IST-2003-511598 (NoE)

COGAIN

Communication by Gaze Interaction

Network of Excellence

Information Society Technologies

D6.1 State of the art report of evaluation methodology

Due date of deliverable: 31.08.2005
Actual submission date: 05.09.2005

Start date of project: 01.09.2004 Duration: 60 months

Risø National Laboratory

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	x
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Andersen, H.K., Alapetite, A., Aula, A., Isokoski, P., Majaranta, P., Itoh, K., Istance, H., and Donegan, M. (2005) ***D6.1 State of the art report of evaluation methodology.*** Communication by Gaze Interaction (COGAIN), IST-2003-511598: Deliverable 6.1. Available at http://www.cogain.org/results/reports/COGAIN-D6.1.pdf

Authors: Hans H. K. Andersen (RISØ)

Alexandre Alapetite (RISØ)

Anne Aula (UTA)

Poika Isokoski (UTA)

Päivi Majaranta (UTA)

Kenji Itoh (TIT)

Howell Istance (DMU)

Mick Donegan (ACE).

Table of Contents

1 Introduction

As stated in Annex 1 of the contract, the motivation for COGAIN is to increase the scientific understanding of usability aspects in the interaction of humans with eye movement tracking systems. The metrics and methods will be tested through user studies in both laboratory experiments and field tests. During the five year period, COGAIN will focus on the following activities: (a) system use and further development of eye movement tracking evaluation methods and metrics; (b) evaluation and analysis of eye movement tracking interfaces (developed within the COGAIN network); and (c) field and laboratory studies of individual user activities when using the eye movement tracking systems.

The first step towards these activities is to discuss the initial user requirements (from Deliverable 3.1 by Donegan et al., 2005) and potential evaluation measures. In addition to text entry research metrics and gaze-specific interaction research methods, most of the general usability evaluation methods can be applied in eye typing research. Some of the metrics are common for both text entry and usability research, e.g., measuring error rates. In addition, the user's subjective satisfaction, perceived cognitive workload, and physical strain (ergonomics and safety) are important issues to consider. Furthermore, the usability design guidelines are also important in eye typing studies. For example, general usability guidelines indicate that feedback on actions should be provided within reasonable time. This also applies for gaze input. The aim of COGAIN is to adapt all this existing knowledge into a common evaluation methodology that can be used to evaluate the systems developed within and outside COGAIN community.

The form and content of this deliverable has been discussed through e-mail and at the Work Package 6 (WP6: “Analysis and evaluation”) research retreat in Copenhagen during the COGAIN Camp. The first section focuses on explaining the need for common evaluation metrics. As documented in the WP3 (“User involvement”) deliverable "D3.1 User requirements report with observations of difficulties users are experiencing" (Donegan et al., 2005), this need is grounded in the gap that exists between users' needs and the capability of the current state of the art eye-gaze control systems. The next two sections determine the objective of evaluating gaze interaction systems for disabled people. First of all, these sections determine *what* and *how* to measure. As a point of departure, the section focuses on the issues that can be learnt from traditional usability evaluation methodology. The section also presents and discusses an initial version of a usability-testing scheme. The following subsections contain short descriptions of common usability evaluation methods. Following the brief descriptions of the different methods, the limitations and possibilities of these methods are discussed in relation to evaluating gaze-controlled interfaces. General usability evaluation methods are valid for evaluating text entry systems. However, efficiency is often the primary factor in determining the suitability of a text entry method. Therefore, methods for evaluating the efficiency of text entry systems are needed. A dedicated subsection deals with this issue. The fourth section presents a case study where usability measures were used to evaluate two different gaze typing interfaces running in Japanese, namely GazeTalk and Dasher. Accessibility is an issue related to many usability aspects. The Appendix A provides an overview of concepts and needed accessibility standards for web sites. Web sites are a domain where accessibility has been studied in detail, and where rules and recommendations have been established. That is why it is a good starting point when studying accessibility.

2 Why measure

As stated in the WP3 deliverable "D3.1 User requirements report with observations of difficulties users are experiencing" it seems that, at present, eye control can only be used effectively to meet a limited range of user requirements. Furthermore, it can only be used effectively by a limited number of people with disabilities who might benefit greatly from it. This also emphasizes the need to develop a set of common evaluation metrics that can be used in standardized ways across the community of people concerned with providing research, aids, care, etc. for the benefit of the disabled. Moreover, WP3 states a number of hardware and software user requirements that also calls for such a development. The main requirements produced in WP3 are cited in the two sections below (Donegan et al., 2005).

2.1 End-users' eye control hardware requirements

It is recommended that a good starting point would be to measure how effectively the eye control technology available can meet the needs of the full range of users who might benefit from it. To achieve this aim, it is recommended that WP3 (User Involvement) should trial as many specialist eye control systems as possible. This will provide an opportunity to feed back to eye control system developers how effectively their technology is meeting the needs of the full range of existing and potential users. In addition, it will provide an opportunity to make observations and suggestions relating to any potential modifications to their systems and/or software that might make it more accessible and/or more effective for more users. As the above information is acquired, to enable users to make an informed choice of which hardware to consider for their eye control needs, it is recommended that WP3 should add the information gathered from the above investigations to the WP5 catalogue of currently available eye trackers. The emphasis of the information provided by WP3 should be specifically related to usability issues related to the requirements of end-users with disabilities, e.g. environmental control, portability issues, mounting issues, 'choice of output methods', 'range of access methods', etc.

2.2 End-users' eye control software requirements

Features of the wide range of assistive software already being successfully used via a range of access methods in addition to eye control include the following: resizable cells and grids; a range of input methods; a wide choice of output methods; a choice of symbols or text output; a wide choice of text styles and colours; a range of speech facilities; a choice of languages, etc. As a result, it is recommended that the following issues be investigated with the involvement of the users themselves. Of the wide range of specialist (non-eye control) software that is already successfully being used by many people with disabilities for access and communication, find out which can be adapted effectively for eye control (e.g. The Grid, SAW). This will enable COGAIN partners to make a comparison of how effectively both the existing range of software specifically designed for eye control and the adapted specialist software compare in terms of terms of their efficacy with eye control systems. As a result, recommendation for modifications can be made to the current range of software that can (or could) be used for eye control, so that it meets as many of the needs of as many existing and potential users as possible. As the above information is acquired, to enable users to make an informed choice of which software to use for eye control, it is recommended that a matrix should be set up on the COGAIN website relating to features of different software that can (or could be) used for eye control. The comparison would be based on features such as those described above, such as 'choice of output methods', 'range of access methods', 'range of multi-modal access', etc.

3 What to measure

This section will focus on determining the objective of evaluation of gaze interaction systems for disabled people.

First, we need to determine what we want to measure. We can learn from traditional usability evaluation methodology, always keeping in mind that the concept of usability within systems for the disabled might have different connotations than what is usable agreed upon within the mainstream human-computer interaction area.

Furthermore, assistive technology evaluations differ from "typical" evaluations. There are virtually no standardized tests to "find out" what kind of technology a disabled person needs to use. Instead, an assistive technology evaluation looks at the results of all recent evaluations, along with the current goals and objectives of assisting the disabled person in his or hers personal life as a whole. Typically, an evaluation team or an evaluator first interviews the disabled person and conducts first trials with the technology. Interviews can also be carried out with the family and other significant people working with the disabled person. Ideally, the evaluation process is based on careful observational work coupled with trials with possible devices from low to high technology. Data are gathered from these trials about the effectiveness of various technologies to meet the needs of the disabled. The environment is carefully examined, especially when the device has to work in a variety of settings. Information is collected concerning the disabled person's abilities and accuracy when using various technologies, including the positioning and settings that work best. The evaluation could also include the disabled person's (could also include family and other involved people) feelings towards the technology as well as the technology. This deliverable will not establish how to conduct this evaluation processes in detail. Instead, we want to examine if it is possible to develop common evaluation measures on a more general level. These general evaluation measures could help the evaluation teams and evaluators prepare for their careful examinations and assessments.

The concept of 'usability' was defined in the ISO standard 9241-11 (1998) as "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use. In this standard, effectiveness was defined as "the accuracy and completeness with which users achieve specified goals", efficiency was defined as "The resources expended in relation to the accuracy and completeness with which users achieve goals", and satisfaction was defined as "The freedom from discomfort, and positive attitudes towards the use of the product". This was later refined in the 2001 standard "ISO/IEC 9126-1" where a broader definition was suggested:

- **Quality in use**: the capability of the software product to enable specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction in specified contexts of use.
- **Functionality**: the capability of the software to provide functions that meets stated and implied needs when the software is used under specified conditions.
- **Reliability**: the capability of the software to maintain its level of performance when used under specified conditions.
- **Usability**: the capability of the software product to be understood, learned, used and liked by the user, when used under specified conditions.
- **Efficiency**: the capability of the software to provide the required performance, relative to the amount of resources used, under stated conditions.

A more narrow definition was suggested by Jakob Nielsen (1993) (see Figure 1).

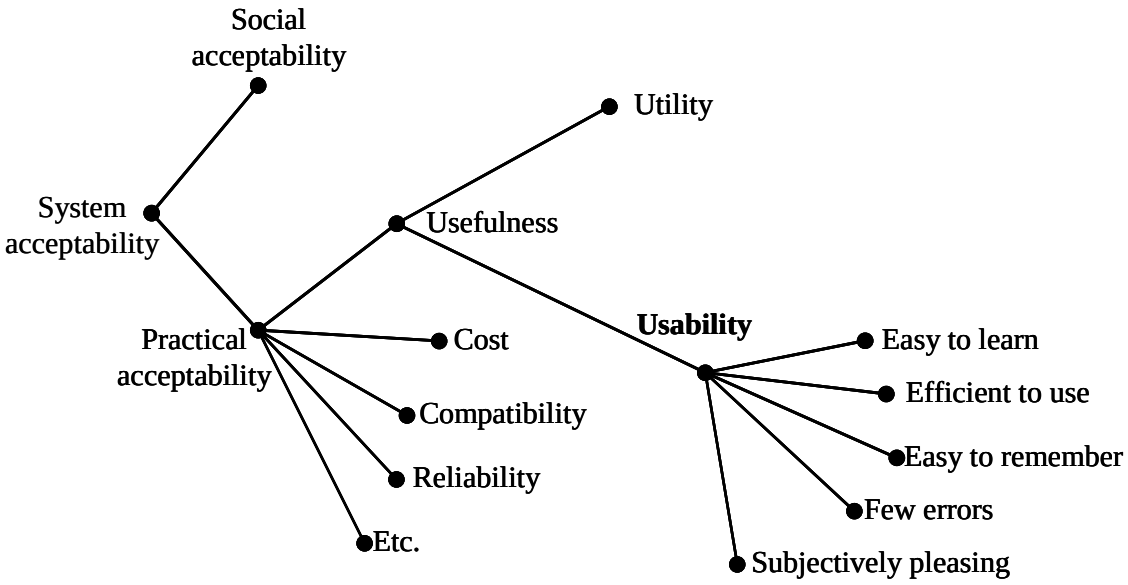

Figure 1. Usability evaluation elements.

4 How to measure

This section focuses on determining a range of possible methods for measuring usability of gaze interaction systems for disabled people. When possible, we relate the common concepts of usability (as defined above) to the domain of gaze interaction systems in general and specifically to systems for the disabled people. However, the intention here is not to describe a complete and unambiguous correct evaluation methodology. That is the final goal of COGAIN WP6, but the topic of this report is to describe the state of the art with its deficiencies and inconsistencies.

4.1 Common evaluation criteria

Any evaluation method or metric of an eye gaze interaction system must meet certain criteria. These include the following:

Validity: This is the extent to which the evaluation method measures what it is presumed to measure. There is a range of validity types that can be addressed (e.g. face validity, concurrent validity, construct validity).

Reliability: This indicates how consistent the method is. The common form is test-retest reliability, which is the ability of a method to provide consistent results on different occasions. Reliability also refers to the requirement that a system should be dependable and provide a high level of uninterrupted operation.

Sensitivity: This refers to the ability of the method to distinguish between different use conditions.

Diagnosticity: How capable is the method to measure demands on specific resources of the user? This refers to the ability of the evaluation method to discern the type or cause of user's workload when using the system, for example sensory, perceptual, cognitive or psychomotor mechanisms.

Intrusiveness: This is the methods ability to measure to which extent an eye gaze interaction systems intrudes on normal use situation or makes the user uncomfortable.

Applicability: This criterion bridges the gap between laboratory and normal use situations in user's own environment. The method should be able to function equally well in both normal and laboratory conditions (i.e. reproduce results in both the field and laboratory studies).

Acceptability: This is the method's ability to measure how acceptable users consider the technique to be and whether it intrudes on other aspects of their lives.

Implementation: This criterion reflects the practicality of the technique. This is the methods capability to evaluate on the practicalities of an eye gaze interaction system. E.g., how portable the equipment is and whether it can physically fit where it is needed and the undue time and effort is needed operate and maintain the system.

4.2 Usability evaluation methodologies

This section contains short descriptions of common usability evaluation methods. A report on usability evaluation methods, edited by Ovaska, Aula and Majaranta (2005; in Finnish) was used as a general background source when writing the descriptions.

These short descriptions are meant to act as examples of the most common usability evaluation methods currently available. There are many variations of the basic methods, and there are also other methods not

described here. Most importantly, the reader should understand that the conventional usability evaluation methods may not work well with special user groups, as noted by Lepistö and Ovaska (2004). It is, however, possible and sometimes necessary to adjust the methods to suit the characteristics of the participants. It may also be beneficial to collect data with several complementary methods.

Following the brief descriptions of the different methods, the limitations and possibilities of these methods are discussed in relation to evaluating gaze-controlled interfaces.

4.2.1 Usability testing and think aloud

Usability testing provides information about how the user *actually* uses the product. A few representatives of the target user group execute tasks, which represent the most common and typical tasks in the interface. The user is often asked to think aloud during solving the tasks. Think aloud provides information about the problems in the interface, how the user perceives the product, and what kind of mental models she has. Data analysis may be based on videotapes or the notes of an observer but may also include automatic log files (e.g., logged keyboard activity or eye tracking data revealing the user's gaze behaviour). Usability testing not only gives concrete information about the problems from the user's point of view but also shows which parts of the system function well enough. Usability testing is typically carried out in a laboratory, thus the situation and the context are artificial and may affect the results. The tests and especially the analysis take a lot of time but produce fairly reliable results.

4.2.2 Field studies

The purpose of field studies, such as studies based on ethnography or contextual inquiry, is to gain an understanding of the physical and social context where the product is used. In these studies, different data collection methods are utilized, such as interviews and observations. Field studies are suitable in different stages of the product lifecycle; they can provide information needed in designing completely new products (user requirements), and they can also be used for studying the usability of an existing product. For field studies to be successful it is important that the researcher and the user truly interact with each other; the users should not be treated as subjects of testing but instead, they should be treated active participants in the process of studying and even in designing the products.

4.2.3 Expert evaluation

In expert evaluation, the usability of the product is evaluated by a group of usability experts. Typically, expert evaluation is based on a list of usability principles or guidelines (e.g., standards) against which the product is being inspected. The most common expert evaluation method is heuristic evaluation (Nielsen and Molich, 1990; Nielsen, 1994a; Nielsen, 1994b) which is based on a list of ten usability heuristics. In expert evaluations, the usability of the product is typically evaluated by a group of about three to five experts each using a couple of hours for the evaluation. Thus, it is not surprising that expert evaluations have been referred to as discount usability methods; they are relatively cheap and fast to conduct while at the same time producing reasonably good results. However, these methods have also received criticism as to whether the evaluators will actually find usability problems that the actual users of the product will face.

4.2.4 Walkthroughs

In walkthroughs, as the term suggests, the evaluator "walks through" the interface, trying to spot potential usability problems. Walkthroughs can be conducted with or without the end user, and there are many variations of the method. Cognitive walkthrough is conducted by a usability expert, who tries to model the user's actions and thinking processes. The main aim is to find out if the interface is easy to learn by applying explorative learning methods. By asking predefined questions, the evaluator makes assessments on how intuitive the interface is. Pluralistic usability walkthrough or group walkthrough involves users, designers and usability experts in the evaluation process. The participants may act as users or observers. The findings are

discussed after the walkthrough. Walkthroughs are especially useful in the early stage of the development, as simple screenshots can be used to evaluate the interface. Since real interaction between the user and the interface is missing, it is not possible to find all kinds of usability problems.

4.2.5 Automatic usability evaluation

Since usability evaluation typically requires a lot of time and the development resources are limited, there have been efforts to make tools that automatically conduct at least part of the evaluation. These include tools for usability testing, inspection, conducting queries, user or interface modelling, and simulation. A significant part of the tool is the automated data collection but the tool may also make analyses and produce suggestions for improvements. Tools for evaluating web pages are especially popular these days; one can fairly easily test if the pages follow web standards (such as HTML or CSS) and accessibility rules. Automatic tools may reduce costs but using the tools may still require a lot of time, and they definitely do not replace a human evaluator. Furthermore, they lack qualitative and subjective feedback from the user.

4.2.6 Questionnaires

In usability studies, questionnaires are commonly used, for example, for collecting background information about the participants and for collecting information about subjective attitudes towards the product being tested. Questionnaires can consist of different types of questions, such as open-ended or closed questions, which provide different kinds of information and which require different methods for analysing the responses. There are several questionnaires already available for usability evaluations, such as Software Usability Measurement Inventory (SUMI) and Questionnaire for User Interaction Satisfaction (QUIS). Questionnaires are commonly used as an additional data collection method in usability studies, although they can also be used as the only data collection method.

4.2.7 Interviews

Interviews are typically used for collecting information about the users' attitudes and experiences towards the product being tested. Interviews can be structured, semi-structured, or open, and they can be conducted as individual, pair or group interviews. The many different possibilities for conducting interviews along with the direct interaction with the interviewee makes interviewing a flexible and rich data collection method, although the analysis of the data is sometimes time-consuming. Typically, interviews are used as an additional data collection method in usability studies (e.g., in usability testing).

4.2.8 Focus groups

Focus groups were originally developed for the needs of marketing research, but currently they are widely used in usability-related studies, as well. Focus group method is a semi-structured discussion with approximately 4-10 participants. The discussion is lead by a moderator. Focus group method is especially suitable for studying the meanings and norms formed in the group, along with individual opinions of the participants. The benefits of this method include the short time needed for the data collection and the flexibility of the interaction due to face-to-face interaction. However, the reliability of the data may suffer from the group dynamics; some participants may not express their opinions honestly because of the pressure caused by the group situation.

4.2.9 Using the methods for evaluating gaze-controlled interfaces

When evaluating gaze-controlled interfaces, the target group consists of people with severe motor disabilities. They may not be able to speak, and thus they often use alternative communication methods. The use of alternative communication methods sometimes means that the commonly used usability evaluation methods are not suitable as such. However, the methods can be modified according to the demands of the situation. In general, working with people with disabilities requires great flexibility and thoughtfulness from the

researcher. Some possible modifications and problems with the existing usability evaluation methods are presented next.

The use of verbal protocols, such as interviews, focus groups and think-aloud protocol, is not possible if the person is not able to speak. However, interviews can also be conducted via alternative communication methods. An assistant may be needed (at least in the beginning) if the researcher is not familiar with the communication method used by the participant (e.g., the participant may use communicative symbols instead of written text). It should also be noted that interviews take more time with the alternative communication methods and thus, the researcher should carefully plan the interview in order to ask relevant questions from the interviewee. The use of think aloud method is not possible when the participant is not able to speak; however, information on the participant's thoughts and opinions can be elicited after the use in an interview.

Usability testing in a usability laboratory may not be possible with people with severe disabilities. Their health may require special medical care, as for example, in the late stage of ALS where constant life support is needed. Thus, field studies are often the best choice when evaluating gaze-based systems. In addition to being an easier method for the participant, field studies provide important information on the context where the product is actually used, as explained above.

In order to successfully conduct expert evaluations and walkthroughs, the evaluator would need to have a clear understanding of the end users' needs and requirements for the system, as well as the limitations for use caused by the disability. In the case of the end users being severely disabled people, the position of the "expert" is extremely demanding in this respect. Thus, the expert evaluation cannot be the only method used for evaluating the usability of the product.

Questionnaires may be a good method for evaluating the usability of gaze-based systems. In the case where the user can control the computer by gaze, the questionnaire can be designed so that the user can fill it independently. If the user is able to use eye typing, the questions can even be open-ended, although it might be better to strive for closed questions with pre-formulated choices.

4.3 Text entry method evaluation

General usability evaluation methods are valid for evaluating text entry systems. However, efficiency is often a primary factor in determining the suitability of a text entry method. For example, the flow of a discussion is often disturbed if one of the participants is much slower in expressing his or her views. This can happen if the text entry system in a communications aid is too slow. Therefore, methods for evaluating the efficiency of text entry systems are needed. We are not aware of methods especially developed for gaze based text entry. However, text entry researchers that do text entry method development for the general public have used two main approaches: modeling and experiments.

Modeling can produce useful results especially for estimating expert performance. Text entry methods are often used extensively, and over time, user's performance approaches the theoretical upper limit. This limit can often be estimated based on knowledge on human performance in similar tasks. A simple example of a useful modeling result is that if one method of text entry requires 1.2 key presses per character on average and another 1.4, it is reasonable to expect that the first one is more efficient in expert use. Models are good for estimating expert motor performance and the efficiency of the overall design, but good models are notoriously difficult to construct for tasks that require cognitive action to be taken by the user. Because of this, it is often necessary to perform experiments to evaluate a text entry system. The experimental approach is often the simplest way to evaluate the efficiency of a text entry system when used by beginners. There are two main experimental approaches. The first disallows or ignores error correction activity and the second uses various metrics to describe the extent of errors and error correction activity.

It is easy and often sufficient to just have users write something and observe the quality of the resulting text and the writing speed. However, to get more reliable and repeatable results, the evaluation method should

account for differences in the speed/accuracy trade-off and differences in the text written. The former requires that errors are tabulated and reported in a standard format. For the latter purpose, it is useful to use a standard text corpus to be written in all evaluations. In recent years, the corpus assembled by Soukoreff and MacKenzie (2003) has been popular in text entry method evaluations. It consists of 500 English phrases and its character frequency distribution matches that of larger corpora rather well. We have observed the following four approaches in conducting text entry method experiments.

No error tabulation: This kind of evaluation does not force the user to write anything specific. This means that it is impossible to tell in automated means whether the resulting text is what was intended. Thus, at least in the case of evaluation with large number of users or long text passages error tabulation becomes expensive because of the large amount of manual work.

Forced synchronization: When the evaluator wishes to lessen the amount of manual work, entering erroneous text is often made impossible. The user is presented a text passage to transcribe and the system prevents the entry of wrong characters and counts the attempts to do so as errors.

The MSD/KSPC method: The MSD/KSPC method involves presenting text to transcribe and allowing normal error correction behavior. After user's activity has been recorded, the Levenshtein's algorithm is used to compare the presented and transcribed strings and the number of errors is tabulated. The number of key presses or other units of input action that is needed for entering the text is compared to the number that the user actually used. This gives the key strokes per character (KSPC) metric, which reflects the amount of excess work done to correct errors. Note the two different uses of KSPC. It can be used a priori to model text entry methods, and a posteriori to characterize user behavior. These two should not be confused.

The input stream method: The input stream method compares the presented text to the input stream produced by the user. In different systems, the input stream consists of different tokens. For illustration, it is convenient to think of the keyboard. Usually text is written by pressing keys corresponding to the alphabet, punctuation, and other symbols that appear in the text. Sometimes, however, other keys such as the backspace key are used. The input stream intended here includes all tokens, not just the entered alphabet. The principle of the input stream method is the same as in the MSD/KSPC method, but some additional algorithms are used to handle the ambiguous situations in the input stream to produce data that is more detailed than the aggregate KSPC figure. The automatic analysis produces information on the kinds of errors that occurred and possibly on incomplete input actions that often occur for example in handwriting recognition. Overall, the input stream method is a more detailed version of the MSD/KSPC method. Its results can be used to compute the MSD/KSPC metrics as well.

4.3.1 Pros and cons of the text entry method evaluation methods

The shortcomings of the 500 phrase corpus by Soukoreff and MacKenzie (2003) are that it is available only in English, its content is not tuned for the kind of phrases that users of gaze-based communications aids would need. The semantic content of the phrases is important in evaluating some types of language prediction systems. The no error tabulation and forced synchronization methods produce inferior data for obvious reasons. The input stream method produces too much data to be easily comprehended, but can give text entry method developers an automated way to get data on the performance of the system that is difficult to attain by any other means. The MSD/KSPC method appears to strike the right balance between detail and comprehensibility of the results. Models are the most efficient way to estimate expert performance when experts are not available. Experiments are often the only way to get reliable data on the performance of beginners and their learning rate.

The methods and metrics described above have been developed in the English speaking HCI community. This means that their usefulness may be limited in other contexts. One of the reasons for including a Japanese case study in this report is the descriptions of Japanese text entry that it contains. One can ask whether the character-based error tabulation methods, for example, are well suited for a language where characters have a somewhat different role.

4.3.2 Recommendations

The MSD/KSPC methodology should be adapted for evaluating gaze based text entry systems when the results are intended for wide dissemination, and the data produced by the input stream method should be used in the development of text entry systems. The MSD/KSPC method and some of its extensions are described in a paper by Soukoreff and Mackenzie (2003). The input stream method is described in a paper by Wobbrock and Myers (2005). A text corpus consisting of typical phrases used in gaze based communications aids should be assembled and translated to several languages. However, this corpus should not be recommended as the only basis of evaluation. Some users have very limited basic communication use for their text entry systems while others use it in many areas of life. The latter group will appreciate systems that perform well with many kinds of text. New text entry systems should first be modeled to understand the limits of their performance. If the modeling results look good, a longitudinal experiment is the most reliable way to evaluate their efficiency. The survey by MacKenzie and Soukoreff (2003) is a good starting point for familiarizing oneself with the state of the art in text entry research.

4.3.3 Pointing device evaluation methods

General usability evaluation methods can be used for evaluating pointing devices. However, pointing is often done repetitively for long periods of time in graphical user interfaces. Therefore, the efficiency of pointing has a large impact on many aspects of usability. Input device researchers have developed many ways of evaluating the performance of pointing devices. However, by far the most popular and rigorous method is based on the Fitts' paradigm of pointing device experiments. This procedure is outlined in many research papers including the recent survey by Soukoreff and MacKenzie (2004). In addition, it is recommended in the informative Annex B of the ISO 9241-9 standard (2000).

4.3.4 Pros and cons of the Fitts' Law based testing of pointing devices in the context of gaze-based user interfaces.

Work on the Fitts' paradigm and eye trackers is relatively rare. We are not aware of any experiments that would have applied the standard methodology according to the recent recommendations by Soukoreff and Mackenzie (2003). Earlier work, while slightly different, suggests that eye-tracker pointing may be effectively modeled with Fitts' law.

4.3.5 Recommendations

A standard protocol for measuring the efficiency of eye tracker as a pointing device, should be developed based on the Fitts' law paradigm that is prevalent in other areas of pointing device research.

4.4 COGAIN usability testing scheme.

To be useful, a usability measurement scheme needs to be relevant to the problem, unambiguous in how it is applied, and needs to produce results that are repeatable, consistent and also readily understandable by the target user audience. In the context of eye-gaze communication systems, this means that the usability attributes addressed in the scheme must be relevant to the needs of the disabled groups of users of gaze-based systems. It also means that a set of repeatable and transparent standard tests has to be devised such these can be applied to any new product and that the results obtained can be compared with other systems on the market. The outcomes of the testing need to be presented in such a way that they can be readily published and understood by a wide range of primary and secondary users, as well as access centres and advisory services. The usability of gaze-based systems is an important attribute, but not the only consideration for many wishing to make use of these types of system. Cost is also a very important consideration and many users will want to consider tradeoffs between cost, functionality, and usability in deciding whether to invest in this type of

assistive technology. The usability information provided about the system needs therefore to be sensitive to the types of tasks and applications that the individual wishes to undertake or use with the system.

A usability testing scheme suitable for gaze-based systems needs to address this set of requirements.

Usage phase	Application area		
	Software Control	*Mobility Control*	*Environmental Control*
Installation			**Procedures** for one or more standard usability tests **Qualitative outcomes**: pros and cons **Quantitative outcomes**: star ratings e.g. ***** less than 5 minutes *** between 5 and 10 minutes * more than 10 minutes
Calibration			
Operation			
Other			

Table 1. The usability testing scheme

The columns in the table above show a coarse categorisation of the applications that gaze-systems are used for, which has come from the WP1 initial review of user requirements. Software control refers to interacting with applications to conduct work of some kind, such as eye typing or interacting with browsers or other screen-based applications. Mobility and environmental control as categories are self-explanatory.

The rows correspond to different phases in the use of a system, and it is important that the usability of all aspects of a gaze-based system are considered, not only performance or preference measures once the system has been set up and calibrated.

The contents of each cell will consist of descriptions for one or more standard usability tests, and a description of how a system would be rated in accordance with these tests. In some cases, usability metrics will be appropriate and levels of these can be equated to a star-based rating system, similar to those found usually in consumer product reviews. In some cases, a qualitative description of usability problems and benefits will complement or replace the quantitative outcomes. An example of this can be seen at the ACE Centre's web site containing a comparison of head pointers. This approach delivers usability metrics that are both comprehensive and also transparent and easily understood by the target audiences of the usability reports.

The testing itself would be carried out by COGAIN-accredited access centres in a variety of European countries. There are parallels in this proposal to the Common Industry Format (CIF) Usability Tests developed in the US under the auspices of NIST (Bevan, 1999). In addition, Douglas et al. (1999) have discussed procedures for testing device performance and user assessment of hand controlled pointing devices in accordance with ISO9241, part 9 (2000). Both of these approaches can be used to inform the design of suitable tests. Finally, there is a range of relevant work to draw on in the development of suitable usability metrics from the seminal early work by Whiteside, Bennett and Holtzblatt (1988) through to the application of these ideas to the assessment of gaze-based pointing devices (Bates and Istance, 2003)

5 A usability case-study of two eye-typing systems

5.1 Objective

The purpose of this part in the deliverable is to present application of usability measures to evaluation of actual gaze interaction systems. We report a case study in which usability comparisons were carried out for two different gaze typing interfaces running in Japanese namely GazeTalk and Dasher. The example presented in this chapter is a work in progress by Kenji Itoh, Hirotaka Aoki and John Paulin Hansen.

5.2 Japanese text typing by gaze Interaction

5.2.1 Text entry in Japanese language

In the Japanese language, we primarily use three systems of Japanese specific characters, i.e., "Hiragana", "Katakana", and "Kanji (Chinese characters)", as well as alphanumeric letters and symbols. The former two character systems ("Hiragana" and "Katakana") are phonetic alphabets, and each system comprises approximately fifty characters. In general, "Katakana" characters are used only for writing words of non-Japanese origin such as persons' names and cities in other countries, and modern words developed in western countries such as computers and video games. We usually divide all of Hiragana or Katakana characters into ten groups in terms of ten consonants, i.e., null, "k", "s", "t", "n", "h", "m", "y", "r", and "w", combined with five vowels, i.e., "a", "i", "u", "e" and "o". For example, the group (line) of "k" includes five phonetic characters, "ka", "ki", "ku", "ke" and "ko" in Hiragana or Katakana. As such, each Hiragana or Katakana character has a Romanization.

As another characteristic of the Japanese language, a written text is composed in combination of Hiragana, Katakana, Kanji, and alphanumeric letters. In making a Japanese sentence in a computer-based text entry system, we first input Hiragana characters, and then convert them into a corresponding representation of mixed Kanji/Hiragana characters using a "Kana-Kanji conversion" programme. Thus, the characters included in the mixed Kanji/Hiragana sentence may be slightly different from those made by other individuals even if it is converted from the same Hiragana characters, depending, for example, on the education level such as adults versus elementary school boys and girls. In addition, it is importance to notice that written sentences that are completely correct in terms of syntax and semantics must be represented in correct combination of the three character sets, since lack of the correctness makes the sentences to be incomprehensible.

5.2.2 GazeTalk: Hierarchical, static menu system

The original version of GazeTalk (Hansen et al., 2001; 2002) was developed for Danish and English language users. It equipped a character prediction function applying a Markov Chain Model, which predicts the most likely six letters subsequent to the last typed character. For this feature, this version employed a dynamic menu system in which a key of any character should be changed in its position dynamically. The present Danish and English versions equip a word prediction function in addition to the character prediction. In contrast, the Japanese version furnishes neither the character nor the word prediction and consequently has no language model since this function does not seem to work well in Japanese due to a completely different language system from European languages, as briefly mentioned above.

As the figures below illustrates, like in the original GazeTalk for Danish/English users, each menu in the system comprises eight large on-screen keys as well as a double-key-space text field in the Japanese version. A user can activate (or "push") each of these on-screen keys by gazing it for a specified "dwell time". As default, the dwell time was usually selected at 500 ms, and then it can be shorter or longer, depending on the user's preference. As a feedback of gaze to a user, in the present version of the GazeTalk, a typeface of a key, which is currently gazed at, is changed in its size (see Figure 2). It starts getting smaller when the user shifts his/her eyes to a key, and the typeface continues to become smaller while he/she is gazing it. The size of the typeface represents a time remained for activation of the key, and the key is activated when its size becomes zero, i.e., when the typeface disappears on the key.

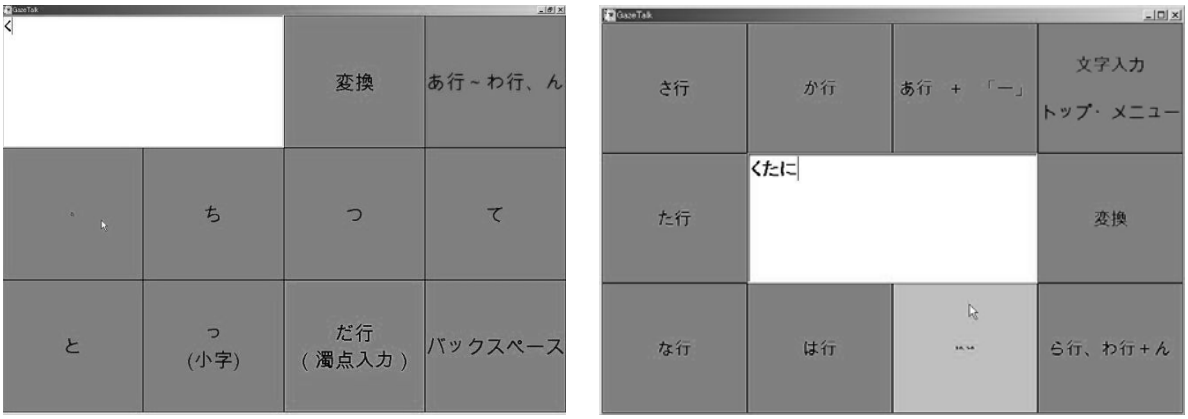

Figure 2. Layout of a menu in the Standard version of the Japanese GazeTalk (character-level menu) (left) and layout of a menu in the centre-text version of the Japanese GazeTalk ("Kana" top menu) (right).

Taking into account the characteristics of the Japanese language mentioned previously, we decided to adopt a static, hierarchical menu structure for the typing interface of the Japanese version. In this study, we implemented two versions of GazeTalk for Japanese users to examine effects of the text field position on gaze typing usability. One is called the Standard version of (Japanese) GazeTalk (abbreviating S-GazeTalk) whose position of the text field is the same as in the original Danish/English GazeTalk, i.e., in the upper-left corner having double space of a key (cf. Figure 2). As its derivative, which is called the Centre-text version of GazeTalk (C-GazeTalk in short), the text field was slightly modified to locate in the middle of the display so that a user can easily and comfortably check input text during gaze typing (cf. Figure 2).

To easily get an idea of the hierarchical menu structure, we describe briefly how to type Japanese text with the GazeTalk interface as follows: At the "Kana" top menu, as shown in Figure 2 for the Centre-text version, an entry for each of the above-mentioned Hiragana groups is allocated to each key. When one presses, for example, the key located in the left of the text field in Figure 2, by gazing it for a specific dwell time, then a next-level menu appears in which five Hiragana characters composing this group are included: "ta", "ti", "tu", "te" and "to", as can be seen in Figure 2 for the Standard version. Then, in this "character-level" menu, a Hiragana character can be typed by fixating a key which one wants to input. Thus, each of most Hiragana characters can be typed by two "gaze" clicks, i.e., one in the "Kana" top menu and the other in the character-level menu. As mentioned previously, the Japanese version also couples with the "Kana-Kanji conversion" programme to produce a usual mixed Kanji/hiragana text. The typing system includes one or more keys relevant to this function in each menu. For example, a key for initiating the conversion is allocated in the right of the text field in Figure.

5.2.3 Dasher: Text entry system using continuous gesture

Dasher (Ward, 2001; Ward and MacKay, 2002) is a text entry system, which has a novel interface incorporating a language model and is driven by continuous two-dimensional gestures such as a mouse,

touch-screen, rollerball, breathing device or eye-tracker. The language model is trained on example documents, i.e., training corpus, and it allows Dasher to predict the probability of each character's occurrence in a given context. The size of space is allocated for each letter and successive characters according to the predicted probability. A screen shot of Dasher is shown in Figure 3. It is reported that, when using a mouse as the steering device with Dasher, novice users can be learned to type at more than 25 words per minute after one hour practice and experts can type at 34 words per minute (Ward and MacKay, 2002) while typing speed for experts with a large (246 mm wide) standard QWERTY keyboard is 32.5 words per minute (Sears et al., 1993).

Dasher works in most languages, and has several derivatives for the Japanese language. Among those we chose the "Hiragana 60" version of Dasher – which has a set of 60 Hiragana characters, plus numerals and special Japanese symbols – and used it with a 380,000 Hiragana character training corpus of "everyday phrases" in this study. Dasher works in most languages, and has several derivatives for the Japanese language. Among those we chose the "Hiragana 60" version of Dasher – which has a set of 60 Hiragana characters, plus numerals and special Japanese symbols – and used it with a 380,000 Hiragana character training corpus of "everyday phrases" in this study.

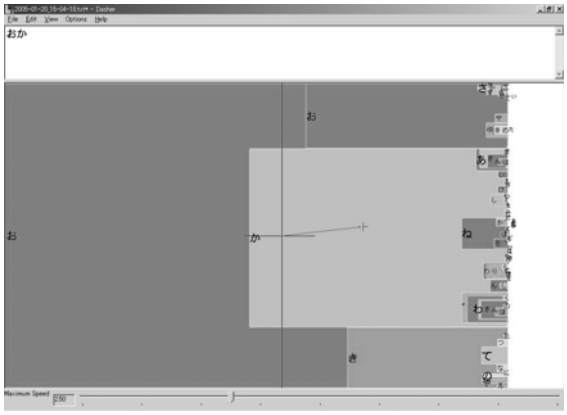

Figure 3. Screen configuration of Dasher.

5.3 Experiment

5.3.1 Subjects

Fifteen Japanese students in Tokyo Institute of Technology participated in the experiment as subjects. Their mean age was 21 years old (ranging 19-25 years old). Subjects were grouped to three groups, each of which performed experimental trials with only one of the typing systems. Each group was arranged to be formed as identically as possible in terms of experience and skills of gaze typing and use of computer. For such homogeneous allocation of subjects to typing systems, we conducted a preliminary test using another Japanese gaze typing system, Hearty Ladder which is an on-screen keyboard of Hiragana characters, to estimate their initial (baseline) skills of gaze typing. Each subject was paid 750 Japanese yen (JPY; approximately 6 euros) per hour for participation in the experiment. We informed him/her that top two typists for each typing system group in terms of combined measure of efficiency and accuracy could receive an extra prize, i.e., 5000 JPY (approximately 40 euros) for the best and 3000 JPY (approximately 25 euros) for the second best typist, so that he or she can keep high motivation for participation in the four day experiment.

5.3.2 Task

The experimental task was to type Japanese-written phrases or sentences of daily conversation only in Hiragana characters – supposing ALS (amyotrophic lateral sclerosis) patients' use of such systems and supporting this kind of communication by a computer-based system need not be used Kanji characters – with a gaze typing system operated with the eye-tracking system. Examples of typed sentences (in translation) were: "Give me water", "Turn on the TV set", and "What time is it?" Each sentence was made up of 8-23 characters (15 on average), including Hiragana characters as well as numerical numbers. The task was performed block by block. Each block comprised 20 sentences, which contained approximately 300 characters. Subjects were instructed to type these sentences as fast and accurate as possible. For the first ten sentences within a block, the eye-tracker was calibrated using the "standard" procedure to obtain good

accuracy of eye-tracking, and the other half of sentences were typed with application of a "deliberate miscalibration" for emulation of a low-cost, low quality tracker, as will be described later. A typed sentence was spoken out by the experimenter. With a long sentence, it was divided into several parts and each part was spoken with the progress of the subject's typing. We gave the subject the instruction that, when he/she forgot a sentence to be typed, he/she would ask the experimenter to speak it. We also instructed him/her that correction of typing error could be made when he/she realised it later at the time of text verification, but this need not be made.

5.3.3 Apparatus

The gaze typing system was run on a personal computer (CPU: 933 MHz) operated by Windows 2000 including the IME Kana-Kanji conversion programme, with a 17-inch colour monitor (1024 x 768 pixels) – the viewing distance from the subject to the screen was 70 cm. The dwell time for key activation with GazeTalk was initially set at 500 ms which was identical to the one in our previous studies (Aoki et al., 2003), and the subject could change it according to his/her preference in any experimental block. We initially set a speed parameter for Dasher (maximum speed) at 1.5, and the subject could also change it at his/her disposal. A QuickGlance system (EyeTech Digital System) was combined with the typing system as an eye-tracking device with a tuning 15 of the update-rate and 7 of the smoothing-factor.

5.3.4 Procedure

In this study, the following three experimental factors were specifically examined: (1) typing systems (between-subject factor), (2) accuracy of eye-tracking systems (within-subject factor) and (3) learning effect (within-subject factor). For the first experimental factor, we used three gaze typing systems mentioned in section 5.2: S-GazeTalk, C-GazeTalk and Dasher. The second factor was controlled by arranging a calibration procedure of the eye-tracker. High accuracy of eye tracking was produced by calibrating the tracker using the standard QuickGlance procedure. Low accuracy was made by applying a deliberate miscalibration procedure in which each fixation point to be calibrated was slightly distorted intentionally from a real calibration target, i.e., with 2 degree error from the target in a direction randomly selected: up, down, right or left.

The learning effect was examined in terms of differences between all or some of seven experimental blocks in usability indices mentioned below. The experiment was carried out with each subject in different four days, taking him or her at longest one and a half hour in each day. Approximately a week prior to starting the experimental session, a preliminary test was conducted for allocating subjects to typing systems, as mentioned in section 5.1. Subsequently, a general instruction was given to the subject: purposes of the experiment, typing systems – how to use each system, experimental procedure, etc.

On Day 1, before the experimental session, a subject performed a training session with a typing system that he/she would use in the experiment for approximately 10 minutes. Then, he/she performed one block of the gaze typing task – which included ten sentences with high accuracy of eye-tracking and other 10 sentences with low accuracy, as mentioned above. Between two conditions of accuracy, a short break (approximately 5 minutes) was given to the subject. At the end of Day 1, he/she responded to a questionnaire on subjective opinions about the system that he/she used in the experiment.

On the other days, Days 2-4, the subject first received a 5-minute warm-up trial, and subsequently they performed two blocks of the typing task. On the last day of the experiment, i.e., Day 4, he or she filled in the same questionnaire as the one in Day 1 for the purpose of checking his/her changes (or not) in subjective opinions about the system after the four day experience of use.

5.4 Analysed measures

Nielsen (1993) suggested usability of a human-machine interface comprises the following five attributes: learnability, efficiency, memorability, errors and satisfaction. We take up four of these five usability attributes except memorability. These attributes and their inter-associations are examined for each gaze typing interface based on usability indices or aspects derived from data concerning typing speed, errors and typing-related attitudes.

Typing speed is typically measured in terms of words per minute (WPM) or characters per minute (CPM). This aspect for the text written in Japanese may be measured appropriately in CPM while WPM has been more frequently applied to European languages. We may be able to set a conversion factor between WPM and CPM, and we conventionally use a factor of 2 when comparing typing efficiency between CPM for Japanese and WPM for English. Thus, efficiency of an interface can be examined in CPM or WPM performed by experts or skilled users. In other cases, efficiency can be estimated by perfect user simulation or extrapolating a learning model. In this study, we estimate efficiency of the gaze typing interfaces for the Japanese language in extrapolated CPM's after long-enough trials by learning models.

Learnability of an interface can be typically evaluated in terms of a learning factor of task time identified by applying the "power law of practice" model to experimental data or in terms of differential rate of typing speed between two different time points in an earlier stage of practice. In addition to the learning factor and the differential rate, in this study, mean CPM's themselves in several earlier experimental blocks are also compared between the gaze typing systems as a measure connecting to learnability.

As error-related measures, several indices, e.g., over-production rate, rate of backspacing, and minimum string distance (MSD; Soukoreff and MacKenzie, 2003), have been suggested in addition to the rate or frequency of errors per character or unit time. The MSD is a sentence-based error measure, which is calculated as how many key-manipulation steps one needs to obtain a target sentence from a typed sentence (including errors). In our performance data collected from the experiment, almost all the sentences were correctly made with or without correction during gaze typing, and therefore it would be impracticable to apply MSD to our data. The over-production rate is referred to as a rate of the actual number of (gaze) clicks or activation over the optimal (least) number of clicks for a given sentence. This index is in particular useful to examine frequency of mistyping when using a hierarchical menu system like the Japanese version of GazeTalk. However, it is difficult to define the number of click actions in Dasher. The rate of backspacing can be calculated by dividing the total number of the backspace key used (or the total number of characters erased prior to the cursor position) by the total number of typed characters.

It is beneficial to classify error modes of gaze typing during making a text. In this study, two types of errors specific to gaze typing were classified based on human characteristics on visual perception (Aoki et al., 2005), and this classification was applied to calculation of error rates. One is a measure closely relating to the famous, so-called "Midas Touch" problem (Jacob, 1991) when applying the dwell time activation. A Midas Touch error is referred to as incorrect "gaze" activation of a not-intended-to-type key. Such a typing error occurs in fixation at an undesirable key while the user is scanning for a key to be typed or shifting his/her fixation to a target key position. An occurrence rate of this type of error can be computed per character. The other error type focused on in this study is called a premature movement error. This type of error was also first recognised by Jacob (1991). He noticed that it could be difficult for some people to stare at will in order to do a dwell time selection. Naturally, the eyes are moved whenever a piece of information has been noticed and a decision to act has been taken. However, if this is done before the end of the dwell time, the selection is cancelled. We counted the number of such unwanted eye movements by determining a threshold of fixation duration for a valid perception. In this study, the threshold was set at 170 ms. The rate of premature movement error is calculated by dividing the number of eye movements away from a correct key position after the threshold duration (170 ms) but before activation (at 500 ms for most subjects) by the number of characters included in the typed sentence.

Regarding self-reportedly items, we included not only ones relating to the usability attribute, satisfaction, but also ones subjectively perceived about task performance such as speed and errors in a questionnaire. These items were described in a five-point SD (semantic differential) scale – a pair of terms having opposite meanings such as 'very fast' and 'very slow'. The following subjectively rating items were included in the questionnaire: perceived typing speed, perceived likelihood of error, interface preference, satisfaction with system, perceived fatigue and motion sickness.

5.5 Results

5.5.1 Typing speed

Data analysis of the typing speed, i.e., CPM, was performed applying 3-way ANOVA with the typing system (S-GazeTalk, C-GazeTalk or Dasher), tracking accuracy (low or high) and learning effect (Block 1-7) as the independent variables (subjects were treated as repetitions). A result of the ANOVA is shown in Table 2. There was a significant difference in CPM between the three typing systems. A learning effect was also observed as a significant difference between seven blocks. The learning effect for each typing system is depicted in Figure 5 in terms of CPM. The typing speed was increased with blocks for each system. The standard version of GazeTalk exhibited slightly better performance in typing speed than the other two systems, between, which were not significantly different. The typing trials performed in this experiment has been only for several hours during entire seven blocks, and therefore the mean CPM in such a short period may be corresponding to a measure of learnability.

Quantitative estimation of the learning effect of typing speed was made by applying the "power law of practice" model to individual subject's data. In Table 3 are indicated results of parameter estimation of the learning model for all the subjects based on the typing systems. It is seen that the "power law of practice" well fits to any subject using S-GazeTalk or C-GazeTalk while the learning effect can be significantly explained by this model only for two of five Dasher users. The learning factors – which themselves may represent a measure of learnability – are alike and reasonably high (about 1.6 on average) for the two versions of GazeTalk. On the other hand, those of Dasher users for whom the power law of practice was well fit to their learning effects are very high, even compared with the best-learned GazeTalk users.

Factor	s.s.	d.f.	V	F_0
System (A)	84.8	2	42.4	3.491*
Accuracy (B)	8.5	1	8.5	0.700
Block (C)	1000.2	6	166.7	13.731**
A×B	3.7	2	1.9	0.152
A×C	62.0	12	5.2	0.426
B×C	146.9	6	24.5	2.016
A×B×C	219.9	12	18.3	1.509
Error	2039.6	168	12.1	
Total	3565.6	209		

*: $p<0.05$, **: $p<0.01$

Table 2. Result of ANOVA on the character per minute (CPM).

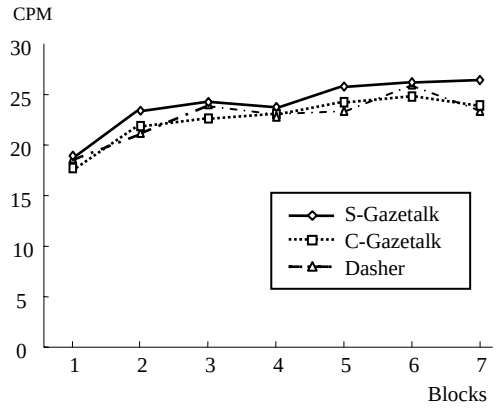

Figure 4. Transitions of CPM with experimental blocks for each typing system

Systems	Subjects	a†	b†	R^2	F_0
S-GazeTalk	S_1	0.047	-0.159	0.761	15.892*
	S_2	0.044	-0.122	0.985	332.857**
	S_3	0.056	-0.121	0.598	7.425*
	S_4	0.047	-0.195	0.892	41.527**
	S_5	0.063	-0.202	0.931	67.966**
Mean of learning factor			-0.160		
C-GazeTalk	S_6	0.052	-0.119	0.720	12.878*
	S_7	0.051	-0.178	0.891	40.971**
	S_8	0.057	-0.149	0.668	10.041*
	S_9	0.052	-0.119	0.644	9.064*
	S_{10}	0.058	-0.229	0.754	15.350*
Mean of learning factor			-0.159		
Dasher	S_{11}	0.079	-0.332	0.884	38.014**
	S_{12}	0.041	-0.066	0.241	1.586
	S_{13}	0.064	-0.289	0.697	11.484*
	S_{14}	0.040	-0.049	0.231	1.504
	S_{15}	0.059	-0.045	0.257	2.079
Mean of learning factor			-0.156		
			-0.310‡		

†$y=ax^b$, where x=blocks practiced (ca. 300 characters/block),
and y=typing time per character = 1/CPM (min./character)
‡mean obtained only from the subjects having significant effect
S-GazeTalk: Standard version of GazeTalk; C-GazeTalk: Centre-text version of GazeTalk

Table 3. Parameter estimation of power law of practice on CPM for each typing system.

The typing speed after a particular amount of trials – which is relating to the usability attribute, efficiency, after a number of trials have been performed – can be estimated by extrapolating the "power law of practice". In Table 4 are shown estimated CPM's of the best learned subject – i.e., one having the greatest learning factor – for each typing system after 7, 10, 20 and 50 blocks are performed (one block includes 300 characters). The CPM of the best Dasher subject is estimated to catch up with that of the best S-GazeTalk user at approximately 34 CPM after 20 blocks of typing performance, i.e., typing 6,000 characters by gaze, which corresponds to nearly a total of three or four hours of typing practice. After this point, typing speed with Dasher is expected to outperform that with S-GazeTalk. The learning model estimates an increased typing speed with Dasher at 46 CPM, which might be equivalent to approximately 23 WPM for European language (as mentioned previously, we used a conventional factor of exchange between Japanese CPM and English WPM: roughly 2), and at 42 CPM with the S-Gazetalk after 50 block trials (i.e., 15,000 character entry). The typing speed estimated with Dasher after 6-7 hour practice is similar to that of an actual one-hour practiced user when combined this system with mouse control (Ward and MacKay, 2002).

	Experiment		Estimated by model			
	n=1	*n*=7	*n*=7	*n*=10	*n*=20	*n*=50
S-Gazetalk	15.57	23.59	23.54	25.29	29.09	35.00
C-Gazetalk	15.37	24.91	26.71	28.98	33.97	41.90
Dasher	12.80	21.04	24.04	27.07	34.07	46.17

n: blocks practiced (each block includes 300 characters);
e.g., *n*=50: a CPM after learning of 15,000 characters typed

Table 4. Estimated CPMs of the best learned user for each typing system at various time points

5.5.2 Typing errors

(1) Over-production rate

As for one of the error-related indices applied in this study, a result of ANOVA for the over-production rate is shown in Table 5. This index may not be appropriate to apply to Dasher, as mentioned in section 5.5, since it is difficult to define the number of clicks with a dynamically and continuously controlled interface. Therefore, the ANOVA tested only two versions of the GazeTalk for the factor of typing systems as well as other two factors, i.e., accuracy levels of eye tracking and blocks. There was a significant difference only between blocks but no significant effects were observed for any other factors. As depicted the over-production rate of each version of GazeTalk in Figure 6, the rate was gradually decreased with blocks. In particular, the learning effect on this index is seen until Block 5, and subsequently the rate seems to become constant.

.

Factor	s.s.	d.f.	V	F_0
System (A)	0.014	1	0.014	0.605
Accuracy (B)	0.013	1	0.013	0.544
Block (C)	0.514	6	0.086	3.685**
A×B	0.027	1	0.027	1.171
A×C	0.177	6	0.030	1.271
B×C	0.215	6	0.036	1.540
A×B×C	0.233	6	0.039	1.672
Error	2.603	112	0.022	
Total	3.800	139		

*: $p<0.05$, **: $p<0.01$

Table 6. Result of ANOVA on the over-production rate.

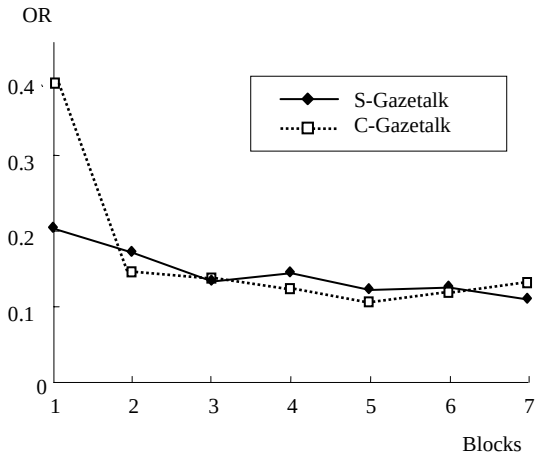

Figure 5. Transitions of over-production rate with experimental blocks for the two versions of GazeTalk

(2) Rate of backspacing

A result of ANOVA for the rate of backspacing with the same three factors is shown in Table 7. For this error-related index, a significant difference was observed only between the typing systems. In particular, as can be seen in Figure 6, the rate of backspacing with Dasher was far higher than GazeTalk. There was no significant difference between the two versions of the latter typing system.

Factor	s.s.	d.f.	V	F_0
System (A)	276.01	2	138.01	15.241**
Accuracy (B)	34.04	1	34.04	3.759
Block (C)	44.72	6	7.45	0.823
A×B	26.35	2	13.17	1.455
A×C	111.53	12	9.29	1.026
B×C	49.16	6	8.19	0.905
A×B×C	57.14	12	4.76	0.526
Error	1521.24	168	9.05	
Total	2120.19	209		

*: $p<0.05$, **: $p<0.01$

Table 7. Result of ANOVA on the frequency of using backspace.

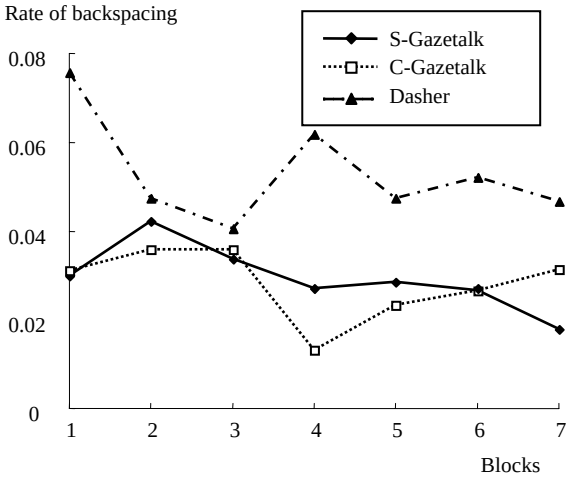

Figure 6. Transitions of the rate of backspacing with experimental blocks for three typing systems

5.5.3 Subjective ratings

Responses of subjective ratings at two different time points of system usage, i.e., at the beginning of the trial (after the first experimental block), and after seven blocks (2,100 character entry), are summarised in Table 8 in terms of percentage agreements of each self-reported item. The percentage agreement is calculated as the rate of agreement strongly or slightly with the left-hand side of an SD scale (i.e., greater than neutral point for the left-hand side term) for the subjects using each typing system. There were only slight differences made by a single subject's response – one response is corresponding to 20% difference of agreement since only five subjects were included in each group – for most self-reported items. In addition, there may be an individual difference in subjective criterion of rating on the items. Therefore, we cannot derive a sound conclusion on subjective satisfaction with each typing interface from these results.

Items		S-Gazetalk	C-Gazetalk	Dasher
Typing speed	after Block 1	20%	40%	20%
(very fast ----- very slow)	after Block 7	20%	20%	40%
Interface preference		80%	20%	80%
(sophisticated ----- difficult)		60%	40%	20%
Error		80%	60%	60%
(unlikely ----- likely to make error)		60%	40%	20%
Satisfaction with system		60%	60%	80%
(very satisfied ----- very dissatisfied)		80%	40%	60%
Fatigue		60%	60%	80%
(very tired ----- not tired at all)		80%	40%	60%
Motion sickness		0%	0%	0%
(felt bad ----- did not feel bad at all)		0%	0%	0%

Table 8. Percentage agreements of subjective ratings on usability issues of typing systems at the beginning and the end of the experiment.

5.6 Discussion and Summary

We conducted usability evaluation of different gaze typing systems, which have two extreme of interface design by applying several indices proposed here or introduced in previous work, which could be useful for the COGAIN project. An interface of one design extreme taken up in this study is GazeTalk, which is a menu-driven system with static, hierarchical menus and has no language model. An example of the other extreme is Dasher, which drives text entry using a language model and by continuous two-dimensional gesture. As a result obtained from the four-day experiment, it is found that learnability of GazeTalk was reasonably high for all the student subjects. The rationale of this result may have been derived from a simple design such as a static, hierarchical menu structure. This design principle may also have contributed to desirable effects on error-related metrics. In contrast, some Dasher users achieved better learnability than any GazeTalk subject although no leaning effects were observed for other Dasher subjects. In addition, Dasher was expected to achieve more "efficient" typing performance after some hour practice.

6 References

Aoki, H., Itoh, K. and Hansen, J.P. (2005) Learning to type Japanese text by gaze interaction in six hours. *Proceedings of the 11th International Conference on Human-Computer Interaction*, Las Vegas, Nevada.

Aoki, H., Itoh, K., Sumitomo, N. and Hansen J.P. (2003) Usability of a Gaze-Interface Compared to Mouse and Head-Tracking Devices in Typing Japanese Texts. *Proceedings of the 15th Triennia lCongres of the International Ergonomics Association, IEA 2003*, Seoul, Korea, August 2003.

Bates, R. and Istance, H.O. (2003) Why are eye mice unpopular? A detailed comparison of head and eye controlled assistive technology pointing devices. *Universal Access in the Information Society* Volume 2, Number 3, October 2003, ISSN: 1615-5289, pp.280–290.

Bevan, N. (1999) Common Industry Format Usability Tests. *Proceedings of UPA'98*, Usability Professionals Association, Scottsdale, Arizona, 29 June – 2 July 1998

Donegan, M., Oosthuizen, L., Bates, R., Daunys, G., Hansen, J.P., Joos, M., Majaranta, P. and Signorile, I. (2005) *D3.1 User requirements report with observations of difficulties users are experiencing*. Communication by Gaze Interaction (COGAIN), IST-2003-511598: Deliverable 3.1. Available at http://www.cogain.org/results/reports/COGAIN-D3.1.pdf

Douglas, S.A., Kirkpatrick, A.E. and MacKenzie, I.S. (1999) Testing Pointing Device Performance and use Assessment with the ISO9241, Part 9 Standard. *Proceedings of CHI'99*, ACM Press.

Hansen, D.W., Hansen, J.P., Nielsen, M., Johansen, A.S. and Stegmann, M.B. (2002) Eye typing using Markov and active appearance models. *IEEE Workshop on Applications on Computer Vision*, pp.132–136.

Hansen, J.P., Hansen, D.W. and Johansen, A.S. (2001) Bringing gaze-based interaction back to basics. *Proceedings of Universal Access in Human-Computer Interaction (UAHCI 2001)*, New Orleans, Louisiana, August.

ISO 9241-9 (2000) ISO 9241-9:2000: Ergonomic requirements for office work with visual display terminals (VDTs), Part 9: Requirements for non-keyboard input devices. International Organisation for Standardisation.

ISO 9241-11 (1998) ISO 9241-11:1998: Ergonomic requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability. International Organisation for Standardisation.

ISO/IEC 9126-1 (2001) ISO/IEC 9126-1:2001: Software engineering – Product quality, Part 1: Quality model. International Organisation for Standardisation.

Jacob, R.K. (1991) The use of eye movements in human-computer interaction techniques: What you look at is what you get. *ACM Transactions on Information Systems*, 9(3), pp.152–169.

Lepistö, A. and Ovaska, S. (2004) Usability evaluation involving participants with cognitive disabilities. *Proceedings of the third Nordic conference on Human-computer interaction (NordiCHI 2004)*, ACM Press, pp.305–308.

MacKenzie, I.S. and Soukoreff, R.W (2003) Phrase sets for evaluating text entry techniques. *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems – CHI 2003*, pp.754–755, ACM Press, 2003.

Nielsen, J. and Molich, R. (1990) Heuristic evaluation of user interfaces. *Proceedings of Human Factors in Computing Systems (CHI 1990)*, ACM Press, pp.249–256.

Nielsen, J. (1993). *Usability engineering*. Academic Press, San Diego, CA.

Nielsen, J. (1994a) Enhancing the explanatory power of usability heuristics. *Proceedings of Human Factors in Computing Systems (CHI 1994)*, ACM Press, pp.152–158.

Nielsen, J. (1994b) Heuristic Evaluation. In Nielsen, J. and Mack, R. L. (Eds.) *Usability Inspection Methods*. New York, NY: John Wiley & Sons, pp. 25–62.

Ovaska S., Aula A. and Majaranta P. (2005) (Eds.) *Käytettävyystutkimuksen menetelmät* [Usability evaluation methods]. Department of Computer Sciences, University of Tampere, Finland. Report B-2005-1, April 2005. 348 pages (in Finnish).

Sears, A., Revis, D., Swatski, J., Crittenden, R. and Shneiderman, B. (1993) Investigating touch screen typing: The effect of keyboard size on typing speed. *Behaviour and Information Technology*, 12, pp.17–22.

Soukoreff, R.W. and MacKenzie, I.S. (2003) Metrics for text entry research: An evaluation of MSD and KSPC, and a new unified error metric. *Proceedings of the Conference on Human Factors in Computing Systems*, 5(1), pp.113–120.

Soukoreff, R.W. and MacKenzie, I.S. (2004) Towards a standard for pointing device evaluation: Perspectives on 27 years of Fitts' law research in HCI. *International Journal of Human-Computer Studies*, 61, pp.751–789.

Ward, D.J. (2001) *Adaptive computer interfaces*. Doctoral Dissertation, University of Cambridge, UK.

Ward, D.J. and MacKay, D.J.C. (2002) Fast hands-free writing by gaze direction. *Nature*, 418, (22nd August 2002), p.838.

Whiteside, J., Bennett, J. and Holtzblatt, K. (1988) Usability Engineering: Our Experience and Evolution. In Helander, M. (Ed.) *Handbook of Human Computer Interaction*. New York: North Holland.

Wobbrock, J.O. and Myers, B.A. (2005) Analyzing the input stream for character-level errors in unconstrained text entry evaluations. Available at http://www-2.cs.cmu.edu/~jrock/.

Appendix A: Accessibility of web documents: overview of concepts and needed standards

Definitions

The concept of accessibility is a measure of the easiness of using something efficiently, for as many types of users as possible. Some recommendations for accessibility focus on people with various disabilities, but in general, all the different types of people, combined with various possible use cases, should be taken into account. Types of users are, for example: novices, experts, seniors, impaired people, non-native speakers, and even software robots. In the case of accessing electronic documents, types of use could be on a desktop computer screen, a video projector, a printer, a mobile electronic device, for urgent professional use, or entertainment; in the dark, or while running to get the bus…

Introduction

Web sites are a domain where accessibility has been studied detail, and where rules and recommendations have been established. That is why it is a good starting point when studying user accessibility more generally.

The year 1994 marks the beginning of standardisation of the Web, with the creation of the W3C (World Wide Web Consortium) [1], which released the first standard recommendation of HTML (the language to design Web pages) [2] in 1995. Since then, standardisation efforts have continued, and been influenced by the evolutions of practices (more users, new communication habits, etc.) and of the technology, both hardware (new screens, computers, input systems, mobile devices, networks, etc.) and software (Web browsers, multimedia, etc.).

It is only when the first technical standards were in place, and began to spread, that the concept of "accessibility" developed. The first step in improving accessibility was the result of a W3C publication in December 1996, aimed to separate text content from the layout structure, with CSS (Cascading Style Sheets) [3]. The real recognition of accessibility needs came in February 1997, when the W3C launched the WAI (Web Accessibility Initiative) [4] that published its first recommendations in 1999 [13] and which is partly funded by the European Commission's Information Society Technologies Programme [14].

Web accessibility factors

On the Web, the accessibility addresses several issues including specific needs of users, electronic devices, and software robots.

Users

Web accessibility is mainly aimed to ensure that people with various disabilities will be able to access the information. Making the content easier to browse and comprehend is also of a great help for people with cognitive limitations, for them that are not good readers, or not fluent in the needed language, and even for users under stressing or time critical circumstances. Not all the users have the same skills, also, offering the possibility of customising the interface, and tools like search facilities, is a plus.

Devices

Not all the browsing devices have the same capabilities, like bandwidth, processing power, human-computer input and output facilities. With new mobile devices becoming more and more popular and diverse, one should be aware of that. Furthermore, not all the client devices have the same goal.

- Most of them try to render on screen exactly what the Webmaster expects, and even that is not always perfect, as shown by the Acid2 browser test [8].

- Other devices try to modify Web documents, in order to adapt them to specific users or situations. For instance, fitting documents designed for a large screen on a small PDA screen. Similarly, some devices read web pages using speech synthesis or render them in Braille for fingertip reading. All these devices work better when documents are standard compliant and follow some basic accessibility rules. Having common open standards is therefore crucial.

Software

Finally, accessibility rules are also beneficial to software and robots. Software and hardware are evolving rapidly, and following accessibility rules improves data perennially. Software robots, like search engines need to access web pages. For them as well as for devices transforming the content before displaying it to the user, accessibility rules are very important. It is easier to automatically extract the semantics, like the title and subtitles, navigation paths, topics, etc. when the pages are standard compliant. Interestingly enough, many webmasters have integrated accessibility rules to improve their ranking and quality of indexation in search engines. Overall, the same accessibility rules are solving a number of problems.

Recommendations and guidelines

When designing a new Web site, as well as when evaluating it, there is an order, which is best to follow. First, one should consider the architecture of the Web site, then validate each part of its structure against the current standards in force, finally use the accessibilities guidelines by order of priority.

Architecture principles

Separation text-layout: one of the first W3C's achievements was to propose a good solution to separate text content from the layout structure. This is important because managing the content and the layout involves different technologies, which can be better validated when separated. It also offers the possibility to have various layouts, for the same content, targeted at different devices or situations, or simply to change the design of a Web site rapidly. Furthermore, a separated description of the layout can be reused on many pages, saving bandwidth and ensuring some layout consistency. Last but not the least, this separation principle is normally associated with a better semantic, good for accessibility and vital for some software (cf. §3.3). The main standard allowing this separation of content and layout is CSS (Cascading Style Sheets) [3, §4.2.3].

International character set: There are in the world many alphabets and other signs. Computers can only deal with numbers, and this implies an encoding convention; that is to say, a conversion table between characters and numbers. Many different incompatible character sets have been developed: only in the European Union, twenty official languages (2005) are using five major encodings, in the ISO-8859-X family [25]. Luckily, since 1991, the Unicode [7] standard is the recommended unification solution; it can deal with most of the needed characters in the World (already more than 96000 characters in version 4) and is widely supported.

Persistent Web addresses: Web addresses should be made in such a scalable way that it will not be needed to rename them when the Web site evolves [17]. If original addresses were badly chosen and need to be modified or removed, that should be done with proper HTTP mechanisms [18, §4.2.1] (like HTTP 301 Moved Permanently [19]) so the user is directed instantly to the new address, and links have a chance to be quickly

and automatically updated (like search engines). Web server log files should be analysed to track broken links.

Meaningful Web addresses: Furthermore, short meaningful human-readable addresses are appreciated. For instance, http://www.acme.com/cgi/index.cgi?st=1526&dp=31 is less explicit and often longer than http://www.acme.com/biology/staff/james-brown/, which can easily be read as "James Brown, member of biology staff of Acme company". This approach is also beneficial for the quality of search engines indexation. The character set should be Unicode UTF-8 [22]; this is important if there are other characters than ASCII [20]. In any case, the address is always encoded with the URI %HH escaping mechanism [23]. Addresses are an important factor of trust, and the name of the site (DNS, Domain Name Server) especially (www.acme.com in the current example). The extension (.org .com .eu) should also be carefully chosen [20]. It is recommended that the main Web site can be reached both from http://acme.com and the traditional http://www.acme.com.

Metadata: Searching the Web and finding relevant information is not always easy. This process should be helped, by giving documents pertinent addresses, titles and subtitles, but also by including extra information, "metadata", like keywords, authors, classification, etc. In addition to the basic metadata defined in the HTML specification, there are some attempts of standardisation of broader metadata, like the Dublin Core Metadata Initiative [27]. Metadata format should be checked, manually or automatically, against a standard that is better chosen at the early stages of a new Web site.

Semantics: RSS, RDF

Standard protocols and formats

Communication protocol: The lowest layer of interest in this document is HTTP (Hypertext Transfer Protocol) [18], which is codifying the dialog between the client device and the Web server, and which is not always well known by webmasters. Normal static documents are normally handled correctly by the Web server, if set up correctly, but one should pay more attention to dynamic Web pages, where a part of this work is transferred to the webmaster, who does not always implement advanced HTTP concepts like date of last modification, negotiation and caching, which are especially useful for devices with limited bandwidth, and various software tools. Among other things, HTTP is also used to convey information about the type of document (MIME type [32]), character coding (cf. §4.1.2) and for addresses redirection (cf. §4.1.3). Those are points to check with any HTTP header analysis tool [33].

Document content: HTML (HyperText Markup Language) [2] is the standard to use when publishing text documents on the Web. HTML is a text-based markup language, like this: this is an important example . The HTML source code is human-readable. Even if the final user usually only sees the rendering of the HTML page in his Web browser, it is possible to fall back to the source code (especially with basic IT skills becoming more common); it is therefore a plus to provide clean, readable source code. HTML offers a variety of tags, used to structure the text. The Webmaster should pay attention to use the most appropriate ones, avoiding generic ones, to clearly identify titles, subtitles, lists, definitions, etc. A rough standalone HTML page using only the default presentation rules (no CSS, cf. §4.2.3) should have an acceptable display. Since 1995, several HTML versions have been released. Some of them have become obsolete, but others are targeted to various uses. Today, "XHTML 1.0 Strict" should be chosen in most cases, and ensure that many essential accessibility points are de facto covered. HTML pages must be checked against a validation tool, such as the W3C Markup Validation Service [26], and tested in several Web browsers, in 'standard' mode (not 'quirks') [31].

Presentation layout: the layout instructions should be separated from the HTML body of text. This offers the possibility to display the rough Web document with any extra design information. The rough document should be easy to read and to navigate, and most of the functionalities should be available. CSS (Cascading Style Sheets) [3]. The user can override some of the CSS rules, like colour and shape of links, background, font, etc. This is mainly aimed to be used by users with specific requirements, and webmasters should

therefore not rely on the design to make available the various functionalities of the Web site. CSS instructions must be checked against a validation tool, such as the W3C CSS Validation Service [28], and tested in several Web browsers.

Dynamic pages: Dynamic behaviours, animations, form validation, etc. can be done on Web pages with a client side programming language: ECMAScript, the standardised version of JavaScript [29] (nothing to do with Java). Whenever possible, for design purposes, CSS should be used instead of JavaScript. Like for the layout, it is better to keep JavaScript code separated from the text, and Web pages should be functional with JavaScript disabled. The webmaster should be careful not to break the HTML structure with JavaScript, so that the rendering mode can still be standard, and do not need to fall back to quirks mode [31]. During development and validation of JavaScript code, proper debugging tools should be used, like Mozilla's JavaScript console and debugger [30]. Here again, tests should be made in various Web browsers, with all the warnings activated.

Multimedia content: Text content is not all, and multimedia takes a big part on Internet. Discussing accessibility and standards for multimedia contents is too large to be reported here in details. However, the basic notions can be introduced. Other good practices are listed in the accessibility guidelines (cf. §4.3), like providing an alternate text, a title and a description for pictures. Markup languages should be used when possible (MathML, SVG), instead, or in addition, of other graphic-based solutions. Standard and/or open formats should be used whenever possible, instead of proprietary formats; it should be available on many systems, with minimal fees for the client. For images, standard PNG [34] should be used for synthetic images (graphics, line drawing, text, etc.), lossless compression, or advanced transparency needs. JPEG/JFIF [35] can be used for photographic pictures, with loss compression. Video formats are less established; the Moving Picture Experts Group (ISO/IEC) [36] proposes some formats suitable for Internet, like MPEG-1 (widely supported, high compression, for small videos) or MPEG-4 (better quality/size ratio, for longer and larger videos). MPEG standards are not free of charge, in countries where software patents apply. XviD [37] is an Open Source free implementation of MPEG-4. Similarly, for sounds on Internet, the most popular format proposed by MPEG is the MP3 (MPEG-1 Audio Layer 3); and the Open Source community often uses the free "Ogg Vorbis" format [38].

Plug-ins and other formats: Many other formats are competing on Internet, under the form of various plug-ins. Each format has its own accessibility features and recommendations. Some of those formats are open and royalty-free, like PDF [39] that can be used in addition to HTML pages, to provide complex and precise printable versions, not achievable with CSS. Proprietary formats and plug-ins should be restricted to a minimal usage, for very specific use. Among other problems, installing a plug-in from a third party is a potential safety breach. When requiring a plug-in, a link to an official place to download the required software should be provided, which should contain a policy declaration about privacy, advertisement intrusions, licence, legal aspects, etc.

Accessibility guidelines

It is only when the architecture and the standard formats used by the Web site have been verified, that it is possible to really concentrate on accessibility rules. Verification with automatic accessibility tools cannot be accurate if the documents are not valid.

Localisation issues: Being ready for internationalisation is always important, even if no translation to other languages is expected. Unicode is a first step, and following internationalisation authoring techniques [24] a second, but localisation is broader than those technical issues. As an example, the date format issue: in an English text, 07/06/05 can be 7th of June 2005, but also 6th of July 2005 (USA style); the ISO-8601 standard proposes 2005-06-07, among other unambiguous solutions. Time formats and calendars, number and currency formatting, alphabetical sorting, etc. are related issues. Addressing those issues is important, to be well understood.

Navigation: Browsing a large Web site is sometimes not trivial, and providing and effective alternative versions, current version. List of priorities, in three levels, from A-level (minimal accessibility) to AAA-level (high accessibility). Logo AAA, Browser, Multimodal interaction. Accessibility audits are, of course, more impartial and accurate when made by an independent team, different from the one, which builds the Web site. General accessibility rules: http://webxact.watchfire.com Recommended colours, colour-blindness accessibility test: http://www.vischeck.com/vischeck/vischeckURL.php, http://colorfilter.wickline.org.

The law

In order to give accessibility recommendations more impact, several countries have issued some resolutions or even some laws to ensure that those good practices can be widely adopted. [41]

USA: US Public Law PL 105-220 of August 7, 1998. Section 508 Amendment to the Rehabilitation Act of 1973 [42]. The WAI recommendations were considered when writing this amendment. It is an effective enforceable law, with financial penalties when transgressed.

Europe: Parliament Resolution P5_TAPROV(2002)0325 on the Commission communication eEurope 2002: Accessibility of Public Web Sites and their Content. [43] (see [44] for current practices at the European commission)

France: Loi n°2005-102 du 11 Février 2005 pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées, Article 47 [45]. "International Internet accessibility rules must be applied to online public communication services".

Limitations

While it is often easy to make Web content accessible, there are some limitations. Here are some of them.

Arts and entertainment

Electronic content that aim to be artistic, funny or challenging is often more difficult to make accessible. In this case, it is even possible to argue that less-accessible content is superior. Challenge is indeed an important factor when designing games that will keep people interested for a long time, and that can also be the case for some Web sites. When those circumstances occur, it is helpful to provide a separate accessible document describing the content. Similarly, literature content might suffer if the author has to stick to basic words and expressions. As an example, the free collaborative encyclopaedia Wikipedia [15] contains a subset of articles in "simple English" [16], even in the "normal English" articles try to be understandable by most English speakers.

Turing anti-robot tests

On the Internet, it is often needed to automatically "tell computers and humans apart" (Captcha, [9]) in order to prevent software robots from accessing some areas of a Web site. In this case, some captcha tests are used that "most humans can pass, and Current computer programs can't pass". They are often based on some difficult text to read, or idea association between photos. They are mainly visual and it is difficult to make them accessible, without making them too easy for a computer. For those tests, accessibility can be increased by providing at least another type of test, (e.g. one visual, one with sound). Other approaches are possible but more cumbersome, with various biometrics or electronic identification like credit cards. [10]

Device independence ongoing work

Accessibility rules help in having one good and accessible version of a Web document, and while this approach is important, it has some limitations. Therefore, in addition to the accessibility rules, there is some ongoing work to facilitate the dialog between the user devices and the server of documents, to provide more personalised versions of the documents. The Device Independence [5] working group is in charge of this topic at the W3C, and has issued recommendations like CC/PP (Composite Capabilities/Preference Profiles) [6], but those are not widely used, at this date.

Summary

Web accessibility is a large topic, and one cannot ask each webmaster to follow all the standards and accessibility rules, but having a clear common target is valuable. As seen in this paper, Web accessibility is partly promoted and rooted in communities of disabled users, but now targets and benefits all users.

References

[1] W3C, World Wide Web Consortium, http://www.w3.org

[2] HTML, HyperText Markup Language, http://www.w3.org/MarkUp/

[3] CSS, Cascading Style Sheets, http://www.w3.org/Style/CSS/

[4] WAI, Web Accessibility Initiative, http://www.w3.org/WAI/

[5] Device Independence, Access to a Unified Web from Any Device in Any Context by Anyone, http://www.w3.org/2001/di/

[6] CC/PP, Composite Capabilities/Preference Profiles, http://www.w3.org/Mobile/CCPP/

[7] Unicode, ISO/IEC 10646, http://www.unicode.org

[8] Acid2 browser test, http://www.webstandards.org/act/acid2/

[9] Captcha, Completely automated public Turing test to tell computers and humans apart, http://www.captcha.net

[10] Inaccessibility of Visually-Oriented Anti-Robot Tests, http://www.w3.org/TR/turingtest/

[11] Wikipedia articles about accessibility, http://en.wikipedia.org/wiki/Category:Accessibility

[12] Dive Into Accessibility, http://diveintoaccessibility.org

[13] Web Content Accessibility Guidelines 1.0, http://www.w3.org/TR/WCAG10/

[14] Europe's Information Society, http://europa.eu.int/information_society/, http://europa.eu.int/information_society/policy/accessibility/

[15] Wikipedia, the free encyclopaedia, http://en.wikipedia.org/wiki/Wikipedia

[16] Wikipedia, simple English, http://simple.wikipedia.org/wiki/Wikipedia:Simple_English_Wikipedia

[17] W3C Style, Cool URIs don't change, http://www.w3.org/Provider/Style/URI.html

[18] HTTP/1.1, Hypertext Transfer Protocol, http://www.w3.org/Protocols/rfc2616/rfc2616.html

[19] HTTP 301 Moved Permanently, http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.3.2

[20] Domain Name System Structure and Delegation, http://www.ietf.org/rfc/rfc1591.txt

[21] ASCII (American Standard Code for Information Interchange) format for Network Interchange, RFC 20, http://www.ietf.org/rfc/rfc20.txt

[22] UTF-8, 8-bit Unicode Transformation Format, RFC 3629, http://www.ietf.org/rfc/rfc3629.txt

[23] Uniform Resource Identifiers (URI): Generic Syntax, RFC 2396, http://www.ietf.org/rfc/rfc2396.txt

[24] Authoring Techniques for XHTML & HTML Internationalization, http://www.w3.org/International/geo/html-tech/outline/html-authoring-outline.html

[25] ISO 8859 character set family, http://en.wikipedia.org/wiki/ISO_8859

[26] W3C Markup Validation Service, http://validator.w3.org

[27] Expressing Dublin Core in HTML/XHTML meta and link elements, http://dublincore.org/documents/dcq-html/

[28] W3C CSS Validation Service, http://jigsaw.w3.org/css-validator/

[29] ECMAScript, http://www.ecma-international.org/publications/standards/Ecma-262.htm

[30] Venkman, Mozilla's JavaScript Debugger, http://www.mozilla.org/projects/venkman/

[31] 'Standards' versus 'Quirks' modes, http://www.w3.org/International/articles/serving-xhtml/#quirks

[32] XHTML Media Types, http://www.w3.org/TR/xhtml-media-types/

[33] Mozilla LiveHTTPHeaders, http://livehttpheaders.mozdev.org

[34] PNG, Portable Network Graphics, http://www.w3.org/Graphics/PNG/

[35] JPEG, Joint Photographic Experts Group, ISO/IEC IS 10918-1 | ITU-T T.81, http://www.jpeg.org/jpeg/ , http://www.w3.org/Graphics/JPEG/

[36] MPEG, Moving Picture Experts Group, http://www.chiariglione.org/mpeg/

[37] XviD, GNU GPL license, ISO MPEG-4 compliant video codec, http://www.xvid.org

[38] Ogg, http://www.ietf.org/rfc/rfc3533.txt ; and Vorbis, http://www.vorbis.com

[39] PDF, Adobe Portable Document Format, ISBN 0201758393, http://partners.adobe.com/public/developer/pdf/index_reference.html

[40] Designing more usable web sites (Trace Center, Collage of Engineering, University of Wisconsin-Madison), http://trace.wisc.edu/world/web/

[41] International Policies on Accessibility, http://www.w3.org/WAI/Policy/

[42] Section 508, the United States legislation, http://www.section508.gov

[43] EU legislation, http://europa.eu.int/information_society/topics/citizens/accessibility/web/wai_2002/ep_res_web_wai_2002/index_en.htm

[44] EC's accessibility statement, http://www.cordis.lu/ist/accessibility-statement.htm

[45] France legislation, http://www.legifrance.gouv.fr/WAspad/UnTexteDeJorf?numjo=SANX0300217L

Appendix B: Common tools for communicating evaluation results

Name of system
Description XXX is a system that uses the pupil/corneal reflection method to measure eye point of gaze in real time.
Procedure A remote eye-tracking device records eye movements using selective mirror. Gaze position can be transferred to other systems in real time.
Advantages XXX is a non-intrusive and accurate system and can be configured with other systems for wider eye gaze interaction
Disadvantages Expensive, difficult to calibrate, complex documentation
Potential use for eye gaze interaction The system's ability to integrate into other systems may make it suitable for eye gaze interaction
Potential users People suffering from…
Potential use situation Typical use situations include…
Other issues Issues, more issues, and even more issues…

Appendix C: European standards

In addition to the methods and measures presented in this deliverable, we also need to how the systems apply to European standards and to the general accepted guidelines promoted by user organisations and other institutions in the field like, for example, charity organisations. Below is a first list COGAIN evaluation related European standards. We will probably not have to deal with any one of them. But least we need to know to some degree how the different standards are intertwined with the concept of usability and to what extent they are important to our area.

EN 12182	Technical aids for disabled persons - General requirements and test methods
EN 614-1	Safety of Machinery, Ergonomic design principles. Part 1: Terminology and general principles
EN 894-3	Safety of machinery - Ergonomics requirements for the design of displays and actuators - Part 3: Control actuators.
EN 563	Safety of machinery - Temperatures of touchable surfaces - Ergonomics data to establish temperature limit values for hot surfaces
EN 60068-2-32	Basic environmental testing procedures: Part 2 Tests: Tests Ed. Free Fall.
EN 60601-1	Medical electrical equipment: Part 1: General requirements for safety.
ISO 11200	Acoustics - Noise emitted by machinery and equipment - Guidelines for the use of basic standards for the determination of emission sound pressure levels at a work station and at other specified positions
ISO 11201	Acoustics - Noise emitted by machinery and equipment - Guidelines for the use of basic standards for the determination of emission sound pressure levels.