

LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN
Department "Institut für Informatik"
Lehr- und Forschungseinheit Medieninformatik
Prof. Dr. Heinrich Hußmann

Projektarbeit

Methods for Evaluating Peripheral Interaction in the Lab

Attila von Thermann
attila_von_thermann@gmx.de

Bearbeitungszeitraum: 24. 10. 2011 bis 1. 6. 2012
Betreuer: Doris Hausen
Verantw. Hochschullehrer: Prof. Butz

Zusammenfassung

In dieser Forschungsarbeit sammeln wir Literatur bezüglich peripherer Interaktion und verwandten Forschungsfeldern, um eine Definition von peripherer Interaktion zu liefern und die Grenzen zwischen peripherer Interaktion, Tangible User Interfaces, Benachrichtigungssystemen und Ambient Information Systems aufzuzeigen. Studienmethoden von Laborstudien im Bereich peripherer Interaktion sowie im Bereich angrenzender Forschungsfelder werden untersucht, anschließend werden Studienkonzepte aus verwandten Gebieten analysiert. Wir werden abstrakte Aufgabenarten auf verschiedene Weisen definieren: Art der Aufgabenstellung, verwendete Input- und Outputkanäle und zehn Aufgabendimensionen.

Der Hauptteil dieser Arbeit ist das Vorstellen einer abstrakten Methode zum Entwurf, zur Vorbereitung und zur Ausführung einer Laborstudie im Bereich der peripheren Interaktion, genannt PILS (“peripheral interaction lab study”). Wir werden Fragen zusammenstellen, die man sich während des Entwurfs der Studie stellen muss und Fragen, die man den Teilnehmern vor und nach der Studie stellen kann. Um unsere abstrakte Methode zu testen, haben wir sie in zwei Pilotstudien angewandt, deren Ergebnisse wir vorstellen werden.

Abstract

In this research paper, we will compile related work from peripheral interaction and neighboring research fields, to define peripheral interaction and to find and visualize borders between peripheral interaction, tangible user interfaces, notification systems and ambient information systems. Study methods of lab studies in peripheral interaction and neighboring research fields will be investigated, followed by a discussion of study methodologies from neighboring research fields. We will define task types in multiple ways: assignment type, used input and output channels and the ten task dimensions.

The key part of this work is the suggestion of an abstract procedure for designing, preparing and carrying out a study dealing with peripheral interaction concepts, labelled PILS (“peripheral interaction lab study”). We will gather questions to ask while designing a study and questions to ask the participants before and after the study. To test our suggested methodology, two pilot studies were carried out according to it and the results of and conclusions from these studies will be presented.

Aufgabenstellung

Peripheral Interaction is interaction in the periphery of our attention. In real life, for example, drinking a cup of coffee while talking to somebody, or walking while taking a photo. Adapted to the digital world it could be controlling your music with media keys while typing a text.

In this thesis, different evaluation methods, e.g. for dual-task studies, shall be analyzed and suitable methods shall be adapted for evaluation of peripheral interaction in the lab.

Tasks:

- Find related work and analyze it
- Adapt evaluation methods from related work concerning peripheral interaction
- Carry out user studies with your adapted evaluation method, already existing prototypes can be used for it (but may need adaption)
- Written thesis and presentation of your work

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbstständig angefertigt, alle Zitate als solche kenntlich gemacht sowie alle benutzten Quellen und Hilfsmittel angegeben habe.

München, 28. Mai 2012

.....

Contents

1	Introduction	1
2	Related Work	3
2.1	What is Peripheral Interaction?	3
2.2	Peripheral Interaction Research	3
2.3	Related Fields of Research	4
2.4	About Lab Studies	5
2.4.1	Studies for Peripheral Interaction	6
2.4.2	Relevant Lab Studies for Ambient Information Systems	6
2.4.3	Existing Study Concepts and Methodologies	7
3	Theoretical Part	9
3.1	Study Tasks, Patterns and Measurements	9
3.1.1	Implicit Measurements	9
3.1.2	Explicit Measurements	10
3.1.3	Analysis of Used Measurement Parameters	10
3.2	Conclusions About PILS	10
3.3	Primary Tasks	11
3.4	Guidance for Designing a PILS	12
3.4.1	Preparations	12
3.4.2	Study Procedure	14
4	Practical Part	15
4.1	Pilot Study 1: Just Visual or With Audio	15
4.1.1	Study Design Process	15
4.1.2	Software Implementation	16
4.1.3	Hypotheses	24
4.1.4	Study Procedure	24
4.1.5	Measured Values and Participant Feedback	25
4.1.6	Examination of Hypotheses	25
4.1.7	Conclusions	26
4.2	Pilot Study 2: Event-based or Continuous	26
4.2.1	Study Design Process	26
4.2.2	Software Implementation	27
4.2.3	Hypotheses	29
4.2.4	Study Procedure	30
4.2.5	Measured Values and Participant Feedback	30
4.2.6	Examination of Hypotheses	31
4.2.7	Conclusions	33
5	Summary and Outlook	35
5.1	Summary	35
5.2	Lessons for PILS	35
5.3	Outlook	36
6	Appendix	37
6.1	List of Analyzed Lab Studies	37
6.2	List of Possible Primary Tasks (By Channels)	38
6.3	List of Primary Task Scenarios	44
6.4	PILS Methodology Overview	45

6.5	Study 1 - Explanation Sheet	46
6.6	Study 1 - Questionnaires	49
6.7	Study 1 - Result Tables	51
6.8	Study 1 - Statistical Results	52
6.9	Study 2 - Explanation Sheet	55
6.10	Study 2 - Questionnaires	58
6.11	Study 2 - Result Tables	60
6.12	Study 2 - Statistical Results	61

1 Introduction

As more and more activities move from the analog to the digital desktop, we as users will find ourselves in multitasking situations. While user interface design for focal activities is an ample field of research, designing devices for peripheral use is still somewhat in an early phase. Since work with the periphery needs acclimatization, it seems like the only suitable type of studies for peripheral interaction is long-term studies. Yet, long-term studies consume a high amount of time and monetary resources. Especially for the early design stage of peripheral devices, it would be very helpful to have a way of receiving viable feedback in short time, with small costs. This is why we are investigating lab studies for peripheral interaction.

In this research paper, we will compile related work from peripheral interaction and neighboring research fields, to define peripheral interaction and to find and visualize borders between peripheral interaction, tangible user interfaces, notification systems and ambient information systems. Study methods of lab studies in peripheral interaction and neighboring research fields will be investigated, followed by a discussion of study methodologies from neighboring research fields. We will define task types in multiple ways: assignment type, used input and output channels and ten task dimensions: interruptibility, mental load, continuity of task, continuity of user input, task focus, inclusion of random factors, dependence on previous knowledge and abilities, penalty, reward and ambidexterity.

The key part of this work is the suggestion of an abstract procedure for designing, preparing and carrying out a study dealing with peripheral interaction concepts, labelled PILS (“peripheral interaction lab study”). We will gather questions to ask while designing a study and questions to ask the participants before and after the study. To test our suggested methodology, two pilot studies were carried out according to it. Additionally, we will describe the software code written for the pilot studies. It is designed in a manner allowing for adaption and extension, to reuse it in other studies dealing with peripheral interaction.

The first pilot study features two primary tasks, differing in the used input / output channels: a task combining visual output with haptic input and a task combining visual and auditive output with haptic input. These two primary tasks are being tested in combination with secondary pseudo-tasks using the periphery device prototype “Ambient Presence Indicator”. Regarding aforementioned task dimensions, both primary tasks should rank as follows: flexible mental load, low continuity of user input, task focus on both accuracy and speed, inclusion of random factors, low dependence on previous knowledge and abilities, no penalty. Both primary tasks are of the assignment type “react to event”.

The first pilot study was conducted with four research assistants, who also attended a group discussion to review the study methods. With the results and observations of the first pilot study and the suggestions of the group discussion in mind, the second pilot study was designed.

The main goal of the second pilot study was to get rid of some weaknesses of the first one and also bring in some variation. To achieve that, a new primary task was to be designed, accompanied by some changes in the procedure of the study. The new primary task, PT3, should differ from the old ones in terms of interruptibility, continuity, ambidexterity and reward. The new primary task is of the assignment type “carry out pre-defined activity”, the participant has to click on certain items to delete them from the screen. The new primary task ranks in the aforementioned task dimensions as follows: High interruptibility, flexible mental load, high continuity of task, low continuity of user input, task focus on accuracy and speed, inclusion of random factors, low dependence on previous knowledge and abilities, no penalty, inclusion of reward in scoring system and two additional rewarding displays, high ambidexterity.

The second pilot study was carried out with eight Media Informatics students and featured two primary tasks, one event-based from pilot study 1 (without audio), and the new continuous primary task, combined with the two secondary tasks from pilot study 1. The results of and conclusions from both of these studies will be presented herein. We will close the paper with some thoughts about the practicability of lab studies for peripheral interaction and an outlook to future work.

2 Related Work

To get a first impression of this thesis' subject, we shall start with a definition of what "peripheral interaction" means, followed by the description of some pieces of research. We will continue with an overview of related fields of research, characteristics of lab studies, closing with the presentation of study methodologies and measurement methods.

2.1 What is Peripheral Interaction?

In his technical report about tangible peripheral interaction, Edge[10] defines that peripheral interaction takes place when users perform "fast, frequent interactions with physical objects on the periphery of their workspace, to create, inspect and update digital information which otherwise resides on the periphery of their attention" [10].

Hausen and Butz[14] extend this definition by stating that in peripheral embodied interaction, it is characteristic to "keep the interaction [...] simple and casual, not requiring precise actions" [14].

2.2 Peripheral Interaction Research

To get a more precise impression of the subject area, we will lay down more features of peripheral interaction.

Hausen and Butz[13] continue their ideas about peripheral interaction by stating that there should be "no need for many commands" [13]. As Hausen[15] states, the goal is to "minimize disruption" to keep the level of cognitive load low; there should only be "occasional active interaction" [15]. Hausen and Butz[14] also propose a classification consisting of five design dimensions: explicitness, input mode, granularity, privacy and proximity.

Olivera et al.[30] want to explore "the possibilities of Peripheral Interaction through physical and simple actions"[30]. They describe peripheral interaction by comparing it with subtle interaction and showing up the differences. They state that peripheral interaction has the same concept as subtle interaction, but examines it from a different perspective. Subtle interaction "uses a concise piece of information to build up a dialog upon context"[30] while being completely focused, but peripheral interaction asks: "How much of our attention do we put into the interaction itself?"[30] The focus is on distributing "the interactions among the channels according to the nature and importance of the tasks" [30].

In 2001, MacIntyre et al.[20] proposed "Kimura", an augmented office environment, that can be seen as an early approach to working with the periphery to support the user's multitasking capabilities. Their project uses interactive peripheral displays to visualize sets of tasks clustered as "working contexts". Kimura helps the user by reminding her of past actions and reflects on every working context's primacy and recency, as well as on relationships between the working contexts. The conclusions drawn from this process influence the representation on the peripheral screen, illustrating the status of background activities.

In a theoretical overview aimed at enhancing the design of peripheral devices, Bakker et al.[3] explain attentional issues related to peripheral interaction. They describe two different approaches to attention: as a mental filter and as a finite mental resource. The selection process, labeled as "priming", is influenced by the current attentional focus, generally relevant topics (such as the own name) and other topics in the back of one's mind. So called "salient" stimuli can pass this filter without referring to one of the mentioned topics. At last, Bakker et al. define attention as "the division of attentional resources over potential activities" [3].

2.3 Related Fields of Research

As there are only few studies and research papers about peripheral interaction, we will also include observations from other fields of research that have to deal with similar problems and whose approaches and results might be applied to peripheral interaction. One of the earliest concepts related to peripheral interaction is calm technology. In a 1996 article describing a prototype called “Dangling String”, Weiser and Brown[38] define: “Calm technology engages both the center and the periphery of our attention, and in fact moves back and forth between the two” [38].

In *figure 2.1*, you can see a simplified Venn diagram of the research fields mentioned in this chapter, arranged in the dimensions interactivity and attentive context.

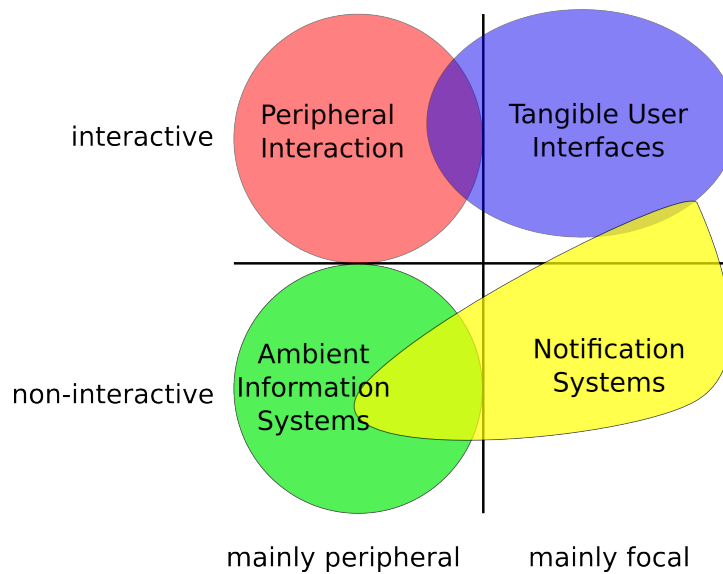


Figure 2.1: An overview of research fields related to peripheral interaction.

Peripheral Interaction can only be located in the interactive and mainly peripheral quarter of the diagram. Tangible User Interfaces can only be in the interactive half of the diagram, with only a minority of Tangible User Interfaces being designed for use in the periphery. Ambient Information Systems’ main task is offering information, so they are non-interactive. As the first word of the name indicates, they are designed for use in the periphery. Notification Systems are not clearly defined in terms of interactivity and peripherality, yet you can say that most notification systems will demand the user’s full attention, but only low levels of interaction, when they have something to report. This is why Notification Systems is mainly located in the focal and non-interactive quarter of the diagram.

Ambient Information Systems

The closest neighbouring research area to peripheral interaction is Ambient Information Systems. Pousman and Stasko[33] provide a taxonomy of ambient information systems, that partially overlaps with our definition of peripheral interaction: Ambient information systems “display information that is important but not critical”, “can move from the periphery to the focus of attention and back again” and “provide subtle changes to reflect updates in information”. They “should not be distracting”, but “aesthetically pleasing and environmentally appropriate” [33]. They identify four archetypes of ambient information systems: Symbolic Sculptural Display, Multiple-Information Consolidators, Information Monitor Display, and High Throughput Textual Display.

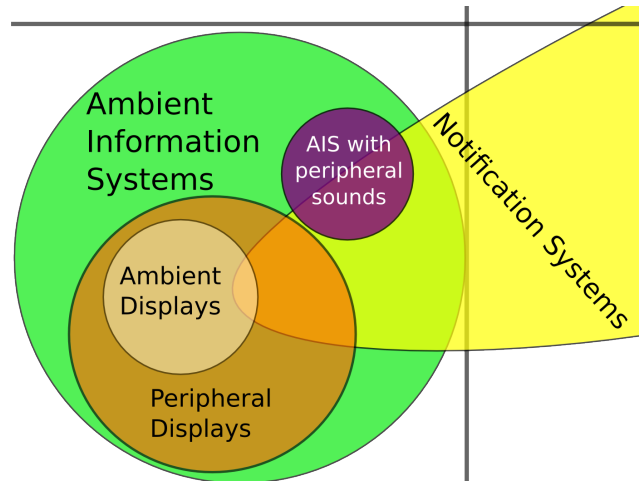


Figure 2.2: Detailed look at Ambient Information Systems.

Peripheral and Ambient Displays

A more concrete subset of ambient information systems is peripheral displays and ambient displays. Pousman and Stasko[33] establish a relation between the two terms by defining that “all ambient displays are peripheral displays, some notification systems are peripheral displays” [33], while ambient displays are the ones focusing on aesthetics. *Figure 2.2* visualizes the connection simultaneously to *figure 2.1*.

With an abstract toolkit, Matthews et al.[24] are aiming to assist at the development stage of the design process of a peripheral display. They identify two general sets of attention issues: user context and attention management, with the emphasis of their paper being the latter. They specify “three issues specific to conveying information on the periphery of human attention” [24]: abstraction of raw input, rules for assigning notification levels to input, and transitions for updating a display when input arrives. Matthews et al.[24] argue that not every information conveyed by the peripheral display is equally important and thus not every incoming information meant to interrupt the user. The designers of the display should address this issue by implementing visual representations and representation changes of different intensity.

In their abstract analysis paper, Matthews et al.[26] categorize peripheral displays in three dimensions with impact on study design: the first one is scope, the amount of activities the display is associated with. The higher the amount, the more complicated the evaluation. The second dimension is classes of supported activities, which can be the primary activity, a secondary activity or pending activities. The last one is criticality of the displayed information. These dimensions can be used to support the mapping from data input to output on the peripheral display.

2.4 About Lab Studies

When considering a lab study for peripheral interaction, one should regard that lab studies have certain characteristics. They can be carried out in short time, with low costs. But they are not appropriate for extensive evaluation. They make most sense when applied in an early stage of the design process, when there are many different concepts, to sort out unsuitable ideas and find usability problems. Similar concepts can be compared in their efficiency by measuring task performance. Lab studies are not a substitute for long-term studies, but can be helpful as an early addition. [34]

A general problem of lab studies for peripheral interaction is that, to move perception to the periphery, it may take a long time of familiarization; in the case of the Bakker et al.[4] study, it took multiple weeks. A typical lab study lasts only for about an hour, so the results of lab study dealing with periphery cannot be anywhere as accurate as results from a long-term study. One should keep in mind that the focus of a peripheral interaction lab study is not accuracy, but finding valuable conclusions in an early stage of the design process.

2.4.1 Studies for Peripheral Interaction

Hausen and Butz[14] carried out lab and long-term studies with the prototypes “Ambient Appointment Projector” and “Tangible Presence Indicator”, but were not satisfied with the results of the lab studies. Olivera et al.[30] carried out a user study with PolyTags, which are physical objects with 2D-codes printed on them. It turned out that the augmented objects lead to the users making fewer errors.

Edge and Blackwell[11] introduce a tangible user interface designed for work in the periphery of an office desk situation and describe its design process as consisting of four analysis stages: context, activity, mapping and meaning analysis. In their long-term study, they especially investigated matters of desk real estate, bimanual coordination and support for social interaction with co-workers.

2.4.2 Relevant Lab Studies for Ambient Information Systems

In a research paper about ambient information systems, Hazlewood et al.[17] state that usability is an important, but not the most important matter in their study: “Usability tells us whether or not certain features are working”. Initially, they explain that lab studies are in general not suitable for work with periphery, as the integration of a new object “cannot be accurately simulated inside a controlled laboratory, which is outside the observer’s everyday environment”. Yet, their approach is, if one decides to still conduct a lab study, the key factor would be to have participants carry out “some sort of task which would distract them enough to force the AIS out of their primary attention” [17].

Ames and Day[1] collect important aspects of studies for ambient displays in their overview paper, e.g.: Do people want/need to know the information the display provides? How often do they want to look? Is the display easier to monitor and understand than the information-gathering technique it is replacing? Does the display show information at the right level of detail? Is the display too peripheral to get people’s attention when needed?

Hsieh and Mankoff[18] conducted a comparative study with two peripheral displays, where they measured three main factors: usability, awareness and distraction. In the lab part of their study, subsequent to a heuristic evaluation, they used “selfreporting of awareness and distraction” and “objective records of performance on questionnaires” (speed and accuracy of sorting emails). For measuring usability, participants “were asked to rate the displays against the heuristics used in the former heuristic evaluation” [18].

Shen et al.[36] describe studies with two different ambient display prototypes: Fisherman and MoneyColor. Their definition of ambient displays elaborates Pousman and Stasko’s[33]: “Ambient displays typically communicate just one, or perhaps a few at the most, pieces of information and the aesthetics and visual appeal of the display are often paramount” [36]. In the Fisherman experiment, they determined the “order of factors by disruptiveness for ambient displays: Animation, Color, Area and Position” [36]. In the MoneyColor experiment, they used a primary task called “SquareClick” to distract users from the periphery ambient display. Squares appear randomly on a screen and have to be deleted by the participant clicking on them; speed and error rate are measured.

Bakker et al.[4] present a paper with three prototypes using peripheral sounds, called RainForecasts, AudioResponse and EntranceSounds. In their long-term studies, they observed that the perception of the sounds by the participants shifted to the periphery after about three weeks. They also determined that “the relevance of the information is related to the extent to which the sound representing it is experienced as disturbing” and that “the systems were most useful when the information conveyed by the sounds differed from what the user expected”. The second observation is closely linked to the assumption that in the overwhelming majority of cases “a sound was played, no new information was conveyed” [4].

2.4.3 Existing Study Concepts and Methodologies

As stated before, there is not much work about peripheral interaction yet. Correspondingly, there are no existing methodologies for evaluating peripheral interaction devices. We will present concepts from neighbouring research fields and try to find approaches applicable to peripheral interaction.

The first study concept we want to present is DECIDE by Preece et al.[34], which consists of six steps and is aimed at user interface evaluation. In the first step, evaluation goals should be determined, based on the life cycle stage of the prototype. Secondly, specific questions of the project should be explored. The questions should be selected in a manner that helps reaching evaluation goals from step one and should be broken down to sub-questions. In the third step, evaluation paradigms and techniques should be chosen. Next, practical issues like site, suitability, flexibility should be identified, before preparing hypotheses, materials and parameters. In the fifth step, ethics issues concerning participants and tasks should be dealt with. Finally, the actual evaluation can take place after a pilot study, followed by evaluation, interpretation and presentation of data.

Shen[37] derives his methodology for evaluating ambient displays from Preece et al.’s[34] DECIDE concept, amending it with abstract input and output information for each step: evaluation progress starts with the prototype and proceeds to general evaluation goals, specific evaluation questions, detailed evaluation attributes and criteria, specific evaluation issues, legal specific evaluation issues, until finally returning the evaluation results. Preece et al.’s[34] first step, “determine the overall goals”, is divided into “identify life cycle stage” and “determine specific evaluation goals” in Shen’s[37] concept. In step three, attributes like distraction, comprehension, memorability and criteria (e.g. efficiency and effectiveness), should be taken into account. Another division into substeps takes place at Preece et al.’s[34] step four, “identify practical issues”, which Shen[37] divides into “pre-consideration” about site, subject and schedule, and “basic consideration” about hypotheses, materials, parameters and tasks. He also adds details regarding ambient displays to each step, like amending the evaluation attributes (step 3) distraction, comprehension, memorability, by the addition of aesthetics.

Matthews and Mankoff[25] developed a toolkit for “rapid prototyping of peripheral awareness displays”. After interviewing professionals from computer hardware companies, they describe five evaluation metrics: Awareness, Distraction, Appeal, Learnability and Effects of Breakdowns. In their context, they define awareness as the “amount of information shown by the display that users are able to recall, understand, or use” [25]. Appeal includes usefulness and aesthetics. Learnability is “the amount of time and effort required for users to learn to get information from the display” [25]. Effects of Breakdowns could be described as “error management”, its goal is “to make breakdowns obvious and recovery easy” [25].

Based on the work of Nielsen and Molich[29], who posited ten usability heuristics for user interfaces in general, Mankoff et al.[23] deduced twelve heuristics for the design of

ambient displays: sufficient information design, consistent and intuitive mapping, match between system and real world, visibility of state, aesthetic and pleasing design, useful and relevant information, visibility of system status, user control and freedom, easy transition to more in-depth information, peripherality of display, error prevention, flexibility and efficiency of use. They compared their set of heuristics to the original ones in a study, substantiating the aptitude of the new heuristics for the field of ambient displays. Apart from that, Mankoff et al.[23] also address the general discrepancy in the prioritisation of the tasks: the secondary/periphery task, which is presented as the one with lower priority to a study participant in a study with ambient displays, is the more important task for the results of the study.

McCrickard et al.[28] provide “a unifying model that can guide evaluation efforts for notification systems” [28], the IRC model. The three parameters cited in the acronym, interruption, reaction and comprehension, are subsequently used as dimensions to categorize and evaluate a collection of existing notification systems. Interruption signifies the degree of need for attention reallocation. Reaction indicates the need for “rapid and accurate response to stimuli”. Comprehension denotes the users’ “goal of remembering and making sense of the information [...] at a later time”. McCrickard et al. want to use their three parameters for a “direct and manageable measure of the design’s ability to serve its purpose” [28]. *Figure 2.3* exemplifies the IRC model in the overview of *figure 2.1*.

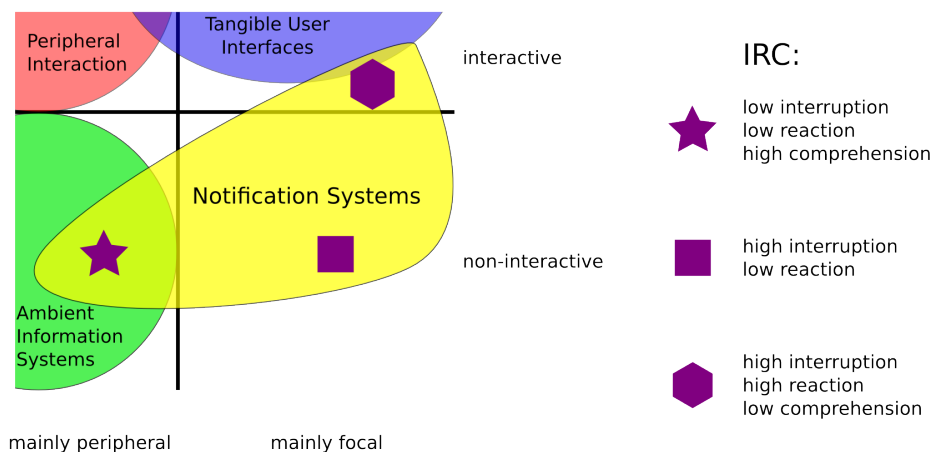


Figure 2.3: Detailed look at Notification Systems, comprising the IRC model.

Shami et al.[35] proposed an evaluation framework for peripheral displays that “depends on active user participation and emphasizes the experience of using peripheral displays” [35] rather than measuring error rates or task completion times. Their “Context of Use Evaluation of Peripheral Displays” consists of three steps, all based on an individual scenario of possible usage. The first one is “scenario building”, a discussion between designer and user, where a scenario is created “that captures how the information presented in the display is accessed”. In the second step, “scenario enactment”, the user performs a primary task while having the opportunity to use the peripheral display, according to the scenario created in step 1. The last step is “scenario reflection”, where the user fills out a questionnaire, rating the display’s “defining attributes”. Shami et al.[35] posit 5 categories of questions: noticeability, comprehension, relevance, division of attention and engagement. In the end, the experimenter analyzes the whole evaluation process.

3 Theoretical Part

In the theoretical part, we will describe various ways to categorize tasks: assignment type, task dimensions and used input and output channels, with a list of combinations and example task scenarios. We will present implicit and explicit measurement methods as well as measurement parameters and describe the results of our meta-analysis of 17 lab studies. The last subsection will give detailed instructions about how to set up and carry out a peripheral interaction lab study.

3.1 Study Tasks, Patterns and Measurements

There are two general ways of extracting results from a lab study. The most obvious one is to measure a participant's performance directly. To quantify a participant's lab study performance in a manner yielding viable results, clearly defined tasks with measurable parameters are needed. We are describing three basic task types suitable for clear definition and offering easily measurable parameters:

3.1.1 Implicit Measurements

- One basic task type described is “find mistakes”. Maglio and Campbell[21] describe their study task, where “errors were introduced into [...] texts by hand according to three rules” [21]. But finding a mistake is not limited to textual information; it can also be applied to photographs, mathematic sequences et cetera. Error/success rate, and speed can be measured and probably be itemized by types of errors.
- Another basic task type would be “react to events”, like in Shen et al.'s[36] task “SquareClick”. Error/success rate and speed can be measured.
- Finally, a very abstract basic task type is “carry out pre-defined activity”. There are two general approaches: giving the participant clear instructions on the task, or giving rough goals and observing the participant's course of finding a solution. Error/success rate and speed can be measured. Additionally, the sequence of actions can be recorded.

The very nature of peripheral interaction requires a multi-tasking approach for evaluation. There should be a primary task, in the focus of the attention, and a secondary task, in the periphery of the attention, whereas the periphery task is the one more important for the results of a study dealing with periphery interaction. A key aspect of multitasking is task switching: When does a user switch tasks, why, and for how long. Benbunan-Fich et al.[7] describe seven multitasking metrics based on activity theory, combining the three domains user, task and computer. Subsequently, they apply these metrics to user self-reports of task switching. In another research report, Benbunan-Fich et al.[6] “demonstrate how to collect and estimate the different multitasking measures” [6] they are proposing.

A common modus operandi of multitasking studies is to work with a baseline. It means that a participant's performance without the validity of a certain condition is compared to the participant's performance during the validity of the condition. In the multitasking case, the participant's performance when processing a primary task only is compared to the participant's performance when processing an additional task concurrently. [34]

One advantage of studies in computer science is that the measurement of parameters can be built into the code of the study software. Especially reaction times and error rates can be extracted comfortably, as well as degradation from the baseline performance. Abstract concepts like a scoring system can be implemented and visualized to give the participant impromptu visual feedback about her performance, which can be motivating.

Other possible parameters like number of focus shifts from focal to peripheral task or comprehension time can only be extracted by visual observation.

Observing the study participant can also be filed under implicit measurements. The observation can be carried out during the study or, if the study has been recorded, afterwards. In most cases, an automatic mensuration of measurement parameters cannot be realized when it comes to observation.

3.1.2 Explicit Measurements

Another way to extract results from a lab study is asking the participant. This can be done with vocal communication or via paper or computerized forms. There are three basic types of implicit measurements: quiz, user impression feedback and technical user feedback.

- In a quiz, the experimenter is asking questions about a media content the participant has been exposed to in an earlier stage of the study, so the main task of the participant is recalling information.
- When obtaining user impression feedback, the experimenter is asking questions about the participant's opinions about the study subject and about the impressions the participant gathered during the course of the experiment. The main task of the participant is recalling feelings and exploring opinions.
- When obtaining technical user feedback, the experimenter is asking the participant about what she would change to improve the study subject. The participant's main task is to think of suggestions for improvement.

3.1.3 Analysis of Used Measurement Parameters

To get a better understanding of which methods and measurements are currently being used, we analyzed 17 lab studies from peripheral interaction and neighbouring research fields. For a complete list of the 17 studies, see *Appendix, 6.1*. Five studies used a quiz, five reported of collecting user impression feedback, five measured error rate or accuracy, four measured task completion time, five measured reaction time or detection time or speed, six used a baseline to compute degradation. Only one of the studies reported to count the number of focus shifts.

3.2 Conclusions About PILS

After working through all the bits of information gathered in the preceding chapters, some characteristics emerge that a Peripheral Interaction Lab Study (PILS) will have to feature. To allow for the participant's maximum possible acclimatization to the peripheral device, training time is paramount. Yet, study time is precious, so a thorough consideration of training time is needed. Another way to enhance acclimatization is the minimizing of unrelated sources of irritation, like sounds from outside of the laboratory, eye-catching items in the lab and display environment, or unnecessary pop-up windows in the study software. It is important that both the primary and the peripheral task are clear and comprehensible; the difficulty of a primary task should not derive from interpretation challenges.

A different way to make a lab study more reliable is to approximate the surrounding conditions to the conditions of an actual use case. If the peripheral device is meant to be used in an office, the participant should be set into an office-like desk scenario. Another part of the surrounding conditions is the primary task. If it is possible, a primary task

similar to the type of tasks a user would carry out when in the predefined use case should be applied.

All in all, the primary task should not be considered as merely a means of distraction. Furthermore, in this work, primary tasks play a major role. The periphery task will be individual for each peripheral device, but we are proposing the implementation of reusable primary tasks. Once an adequate collection of reusable primary tasks is available, designing a new Peripheral Interaction Lab Study will be a lot easier.

Some important questions, based on Matthews et al.'s [24] research, that a PILS might answer, are “Is the peripheral device suitable to the user’s context?”, “Is the user’s attention managed in a suitable way?”, “Should the device be interrupting?” and “Should it be interrupting at every occasion?”. Two important questions about a peripheral device derived from Bakker et al.’s [3] paper would be “Does the peripheral device make its way through attention as the user’s mental filter?” and “Does the peripheral device consume the right amount of attention as a finite mental resource?”.

3.3 Primary Tasks

To get an overview of possible types of primary tasks, multiple approaches are possible. Our first classification divides by input and output modalities, using auditive, haptic and visual. When leaving out combinations with multiple computer input modalities, 27 more or less meaningful task types are found (e.g. “haptic input combined with visual output”). For a complete list of the compilation, see *Appendix, 6.2*.

The reason why we left out combinations with multiple computer input modalities is that those combinations can be easily derived from combinations with a single computer input modality, and including combinations with multiple computer input modalities would have made the list too long to use in a reasonable manner.

In the second step, we tried to find an abstract example task for each combination, which you can also see in *Appendix, 6.2*. Third, we combined similar example tasks to boil down 27 abstract example tasks to seven exemplary primary task scenarios (with variations). With these seven scenarios as starting points, most primary task needs can be met when it comes to input and output modalities. For a complete list of the scenarios, see *Appendix, 6.3*.

Apart from input and output modalities, we also want to explore possibilities in primary task design and selection by investigating dimensions of tasks. We collected ten task dimensions:

- **Interruptibility.** What effect does a focus shift have on the execution of the primary task? Low interruptibility: Focus shift has large detrimental effect on task performance. High interruptibility: Focus shift has no or marginal negative effect on task performance.
- **Mental load.** How immersed will the participant be in the primary task? Low mental load: The primary task only requires a low amount of attention from the participant. High mental load: The primary task requires a high amount of attention from the participant. Leung et al.[19] use Signal Detection Theory to “quantify how heavily participants were loaded by the [...] tasks” [19].
- **Continuity of task.** Will the participant be confronted with the permanent opportunity to carry out task-related operations? Low continuity of task: The participant’s assignment does not require her to carry out operations all the time. High continuity of task: The participant’s assignment requires her to carry out operations all the time.

- **Continuity of user input.** Will the participant be confronted with the demand to permanently use input devices? Low continuity of user input: The participant does not have to operate input devices all the time. High continuity of user input: The participant has to operate input devices all the time.
- **Task focus:** Is the main focus of the task accuracy or speed or a combination of the two?
- **Inclusion of random factors.** Are random factors included into the task course?
- **Dependence on previous knowledge and abilities.** Will a participant with a certain knowledge or a certain ability perform better than another participant?
- **Penalty.** Do penalty factors for incorrect or missing user input actions have influence on the measured performance? Are there additional penalty factors like sounds, pop-ups, text displays?
- **Reward.** Do reward factors for correct user input actions have influence on the measured performance? Are there additional rewarding factors like sounds, pop-ups, text displays?
- **Ambidexterity.** This applies to haptic input only. Will the participant's assignment be accomplishable with the use of only one hand? Low ambidexterity: The assignment can be accomplished with only one hand. High ambidexterity: Both hands need to be used to accomplish the assignment.

When it comes to penalty and reward, it is obvious that at least one of the factors must be included to measure performance at all. Yet, an additional inclusion of the other factor could make the performance measurement more meaningful and the task more motivating for the participant.

3.4 Guidance for Designing a PILS

In this section we want to collect all the hints and advices for designing a Peripheral Interaction Lab Study (PILS).

3.4.1 Preparations

The first part of the preparations is to become clear of the focus of the study. Is it about usability, acceptance or comparison? The second part is specifying the peripheral task and choosing or implementing a primary task; is a quiz-type or a performance-type primary task more suitable? Should the same input and output channels as in the secondary task be used or different ones? How should the primary task rank in the ten primary task dimensions? The next step is forming your study hypotheses. They heavily depend on the focus of your study and the nature of the primary and secondary task. Some abstract examples for hypotheses: Use of device improves multitasking performance by at least x . Device fulfills usability standards X . Users accept device. Device degrades PT performance by less than x .

As much as in other computer science evaluation studies, questionnaires or interviews are a valuable way of complementing deductions from measurements and thus gathering information in PILS. The next step is to construct your questionnaires or interview questions. Will you be having a pre- and a post-study questionnaire/interview or just one of them? We gathered some of the most important questions for PILS:

- **Multitasking behavior.** How does the participant estimate her own multitasking behavior in working with computers?
- **Usage of input devices.** Is the participant used to operating periphery input devices?
- **Experience with secondary task.** Is the participant used to carrying out tasks similar to the secondary task chosen for the study?
- **Satisfaction with past interaction type.** If the participant is used to carrying out tasks similar to the secondary task, is she satisfied with the hardware/software she typically uses?
- **Importance of task type and related information.** If the participant is used to carrying out tasks similar to the secondary task, is the task and the information it provides important to her?

Questions only applicable in a post-study questionnaire:

- **Amount of attention.** How does the participant feel about the amount of attention she allocated to the periphery task? Did she find the peripheral device distracting? Did she find the peripheral device perceivable?
- **Satisfaction with peripheral device.** Did the device do what the participant wanted it to do? [34] Did the participant like the device? [34] Did the aesthetic design appeal to the participant? [34] Does the participant want to use the peripheral device again? [34]
- **Problems.** Did the participant have problems using the peripheral device? [34] Did the participant feel comfortable or overextended while multitasking?
- **Learning curve.** Did the participant find it hard to learn how to use the peripheral device?

To complete the capturing of the participant's opinion about the peripheral device, the participant could be asked to rate the device against employable heuristics. In the following, we will list a selection of Mankoff et al.'s[23] heuristics for ambient displays:

- **Sufficient information design.** Does the device convey “just enough” information?
- **Consistent and intuitive mapping.** Is the state of the device easy enough to understand?
- **Match between system and real world.** Does the participant understand symbols and words?
- **Visibility of state.** Are transitions from state to state noticeable?
- **Aesthetic and pleasing design.**
- **Useful and relevant information.**
- **Visibility of system status.** Does the device give appropriate feedback within reasonable time?
- **User control and freedom.** Are there enough “emergency exits”?

- **Easy transition to more in-depth information.**
- **Peripherality of display.** Is the device easily monitorable?
- **Flexibility and efficiency of use.** Are there accelerators for experienced users?

It is also important to leave open space for annotations, both in an interview and on paper. Sometimes, the most important bits of information come from seemingly negligible side notes. The last step of the preparation would be to determine the way of capturing the study. The most common possibilities are capturing video, audio, photos or taking notes.

3.4.2 Study Procedure

Finally, we want to describe a general process flow for a Peripheral Interaction Lab Study. For a single page overview, see *Appendix, 6.4* The room in which the study takes place should be properly prepared, with all devices, for the study and for the recording, arranged in a manner suitable for carrying out the study smoothly. The experimenter should have a seat close to the participant, where she can observe the participant and her actions easily without being in the participant's direct field of view. The study starts with a short briefing of the participant, where she gets some background information about the study and a rough description of the tasks she will be going through.

Another important point is to tell the participant that during the study, she should not consider incorrect actions as her mistakes, but as her help to the experimenter to find weaknesses of the tested devices and software. After a pre-study questionnaire, the training begins. To make accustomization easier for the participant and lowering her stress level, the experimenter should emphasize that performance in the training rounds will not be regarded. The next step is the baseline round, followed by the main test. Afterwards, the post-study questionnaire completes the collection of data.

4 Practical Part

In the practical part, we will describe the preparations, software code, execution and results of two pilot studies developed according to the proposed Peripheral Interaction Lab Study approach. The pilot studies were carried out to investigate the usefulness of PILS. Four conditions were tested in each study in within-subjects manner, using latin square allocation for serializing the task combinations.

4.1 Pilot Study 1: Just Visual or With Audio

The first pilot study was carried out in January 2012 with four research assistants from the Media Informatics Department of LMU. It featured two primary tasks, one with visual and audio, one just with visual output modality, and two secondary tasks comprising the peripheral device “Ambient Presence Indicator”.

4.1.1 Study Design Process

The key matter of designing this study was defining and implementing suitable primary tasks. Based on the 27 input and output modality combinations described earlier (see *Appendix, 6.2*), two combinations prevalent in actual computer use were selected: haptic input combined with visual output; and haptic input combined with visual and auditive output. Predefined goals of the primary task design process were designing a well-defined task that is easily measurable, customizable in its level of difficulty, expandable and usable for both input / output combinations.

Regarding the task dimensions in *section 3.3*, the primary tasks should rank as follows: flexible mental load, low continuity of user input, task focus on both accuracy and speed, inclusion of random factors, low dependence on previous knowledge and abilities, no penalty. Task dimensions not concretely regarded in the design process of this pilot study were interruptibility, continuity of task, reward, ambidexterity.

As a starting point, similar to Scenario 1a in the list of primary task scenarios (see *Appendix, 6.3*), Shen et al.’s[36] “SquareClick” task was chosen. In SquareClick, items appear randomly on the screen and the participant has to click on them with a mouse as soon as possible. A problem of SquareClick is its dependence on Fitts’ law [12], as the participant has to move the mouse cursor every time a new item appears, and thus the spatial distance between two items appearing subsequently is an influential factor. To get rid of this problem, the task was changed: instead of moving the mouse, the participant has to press a button dependent on the shape and the color (just visual task) or the shape and a sound (visual and auditive task) when an item appears. The position of the appearing item is no longer a disruptive factor. Accuracy, as the amount of correct button presses compared to the amount of all button presses, and reaction time are measured. The mapping of item attributes to keys can be seen in the explanation sheet, *Appendix, 6.5*. For a screenshot of the primary tasks, see *figure 4.1*.

For the secondary task, a classic reaction task was chosen. The four “colleague lights” of the Ambient Presence Indicator are randomly changing their color. When a certain condition dependent on the color of the four colleague lights occurs, the “main light” knob has to be turned, so it also changes its light to a certain color. For secondary task 1, the reaction condition is “dominant color of the colleague lights changes”, so the participant has to keep the main light shining in the same color as the majority of the colleague lights. For secondary task 2, the reaction condition is “bottom colleague light changes color”, so the participant has to keep the main light shining in the same color as the bottom colleague light. The participant is granted a maximum reaction time. If she does not turn the main light to the correct color within this time frame, the subtask will be counted as failed,

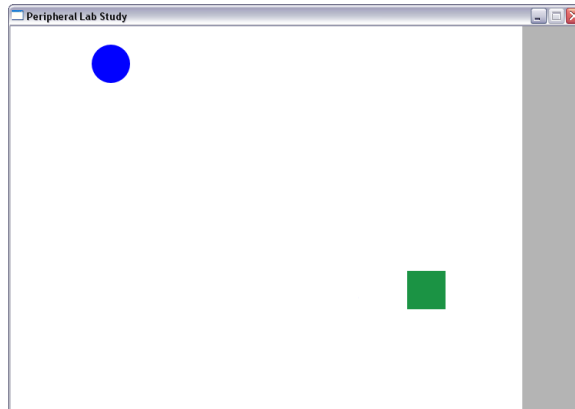


Figure 4.1: Screenshot of a primary task in the first pilot study

incorporated in the error rate. If she does turn the main light to the correct color within the time frame, the reaction time is measured.

For the pre-study and post-study questionnaires, questions were selected from the collection in *section 3.3*. The questionnaires can be found at *Appendix, 6.6*. In the pre-study questionnaire, a short description of upcoming tasks was given, and experiences with input devices and multitasking were determined. In the post-study questionnaire, demographic information, the participant's opinion about attention distribution, peripherality and a non-embodied alternative to the peripheral device were traced. Finally, the participant was asked to rate the peripheral device against heuristics.

4.1.2 Software Implementation

The code for this study was written in Python and Pygame. For the communication with Clara Lueling and Simone Rodestock's[16] arduino code of the Ambient Presence Indicator, pySerial was used. The classes are separated in a manner allowing for a simple substitution of primary and secondary tasks, there are 5 files: Main.py, PriTask.py, SecTask.py, SerialManager.py and Util.py. Main.py is the main file used for starting, offering variables and incorporating the main loop and display. PriTask.py and SecTask.py handle the respective tasks, SerialManager.py carries out the communication with the Ambient Presence Indicator and Util.py provides help and calculation functions.

The main loop of the study implements a round concept with separate rounds for primary task and secondary task. The while loop is basically doing nothing but checking if the round time is over. When the round time is detected to be over, the algorithm is initiating actions: a change can happen with a predefined chance, e.g. a primary task item could appear. The predefined chances of change are also influenced by other factors, e.g. amount of rounds since last change, number of items visible.

The code automatically collects data during the course of the test and writes it into a text file. After one study round of carrying out a task mode has been completed, evaluation takes place. The text files written during the study are being read in, and a new text file with results is being created in the same directory. Thus, study and evaluation are independent from each other.

The code of all files will be explained separately, starting with Main.py. There will be an abstract overview of the contents of each file, followed by a closer look into implementation details.

Main.py

```

#START VARIABLES
#DEBUGGING VARIABLES
#TUNING VARIABLES
#ALGORITHM VARIABLES
#PREPARATIONS FOR TXT FILES, PYGAME DISPLAY+SOUND
#INITIALISING TASK MANAGERS
#WAITING LOOP FOR START SIGNAL
#MAIN STUDY LOOP
#CLOSING TXT FILES, REREAD TXT FILES
#EVALUATION
#WRITE+CLOSE RESULT CSV FILES

```

This is the abstract structure of Main.py, the main file for running the study software. It offers plenty of manageable variables to adapt study conditions, especially for the difficulty level of the study. We will explain all parts of the code in chronological order, beginning with the “start variables”:

```

#START VARIABLES
username = "USER_4"
taskMode = Util.TASK_MODE_A
portName = "COM3"

```

The first part of Main.py offers “start variables”, which are those variables that need to be adjusted most likely before each start. User name and task mode influence the name of the folder result data will be written into, as well as time and date of running Main.py. Task mode can have the values “Util.TASK_MODE_A”, “Util.TASK_MODE_B”, “Util.TASK_MODE_C”, “Util.TASK_MODE_D”, “Util.TRAINING_MODE_A”, “Util.TRAINING_MODE_B”, “Util.TRAINING_MODE_C” and “Util.TRAINING_MODE_D”. PortName refers to the COM-port, to which the Ambient Presence Indicator is connected. This can vary from system to system and should be determined before the study, e.g. by looking it up in the device manager.

```

#DEBUGGING VARIABLES
debug = False
debug_buttons = False
debug_rounds = False
debug_changes = False
debug_lights = False
debug_secInput = False
debug_eval = False

```

Debugging variables can be used to debug certain parts of the code, like button input, round values and intermediate results of evaluation.

```

#TUNING VARIABLES
screenwidth=667
screenheight=500
symbolsize=50
maxTestTime=300.0
endTimeForThinking = 3.0
lightness = 50

```

In the tuning variables, pivotal values for the study are configured. The difficulty level of the study can be heavily influenced with these values and, once adapted, they must stay the same for each participant. The only exception is the lightness of the Ambient Presence Indicator, which must be adjusted manually between values ranging from 0 to 64 to fit changing light conditions. Symbol size defines the maximum height and width of appearing visual objects, in pixels. MaxTestTime is the time for each study round in seconds. EndTimeForThinking assesses a timespan at the end where no new items appear.

```
#TUNING VARIABLES (cont.)
baseRoundTime_p=1.5
maxRoundExtraTime_p=0.5
baseRoundTime_s=2.7
maxRoundExtraTime_s=1.0
estimatedMaxReactionTime_p=4.0
maxErrorsBeforeSearchCancels_p=4
estimatedMaxReactionTime_s=9.0
```

In the second section of the tuning variables, round times are defined. BaseRoundTime_p declares the minimum time for a primary task round in seconds. A random fraction of maxRoundExtraTime_p will be added to find out the exact length for each round. The same concept is applied to the secondary task rounds with baseRoundTime_s and maxRoundExtraTime_s.

With estimatedMaxReactionTime, the timespan the participant has to react to an event is defined. If it takes the participant longer than this variable to react, the event will be counted as “not correctly reacted to”, both for primary and for secondary task. In primary task evaluation, there is the additional concept of maxErrorsBeforeSearchCancels_p. If this was not built in, the participant could just press all buttons to react correctly to an event. With this concept, only the first x (int value of maxErrorsBeforeSearchCancels_p) button presses recognized after the event will be regarded for evaluation. For the secondary task evaluation, only the last turning action’s result will be regarded.

EstimatedMaxReactionTime_s also influences the generation of secondary events. As long as the participant still has time to react to a recent event, no new event requiring a reaction will be generated.

```
#TUNING VARIABLES (cont.)
survivalPercentageInRound=[100,96,92,90,85,80,75,70,60,50,20,10,5,2,1,0]
creationPercentageWhenObjects=[60,40,30,10,0]
changePercentageInRound=[0,0,3,5,8,10,12,15,20,25,30,35,40,45,50,100,...]
```

In the last section of the tuning variables, probabilities are defined. SurvivalPercentageInRound indicates the chance that a primary task object will stay alive when its lifetime has surpassed the amount of rounds used as index. CreationPercentageWhenObjects has an influence on the calculation determining if a new primary task object will appear. It declares the chance that a new primary task object is created when amount of alive objects as index exist. ChangePercentageInRound is for the secondary task, it configures the probability of a light changing when it has not changed for x rounds, with x as index.

```
#ALGORITHM VARIABLES
isTestRunning=False
secChangeBlockingRoundAmount = int(est.MaxReact.Time_s / baseR.Time_s) + 1
endTimeOfLastRound_p=0
endTimeOfLastRound_s=0
```



```

timeNow=0
thisRoundLifeTime_p=baseRoundTime_p+(maxRoundExtraTime_p/2)
thisRoundLifeTime_s=baseRoundTime_s+(maxRoundExtraTime_s/2)
roundNumber_p=0
roundNumber_s=0
changes = True
isThereSecTask = True
performanceMeasure = 0

```

Algorithm variables are the variables used for implementing the algorithm. They should not be altered, but we still want to explain how they work. `IsTestRunning` is a boolean variable that signifies if the study is running at the moment. `SecChangeBlockingRoundAmount` calculates the amount of rounds after a secondary task event requiring a reaction, where no new secondary task event requiring a reaction may be generated. `EndTimeOfLastRound` saves the exact time passed since the start of the study, up to the moment when the last round ended, both for primary and secondary task. `TimeNow` holds the exact time passed since the start of the study. `ThisRoundLifeTime` contains the exact round length for the current round, both for primary and secondary task. In `roundNumber`, the amount of rounds completed is counted for both tasks.

In each pass of the main while loop, `changes` is initially set to `False`. If there is a change detected, e.g. a new item appears or an old one disappears, `changes` is set to `True`, so the display part of the main loop only has to be gone through when needed. `IsThereSecTask` is used for performance improvement; the existence of a secondary task is only checked once and written into this variable for further usage. `PerformanceMeasure` has been implemented to check if the computer system is running the program fast enough. Every time the while loop runs through without having a “round”, this counter increases. A low value, e.g. beneath 200 for a 3-minute test, would suggest that the computer system is running too slow.

There is an array called `symbolSurfaces` that holds all primary task object symbols, loaded in from png files. It is filled depending on the chosen primary task. The most important objects for the graphic part are `screen`, `mainsurface` and `background_surf`, amended with `background_surf_beforestart` to have a different background before the study starts.

```

screen = pygame.display.set_mode(Rect(0,0,screenwidth,screenheight).size)
pygame.display.set_caption("Peripheral Lab Study")
mainsurface = pygame.display.get_surface()
background_surf = pygame.image.load("res/bg.png")
background_surf_beforestart = pygame.image.load("res/bg_bs.png")

```

The pygame sound infrastructure and the task managers (with their code located in `PriTask.py` and `SecTask.py`) are initialized depending on the chosen task modes.

```

if priTaskType == PriTask.VISUAL_AND_AUDIO:
    pygame.mixer.init()

primaryTaskManager = PriTask.PrimaryTaskManager(priTaskType, foldername,
debug, screenwidth, screenheight, symbolsize, creationPercentageWhenObjects,
survivalPercentageInRound)

if isThereSecTask:
    secTaskManager = SecTask.SecondaryTaskManager(secTaskType, portName,
changePercentageInRound, debug_lights, lightness, debug_secInput,
secChangeBlockingRoundAmount)

```

In the beginning, there is a while loop waiting for an ENTER keypress, so the study does not start before the key is pressed. This provides the participant with the decision when to exactly start the study. Once the key is pressed, the display is being cleared and the main study loop, “while isTestRunning:” comes into action. When timeNow surpasses maxTestTime, isTestRunning is set to False and the program thus exits the main study loop. The main study loop’s basic structure is as follows, with shortened code in caps:

```
while isTestRunning:
    timeNow = time.clock() - startClock
    changes = False
    isItARound = False

    if timeNow > maxTestTime:
        isTestRunning = False

    for event in pygame.event.get():
        # REACT TO KEYPRESS (for primary task)

        # SECONDARY TASK
        if isThereSecTask:
            secondaryUserInput = secTaskManager.checkForUserInput()
            # REACT TO SECONDARY TASK USER INPUT
            # CHECK IF SECONDARY ROUND IS OVER
            # INITIALIZE NEWROUND MECHANISM IF NEEDED
            # NEWROUND MECHANISM: COMPUTE NEW ROUND TIME
            # NEWROUND MECHANISM: COMPUTE NEW EVENTS
            # CHANGE LIGHT COLORS IF NEEDED

        # PRIMARY TASK
        # CHECK IF PRIMARY ROUND IS OVER
        # INITIALIZE NEWROUND MECHANISM IF NEEDED
        # NEWROUND MECHANISM: COMPUTE NEW ROUND TIME
        # NEWROUND MECHANISM: COMPUTE NEW EVENTS
        # UPDATE CHANGES VARIABLE IF NEEDED

    # SCREEN UPDATE (IF CHANGES IS TRUE)
```

After the main study loop, the text files with event and reaction information are finalized before evaluation commences. In the first step, events generated by the code are read into a list called botEventList and events generated by user input are read into a list called userEventList. These entries comprise time of the event, type of the event and two boolean values for use in the evaluation process later on. Next, the algorithm goes through the lists to sort out user events that happened before the first symbol appeared, by calling Util.sortOutEarlyEvents(). In the main step, the algorithm searches through all program events to find corresponding user input events, by calling Util.searchCorrespondingUserEvent(). The results of this function are saved in result sets to comfortably map them to variables in the next step. Average reaction time and error rate are then calculated with these variables. Finally, the values are printed to the console and written into an individual csv file as well as into “all_results.csv”.

PriTask.py

The basic structure of PriTask.py can be abstracted as follows:

```
#GLOBAL VARIABLES
#CLASS DEFINITION
#VARIABLES
#CONSTRUCTOR
#STATE READER FUNCTIONS
#PRIMARY TASK EVENT MANAGEMENT FUNCTIONS
```

The global variables are `JUST_VISUAL` and `VISUAL_AND_AUDIO`, helping to identify the two possible primary task modes. In the constructor, a lot of variables are being received from `Main.py`, like screen size or symbol size, and written into local variables. The temporary text files, used later for evaluation by `Main.py`, are managed in the `PriTask` object. Most functions inside `PriTask.py` are designed for retrieving information about the state of visual symbols, like `getNumberOfAliveObjects()`, `findOutIfObjectIsAlive(objectNr)` or `getObjectType(objectNr)`. Still, there are also two functions altering the state of visual symbols: `checkNewItemCreation(timeNow)`, which checks probabilities for creating a new visual object and also creates the new object if needed, and `checkOldItemDeletion()`, that does the same thing for deleting visual objects. We will first have a look at `checkNewItemCreation(timeNow)`, with shortened code in caps:

```
def checkNewItemCreation(self,timeNow):
    changes = False

    #maybe create new object
    if self.aliveObjects < 5:
        megaDiceRoll = random.randint(0, 100)

        #if diceroll is higher than perc, no new item
        if self.creationP_W_0_[self.aliveObjects] >= megaDiceRoll:
            #SEARCH FOR A SLOT WITHOUT ALIVE OBJECT AND CREATE IF NEEDED
            #GENERATE RANDOM COORDINATES UNTIL NEW COORDS WITHOUT COLLISION
            #PLAY SOUNDS IF NEEDED
            #WRITE NEW VALUES INTO VARIABLES
    return changes
```

The function returns a boolean as a result that signifies if anything has been changed. This boolean is later combined with the boolean result of `checkOldItemDeletion()`:

```
def checkOldItemDeletion(self):
    changes = False
    #check all alive objects if they survive
    for i in range(0,5):
        if self.objectAlive[i]:
            #object turns 1 round older
            tempObjectLifetime = self.objectLifetimes[i]
            tempObjectLifetime = tempObjectLifetime + 1
            self.objectLifetimes[i] = tempObjectLifetime

            #if so, object dies
```

```

        gigaDiceRoll = random.randint(0, 100)
        if self.surv.Perc.InRound[tempObj.Lifet.] <= gigaDiceRoll:
            self.objectAlive[i] = False
            self.aliveObjects = self.aliveObjects -1
            changes = True
    return changes

```

SecTask.py

The basic structure of SecTask.py can be abstracted as follows:

```

#GLOBAL VARIABLES
#CLASS DEFINITION
    #OBJECT VARIABLES
    #CONSTRUCTOR
    #FUNCTIONS FOR COMMUNICATION WITH SERIALMANAGER
    #SECONDARY TASK EVENT MANAGEMENT

```

The global variables in SecTask.py are for identification of tasks (NO_TASK, TASK_1, TASK_2), of colors (RED, GREEN, BLUE) and of the starting color and the relevant light for task 2. The SecTask objects also stores variables retrieved from Main.py, like lightness or changePercentageInRound, and object of the type SerialManager. For event management, checkChangeCreation(timeNow) and checkForUserInput() are the most important functions. CheckForUserInput() is being called from Main.py at the beginning of every round and reads data from the SerialManager object.

```

def checkForUserInput(self):
    if self.lastLineAmount < self.serialManager.getLineAmount():
        self.lastLineAmount = self.serialManager.getLineAmount()
        data = self.serialManager.getLastLine()
        if data == 1: return self.changeMainLight(True)
        elif data == -1: return self.changeMainLight(False)
    return 100

```

To find out if any new data is available, there is a counter for lines received that is being checked against the SerialManager object's counter. The function delivers a detected turning of the main light knob to the changeMainLight function, with a boolean signifying the direction of the turning. ChangeMainLight returns a value when the light state has been successfully changed, which in turn checkForUserInput returns to Main.py. The result 100 informs Main.py that no change has been detected. CheckChangeCreation(timeNow) works simultaneously to checkNewItemCreation(timeNow) for the primary task.

```

def checkChangeCreation(self,timeNow):
    self.roundAm_SinceLastR_Change = self.roundAm_SinceLastR_Change + 1
    changes = False
    for i in range(1,5):
        #light turns 1 round older
        tempLightLifetime = self.lightLifetimes[i]
        tempLightLifetime = tempLightLifetime + 1
        self.lightLifetimes[i] = tempLightLifetime

        if changes == False:

```

```

gigaDiceRoll = random.randint(0, 100)
#if so, light changes
if self.changeP_InRound[tempLightLifetime] >= gigaDiceRoll:

    #find out if it is relevant for blocking
    #or blocking is already over
    if ((self.roundAm_SinceL_R_Change > self.secC_B_RoundAm_)
        or (not self.isThisABlockedLight(i))):
        newColour = self.changeColleagueLight(i)
        changes = True
        result = (100*i) + newColour
        return result

return 999

```

CheckChangeCreation also returns an integer value to Main.py, signifying if there was a change, and which change took place.

SerialManager.py

The basic structure of SerialManager.py can be abstracted as follows:

```

#CLASS DEFINITION
#VARIABLES
#CONSTRUCTOR
#WRITE TO COMPORT FUNCTION
#THREAD FUNCTION LISTENING FOR INPUT
#READ FUNCTIONS FOR SECTASK.PY

```

The thread function run() features a while-loop that reads input coming from the arduino device and saves this input in a variable. Another variable is counted up with each new input, so the SecTask object can find out if there is new input. To ensure correct communication with the arduino device, the write function has to reformat output data. The function accepts an integer, places it in an array with a single item, and transforms the integer array to a byte array.

```

def write(self, i):
    if self.usable:
        output = bytearray([i])
        self.ser.write(output)

```

Util.py

In Util.py, several utility functions are implemented and the task modes (combinations of a primary task with a secondary task) are being held as global variables. It has four sections:

```

#GLOBAL TASK MODE VARIABLES
#EVALUATION HELP FUNCTIONS
#SECTASK COLOR HELP FUNCTION
#FOLDERNAME FUNCTION

```

4.1.3 Hypotheses

After designing and implementing the tasks, hypotheses were put forward:

- H1: The audio and visual primary task reaction time will be shorter than the reaction time in just visual primary task.
- H2: Just visual primary task error rate will be lower than audio and visual primary task error rate.
- H3: Secondary task 1 error rate and reaction time will be inferior to secondary task 2's.

Since this is not a study testing the actual use of a peripheral device but a study testing the study, hypotheses suitable for testing the study have been chosen. The idea behind H1 is that supplementary auditive cues make it easier to notice something happened, especially when the participant was busy with secondary task. For H2, if participant misses to get the right connection of audio and symbol, she can't look it up, whereas in just visual mode all information is visible on the screen. The reason for H3 is that in secondary task 2, only a single information has to be checked, while in secondary task 1, the dominant color has to be determined, which might be harder.

4.1.4 Study Procedure

The study was carried out in a medium-sized conference room, with the participant sitting at a large desk and the experimenter sitting at the left hand side of the participant. The participants were working with a 15,4" laptop, an additional USB keyboard and the prototype of the Ambient Presence Indicator. The USB keyboard was on the desk between participant and laptop, with the Ambient Presence Indicator positioned on the right hand side of the USB keyboard. The mapping explanation mini-sheet was slotted in the USB keyboard above the numpad. The room lighting and Ambient Presence Indicator brightness were adjusted manually to adapt to changing weather conditions. For a picture of the exact alignment of devices, see *figure 4.2*. Training time was 2 minutes for each primary task, main test time was 5 minutes for each task combination. During the test, the experimenter manually took notes of focus shift frequency, hand movement and other usable observations.

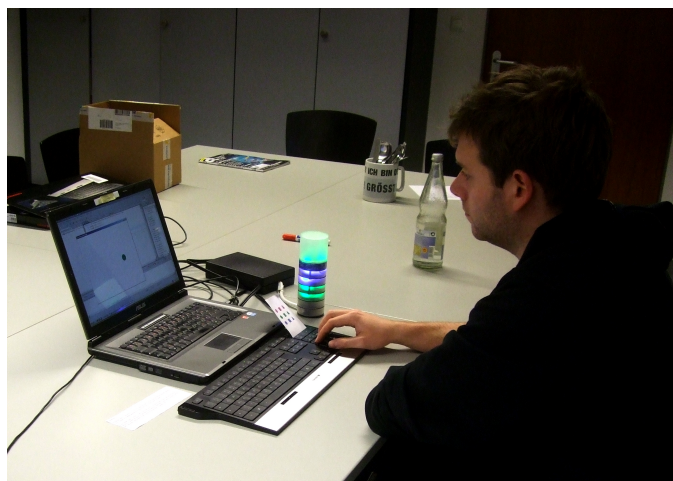


Figure 4.2: Picture of a participant during the first pilot study

4.1.5 Measured Values and Participant Feedback

After analyzing result data, only a scarce outcome was achieved. The only striking data is that the participants had a shorter reaction time and a lower error rate in primary task 1, but only the first of these observations is statistically significant ($p < 0.05$). Comparing the secondary tasks, average reaction times and error rates are similar to each other. For a table of result values, see *Appendix, 6.7*.

When examining user feedback from the questionnaire, more interesting facts could be gained. All participants estimated the primary task performance degradation caused by multitasking with the peripheral device acceptable. Still, most of them found the peripheral noticeability insufficient. A usability problem was found during the course of the experiment: sometimes the peripheral device does not react to the turning of its main knob.

One peculiar fact arising when comparing result data and user feedback is that the primary task with audio was ranked as the easier one by the participants, but came out as the harder one in the result data. Apart from insufficient functioning of peripherality, user feedback was generally positive about the usability of the device.

4.1.6 Examination of Hypotheses

- H1: The audio and visual primary task reaction time will be shorter than the reaction time in just visual primary task. This hypothesis was tested with a paired samples t-test.

Result: refuted

Mean PT1 reaction time (just visual, task modes A and C): 1.67 seconds.

Mean PT2 reaction time (audio and visual, task modes B and D): 2.05 seconds.

$p = 0.00$

For detailed values, see *Appendix, 6.8*. The basic assumption of the hypothesis complies with participant feedback. Yet, the values show a different result. Even when keeping in mind that the number of participants in this pilot study is extremely low, the high statistical significance in the refutation of the hypothesis is surprising. The reason for this fact is subject to speculations. A possible reason is that participants waited for the sound to end before they started reacting, with sound durations being just under the half second mark.

- H2: Just visual primary task error rate will be lower than audio and visual primary task error rate. This hypothesis was tested with a paired samples t-test.

Result: not confirmed

Mean PT1 error rate (just visual, task modes A and C): 6%.

Mean PT2 error rate (audio and visual, task modes B and D): 9%.

$p = 0.25$

For detailed values, see *Appendix, 6.8*. A possible reason for the lacking of statistical significance could be the generally low error rates in both primary tasks.

- H3: Secondary task 1 error rate and reaction time will be inferior to secondary task 2's. This hypothesis can be split up into 8 sub-hypotheses, each tested with a paired samples t-test:

- H3a: Secondary task 1 error rate will be inferior to secondary task 2's

Result: not confirmed

Mean ST1 error rate (task modes A and D): 28%.

Mean ST2 error rate (task modes B and C): 28%.

$p = 0.99$

- H3b: Secondary task 1 reaction time will be inferior to secondary task 2's

Result: not confirmed

Mean ST1 reaction time (task modes A and D): 3.91 seconds.

Mean ST2 reaction time (task modes B and C): 3.40 seconds.

$p = 0.31$

For detailed values, see *Appendix, 6.8*. The result of both sub-hypotheses being not statistically significant shows that the secondary tasks are at the same level of difficulty. This complies with participant feedback from the post-study questionnaires.

4.1.7 Conclusions

In a feedback interview with the study participants, who are research assistants experienced with designing and carrying out usability studies, several weaknesses of the first pilot study were discussed. At the beginning of the discussion, details of the first pilot study were reviewed. The participants reported that they found the first pilot study's combination of primary tasks monotonous and tiring, especially since there was no motivating feedback.

One of the main suggestions in the discussion was designing a continuous rather than event-based primary task. The research assistants argued that a study about peripheral interaction would produce more valuable results if the participants can distribute their attention themselves, which is the case in a continuous task that can be manually disrupted and continued in any moment.

The second very helpful suggestion was to further formalize task criteria and define them more accurately. The ideas collected in the course of the discussion heavily influenced the task criteria in *section 3.3*. The final topic was the design of the new continuous task for the second pilot study. One of the many suggestions for a redesign was particularly helpful for the new task, which will be described in the next section.

4.2 Pilot Study 2: Event-based or Continuous

The second pilot study was carried out in February 2012 with eight Media Informatics students. It featured two primary tasks, one event-based, and one continuous, and two secondary tasks comprising the peripheral device "Ambient Presence Indicator".

4.2.1 Study Design Process

The main goal of the second pilot study was to get rid of some weaknesses of the first one and also bring in some variation. To achieve that, a new primary task was to be designed, accompanied by some changes in the procedure of the study. The new primary task, PT3, should differ from the old ones in terms of interruptibility, continuity, ambidexterity and reward.

Correspondingly, the nature of the task had to be changed. Instead of just pressing a button to react to an event, the participant has to press a button and click with a mouse to achieve a sub-goal in PT3. To make the task continuous instead of event-based, the screen is filled with 30 random items at the beginning of a round. The participant has the assignment of removing all items of a certain color, by clicking them while pressing the key

for the item. When the assignment is achieved, a new round begins with a new random arrangement of items. The color relevant for the current assignment and the number of items removed is displayed at the right edge of the window, as well as a text display saying “Good Job!” when a sub-goal is achieved. While Fitts’ Law [12] has no effect in the old primary tasks, it has a small effect in the new one. The study participant will have to move the mouse from one target object to the next one. But the effect remains restricted as the participant is expected to navigate through the tasks with movement routes as short as possible.

For a screenshot of the new primary task, see *figure 4.3*. The number of items removed, the number of rounds completed, and the number of incorrect input actions is measured automatically. Since there are no events to react to, the concept for reaction time in PT3 is measuring the timespan between two correct user actions. The new primary task ranks in the task dimensions of *section 3.3* as follows:

High interruptibility, flexible mental load, high continuity of task, low continuity of user input, task focus on accuracy and speed, inclusion of random factors, low dependence on previous knowledge and abilities, no penalty, inclusion of reward in scoring system and two additional rewarding displays, high ambidexterity.

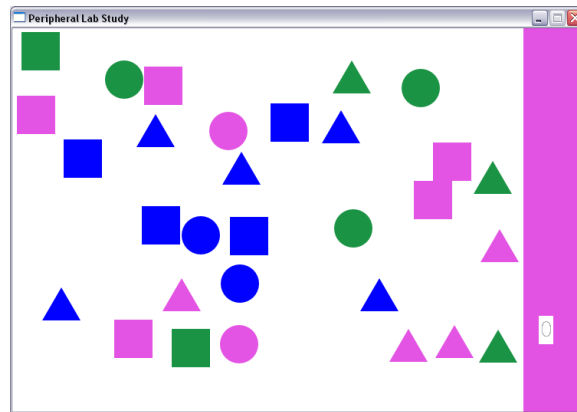


Figure 4.3: Screenshot of the new primary task in the second pilot study

The observation of differences between just visual output and visual output combined with audio output were not continued in this pilot study. The questionnaires were only altered in details and can be found at *Appendix, 6.10*.

4.2.2 Software Implementation

Apart from the new primary task, some other changes have also been applied to study procedure and software code. Between training and study main part, baseline tests were to be carried out with both primary tasks. For all values, standard deviations were to be computed and saved in the results.

Changes to Main.py

To implement the new primary task, even Main.py had to undergo some changes. In the start variables, there are additional task modes available: Util.BASELINE_MODE_1 and Util.BASELINE_MODE_3. In the tuning variables, screenwidthbonus has been added, indicating the width of extra space on the right edge of the window. This extra space is needed for signaling the active color of the new primary task, showing the hit counter and the new label saying “Good Job!”. Hitboxbonus increases the area of the hitbox, so even when the user clicks 3 pixels off the object, it will still be counted as a hit. MaxObjectsPt3

defines how many objects will be randomly distributed on the screen at the beginning of a round. `Pt3goodJobTime` assesses the timespan the “Good Job!” is being shown after a hit.

```
#TUNING VARIABLES
screenwidthbonus=80
hitboxbonus=3
maxObjectsPt3 = 30
pt3goodJobTime = 0.5
```

In the algorithm variables, colors are being defined. Additionally, the last pressed key is saved in `pt3lastKey`. The dominant color is being held in `pt3dominantColour`. In the new primary task, hits and errors are already counted during the course of the study, in the variables `pt3errorCount` and `pt3hitCount`. Accordingly, reaction times as the times between two hits are also collected live, in `pt3reactionTimesList`, with the help of the time of the last saved hit (`pt3lastHit`). The state of the mouse button is saved in `buttonPressed`. The state of the “Good Job!” sign is held in `pt3goodJobIsThere`. `NumLockStateIsIncorrect` is used for notifying user when the num lock button is pressed to prevent the system from not noticing button presses. `Pt3objectCreationThreshold` is the number of remaining objects that will trigger a new distribution of objects.

```
greycolor_rgba = pygame.Color(180,180,180,255)
whitecolor_rgba = pygame.Color(255,255,255,1)
greycolor = pygame.Color(180,180,180)
violetcolor = pygame.Color(227,83,227)
bluecolor = pygame.Color(0,0,255)
greencolor = pygame.Color(26,146,67)
pt3lastKey = 0
pt3dominantColour = 999
pt3errorCount = 0
pt3hitCount = 0
buttonPressed = False
pt3lastHit = 0.0
pt3reactionTimesList = []
pt3goodJobIsThere = False
numLockStateIsIncorrect = False
pt3objectCreationThreshold = 0
```

The for loop handling key presses had to be adapted to the new need of recognizing buttons held pressed. The part of the main loop, where primary task rounds are checked, is now in the else-part of this if-statement:

```
if priTaskType == PriTask.JUST_VISUAL_CONT:

    if timeNow > pt3goodJobTime + pt3lastHit and pt3goodJobIsThere:
        pt3goodJobIsThere = False
        changes = True

    if primaryTaskManager.aliveObjects < pt3objectCreationThreshold:
        roundNumber_p = roundNumber_p +1
        # create objects:
        # set new dominant color, adapt creation threshold to new situation
```

```

pt3creationResult = primaryTaskManager.fillWithObjects(timeNow)
pt3dominantColour = pt3creationResult[0]
pt3objectCr_Threshold = maxObjectsPt3 +1 - pt3creationResult[1]

if pt3dominantColour == PriTask.PT3BLUE:
    bgcolor = bluecolor
elif pt3dominantColour == PriTask.PT3GREEN:
    bgcolor = greencolor
elif pt3dominantColour == PriTask.PT3VIOLET:
    bgcolor = violetcolor
changes = True

```

The render section had to be complemented with some lines for rendering the number of hits and the “Good Job!” sign. In the evaluation section, two new file writers have been included to save reaction times into separate text files. Apart from that, standard deviation has been included into the results, while the “total” part of the results has been excluded.

Changes to PriTask.py and Util.py

PriTask.py has gotten three new global variables: integer codes for the colors in the new primary task. MaxObjectsAmount and hitBoxBonus are taken over from Main.py. There are five new functions: checkSingleDimensionCollision checks if two rectangles collide in either x or y dimension. CheckAreaCollision uses checkSingleDimensionCollision to investigate if two rectangles collide. FillWithObjects removes all old objects and randomly distributes new objects on the screen. MouseButtonClick returns True as result if a symbol has been hit with a mouseclick and calls killObject to remove the object if it has the right color.

There are two new functions in Util.py to assist Main.py at the evaluation, computeAverage and computeStandardDeviation.

4.2.3 Hypotheses

Just like in the first pilot study, this is not a study testing the actual use of a peripheral device but a study testing the study. Thus, following hypotheses suitable for testing the study have been chosen:

- H1: Primary task reaction time and error rate with multitasking will be inferior to primary task reaction time and error rate without multitasking.
- H2: Secondary task reaction time and error rate in combination with PT 1 (event) will be inferior to secondary task reaction time and error rate in combination with PT 3 (continuous).
- H3: Secondary task 1 error rate and reaction time will be inferior to secondary task 2's.

The reason for H2 is that in PT3 (continuous), the participant can choose when to stop and turn to the secondary task. As for H3, in secondary task 2 only a single information has to be checked, while in secondary task 1, the dominant color has to be determined, resulting in higher need for attention.

4.2.4 Study Procedure

The study was carried out in a medium-sized office room, with the participant sitting at a desk and the experimenter sitting at the right hand side of the participant. The participants were working with a 15,4" laptop, an additional USB keyboard and mouse and the prototype of the Ambient Presence Indicator. The USB keyboard was on the desk between participant and laptop, with the mouse positioned on the right hand side of the USB keyboard and the Ambient Presence Indicator positioned between the two. The mapping explanation mini-sheet was slotted in the USB keyboard above the numpad. The room lighting and Ambient Presence Indicator brightness were adjusted manually to adapt to changing weather conditions. For a picture of the exact alignment of devices, see *figure 4.4*. Training time was 1 minute for each task combination, baseline tests were 5 minutes for each primary task, main test time was 5 minutes for each task combination. During the test, the experimenter manually took notes of focus shift frequency, hand movement and other usable observations.



Figure 4.4: Picture of a participant during the second pilot study

4.2.5 Measured Values and Participant Feedback

During the study, a usability problem was found: sometimes, the periphery device reacted hyper-sensitive to a participant turning its main light knob.

When analyzing results data, a striking fact is that primary task performance degradation rates vary to high extent. In general, error rates are lower in PT1 (event-based). To find out if training times were sufficient, the individual user performance were investigated in chronologic order; a continuous, discernible increase in performance was to be found at two participants' reaction times and at four participants error rates. This fact commends for longer training timeslots. The primary task 1 error rate in multitasking condition was below 5% in five participants' performances, which would require adapted values to make PT1 harder the next time it is being used. For a table of result values, see *Appendix, 6.11*.

When it comes to user feedback, participants report that they judge the degradation of their primary task performance as acceptable, but the peripheral noticeability as insufficient. Grouping participants by their self-assessment regarding multitasking and satisfaction with device, no performance differences between the groups could be found.

4.2.6 Examination of Hypotheses

- H1: Primary task reaction time and error rate with multitasking will be inferior to primary task reaction time and error rate without multitasking. This hypothesis can be split up into 8 sub-hypotheses, each tested with a paired samples t-test:
 - H1a: Primary task 1 reaction time with multitasking (task mode A) will be inferior to primary task 1 reaction time without multitasking.
Result: confirmed.
 Mean PT1 reaction time (multitasking): 1.59 seconds.
 Mean PT1 reaction time (baseline): 1.40 seconds.
 $p = 0.02$
 - H1b: Primary task 1 reaction time with multitasking (task mode C) will be inferior to primary task 1 reaction time without multitasking.
Result: confirmed.
 Mean PT1 reaction time (multitasking): 1.56 seconds.
 Mean PT1 reaction time (baseline): 1.40 seconds.
 $p = 0.00$
 - H1c: Primary task 3 reaction time with multitasking (task mode B) will be inferior to primary task 3 reaction time without multitasking.
Result: not confirmed.
 Mean PT3 reaction time (multitasking): .85 seconds.
 Mean PT3 reaction time (baseline): .78 seconds.
 $p = 0.05$
 - H1d: Primary task 3 reaction time with multitasking (task mode D) will be inferior to primary task 3 reaction time without multitasking.
Result: confirmed.
 Mean PT3 reaction time (multitasking): .85 seconds.
 Mean PT3 reaction time (baseline): .78 seconds.
 $p = 0.00$
 - H1e: Primary task 1 error rate with multitasking (task mode A) will be inferior to primary task 1 error rate without multitasking.
Result: not confirmed.
 Mean PT1 error rate (multitasking): 7%.
 Mean PT1 error rate (baseline): 2%.
 $p = 0.05$
 - H1f: Primary task 1 error rate with multitasking (task mode C) will be inferior to primary task 1 error rate without multitasking.
Result: not confirmed.
 Mean PT1 error rate (multitasking): 5%.
 Mean PT1 error rate (baseline): 2%.
 $p = 0.06$
 - H1g: Primary task 3 error rate with multitasking (task mode B) will be inferior to primary task 3 error rate without multitasking.
Result: not confirmed.
 Mean PT3 error rate (multitasking): 11%.
 Mean PT3 error rate (baseline): 9%.
 $p = 0.10$

- H1h: Primary task 3 error rate with multitasking (task mode D) will be inferior to primary task 3 error rate without multitasking.

Result: not confirmed.

Mean PT3 error rate (multitasking): 10%.

Mean PT3 error rate (baseline): 9%.

$p = 0.50$

For detailed values, see *Appendix, 6.12*. With three of four sub-hypotheses confirmed, hypothesis 1 can be regarded as mostly confirmed for reaction times, yet not confirmable for error rates.

- H2: Secondary task reaction time and error rate in combination with PT 1 (event) will be inferior to secondary task reaction time and error rate in combination with PT 3 (continuous). This hypothesis can be split up into 2 sub-hypotheses, each tested with a paired samples t-test:

- H2a: Secondary task reaction time in combination with PT 1 (event) will be inferior to secondary task reaction time in combination with PT 3 (continuous).

Result: not confirmed.

Mean ST reaction time (in combination with PT1, task modes A and C): 3.38 seconds.

Mean ST reaction time (in combination with PT3, task modes B and D): 3.76 seconds.

$p = 0.15$

- H2b: Secondary task error rate in combination with PT 1 (event) will be inferior to secondary task error rate in combination with PT 3 (continuous).

Result: not confirmed.

Mean ST error rate (in combination with PT1, task modes A and C): 29%.

Mean ST error rate (in combination with PT3, task modes B and D): 27%.

$p = 0.70$

For detailed values, see *Appendix, 6.12*. While mean error rates are similar to each other, mean reaction times differ more, but show a tendency suggesting the opposite of the sub-hypotheses being the case. Still, none of the sub-hypotheses have reached a result of statistic significance, so the hypothesis can be regarded as not confirmed.

- H3: Secondary task 1 error rate and reaction time will be inferior to secondary task 2's. This hypothesis can be split up into 2 sub-hypotheses, each tested with a paired samples t-test:

- H3a: Secondary task 1 reaction time will be inferior to secondary task 2's.

Result: not confirmed.

Mean ST1 reaction time (task modes A and D): 3.54 seconds.

Mean ST2 reaction time (task modes B and C): 3.61 seconds.

$p = 0.80$

- H3a: Secondary task 1 error rate will be inferior to secondary task 2's.

Result: not confirmed.

Mean ST1 error rate (task modes A and D): 28%.

Mean ST2 error rate (task modes B and C): 27%.

$p = 0.87$

For detailed values, see *Appendix, 6.12*. For this hypothesis, both reaction time and error rate means are very similar. Accordingly, none of the sub-hypotheses have reached a result of statistic significance, so the hypothesis can be regarded as not confirmed.

4.2.7 Conclusions

In retrospect, the study could have been slightly improved with two changes - the adaptation of code values to make primary task 1 harder and an extension of training timeslots. Still, the pilot study had two of eight participants using periphery view to keep track of the peripheral device, which is surprising for the scarce acclimatization time of a lab study. The observation of participants also revealed that most of them used the end of a PT3 round - when all objective items were removed and new ones distributed - as a break to shift their attention focus to the peripheral device, even with there being no delay to keep them from carrying on with their focal activity. It shows that although PT3 was designed to be a continuous task and although it is possible to operate it as a continuous task, participants tend to interpret it in a way that gives them convenient breaks for focus shifts.

5 Summary and Outlook

In the final chapter, we want to present a roundup of the information gathered in various stages of this research work: The fundamentals from related work, initial thoughts gathered before and the lessons learned from the pilot studies. We will also denote where research in our institute is heading in the near future.

5.1 Summary

In this research paper, we have compiled related work from peripheral interaction and neighboring research fields, to define peripheral interaction and to find and visualize borders between peripheral interaction, tangible user interfaces, notification systems and ambient information systems.

Study methods of lab studies in peripheral interaction and neighboring research fields were investigated in 17 cases, comparing study setup, used study patterns, used measurement parameters. This was followed by a discussion of six study methodologies from neighboring research fields: “DECIDE” for user interface studies by Preece et al.[34], an adapted version of “DECIDE” for ambient displays by Shen[37], a “toolkit for rapid prototyping of peripheral awareness displays” by Matthews and Mankoff[25], an adaptation of the Nielsen and Molich[29] usability heuristics to ambient displays by Mankoff et al.[23], the IRC model by McCrickard et al.[28] and the peripheral displays evaluation framework by Shami et al.[35].

We have collected means of measurement applicable in a lab study, dividing them into explicit measurement and implicit measurement methods, with explicit measurement methods further divided into quiz, user impression feedback and technical user feedback. We have defined task types in multiple ways. The first one is assignment type with the options being “react to event”, “find mistake” and “carry out pre-defined activity”. The second way is used input and output channels, with auditive, visual and haptic input and output channels. Finally, we posited ten task dimensions: Interruptibility, mental load, continuity of task, continuity of user input, task focus, inclusion of random factors, dependence on previous knowledge and abilities, penalty, reward and ambidexterity.

The key part of this work is the suggestion of an abstract procedure for designing, preparing and carrying out a study dealing with peripheral interaction concepts. We have gathered questions to ask while designing a study and questions to ask the participants before and after the study. Two pilot studies were carried out according to our suggested methodology and the results of and conclusions from these studies presented.

In the pilot studies, acclimatization to working with the periphery could only rarely be achieved by the participants. Still, we found two usability problems: a state of lacking reaction and a state of hypersensitivity of the peripheral device. Thus, we have achieved or goal of supporting the early design process of peripheral devices with the revelation of weaknesses. Apart from that, the degradation of primary task performance could be approximated, which is also helpful for finding out which ones of the early concepts are too distracting.

5.2 Lessons for PILS

Peripheral interaction lab studies are meant to assist the design of a peripheral device in an early stage of the design process. They provide early suggestions and hints about which concepts might work, yet they can not replace long-term studies, that will take place in a later stage of the development cycle.

The most basic fact about a peripheral interaction lab study is that it has to be a study based on multitasking. The peripheral device to be tested can be only be built in

into a secondary task, with a different, primary task being in the focus of the participants' attention. There is a slight antagonism of the peripheral activity being the more important one for the outcome of the study.

In general, attention takes a major role in every possible peripheral interaction lab study. The study could be observing the distribution of attention or the characteristics of the participants' focus shifts, or it could be analyzing if the peripheral device is granted the right amount of attention by the participant. Too much attention for the peripheral task would result in an unacceptable performance and usage comfort degradation, while too little attention would render the usage of the peripheral device ineffective.

An important decision in designing a peripheral interaction lab study is which perception channels to use in the primary task. We found that reaction times suffer when using multiple channels, yet this result could also be caused by other factors.

Altogether, a peripheral interaction lab study should be designed to meet the needs of what is meant to be the focus of the study. The focus of the study should influence the structure, the chosen tasks and the questionnaire. But it should also have an impact on details, like: "Which parameters will be measured?" or "Should the primary task offer break points for participants to switch their focus?". The selection of a primary task should be carried out carefully, and it should be tested before the study to make sure its values are correctly adjusted. The allocation of study time to individual parts of the study is important, to grant enough training and test time without making the test too long, which would cause a lack of motivation on the participants' side.

The setting of the study should be similar to an actual use case as much as possible. Any chances to make the lab study feel more like a real use case and to make the peripheral task feel more like a peripheral task should be taken. Feedback about the participants' assessment of attention and other impressions should be obtained, still especially the participants' assessment of attention should not be regarded as thoroughly accurate.

A major problem of peripheral interaction lab studies that cannot be solved is peripheral perception. To make the participants' perception move to the periphery, a long time of acclimatization is needed, which cannot be provided in a lab study. Some participants may achieve a status with peripheral perception in the course of the study, but they will rather be an exception than the usual case.

5.3 Outlook

The next step of developing the study methodology further is comparing a lab study's results with those of a long-term study testing the same peripheral device to find out how much the results differ. In such a study, the study methodology will also be used with a "real" secondary task instead of made-up pseudo tasks, which will be another chance to improve the methodology.

In the future, our methodology should be applied to create more studies in an easier way. As for the code, the implementation of new primary tasks compatible with the software base could lead to it becoming a reusable starting point for a variety of new studies.

6 Appendix

6.1 List of Analyzed Lab Studies

Arroyo et al.	2002	Interruptions	[2]
Bartram et al.	2001	Moving Icons	[5]
Chewar et al.	2002	Secondary Task Displays	[8]
Czerwinski et al.	2000	IM	[9]
Hausen et al.	2011	Ambient Appointment Projection	[13]
Hsieh et al.	2003	Mail Orb	[18]
Leung et al.	2007	Haptically Augmented Touchscreen GUI	[19]
Maglio et al.	2000	Ticker	[21]
Mamykina et al.	2001	Time Aura	[22]
McCrickard et al.	2001	Animation	[27]
McCrickard et al.	2003	Scope IRC	[28]
Olivera et al.	2011	Twitter Mood PolyTag	[30]
Oulasvirta et al.	2011	Pointing / Text	[31]
Pacey et al.	2001	Auditory Cues	[32]
Shami et al.	2005	Weather Watcher	[35]
Shen	2007	MoneyColor	[36]
Weissgerber et al.	2004	Vispad	[39]

6.2 List of Possible Primary Tasks (By Channels)

Channels	Input / Output (into / from computer)		Possible Assignments	Measurable parameters
visual	v	v	Scenario 1a: Elements appear randomly on the screen, some of them move - when a new / certain element appears / changes status, look at it / its symbol	reaction time, error rate
auditive	-	v	Scenario 1a: Elements appear randomly on the screen, some of them move - in which portion of the screen did symbol XY appear - how many XY appeared - which color had XY - in which order appeared symbols XY, XZ, etc - in which direction moved symbol XY Scenario 4a: replay of a video without sound - in which portion of the screen did symbol XY appear - how many XY appeared - which color had XY - in which order appeared symbols XY, XZ, etc - in which direction moved symbol XY	Quiz
	a	a	Scenario 2a: audio replay of a conversation with background sounds - voice command XY, when audio symbol XY occurs	reaction time, error rate
haptic	-	a	Scenario 2a: audio replay of a conversation with background sounds - StereoSwap: how often did conversational partners switch sides - questions about contents of conversation, order of topics - how often appeared audio symbol XY - in which direction moved audio symbol XY	Quiz
	h	h	Scenario 3: replay of a haptic feedback sequence - when haptic symbol XY, press button XY - when haptic symbol XY, make gesture XY	reaction time, error rate
visual + auditive	-	h	Scenario 3: replay of a haptic feedback sequence - how often did haptic symbol XY appear	Quiz
	a	v	Scenario 1a: Elements appear randomly on the screen, some of them move - voice command, when symbol XY appears - voice command, when certain amount of symbols of type XY visible - voice command XY, when symbol XY changes status / appears / changes direction	reaction time, error rate

v	a	<p>Scenario 2a: audio replay of a conversation with background sounds</p> <ul style="list-style-type: none"> - look at visual symbol XY, when audio symbol XY occurs - look at visual symbol XY, when person XY talks - look at visual symbol XY, when XY is the topic - look at visual symbol XY, when conversational partners switch sides 	<p>reaction time, error rate</p>
a	a+v	<p>Scenario 2b: Video replay of a conversation with background items and sounds</p> <ul style="list-style-type: none"> - voice command, when (visual or audio) symbol XY appears - voice command, when certain amount of symbols of type XY visible - voice command XY, when symbol XY changes status / appears / changes direction - voice command, when conflict between image and sound - voice command XY, when event XY occurs 	<p>reaction time, error rate</p>
v	a+v	<p>Scenario 2c: Video replay of a conversation with background items and sounds; static visual symbols at screen borders or edges</p> <ul style="list-style-type: none"> - look at visual symbol XY, when (visual or audio) symbol XY appears - look at visual symbol XY, when Person XY talks - look at visual symbol XY, when XY is the topic - look at visual symbol XY, when conversational partners switch sides - when new / certain Element appears / changes status, look at its symbol 	<p>reaction time, error rate</p>
-	a+v	<p>Scenario 2b: Videoreplay of a conversation with background items and sounds</p> <ul style="list-style-type: none"> - how often did audio symbol XY - how often did conversational partners switch sides - how often conflict between image and sound - how many XY were visible - which color had XY - in which order appeared symbols XY, XZ, etc - questions about contents of conversation, order of topics - how often appeared audio symbol XY 	<p>Quiz</p>

haptic + au- ditive	a	h	Scenario 3: replay of a haptic feedback sequence - when haptic symbol XY, then voice command XY	reaction time, error rate
	h	a	Scenario 2a: audio replay of a conversation with back- ground sounds - when conversational partners switch sides, press but- ton - when conversational partners switch sides, make ges- ture - when audio symbol XY occurs, press button XY - when audio symbol XY occurs, make gesture XY - when Person XY talks, press button XY - when Person XY talks, make gesture XY - when XY is the topic, press button XY - when XY is the topic, make gesture XY Scenario 6a: audio replay of several sentences - type sentences (without visual feedback)	reaction time, error rate
	a	a+h	Scenario 2d: audio replay of a conversation with back- ground sounds, combined with replay of a random hap- tic feedback sequence - voice command, when audio symbol XY and haptic symbol XZ appear at the same time Scenario 2e: audio replay of a conversation with back- ground sounds and haptic feedback, whereas haptic feedback correlates with volume - voice command, when conflict between audio and hap- tic feedback	reaction time, error rate
	h	a+h	Scenario 2d: audio replay of a conversation with back- ground sounds, combined with replay of a random hap- tic feedback sequence - when audio symbol XY and haptic symbol XZ appear at the same time, press button XW - when audio symbol XY and haptic symbol XZ appear at the same time, make gesture XW Scenario 2e: audio replay of a conversation with back- ground sounds and haptic feedback, whereas haptic feedback correlates with volume - when conflict between audio and haptic feedback, press button - when conflict between audio and haptic feedback, make gesture	reaction time, error rate

	-	a+h	<p>Scenario 2d: audio replay of a conversation with background sounds, combined with replay of a random haptic feedback sequence</p> <ul style="list-style-type: none"> - how often did audio symbol XY and haptic symbol XZ appear at the same time - what was the topic when haptic symbol XY appeared <p>Scenario 2e: audio replay of a conversation with background sounds and haptic feedback, whereas haptic feedback correlates with volume</p> <ul style="list-style-type: none"> - how often conflict between audio and haptic feedback 	Quiz
visual + haptic	h	v	<p>Scenario 1a: Elements appear randomly only the screen, some of them move</p> <ul style="list-style-type: none"> - when symbol XY appears, press button XZ - when n symbols XY appear, press button XZ n times - when new symbol XY appears, press button XZ and subsequently arrow key for its direction - when symbol XY appears, make gesture XZ - when symbol of type XY appears, make gesture XY <p>Scenario 6c: Displaying several sentences on the screen</p> <ul style="list-style-type: none"> - type sentences without visual feedback 	reaction time, error rate
	v	h	<p>Scenario 7: Displaying several sentences on the screen</p> <ul style="list-style-type: none"> - correct mistakes <p>Scenario 3: replay of a haptic feedback sequence</p> <ul style="list-style-type: none"> - when haptic symbol XY, then look at symbol XZ 	Completion time, error rate reaction time, error rate
	h	h+v	<p>Scenario 4b: replay of a video without sound, but with corresponding haptic feedback</p> <ul style="list-style-type: none"> - when conflict between video and haptic feedback, press button XY - when conflict between video and haptic feedback, make gesture XY 	reaction time, error rate
	v	h+v	<p>Scenario 4b: replay of a video without sound, but with corresponding haptic feedback</p> <ul style="list-style-type: none"> - when conflict between video and haptic feedback, look at conflict symbol 	reaction time, error rate
	-	h+v	<p>Scenario 4b: replay of a video without sound, but with corresponding haptic feedback</p> <ul style="list-style-type: none"> - how often conflict between video and haptic feedback <p>Scenario 5: repla of a video without sound, with many different scenes and randomly appearing haptic symbols</p> <ul style="list-style-type: none"> - describe what happened in the video when haptic symbol XY appeared <p>Scenario 1b: Elements appear randomly on the screen, some of them move, combined with replay of a constant haptic feedback sequence with sudden peaks</p> <ul style="list-style-type: none"> - which visual symbols appeared when haptic peaks appeared 	Quiz

auditive + visual + haptic	h	v+a	Scenario 2b: Videoreplay of a conversation with background items and sounds press button or make gesture, - when (visual or audio-) symbol XY appears - when person XY talks - when XY is the topic - when conflict between video and stereo sound - when background item is activated	reaction time, error rate
	a	v+h	Scenario 4b: replay of a video without sound, but with corresponding haptic feedback - voice command when conflict between video and haptic feedback Scenario 1b: Elements appear randomly on the screen, some of them move, combined with replay of a constant haptic feedback sequence with sudden peaks - voice command XY, when visual symbol XY appears during a haptic feedback peak	reaction time, error rate
	h	v+a+h	Scenario 2f: Videoreplay of a conversation with background items and sounds and corresponding haptic feedback - when haptic feedback appears with every symbol XY, press button XY - when haptic feedback appears with every symbol XY, make gesture XY Scenario 6b: audio replay of several sentences also shown on the screen - type sentences without visual feedback	reaction time, error rate
	a	v+a+h	Scenario 2f: Videoreplay of a conversation with background items and sounds and rarely appearing, corresponding haptic feedback - when haptic feedback appears with every symbol XY, voice command XY	reaction time, error rate
	v	a+h	Scenario 2g: audio replay of a conversation with background sounds and haptic feedback, whereas haptic feedback correlates with volume, combined with visual symbols - when conflict between stereo sound and haptic feedback, look at conflict symbol Scenario 2h: audio replay of a conversation with background sounds, combined with replay of a random haptic feedback sequence, combined with visual symbols - when haptic feedback during audio symbol XY, look at visual symbol XY	reaction time, error rate

v	v+a+h	<p>Scenario 2i: Video replay of a conversation with background items and sounds and corresponding haptic feedback, combined with visual symbols at screen borders or edges</p> <p>- when haptic feedback appears with every symbol XY, look at visual symbol XY</p>	<p>reaction time, error rate</p>
-	v+a+h	<p>Scenario 2f: Video replay of a conversation with background items and sounds and rarely appearing, corresponding haptic feedback</p> <p>- in which situations did haptic feedback correlate with which background items</p> <p>Scenario 2j: Video replay of a conversation with background items and sounds and random haptic feedback symbols</p> <p>- describe what happened and what was said in the video when haptic symbol XY appeared</p>	<p>Quiz</p>

6.3 List of Primary Task Scenarios

Scenario 1a: Elements appear randomly on the screen, some of them move
Scenario 1b: Elements appear randomly on the screen, some of them move, combined with replay of a constant haptic feedback sequence with sudden peaks
Scenario 2a: audio replay of a conversation with background sounds
Scenario 2b: video replay of a conversation with background items and sounds
Scenario 2c: video replay of a conversation with background items and sounds; static visual symbols at screen borders or edges
Scenario 2d: audio replay of a conversation with background sounds, combined with replay of a random haptic feedback sequence
Scenario 2e: audio replay of a conversation with background sounds and haptic feedback, whereas haptic feedback correlates with volume
Scenario 2f: video replay of a conversation with background items and sounds and corresponding haptic feedback
Scenario 2g: audio replay of a conversation with background sounds and haptic feedback, whereas haptic feedback correlates with volume, combined with visual symbols
Scenario 2h: audio replay of a conversation with background sounds, combined with replay of a random haptic feedback sequence, combined with visual symbols
Scenario 2i: video replay of a conversation with background items and sounds and corresponding haptic feedback, combined with visual symbols at screen borders or edges
Scenario 2j: video replay of a conversation with background items and sounds and random haptic feedback symbols
Scenario 3: replay of a haptic feedback sequence
Scenario 4a: replay of a video without sound
Scenario 4b: replay of a video without sound, but with corresponding haptic feedback
Scenario 5: replay of a video without sound, with many different scenes and randomly appearing haptic symbols
Scenario 6a: audio replay of several sentences
Scenario 6b: audio replay of several sentences also shown on the screen
Scenario 6c: Displaying several sentences on the screen
Scenario 7: Displaying several sentences on the screen

6.4 PILS Methodology Overview

Initial Preparations

- Goals: What is the main focus of the study?
- Hypotheses: e.g. Device fulfils usability standards XY. Users accept device. Device degrades user performance by less than x. Device improves multitasking performance by at least x.
- User observation: notes or video capture or both?

Study Setup

1. Pre-Questionnaire: past experiences, satisfaction with past interaction type, importance of task type and the information provided
2. Baseline test primary task
3. Test primary with secondary task
4. Post-Questionnaire
 - does user feel device is distracting?
 - Has user changed opinions since pre-questionnaire?
 - Did user feel like spending too much or too little awareness on device?
 - Did user feel uncomfortable while multitasking?
 - does the device do what user wants (Preece et al. 2002[34])
 - does user like it (Preece et al 2002)
 - did user have problems using it (Preece et al. 2002[34])
 - does user want to use it again (Preece et al. 2002[34])
 - did user find it hard to learn how to use device
 - rate against heuristics (Mankoff et al 2003[23])
 - sufficient information design (conveying “just enough” information?)
 - consistent and intuitive mapping (state of device easy to understand?)
 - match between system and real world (symbols / words comprehensible?)
 - visibility of state (transitions from state to state noticeable?)
 - aesthetic and pleasing design
 - useful and relevant information
 - visibility of system status (appropriate feedback within reasonable time?)
 - user control and freedom (enough “emergency exits”?)
 - easy transition to more in-depth information
 - peripherality of display (is device easily monitorable?)
 - flexibility and efficiency of use (accelerators for experienced users?)

6.5 Study 1 - Explanation Sheet

Peripheral Interaction Lab Study

Code uses Python 2.7.2, Pygame 1.9.2 (for 2.7), pySerial 2.6. (for 2.7, install with `easy_install / PyPi`). Ambient Presence Indicator Arduino code written by Clara Lülling and Simone Rodestock

Car sound taken from
<http://www.freesound.org/people/han1/sounds/19025/>
 (Creative Commons)

Telephone sound taken from
<http://www.freesound.org/people/transitking/sounds/15826/>
 (Creative Commons)

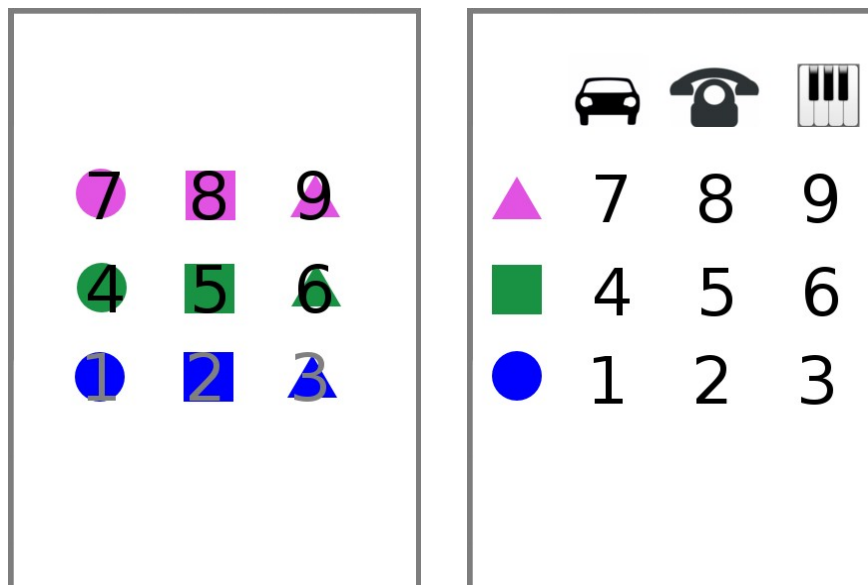
Piano sound made with FL Studio 9 and FL Keys plug-in

Telephone Symbol taken from
<http://www.easyvectors.com/browse/other/6f931f29ddb8ecd21a51a489b4c35786-phone-icon-clip-art>
 (free for commercial use)

Car symbol taken from
<http://www.easyvectors.com/browse/other/car-icon-for-use-with-signs-or-buttons-clip-art>
 (free for commercial use)

Piano symbol taken from
<http://cheeseenthusiast.deviantart.com/art/Piano-icon-126310417>
 (Creative Commons)

Print and cut out button maps for the test. They should fit above the standard keyboard numblock.



Test Setups

<p>Test setup A</p> <p>In primary task 1, called JUST_VISUAL, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen and the left table on page 1.</p> <p>In secondary task 1, called TASK_1, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the majority of the four colleague lights.</p>
<p>Test setup B</p> <p>In primary task 2, called VISUAL_AND_AUDIO, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen, the sound played on its appearing and the right table on page 1.</p> <p>In secondary task 2, called TASK_2, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the bottom colleague light.</p>
<p>Test setup C</p> <p>In primary task 1, called JUST_VISUAL, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen and the left table on page 1.</p> <p>In secondary task 2, called TASK_2, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the bottom colleague light.</p>
<p>Test setup D</p> <p>In primary task 2, called VISUAL_AND_AUDIO, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen, the sound played on its appearing and the right table on page 1.</p> <p>In secondary task 1, called TASK_1, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the majority of the four colleague lights.</p>

Study participants test order

P1: Training Test setup A – P1 S1 Test setup B – P2 S2 Test setup C – P1 S2 Test setup D – P2 S1	P2: Training Test setup B – P2 S2 Test setup A – P1 S1 Test setup D – P2 S1 Test setup C – P1 S2
P3: Training Test setup C – P1 S2 Test setup D – P2 S1 Test setup A – P1 S1 Test setup B – P2 S2	P4: Training Test setup D – P2 S1 Test setup C – P1 S2 Test setup B – P2 S2 Test setup A – P1 S1

Main focus of the study:

- Peripheral device usability
- Peripheral device usability study performance test

Study hypotheses:

H1: Primary task reaction time and error rate without multitasking will be inferior to primary task reaction time with multitasking.

H2: The audio and visual primary task reaction time will be shorter than the reaction time in just visual primary task. (supplementary auditive cues make it easier to notice something happened, especially when user was busy with secondary task)

H3: Just visual primary task error rate will be lower than audio and visual primary task error rate. (if user misses to get the right connection of audio and symbol, she can't look it up, whereas in just visual mode all information is visible on the screen)

H4: Secondary task 1 error rate and reaction time will be inferior to secondary task 2's. (in ST 2, only a single information has to be checked, while in ST 1, the dominant colour has to be determined)

H5: Audio and visual primary task's error rate and reaction time will, in multitasking condition, be lower than just visual primary task's error rate and reaction time. (mental load is distributed between senses)

6.6 Study 1 - Questionnaires

Explanatory notes for study participants

In this study we want to evaluate the use of a so called „peripheral“ device. The study features multiple rounds, where two types of tasks, primary and secondary, will have to be executed at the same time. The primary task will be in the focus of your attention and the secondary in the periphery.

Training (10 min.) and Study (20 min.)

We will start off with a training round, where you will learn how to execute the primary and secondary tasks. For primary tasks, you will have to press a certain button every time a certain symbol appears on the screen, in one of the tasks combined with a sound to be heard. Which button you will have to press for which symbol will be visible on a small card applied to the keyboard. The symbols will disappear after a random time span. Try to press the right button as soon as possible. Even if you press a wrong button, you can still get it right by pressing the correct button afterwards.

The secondary task uses the peripheral device. At certain occasions, you will have to interact with the peripheral device to execute the secondary task. The peripheral device features a main light (on the top), which is turnable, and four so called “colleague lights”. For the secondary task, every once in a while you will have to turn the main light until it shines in a certain colour. Details for all tasks will be explained before each of the 4 test rounds and visible on another small card applied to the keyboard. The need for an interaction with the peripheral device will only arise sporadically.

Once we are through the training of both of the possible primary tasks, the actual study begins, featuring the same tasks. Both tasks should be considered almost equally important, with the primary task having a slight priority.

Pre-Questionnaire

Which types of computer input devices do you use?

- Keyboard and mouse
- Touch screen
- Graphics tablet
- Game pad / joystick / racing wheel
- Other: _____

How often do you execute multiple tasks on your computer at the same time?

- Most of the time
- Often
- Every once in a while
- Seldom
- Never

When you multitask, do you categorize your tasks in “important” and “less important”?

- Yes
- Sometimes
- No

Main questionnaire

Age: _____ Gender: _____ Occupation: _____

In retrospect, which task do you think you prioritized when multitasking?

- Primary task (symbols, sounds, keyboard input)
- Secondary task (peripheral device)
- None

In retrospect, do you think your performance in the primary task has suffered when multitasking, compared to when not multitasking?

- Yes. I could not concentrate enough on the primary task
- Yes. But I still have the feeling I performed well enough on the primary task
- I'm not sure
- No

In retrospect, what do you think about the amount of attention you paid to the peripheral device?

- I looked over to the peripheral device way too often.
- I looked over to the peripheral device too seldom.
- I checked the peripheral device in an adequate frequency
- I perceived changes of the peripheral device easily from the corner of my eye.
- I tried to perceive changes of the peripheral device from the corner of my eye, but it didn't work the way I wanted it to

Do you think you could have executed the secondary task better if it had been a display on the screen and instead of turning the device, you would have had to press buttons?

- Yes
- I'm not sure
- No

Please rate the peripheral device:	yes				no
Did you feel comfortable while multitasking with the device?					
Was the state of the device easy enough to understand?					
Are transitions from one state to another state noticeable?					
Is the design aesthetic and pleasing?					
Does the device give appropriate feedback within reasonable time?					
Is the device easily monitorable?					
Did the peripheral device work the way you wanted it to?					
Would you use the device with your office or home computer?					
Did you find it hard to learn how to use the peripheral device?					
Was the primary task with sounds was easier than the one without?					
Was the secondary task with the majority of lights harder than the one with the bottom light?					

Did you have problems using the peripheral device?

6.7 Study 1 - Result Tables**By Users**

user name	total avg re-action time	total error rate	PT: avg re-action time	PT: error rate	ST: avg re-action time	ST: error rate
User 1	2,439528321	0,1686653971	2,2102411313	0,1500609068	4,3017660622	0,28125
User 2	1,9353631082	0,0782636718	1,7633154953	0,0641956505	3,4581832361	0,1875
User 3	1,8741122271	0,0816357198	1,7257358207	0,0465963489	3,3248681431	0,3392857143
User 4	1,9007757118	0,0757635668	1,7305080118	0,0407469685	3,5351091431	0,3095238095

By Task Setup

task setup	total avg re-action time	total error rate	PT: avg re-action time	PT: error rate	ST: avg re-action time	ST: error rate
Mode A	1,9453742718	0,1024975707	1,6901342822	0,0551219737	4,3030030656	0,3819444444
Mode B	2,1293987268	0,1104192938	2,0102143329	0,0851489519	3,1965122146	0,308531746
Mode C	1,8449060479	0,0946827626	1,6434892847	0,0747960563	3,6010037365	0,2514880952
Mode D	2,2301003215	0,0967287284	2,0859625594	0,0865328929	3,5194075678	0,1755952381

Task Setups:

- MODE A = Primary task: 1 - JUST_VISUAL; Secondary task: 1 - majority
- MODE B = Primary task: 2 - VISUAL_AND_AUDIO; Secondary task: 2 - bottom
- MODE C = Primary task: 1 - JUST_VISUAL; Secondary task: 2 - bottom
- MODE D = Primary task: 2 - VISUAL_AND_AUDIO; Secondary task: 1 - majority

6.8 Study 1 - Statistical Results

1.1 T-TEST. Paired Sample Statistics Hypothesis 1

Pair 0: Comparing reaction times of primary task 1 with reaction times of primary task 2

```
#####
#           #Mean|N|Std. Deviation|SE. Mean#
#####
#Pair 0 PT_RT_AC#1.67|8|           .20|      .07#
#       PT_RT_BD#2.05|8|           .31|      .11#
#####
```

1.2 T-TEST. Paired Samples Correlations

```
#####
# |           #N|Correlation|Sig.#
#####
#Pair 0|PT_RT_AC & PT_RT_BD#8|           .69| .06#
#####
```

5.3(1:2) T-TEST. Paired Samples Test

```
#####
#           #           Paired Differences           #
#           #-----+-----+-----+-----#
#           # |           |           |           | 95% #
#           # |           |           |           | +----#
#           #Mean|Std. Deviation|Std. Error Mean|Lower#
#####
#Pair 0PT_RT_AC - PT_RT_BD#-.38|           .22|           .08| -.57#
#####
```

1.3(2) T-TEST. Paired Samples Test

```
#####
# | | |           #
#----+ | |           #
# | | |           #
#----+ | |           #
#Upper| t |df|Sig. (2-tailed)#
#####
# -.20|-4.87| 7|           .00#
#####
```

2.1 T-TEST. Paired Sample Statistics
Hypothesis 2

Pair 0: Comparing error rates of primary task 1 with error rates of primary task 2

```
#####
#                               #Mean|N|Std. Deviation|SE. Mean#
#####
#Pair 0 PT_ER_AC# .06|8|          .03|          .01#
#          PT_ER_BD# .09|8|          .07|          .02#
#####
```

6.2 T-TEST. Paired Samples Correlations

```
#####
# |                               #N|Correlation|Sig.#
#####
#Pair 0|PT_ER_AC & PT_ER_BD#8|          .82| .01#
#####
```

2.3(1:2) T-TEST. Paired Samples Test

```
#####
#                               # Paired Differences                               #
#                               #-----+-----+-----+-----#
#                               # |                               | 95% #
#                               # |                               |-----#
#                               #Mean|Std. Deviation|Std. Error Mean|Lower#
#####
#Pair 0|PT_ER_AC - PT_ER_BD#-.02|          .05|          .02| -.06#
#####
```

2.3(2) T-TEST. Paired Samples Test

```
#####
# | | |                               #
#-----+ | |                               #
# | | |                               #
#-----+ | |                               #
#Upper| t |df|Sig. (2-tailed)#
#####
# .02|-1.25| 7|          .25#
#####
```

3.1 T-TEST. Paired Sample Statistics
Hypothesis 3

Pair 0: Comparing reaction times of secondary task 1 with reaction times of secondary task 2

Pair 1: Comparing error rates of secondary task 1 with error rates of secondary task 2

```
#####
#                               #
#           #Mean|N|Std. Deviation|SE. Mean#
#                               #
#Pair 0 ST_RT_AD#3.91|8|           .77|      .27#
#           ST_RT_BC#3.40|8|           .82|      .29#
#Pair 1 ST_ER_AD# .28|8|           .16|      .06#
#           ST_ER_BC# .28|8|           .11|      .04#
#####
```

3.2 T-TEST. Paired Samples Correlations

```
#####
#                               #
#           |           #N|Correlation|Sig.#
#                               #
#Pair 0|ST_RT_AD & ST_RT_BC#8|           -.38| .35#
#Pair 1|ST_ER_AD & ST_ER_BC#8|           .08| .85#
#####
```

3.3(1:2) T-TEST. Paired Samples Test

```
#####
#                               #
#           #           Paired Differences           #
#           #-----+-----+-----+-----+-----#
#           #           |           |           | 95% #
#           #           |           |           |-----#
#           #           #Mean|Std. Deviation|Std. Error Mean|Lower#
#                               #
#Pair 0ST_RT_AD - ST_RT_BC# .51|           1.32|           .47| -.59#
#Pair 1ST_ER_AD - ST_ER_BC# .00|           .19|           .07| -.16#
#####
```

3.3(2:2) T-TEST. Paired Samples Test

```
#####
#           |           |           |           #
#-----+ | | |           #
#           |           |           |           #
#-----+ | | |           #
#Upper| t |df|Sig. (2-tailed)#
#####
# 1.62|1.10| 7|           .31#
# .15|-.02| 7|           .99#
#####
```

6.9 Study 2 - Explanation Sheet

Peripheral Interaction Lab Study

Code uses Python 2.7.2, Pygame 1.9.2 (for 2.7), pySerial 2.6. (for 2.7, install with easy_install / PyPi), statlib 1.1.0. Ambient Presence Indicator Arduino code written by Clara Lüling and Simone Rodestock

Explanatory Notes

Peripheral Lab Study is a program to evaluate the multitasking use of a peripheral device in the lab. There are two types of tasks, primary and secondary. The primary task is obligatory and used to distract the user from the peripheral device. The secondary task is optional and uses the peripheral device. Random events are being generated by the program, some of which the user has to react to with a key press (for primary tasks) or an interaction with the peripheral device (for secondary tasks).

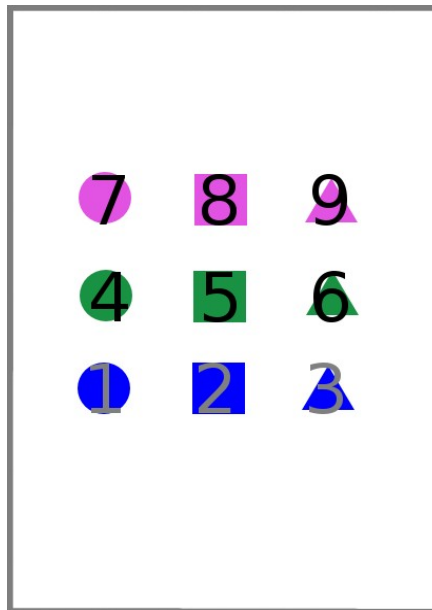
In primary task 1, called JUST_VISUAL, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen and the table¹ on page 1.

In primary task 2, called JUST_VISUAL_CONT, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen and the table on page 1, and click the object with the mouse. Only objects in the colour of the right edge of the window are clickable.

In secondary task 1, called TASK_1, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the majority of the four colleague lights.

In secondary task 2, called TASK_2, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of one special colleague lights, which is number 4 by default

1



Print and cut out button map for the test. It should fit above the standard keyboard numblock.

Test setup A

In primary task 1, called JUST_VISUAL, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen and the table on page 1.

In secondary task 1, called TASK_1, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the majority of the four colleague lights.

Test setup B

In primary task 2, called JUST_VISUAL_CONT, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen and the table on page 1, and click the object with the mouse. Only objects in the colour of the right edge of the window are clickable.

In secondary task 2, called TASK_2, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the bottom colleague light.

Test setup C

In primary task 1, called JUST_VISUAL, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen and the table on page 1.

In secondary task 2, called TASK_2, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the bottom colleague light.

Test setup D

In primary task 2, called JUST_VISUAL_CONT, the user has to press a key on the keypad according to a circle, square or triangle appearing on the screen and the table on page 1, and click the object with the mouse. Only objects in the colour of the right edge of the window are clickable.

In secondary task 1, called TASK_1, the user has to turn an element of the Ambient Presence Indicator, a prototype piece of hardware using arduino featuring 5 lights and a turnable button, until the main light shines in the colour of the majority of the four colleague lights.

Main focus of the study:

- Peripheral device usability
- Peripheral device usability study performance test

Study participants test order

Briefing (3min) + 4 x Training (4x1 + 3 = 6min) + 2x Baseline (2x5 + 1 = 11min) + 4x Test (4x5 + 3 = 23 min) + Questionnaire (5min) = 3 + 6 + 11 + 23 + 5 = 48

P1: Training A B C D Baseline 1, Baseline 3 Test setup A – P1 S1 (5 min) Test setup B – P2 S2 (5 min) Test setup C – P1 S2 (5 min) Test setup D – P2 S1 (5 min)	P2: Training B A D C Baseline 1, Baseline 3 Test setup B – P2 S2 Test setup A – P1 S1 Test setup D – P2 S1 Test setup C – P1 S2
P3: Training C D A B Baseline 1, Baseline 3 Test setup C – P1 S2 Test setup D – P2 S1 Test setup A – P1 S1 Test setup B – P2 S2	P4: Training D C B A Baseline 1, Baseline 3 Test setup D – P2 S1 Test setup C – P1 S2 Test setup B – P2 S2 Test setup A – P1 S1
P5: Training A B C D Baseline 3, Baseline 1 Test setup A – P1 S1 (5 min) Test setup B – P2 S2 (5 min) Test setup C – P1 S2 (5 min) Test setup D – P2 S1 (5 min)	P6: Training B A D C Baseline 3, Baseline 1 Test setup B – P2 S2 Test setup A – P1 S1 Test setup D – P2 S1 Test setup C – P1 S2
P7: Training C D A B Baseline 3, Baseline 1 Test setup C – P1 S2 Test setup D – P2 S1 Test setup A – P1 S1 Test setup B – P2 S2	P8: Training D C B A Baseline 3, Baseline 1 Test setup D – P2 S1 Test setup C – P1 S2 Test setup B – P2 S2 Test setup A – P1 S1

Study hypotheses:

H1: Primary task reaction time and error rate with multitasking will be inferior to primary task reaction time without multitasking.

H2: Secondary task reaction time and error rate with PT 1 (event) will be inferior to secondary task reaction time and error rate with PT 2 (continuous). (user can choose when to stop and turn to ST)

H3: Secondary task 1 error rate and reaction time will be inferior to secondary task 2's. (in ST 2, only a single information has to be checked, while in ST 1, the dominant colour has to be determined)

6.10 Study 2 - Questionnaires

Explanatory notes for study participants

In this study we want to evaluate the use of a so called „peripheral“ device. The study features multiple rounds, where two types of tasks, primary and secondary, will have to be executed at the same time. The primary task will be in the focus of your attention and the secondary in the periphery.

Training (10 min.), Baseline (12 min.) and Study (25 min.)

We will start off with a training round, where you will learn how to execute the primary and secondary tasks. In one of the primary tasks, you will have to press a certain button very time a certain symbol appears on the screen (with symbols disappearing after a random time span), in the other you will have to remove symbols by pressing a button and clicking with the mouse. Only the objects in the colour of the right edge of the screen will be clickable. Which button you will have to press for which symbol will be visible on a small card applied to the keyboard. Try to press the right button as soon as possible. Even if you press a wrong button, you can still get it right by pressing the correct button afterwards.

The secondary task uses the peripheral device. At certain occasions, you will have to interact with the peripheral device to execute the secondary task. Before using, please arrange the peripheral device in a position that seems suitable for you. The peripheral device features a main light (on the top), which is turnable, and four so called “colleague lights”. For the secondary task, every once in a while you will have to turn the main light until it shines in a certain colour. Details for all tasks will be explained before each of the 4 test rounds and visible on another small card applied to the keyboard. The need for an interaction with the peripheral device will only arise sporadically.

Once we are through the training of both of the possible primary tasks, the actual study begins, featuring the same tasks. Both tasks should be considered almost equally important, with the primary task having a slight priority.

Pre-Questionnaire

Which types of computer input devices do you use?

- Keyboard and mouse
- Touch screen
- Graphics tablet
- Game pad / joystick / racing wheel
- Other: _____

How often do you execute multiple tasks (e.g. browsing, working on a document, instant messaging) on your computer at the same time?

- Most of the time
- Often
- Every once in a while
- Seldom
- Never

When you multitask, do you categorize your tasks in “important” and “less important”?

- Yes
- Sometimes
- No

Main questionnaire:

Age: _____ Gender: _____ Occupation: _____

In retrospect, which task do you think you prioritized when multitasking?

- Primary task (symbols, keyboard input)
- Secondary task (peripheral device)
- None

In retrospect, do you think your performance in the primary task has suffered when multitasking, compared to when not multitasking?

- Yes. I could not concentrate enough on the primary task
- Yes. But I still have the feeling I performed well enough on the primary task
- I'm not sure
- No

In retrospect, what do you think about the amount of attention you paid to the peripheral device?

- I looked over to the peripheral device way too often.
- I looked over to the peripheral device too seldom.
- I checked the peripheral device in an adequate frequency
- I perceived changes of the peripheral device easily from the corner of my eye.
- I tried to perceive changes of the peripheral device from the corner of my eye, but it didn't work the way I wanted it to

Do you think you could have executed the secondary task better if it had been a display on the screen and instead of turning the device, you would have had to press buttons?

- Yes
- I'm not sure
- No

Please rate the peripheral device:	yes				no
Did you feel comfortable while multitasking with the device?					
Was the state of the device easy enough to understand?					
Are transitions from one state to another state noticeable?					
Is the design aesthetic and pleasing?					
Does the device give appropriate feedback within reasonable time?					
Is the device easily monitorable?					
Did the peripheral device work the way you wanted it to?					
Would you use the device with your office or home computer?					
Did you find it hard to learn how to use the peripheral device?					
Was the primary task with the randomly appearing symbols harder than the one with removing symbols?					
Was the secondary task with the majority of lights harder than the one with the bottom light?					

Did you have problems using the peripheral device?

6.11 Study 2 - Result Tables

Results By Task Setup

Name	PT: avg re- action time	PT: reaction time sdev	PT1: error rate in rela- tion to all P events	PT1: error rate in relation to all U events	ST: avg re- action time	ST: reac- tion time sdev	ST: error rate	
BL PT1 (PT1 ST1)	1,3967064604	0,3810802499	0,0180518987	0,0556364795	3,2888075556	1,2644682499	0,271577381	
(PT1 ST1)	1,5924752616	0,588061989	0,0659020085	0,09817015	3,4768539885	1,4308696911	0,311656746	
(PT1 ST2)	1,5552343687	0,506462293	0,0453046648	0,1005010924	3,4768539885	1,4308696911	0,311656746	
Name	PT: avg re- action time	PT: reaction time sdev	PT3: error rate in rela- tion to all U events	PT3: Hit Count	PT3: error count	ST: avg re- action time	ST: reac- tion time sdev	ST: error rate
BL PT3 (PT3 ST2)	0,7820568422	0,735009505	0,0897765566	387,25	39,125	3,7394326564	1,3198047425	0,2373511905
(PT3 ST2)	0,8493462201	0,8430538376	0,1127432324	361	46,625	3,7394326564	1,3198047425	0,2373511905
(PT3 ST1)	0,851756034	0,8745058971	0,0987929761	357,625	40,5	3,7860048497	1,3613984853	0,2980654762

6.12 Study 2 - Statistical Results

1.1 T-TEST. Paired Sample Statistics
Hypothesis 1: Reaction Times

Pair 0: Comparing primary task reaction times of task mode A to primary task reaction times of baseline test with primary task 1

Pair 1: Comparing primary task reaction times of task mode C to primary task reaction times of baseline test with primary task 1

Pair 2: Comparing primary task reaction times of task mode B to primary task reaction times of baseline test with primary task 3

Pair 3: Comparing primary task reaction times of task mode D to primary task reaction times of baseline test with primary task 3

```
#####
#           #Mean|N|Std. Deviation|SE. Mean#
#####
#Pair 0 Mode_A #1.59|8| .27| .10#
# Baseline_1#1.40|8| .21| .08#
#Pair 1 Mode_C #1.56|8| .17| .06#
# Baseline_1#1.40|8| .21| .08#
#Pair 2 Mode_B # .85|8| .15| .05#
# Baseline_3# .78|8| .09| .03#
#Pair 3 Mode_D # .85|8| .12| .04#
# Baseline_3# .78|8| .09| .03#
#####
```

1.2(2) T-TEST. Paired Samples Correlations

```
#####
# | #N|Correlation|Sig.#
#####
#Pair 0|Mode_A & Baseline_1#8| .75| .03#
#Pair 1|Mode_C & Baseline_1#8| .91| .00#
#Pair 2|Mode_B & Baseline_3#8| .90| .00#
#Pair 3|Mode_D & Baseline_3#8| .93| .00#
#####
```

1.3(1:2) T-TEST. Paired Samples Test

```
#####
# # Paired Differences#
# #-----#
# # | #
# # | #
# #Mean|Std. Deviation#
#####
#Pair 0Mode_A - Baseline_1# .20| .18#
#Pair 1Mode_C - Baseline_1# .16| .09#
#Pair 2Mode_B - Baseline_3# .07| .08#
#Pair 3Mode_D - Baseline_3# .07| .05#
#####
```

1.3(2:2) T-TEST. Paired Samples Test

```
#####
# | | | #
#-----+-----+ | | #
# | 95% | | | #
# +-----+-----+ | | #
#Std. Error Mean|Lower|Upper| t |df|Sig. (2-tailed)#
#####
# .06| .05| .35|3.09| 7| .02#
# .03| .08| .24|4.84| 7| .00#
# .03| .00| .14|2.35| 7| .05#
# .02| .02| .12|3.62| 7| .01#
#####
```

2.1 T-TEST. Paired Sample Statistics

Hypothesis 1: Error Rates

Pair 0: Comparing primary task error rates of task mode A to primary task error rates of baseline test with primary task 1

Pair 1: Comparing primary task error rates of task mode C to primary task error rates of baseline test with primary task 1

Pair 2: Comparing primary task error rates of task mode B to primary task error rates of baseline test with primary task 3

Pair 3: Comparing primary task error rates of task mode D to primary task error rates of baseline test with primary task 3

```
#####
#                               #Mean|N|Std. Deviation|SE. Mean#
#####
#Pair 0 Mode_A  # .07|8|          .07|     .02#
#   Baseline_1# .02|8|          .02|     .01#
#Pair 1 Mode_C  # .05|8|          .05|     .02#
#   Baseline_1# .02|8|          .02|     .01#
#Pair 2 Mode_B  # .11|8|          .07|     .02#
#   Baseline_3# .09|8|          .05|     .02#
#Pair 3 Mode_D  # .10|8|          .07|     .03#
#   Baseline_3# .09|8|          .05|     .02#
#####
```

2.2 T-TEST. Paired Samples Correlations

```
#####
#                               #N|Correlation|Sig.#
#####
#Pair 0|Mode_A & Baseline_1#8|          .75| .03#
#Pair 1|Mode_C & Baseline_1#8|          .81| .01#
#Pair 2|Mode_B & Baseline_3#8|          .87| .01#
#Pair 3|Mode_D & Baseline_3#8|          .91| .00#
#####
```

2.3(2) T-TEST. Paired Samples Test

```
#####
#                               #Paired Differences  | | |
#                               #-----+-----+-----+
#                               #       |       |       | 95% |       |
#                               #       |       |       |-----+-----+
#                               #Mean|Std. Deviation|Std. Error Mean|Lower|Upper|  t |df|Sig. (2-tailed)#
#####
#Pair 0|Mode_A - Baseline_1# .05|          .06|          .02| .00| .10|2.31| 7|          .05#
#Pair 1|Mode_C - Baseline_1# .03|          .03|          .01| .00| .06|2.23| 7|          .06#
#Pair 2|Mode_B - Baseline_3# .02|          .03|          .01| -.01| .05|1.89| 7|          .10#
#Pair 3|Mode_D - Baseline_3# .01|          .04|          .01| -.02| .04| .72| 7|          .50#
#####
```

3.1 T-TEST. Paired Sample Statistics
Hypothesis 2

Pair 0: Comparing reaction times of secondary tasks when combined with PT1 (event-based) to reaction times of secondary tasks when combined with PT3 (continuous)

Pair 1: Comparing error rates of secondary tasks when combined with PT1 (event-based) to error rates of secondary tasks when combined with PT3 (continuous)

60

```
#####
#                               #
#           #Mean| N|Std. Deviation|SE. Mean#
#                               #
#Pair 0 ST_RT_AC#3.38|16|          .89|    .22#
#           ST_RT_BD#3.76|16|          .70|    .17#
#Pair 1 ST_ER_AC# .29|16|          .15|    .04#
#           ST_ER_BD# .27|16|          .17|    .04#
#####
```

3.2 T-TEST. Paired Samples Correlations

```
#####
# |                               # N|Correlation|Sig.#
# |                               #-----#-----#
#Pair 0|ST_RT_AC & ST_RT_BD#16|          .22| .42#
#Pair 1|ST_ER_AC & ST_ER_BD#16|          -.06| .81#
#####
```

3.3(1:2) T-TEST. Paired Samples Test

```
#####
#                               # Paired Differences #
#                               #-----+-----+-----#
#                               # | | | 95% #
#                               # | | | +-----#
#                               #Mean|Std. Deviation|Std. Error Mean|Lower#
#                               #-----#-----#-----#
#Pair 0ST_RT_AC - ST_RT_BD#-.38|          1.01|          .25| -.92#
#Pair 1ST_ER_AC - ST_ER_BD# .02|          .24|          .06| -.10#
#####
```

3.3(2:2) T-TEST. Paired Samples Test

```
#####
# | | | | #
#-----+ | | | #
# | | | | #
#-----+ | | | #
#Upper| t |df|Sig. (2-tailed)#
#-----#-----#-----#
# .16|-1.51|15|          .15#
# .15| .40|15|          .70#
#####
```

4.1 T-TEST. Paired Sample Statistics
Hypothesis 3

Pair 0: Comparing reaction times of secondary tasks

Pair 1: Comparing error rates of secondary tasks

```
#=====#=====#=====#=====#=====#
#                #Mean| N|Std. Deviation|SE. Mean#
#=====#=====#=====#=====#=====#
#Pair 0 ST1_RT#3.54|16|                .78|   .20#
#        ST2_RT#3.61|16|                .86|   .22#
#Pair 1 ST1_ER# .28|16|                .15|   .04#
#        ST2_ER# .27|16|                .18|   .05#
#=====#=====#=====#=====#=====#
```

4.2 T-TEST. Paired Samples Correlations

```
#=====#=====#=====#=====#
#      |                # N|Correlation|Sig.#
#=====#=====#=====#=====#
#Pair 0|ST1_RT & ST2_RT#16|                .14| .59#
#Pair 1|ST1_ER & ST2_ER#16|                -.07| .80#
#=====#=====#=====#=====#
```

4.3(1:2) T-TEST. Paired Samples Test

```
#=====#=====#=====#=====#
#                #                Paired Differences                #
#                #-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----#
#                # |                |                |                |                |                |                |                |                |                |                |                |                |                #
#                # |                |                |                |                |                |                |                |                |                |                |                |                |                #
#                # |                |                |                |                |                |                |                |                |                |                |                |                |                #
#                #Mean|Std. Deviation|Std. Error Mean|Lower|Upper#
#=====#=====#=====#=====#
#Pair 0ST1_RT - ST2_RT#-.07|                1.08|                .27| -.64| .50#
#Pair 1ST1_ER - ST2_ER# .01|                .24|                .06| -.12| .14#
#=====#=====#=====#=====#
```

4.3(2:2) T-TEST. Paired Samples Test

```
#=====#=====#=====#
#      | |                #
#      | |                #
#      | |                #
#      | |                #
# t |df|Sig. (2-tailed)#
#=====#=====#=====#
#-.26|15|                .80#
# .17|15|                .87#
#=====#=====#=====#
```

Inhalt der beigelegten CD

- FOLDER: code1 (code2 has files and folders with identical names)
 - FOLDER: res
 - 15826 __transitking__telephoning.ogg
 - 19025 __han1__car-start-and-drive.ogg
 - bg.png
 - bg_bs.png
 - bg_ec.png
 - bg_ev.png
 - piano.ogg
 - symbol_bc.png
 - symbol_bs.png
 - symbol_bt.png
 - symbol_gc.png
 - symbol_gs.png
 - symbol_gt.png
 - symbol_vc.png
 - symbol_vs.png
 - symbol_vt.png
 - FOLDER: txt
 - all_results.csv
 - all_results.csv
 - Main.py
 - PriTask.py
 - SecTask.py
 - SerialManager.py
 - Util.py
- FOLDER: graph
 - handout-1-1.pdf
 - handout-1-5.pdf
 - handout-1-6.pdf
 - handout-1-7.pdf
 - handout-1-8.pdf
 - handout-2-1.pdf
 - handout-2-5.pdf
 - handout-2-6.pdf
 - handout-2-7.pdf
 - handout-2-8.pdf
 - research_fields.png
 - research_fields_ais.png

- research_fields_nots.png
- study1-part.jpg
- study1-task.png
- study1_hypothesen_1.pdf
- study1_hypothesen_2.pdf
- study1_hypothesen_3.pdf
- study2-part.jpg
- study2-task.png
- study2_hypothesen_1.pdf
- study2_hypothesen_2.pdf
- study2_hypothesen_3.pdf
- study2_hypothesen_4.pdf

- FOLDER: lit
 - ames-ambient evaluation.pdf
 - arroyo-interruptions.pdf
 - bakker-design.pdf
 - bakker-interactive.pdf
 - bartram-icons.pdf
 - Benbunan-Fich-Measuring.pdf
 - Benbunan-Fich-metrics.pdf
 - chewar-secondary.pdf
 - czerwinski-interruption.pdf
 - edge-peripheral.pdf
 - edge-tangible peripheral.pdf
 - fitts-information.pdf
 - hausen-entangling ambient.pdf
 - hausen-extending interaction.pdf
 - hausen-Klassifikation.png
 - hausen-sectasks.pdf
 - hausen-statube.pdf
 - hazlewood-exploring.pdf
 - hsieh-comparison.pdf
 - Leung-Evaluation of haptically augmented touchscreen gui elements.pdf
 - macintyre-support.pdf
 - maglio-tradeoffs.pdf
 - mamykina-timeaura.pdf
 - mankoff-Heuristic evaluation of ambient display.pdf
 - matthews-attention.pdf
 - matthews-defining.pdf

- matthews-peripheral.pdf
- mccrickard-animation.pdf
- McCrickard-model.pdf
- Nielsen-Heuristic Evaluation.pdf
- olivera-Do Not Disturb - Physical Interfaces.pdf
- oulasvirta-juggling.pdf
- pacey-auditory.txt
- pousman-taxonomy.pdf
- shami-context of use.pdf
- shen-Intrusive and Non-intrusive Evaluation of Ambient Displays.pdf
- shen-methodology.pdf
- weiser-calm.pdf
- weissgerber-vispad.pdf

- FOLDER: study_result_data
 - study1_all_results.xls
 - study1_h1.csv
 - study1_h2.csv
 - study1_h3.csv
 - study2_all_results.xls
 - study2_h1_er.csv
 - study2_h1_rt.csv
 - study2_h2.csv
 - study2_h3.csv

- thermann-pils-presentation.pdf
- thermann-pils.pdf
- thermann-pils.tex

References

- [1] M. Ames, A. Dey: Description of Design Dimensions and Evaluation for Ambient Displays. University of Berkeley Technical Report, 2002
- [2] E. Arroyo, T. Selker, A. Stouffs: Interruptions as Multimodal Outputs: Which are the Less Disruptive?. Proceedings of ICMI 2002, 2002
- [3] S. Bakker, E. van den Hoven, B. Eggen: Design for the Periphery. Proceedings of EuroHaptics 2010, 2010
- [4] S. Bakker, E. van den Hoven, B. Eggen: Exploring Interactive Systems Using Peripheral Sounds. Haptic and Audio Interaction Design 2010, LNCS 6306, 2010
- [5] L. Bartram, C. Ware, T. Calvert: Moving Icons: Detection And Distraction. Proceedings of Interact 2001, 2001
- [6] R. Benbunan-Fich, R.F. Daler, T. Mavlanova: Towards New Metrics for Multitasking Behavior. Proceedings of CHI 2009, 2009
- [7] R. Benbunan-Fich, R.F. Daler, T. Mavlanova: Measuring Multitasking Behavior With Activity-based Metrics. ACM Transactions on Computer-Human Interaction, Vol. 18, No. 2, 2011
- [8] C.M. Chewar, D.S. McCrickard, A. Ndiwalana, C. North, J. Pryor, D. Tessendorf: Secondary Task Display Attributes - Optimizing Visualizations for Cognitive Task Suitability and Interference Avoidance. Proceedings of VISSYM 2002, 2002
- [9] M. Czerwinski, E. Cutrell, E. Horvitz: Instant Messaging and Interruption: Influence of Task Type on Performance. Proceedings of OZCHI 2000, 2000
- [10] D. Edge: Tangible User Interfaces for Peripheral Interaction. Technical Report Number 733: University of Cambridge Computer Laboratory, 2008
- [11] D. Edge, A.F. Blackwell: Peripheral Tangible Interaction by Analytic Design. Proceedings of TEI 2009, 2009
- [12] P.M. Fitts: The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement. Journal of Experimental Psychology, volume 47, number 6, 1954
- [13] D. Hausen, A. Butz: Entangling Ambient Information with Peripheral Interaction. Proceedings of TEI 2011, 2011
- [14] D. Hausen, A. Butz: Extending Interaction to the Periphery. Proceedings of CHI 2011, 2011
- [15] D. Hausen: Peripheral Interaction: Facilitating Interaction with Secondary Tasks. Proceedings of TEI 2012, 2012
- [16] D. Hausen, S. Boring, C. Luelling, S. Rodestock, A. Butz: StaTube: Facilitating State Management in Instant Messaging Systems. Proceedings of TEI 2012, 2012
- [17] W.R. Hazlewood, Y.K. Lim, K. Connelly, K. Makice: Exploring Evaluation Methods for Ambient Information Systems. Proceedings of CHI 2008, 2008

- [18] G. Hsieh, J. Mankoff: A Comparison of Two Peripheral Displays for Monitoring Email: Measuring Usability, Awareness, and Distraction. University of Berkeley Technical Reports, 2003
- [19] R. Leung, K. MacLean, M.B. Bertelsen, M. Saubhasik: Evaluation of Haptically Augmented Touchscreen GUI Elements Under Cognitive Load. Proceedings of ICMI 2007, 2007
- [20] B. MacIntyre, E.D. Mynatt, S. Volda, K.M. Hansen, J. Tullio, G.M. Corso: Support For Multitasking and Background Awareness Using Interactive Peripheral Displays. Proceedings of UIST 2001, 2001
- [21] P.P. Maglio, C.S. Campbell: Tradeoffs in Displaying Peripheral Information. Proceedings of CHI 2000, 2000
- [22] L. Mamykina, E. Mynatt, M.A. Terry: Time Aura: Interfaces for Pacing. Proceedings of CHI 2001, 2001
- [23] J. Mankoff, A.K. Dey, G. Hsieh, J. Kientz, S. Lederer, M. Ames: Heuristic Evaluation of Ambient Displays. Proceedings of CHI 2003, 2003
- [24] T. Matthews, A.K. Dey, J. Mankoff, S. Carter, T. Rattenbury: A Toolkit for Managing User Attention in Peripheral Displays. Proceedings of UIST 2004, 2004
- [25] T. Matthews, J. Mankoff: A Toolkit for Evaluating Peripheral Awareness Displays. Proceedings of CHI 2005 Workshop, 2005
- [26] T. Matthews, T. Rattenbury, S. Carter: Defining, Designing, and Evaluating Peripheral Displays: An Analysis Using Activity Theory. Human-Computer Interaction Journal Volume 22 Issue 1, 2007
- [27] D.S. McCrickard, J.T. Stasko, R. Catrambone: Evaluating Animation as a Mechanism for Maintaining Peripheral Awareness. Proceedings of Interact 2001, 2001
- [28] D.S. McCrickard, C.M. Chewar, J.P. Somervell, A. Ndiwalana: A Model for Notification Systems Evaluation - Assessing User Goals for Multitasking Activity. ACM Transactions on Computer-Human Interaction, Vol. 10, No. 4, 2003
- [29] J. Nielsen, R. Molich: Heuristic Evaluation of User Interfaces. Proceedings of CHI 1990, 1990
- [30] F. Olivera, M. García-Herranz, P.A. Haya, P. Llinás: Do Not Disturb: Physical Interfaces for Parallel Peripheral Interactions. Proceedings of Interact 2011, 2011
- [31] A. Oulasvirta, J. Bergstrom-Lehtovirta: Ease of Juggling: Studying the Effects of Manual Multitasking. Proceedings of CHI 2011, 2011
- [32] M. Pacey, C. MacGregor: Auditory Cues For Monitoring a Background Process: A Comparative Evaluation. Proceedings of INTERACT 2001, 2001
- [33] Z. Pousman, J. Stasko: A Taxonomy of Ambient Information Systems: Four Patterns of Design. Proceedings of AVI 2006, 2006
- [34] J. Preece, Y. Rogers, H. Sharp: Interaction Design: Beyond Human-Computer Interaction. Wiley, 2002

- [35] N.S. Shami, G. Leshed, D. Klein: Context of Use Evaluation of Peripheral Displays. Proceedings of Interact 2005, 2005
- [36] X. Shen, P. Eades, S. Hong, A. Vande Moere: Intrusive and Non-intrusive Evaluation of Ambient Displays. Workshop at Pervasive 2007: W9 - Ambient Information Systems, 2007
- [37] X. Shen: An Evaluation Methodology for Ambient Displays. Journal of Engineering, Computing and Architecture, Volume 1, Issue 2, 2007
- [38] M. Weiser, J.S. Brown: Designing Calm Technology. Powergrid Journal, 1996
- [39] D. Weissgerber, B. Bridgeman, A. Pang: VisPad: A Novel Device for Vibrotactile Force Feedback. Proceedings of Haptics 2004, 2004