schneider, N., Schreiber, S., Wilkes, J., Grandt, M., & Schlick, C. (2008). Foundations of an age-differentiated adaptation of the human-computer-interface. *Behaviour & Information Technology*, 27(4), 319-324.
- Schröder, S., & Ziefle, M. (2005). Semantic transparency of ellular phone menus. Comparing users from different age groups. *Computer Studies in Language and Speech* (pp. 302-315). Frankfurt, Germany: Lang.
- Shneiderman, B. (2004). *Designing the User Interface: Strategies for Effective Human-Computer Interaction* Amsterdam: Addison-Wesley.
- Vercryssen, M. (1997). Movement control and speed of behavior. In A. D. Fisk and W. A. Rogers (Eds). *Human factors and the older adult* (pp. 55-86). San Diego: Academic.
- Wagner, J. (2002). *Mensch-Computer-Interaktion: sprachwissenschaftliche Aspekte*: [Human-Computer- Interaction: linguistic aspects]. Frankfurt, Germany: Lang.
- Zajicek, M. (2001). *Interface Design for Older Adults*. Paper presented at the EC/NSF workshop on Universal accessibility of ubiquitous computing: providing for the elderly.
- Zhao, H. (2001). Universal Usability Web Design Guidelines for the Elderly (Age 65 and Older). www.otal.umd.edu/uupractice/elderly/
- Ziefle, M. & Bay, S. (2004). Mental models of Cellular Phones Menu. Comparing older and younger novice users. In S. Brewster & M. Dunlop (eds.). *Mobile Human Computer Interaction* (pp. 25-37). Berlin: Springer.
- Ziefle, M. & Bay, S. (2005). How older adults meet cognitive complexity: Aging effects on the usability of different cellular phones. *Behaviour & Information Technology*, 24(5), 375-389.
- Ziefle, M. & Bay, S. (2006). How to overcome disorientation in mobile phone menus: A comparison of two different types of navigation aids. *Human Computer Interaction*, 21(4), 393-432.
- Ziefle, M. & Bay, S. (2008). Transgenerational Designs in Mobile Technology. In J. Lumsden (Ed.). *Handbook of Research on User Interface Design and Evaluation for Mobile Technology* (pp. 122-140). IGI Global.