Human–Computer Interaction — INTERACT'99
Angela Sasse and Chris Johnson (Editors)
Published by IOS Press, © IFIP TC.13, 1999

Evaluating Gedrics: Usability of a Pen-centric Interface

Jörg Geißler[1], Michele Gauler[2] & Norbert A Streitz[3]

[1] Plenum Systeme GmbH, Hagenauer Straße 53, 65203 Wiesbaden, Germany.
[2] Johannes Gutenberg University, Institute of Psychology, Staudingerweg 9, 55099 Mainz, Germany.
[3] GMD-IPSI, Dolivostraße 15, 64293 Darmstadt, Germany.

Joerg.Geissler@plenum.de, gaulm000@mail.uni-mainz.de, Norbert.Streitz@gmd.de

Abstract: Many users of today's pen computers have an ambiguous attitude towards these devices. On the one hand, they like the ease of use, especially in the beginning. On the other hand, after some time, they often feel hampered by the systems since the user interfaces do not reflect the users' individual skills, experiences, and preferences. Pen interfaces treat all users in the same way — like novices. Becoming an expert or 'power' user is quite difficult. In this paper, we report on the *gedric* approach (Geißler, 1995) to this problem and evaluate an application with a so-called *pen-centric user interface*(Geißler, to appear). We will show that such an interface efficiently supports experienced as well as novice users. By having the freedom to choose from two popular interaction styles — menus and gestures — and to mix them arbitrarily, gedrics support a wide range of user preferences and skills. This results not only in efficient individual working styles but also in a high user satisfaction.

Keywords: pen computing, gestures, menu interaction, gedrics, usability, study evaluation.

1 Introduction

A widely used statement in the research field of pen computing (Meyer, 1995) says that the pen is mightier than the mouse. In addition to pointing, people are able to sketch, annotate, gesture, and even write with an electronic stylus as they can do with pen and paper (Wolf et al., 1989). Unfortunately, today's user interfaces for pen computers — so-called pen-based interfaces — only use a fraction of the capability of the pen. A small number of gestures is used for simple editing tasks but most of the interface is still based on traditional concepts like push buttons, pull-down menus or other point-and-click components. Strictly speaking, in these interfaces, the pen primarily simulates the mouse and just offers some neat add-ons in terms of gestures. Pen interfaces seem to ignore that pen interaction primarily is stroke-based, and not tap-based (Geißler, 1995).

In the long term, those 'gesture-enhanced point-and-click interfaces' obstruct users who have become experienced or even experts because there are no short cuts for all the functions that are available. Users are forced to continue working with inappropriate interface components — see also (Grudin, 1989). Current pen interfaces may be attractive for inexperienced users, but they do not support their learning process and finally they still treat skilled users like novices.

Extending the influence of gestures on the overall interface design is not that easy. Most of the time, studies about the usability of pen gestures restricted the number of gestures deliberately to ten or even less (Geißler, to appear). Gestures are only useful if they are easy to remember and this depends on their 'naturalness' or familiarity to users. For this reason, only very specific tasks have been observed. Evaluators admitted that the usability of gestures in other, more complex tasks or even in applications that are completely based on gestures, is not yet clear and require further observations (Wolf, 1992).

An attempt to overcome these shortcomings is a user interface component that combines traditional interface concepts with pen gestures. Therefore, we developed the *gedric* (Geißler, 1995). It is part of a so-called *pen-centric interface* (Geißler, to appear) that also covers aspects like ubiquitous annotations, handling user input tolerantly, and more. Gedrics provide an efficient way of interacting with a pen system while engendering a high level of user satisfaction.

In this paper, we briefly describe the concept of gedrics and present an application that makes use of gedrics. Then, we report on an evaluation of this application as well as its results. After a discussion, we will conclude and provide an outlook onto our future work.

2 Gedrics

A gedric (Geißler, 1995) is a *ge*sture-*dr*iven *ic*on, i.e. it is an interface component that has an iconic form and on that users draw strokes with a pen. These strokes are interpreted by the gedric and transformed into commands. Gedrics belong to the functional parts of a pen interface, like tool bars or pull-down menus in mouse interfaces.

A gedric bundles many functions of an application in a small display area. Users can access this functionality not only directly by gestures but also in a way similar to pop-up menus: by tapping on them, a menu appears on top of the gedric and users can select one of its items. Each function of the gedric is displayed in the menu next to a visualization of the corresponding gesture. In addition, the shape of a gesture always corresponds to the gedric's image, making the gesture more intuitive to users. The examples mentioned in the following section will show this clearly.

Gedrics enable developers of pen-based software to create complex applications with a simple and clear interface instead of using cluttered tool bars that provide one button for each function. Moreover, several types of users are supported by gedrics. Inexperienced users may first interact via pop-up menus. Later on, they will use gestures as short cuts to application functionality. With gedrics, it is possible to become a 'power user' of a pen application by incremental learning. Most of the time, any user is experienced in a certain subset of the application's functionality and a novice with respect to the rest of the functions. By providing two interactions styles — menu-based interaction as well as gesture interaction — gedrics support those partial experts.

At first glance, gedrics seem to be similar to the concept of pie menus (Kurtenbach & Buxton, 1991; Kurtenbach & Buxton, 1993). But in contrast to this interface element, gedrics can provide free-form input shapes as gestures and are not restricted to straight lines (Geißler, to appear). Furthermore, pie-menu gestures are only based on the position of the corresponding menu items whereas gedric gestures are semantically related to the gedric image. This supports users to remember them.

3 Sample Application

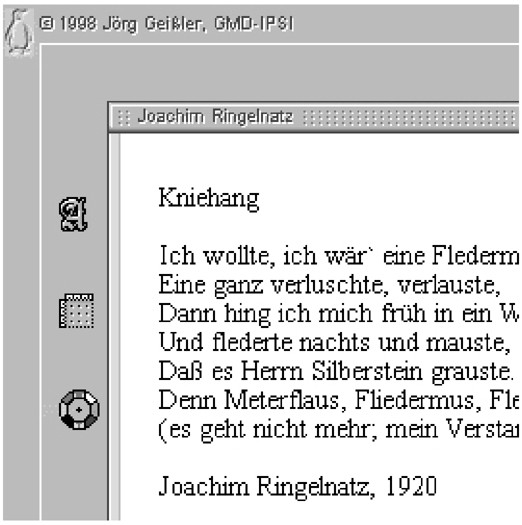

Figure 1: Cropped screen-dump of the sample application.

In order to demonstrate the power of gedrics, a sample application has been developed. This application makes use of several gedrics and hereby illustrates the main aspects of the gedric concept. In order to reduce the complexity of the following evaluation, only three stroke-based gedrics with limited functionality were actually used (Figure 1). The full power of the application is described in (Geißler, to appear).

The purpose of the sample application is to enable users to format an existing text by changing font attributes, like style, size, and colour, as well as aligning passages of the text. This functionality is accessible through the gedrics called TYPOGRAPHER, LAYOUTER, and PAINTER.

First, users select a portion of the text by underlining phrases or by marking lines or paragraphs at their sides. Then, they use the gedric to perform a format operation on that selection. This working style corresponds to traditional direct-manipulation interfaces. However, since each word and even each character shown in the window is a component of the pen-centric interface, i.e. developed in a gesture-sensitive way — similar to gedrics, the application also supports direct content-oriented manipulation of text, e.g. insertion and deletion of phrases. As mentioned above, this functionality was not available in the evaluation.

In the following sections, we describe the three gedrics and the functions used in the evaluation in more detail.

3.1 Font Manipulation: The TYPOGRAPHER

The TYPOGRAPHER controls the font attributes of selected text. Users change those attributes by drawing one of the TYPOGRAPHER'S font-changing strokes on top of it. In the evaluation, only those gestures were available that change the size and the style of the font (Figure 2). Setting the font family by writing the name onto the gedric as well as combinations of strokes were not available. As a reaction to the input strokes, the gedric modifies the selected text accordingly. The icon of the TYPOGRAPHER shows the letter 'A' in an ornate style, indicating the typographic task of the gedric.

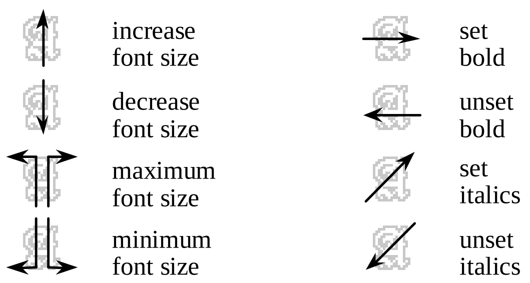

Figure 2: Gestures of the TYPOGRAPHER.

Font Size The font size is changed by drawing vertical strokes onto the TYPOGRAPHER. Strokes that go straight up increase the font size, those going straight down decrease it. Thus, these pen movements correspond to the application feedback, since it makes the characters larger (pen up) or smaller (pen down).

All in all, the application supports five different font sizes. An additional pair of gestures has been added to increase the efficiency of work: two L-like strokes, directed mainly up or down. As used for other components of the pen-centric interface (Geißler, 1998), these L-like strokes indicate functions that maximize (pen up) or minimize (pen down) an attribute. In this case, they set the maximum and minimum font size that is available for the selected text.

In the evaluation, users were not able to write absolute size values onto the TYPOGRAPHER, which is normally possible.

Font Style Changing the font style is done by drawing horizontal or diagonal strokes onto the TYPOGRAPHER. A horizontal stroke from left to right sets the bold attribute, indicating that the modified text will require more horizontal space to the right. The same stroke but with the opposite direction — from right to left — is used for unsetting this attribute. The italics attribute is set by diagonal strokes. A stroke from the lower left to the upper right is used for setting this attribute, indicating that the resulting text will be tilted like the input stroke is. Again, the opposite direction will unset this attribute. Gestures for underlining, for additional styles as well as combinations of strokes were not available in the evaluation.

3.2 Alignment of Lines and Paragraphs: The LAYOUTER

The second gedric used in the sample application is the LAYOUTER. Originally designed for any layout-related task, it is used in this context simply as a tool for aligning text lines and paragraphs. All major alignment operations are supported by the LAYOUTER (Figure 3). Its icon shows a grid with two rulers, one at the top and one at the left.

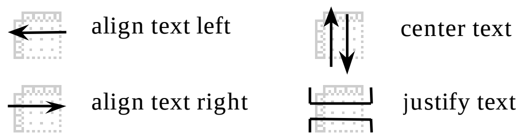

Figure 3: Gestures of the LAYOUTER.

The most simple alignment operation is achieved by straight horizontal strokes onto the LAYOUTER. A stroke to the left results in left alignment of the selected text portion, a stroke to the right aligns the text right. Vertical strokes, either straight up or down, let users centre lines or paragraphs. Bracket-like input will justify it. Again, the pen movement corresponds with the following application feedback.

3.3 Colour Settings: The PAINTER

Similar to the LAYOUTER, the functionality of the PAINTER was specified for a broader application scenario and was limited for the evaluation. Normally, any colour-related aspect of the interface is handled by this gedric. With respect to the experimental setup, the PAINTER was used only to colour selected text portions.

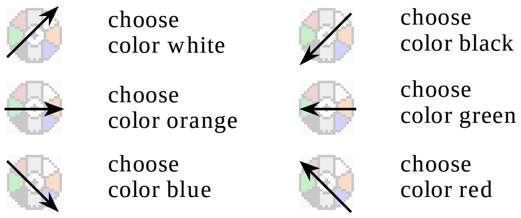

Figure 4: Gestures of the PAINTER.

The icon of the PAINTER shows a colour wheel from which users can choose one of six pre-defined colours. This is done by drawing strokes from the middle of the gedric to the direction the colour is available on the colour wheel (Figure 4). e.g. to colour a selection in red, which is available in the upper left corner of the gedric, users draw a diagonal stroke from the lower right to the upper left onto the gedric. Functions for additional colours, for setting the six colours, and for changing the brightness of colours were not available in the evaluation.

4 Experiment

In this section, we describe an experiment we carried out to investigate the usability of the gedric concept.

4.1 Questions and Hypotheses

A usable interface is "easy to learn", "efficient to use", "easy to remember", "pleasant to use", and has "few errors"(Nielsen, 1993). An empirical user study should show whether gedrics have these qualities or not.

The two interaction styles An important attribute of a gedric is that it offers two possible interaction styles. On the one hand, the gedric can be compared with a traditional icon that has a pop-up menu. By tapping on it, the menu opens and a command can be selected (also by tapping on the menu item). On the other hand, the gedric command can be executed by using the corresponding gesture. We assumed that offering these two interaction alternatives for all commands would increase the users satisfaction (Shneiderman, 1997) with the interface. Users are not forced to use gestures if they find them difficult to learn or to perform. We were interested in finding out which interaction style the users would prefer, if they would show interest in using the gestures or if they would stick to the pop-up menus.

Gesture interaction Using a gesture on a gedric is a short cut for the corresponding command. It speeds up the interaction with the gedrics because it requires fewer interaction steps. But, of course, for most users, the gesture interaction represents a novel form of interaction and the gestures have to be learned first to have the effect of speeding up. Therefore, we were especially interested in the usability of the gestures in comparison to the usage of the pop-up menus. We supposed that for novices the gesture interaction would be more difficult than using the pop-up menus.

The underlying concepts of the gedrics and gestures As described above, each gedric is a combination of several carefully designed aspects:

- The chosen icon of a gedric should match with its functionality.
- The commands of each gedric should correspond to its functionality.
- The order of the commands in the pop-up menu and the association of the gestures to the commands should be logical.

For example, the gesture which changes text into italics indicates the sloping position of italics. This logical design should facilitate the interaction with the gedrics and accelerate the learning process.

We supposed that the logical design of the gedrics would especially lead to less confusion between the different gedrics. During gesture interaction, one gesture can release different commands dependent on which gedric it is performed on. For instance, a stroke from the left to the right performed on the TYPOGRAPHER sets the bold font style. The same gesture performed on the LAYOUTER aligns objects to the right. With this gesture on the PAINTER the colour orange is chosen. An elaborated introduction to the logic of the functionality underlying each gedric and the classing of gestures with commands should lead to fewer confusions with respect to the multiple usage of the same gestures.

In contrast to similar implementations, like 'marking menus' (Kurtenbach & Buxton, 1991; Kurtenbach & Buxton, 1993), where the direction of the gesture has no logical association with the semantic of the corresponding command, we believe that these underlying concepts of the gedrics and their gestures make it easier to learn the gestures. We supposed that an elaborated introduction into the concepts would increase usability.

4.2 Method

Experimental Design We used a between-subject design in order to investigate the usability of the two interaction styles and the need of a specific introduction to the gedric concept.

The two independent variables are: *interaction style* with three values and *introduction* with two values. Subjects were assigned randomly to the experimental conditions.

The three values of the *interaction style* variable are a *combined condition*, and two other conditions with only one interaction style serving as control conditions:

- In the *combined condition* (C), the gedrics were presented with their full functionality, i.e. interaction with pop-up menus and via gestures. It was explained that both interaction styles are equivalent and can be used according to personal preferences.
- In the *gesture condition* (G), subjects could only interact with the gedrics by using gestures. The pop-up menus were used only as passive help menus showing the list of commands and their associated gestures.
- In the *menu condition* (M), the gedrics could only be used by opening the pop-up menus and by tapping on the commands in the menus. The gedrics in this condition are equivalent to traditional icons with pop-up menus.

The *introduction* variable had two values:

- Subjects in the *elaborated introduction condition* were introduced to above described underlying concepts of each gedric.
- In the *simple introduction condition* all gedrics, commands, and gestures were explained without reference to underlying concepts.

Subjects had to work on the same task twice resulting in a 3×2×2 experimental design with 'task' as the repeated measurement factor.

Setting The experiment took place in July and August of 1998 in the AMBIENTE Lab at GMD-IPSI. In this room, so called *roomware* components (Streitz & Geißler, 1998; Streitz et al., 1999) can be found. We used one of these components, the DynaWall (Geißler, 1998), consisting of three segments, each realized by an interactive rear-projection whiteboard. A pen was used for interacting with the DynaWall. We used the DynaWall because large interactive displays represent one of the typical touch-sensitive devices. Furthermore, the DynaWall provided us with excellent conditions of observing the users' performance.

The sample application was running on the middle segment while the right segment showed the instructions in parallel. The experimenter was in the room. All actions of the subjects were recorded by a video camera. Figure 5 shows the setup as recorded by the camera.

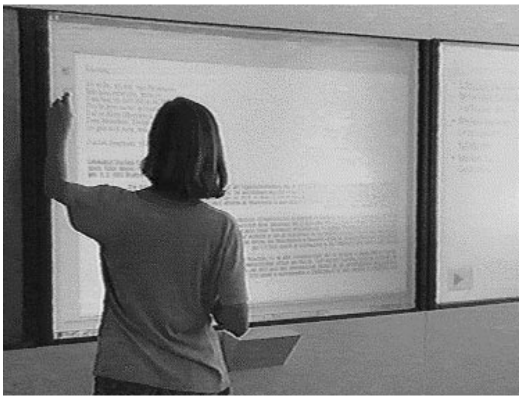

Figure 5: Still picture from the video recordings showing a subject working with the gedrics at the DynaWall.

Subjects A total of 54 subjects participated in the experiment, 27 men and 27 women. Their age varied between 17 and 31. Most subjects were students and had basic computer skills, primary in the area of text processing, drawing programs, and e-mail.

Procedure After filling in a demographic questionnaire, subjects were introduced to the procedure of the experiment and to the software.

Subjects then worked twice on a text-formatting task by using the three gedrics described before. In 40 subtasks, they were asked to reformat a text displayed in a text editing window. Each subtask concerned a part of the text to be formatted using one of the 16 possible gedric operations. The TYPOGRAPHER was used 19 times, the LAYOUTER 11 times and the PAINTER 10 times. The operations were used in no systematic order.

The instructions for the subtasks were given on the right segment of the DynaWall in sentences of the following character: "Please change the colour of the text of the third paragraph to blue". Three to five of such instructions were given per presentation slide.

After each task session, the subjects filled in a questionnaire in which they were asked to assess different aspects of the software. The whole procedure took about one hour, the work on each task took about 15 minutes.

4.2.1 Measures

Performance variables Several interaction and performance measures were extracted:

- *Confusion of gedrics:* A wrong gedric was activated either by opening the menu or by performing a gesture. We recorded also which of the three gedrics were mixed up.
- *Error:* The number of errors, which were made during the performance of the task. Possible errors were: selecting a wrong menu command, using a wrong gesture, and confusing gedrics.
- *Time on task:* Time on task was recorded by monitoring the start and end of the interaction.

The following variables reflect the learning of the gestures in the interaction conditions G and C:

- Correct gesture without opening the menu: A correct gesture is performed without opening the menu of the gedric first. In this case, the user has learned the gesture and performs it
- Correct gesture after first opening the menu: A correct gesture is performed immediately after opening the menu of the gedric. In this case, the user has not learned the gesture yet.

In condition C, the proportion of the *number of menu commands used* and the *number of gestures used* was recorded.

User Satisfaction The questionnaire consisted of 18 questions in which the subjects were asked to give their opinion about the task, the gedrics, the menus and the gestures.

The subjects in condition C were asked after both task performances, whether they preferred the gestures or the interaction with the menus.

5 Results

For analysing the data of this experiment, we used an analysis of variance with three factors (one factor being the repeated measurement). We were using levels of significance of 1% or 5%.

5.1 The Two Interaction Styles

In condition C, there was a significant switch from the dominance of *menu commands used* in the first performance of the task to a dominance of *gestures used* in the second ($F(1,18) = 11.911, p < .05$). This development is shown in Figure 6.

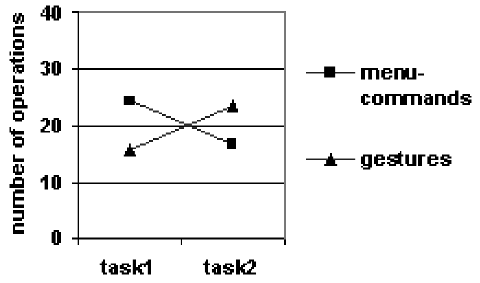

Figure 6: Switch from menu commands to gestures in condition C.

In the questionnaire, the subjects significantly changed their preferences of interaction styles from liking the menu interaction most in the first task to liking the gesture interaction most in the second task ($F(1,18) = 4.571, p < .05$).

In comparison to the subjects in condition G, the subjects in condition C using gestures found them significantly more illogical ($F(1,36) = 6.415, p < .05$), and highly significant more difficult to remember ($F(1,36) = 7.935, p < .01$).

An analysis of the different relations of *used gestures* and *used menu commands* in condition C shows three interaction profiles:

- Subjects who — on the average — performed 0 to 13 of the 40 operations by gestures in both tasks.
- Subjects who used 13 to 26 gestures.
- Subjects who used 26 to 40 gestures.

The 18 subjects of condition C are distributed almost equally over these three groups: six subjects used few gestures, seven subjects used gestures and menu commands equally, and five subjects used more gestures than menu commands.

5.2 Gesture Interaction

The comparison of the three interaction conditions C, G, and M showed no significant differences between the groups concerning *time on task, error, confusion of gedrics,* and *user satisfaction*.

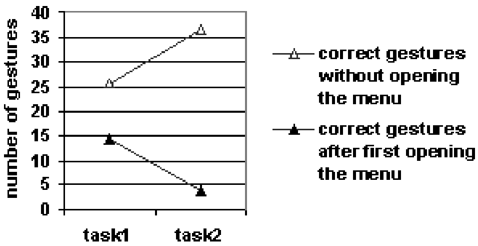

Figure 7: Gestures usage in condition G.

In condition G, the analysis of the performance of gestures shows that the number of *correct gestures without opening the menu* increased highly significant ($F(1,18) = 39.859, p < .01$) and the number of *correct gestures after first opening the menu* decreased significantly ($F(1,18) = 43.768, p < .01$) in the second performance of the task (Figure 7).

Also in condition C, the number of *correct gestures without opening the menu* increased very significantly ($F(1,18) = 29.638, p < .01$) in the second time on the task.

5.3 The Introduction Variable

Subjects in condition G with an *elaborated introduction* performed significantly more *correct gestures without opening the menu* ($F(1,18) = 5.699, p < .05$) than subjects in condition G who belonged to the *simple introduction* condition. These subjects performed significantly more *correct gestures after first opening the menu* ($F(1,18) = 5.877, p < .05$).

The confusion between the gedrics varied in general over all conditions. Most confusions occurred between the TYPOGRAPHER and LAYOUTER (mean(task1) = 3.43, std(task1) = 3.17 and mean(task2) = 1.89, std(task2) = 2.42). The TYPOGRAPHER and PAINTER as well as the LAYOUTER and PAINTER were hardly mixed up. The influence of the introduction condition was significant on the confusion of the TYPOGRAPHER and LAYOUTER ($F(1,53) = 4.105, p < .05$) — an *elaborated introduction* led to fewer confusions. In the questionnaire subjects in the *elaborated introduction* condition judged the gedrics to be easier to differentiate than did the subjects in the *simple interaction* condition ($F(1,54) = 5.142, p < .05$).

6 Discussion

6.1 The Two Interaction Styles

It can be seen that the possibility of interacting with the gedrics by using the pop-up menus or the gestures offer a smooth transition from novice to expert performance with the sample application. In addition, it supports personal preferences and individual skills without causing any deficits in usability. On the average, a preference for gesture interaction can be stated, although this interaction style was new to all subjects.

Results show that subjects switched from a dominance of the usage of the pop-up menus in the first task to a dominance of gesture interaction in the second task. Furthermore, they also switched in their preferences from liking the menu interaction most in the first task to liking the gestures most in the second task. The preference of the gestures was stated although in some aspects subjects in condition C perceived the gestures as more difficult than subjects in condition G. This switch in performance and preference is even more interesting under the aspect that subjects were not instructed to use the gestures but did it spontaneously.

The analysis of the performance characteristics shows that there seems to exist personal preferences of using the pop-up menus or the gestures. Not all subjects switch to gestures in the second task and not all operations were performed by gesture. All in all, these personal preferences do not lead to any interaction constraints compared to the other two interaction conditions. The power of the combined interaction should increase by the increase of the number of gedrics and commands.

6.2 Gesture Interaction

Analysing the subjects' performance of gestures and keeping in mind that the gestures were a novel form of interaction for subjects, we found satisfactory results for the usability of gesture interaction with the gedrics.

The three interaction conditions did not show any significant differences in any of the three interaction variables *time on the task*, *error* or *confusion of gedrics*. The novelty of the gestures and the higher effort of learning the gestures does not seem to affect these aspects of usability. Concerning the *time on the task*, the effects of higher learning effort (for gestures) and higher number of interaction steps (for menu commands) seem to compensate each other. We assume that gesture interaction will become more efficient the more experienced a user is.

The execution of a gesture without the prior opening of the gedrics' menu shows that the respective gesture is performed by rote. In condition G, in the second task on an average 36.33 gestures were executed without opening the menu (std = 7.75). The subjects were able to learn almost all the gestures. Also in condition C, where subjects were not instructed to use the gestures, they showed a highly significant learning effect of the gestures. This is a pleasing result and shows that the gesture interaction is easy to learn in an interaction period of about 30 minutes. Of course, the degree of learnability may change with an increasing number of possible gesture operations.

6.3 The Introduction Variable

The type of introduction (*elaborated* or *simple*) had influences on the learning of the gestures and on the confusion between the gedrics.

In condition G, an *elaborated introduction* was helpful for learning the gestures. Subjects performed significantly more *correct gestures without opening the menu* which indicates that the gesture has been learned. The *confusion of gedrics* over all interaction conditions was reduced by giving subjects an *elaborated introduction*. And subjects in the *elaborated introduction* condition themselves stated the gedrics to be more easy to differentiate than those in the *simple introduction* condition.

The higher number of confusions between the TYPOGRAPHER and the LAYOUTER than between each of them and the PAINTER shows that there is not a clear distinction of the functionality of these two gedrics. If it is not easy to differentiate the gedrics, the number of mistakes increases. These results — the increase of usability caused by the *elaborated introduction* — show that the logical concepts are also evident to the user and not only to the designer of the software.

7 Conclusions and Future Work

In this paper, we introduced a simple application that makes use of gedrics — gesture-sensitive icons as part of a pen-centric interface. We evaluated this application with respect to several usability aspects. It turned out that users like the different interaction styles provided by gedrics very much and that they can work with interfaces like this efficiently, independent from their skills, experiences, and preferences.

The results of this evaluation are the starting point for further experiments. Especially with respect to complexity and scaling to higher numbers of gedrics, the gedric approach has to be observed in more detail. In our sample application, only three gedrics with a total of 16 functions were available. Although the subjects were able to learn these functions and the corresponding gestures very fast, more gedrics with more functionality for each gedric — as specified in (Geißler, 1995; Geißler, to appear) — may cause problems. We are currently in the process of setting up an experiment with less variables to examine but with more functionality to be learned by the subjects, because we believe that especially free-form input shapes as gestures offer more possibilities for interaction.

The conceptual clarity in the design of gedrics, i.e. choosing the right set of functions they represent, their appearance as an icon, a good menu design, as well as intuitive gestures that correspond with the functions and the icon, is the most important factor for the usability of gedrics. All in all, the TYPOGRAPHER, the LAYOUTER, and the PAINTER seem to be designed quite well but there is still space for improvements. Especially the distinguishability of gedrics with respect to their task as well as their visual appearance has to be taken into account.

However, the findings we got from our evaluation encourages us to design and implement the basic interaction concept of gedrics — gestures and menus — on a larger scale. Not only icons can be made gesture-sensitive that way but any interface component. At the moment, we are improving a set of gesture-sensitive widgets (Geißler, 1998) and basic data elements like characters and text that react directly on pen input in the same way (Geißler, to appear). This set of components will be an important step from traditional pen-*based* interfaces towards pen-*centric* interfaces. With these interfaces, users always will have the freedom of choosing that mix of interaction styles that is most appropriate for them.

Acknowledgements

At the time the work described in this paper took place, all authors were affiliated to the Integrated Publication and Information Systems Institute (IPSI) of the German National Research Center for Information Technology (GMD).

We would like to thank the members of the AMBIENTE team at GMD-IPSI for making the system development and this evaluation possible. Special thanks go to Torsten Holmer who developed software for recording user actions at the DynaWall.

References

Geißler, J. (1995), Gedrics: The Next Generation of Icons, *in* K. Nordby, P. H. Helmersen, D. J. Gilmore & S. A. Arnessen (eds.), *Human–Computer Interaction — INTERACT'95: Proceedings of the Fifth IFIP Conference on Human–Computer Interaction*, Chapman & Hall, pp.73–8.

Geißler, J. (1998), Shuffle, Throw or Take it! Working Efficiently with an Interactive Wall, *in* ***EDITOR*** (ed.), *Proceedings of CHI'98: Human Factors in Computing Systems*, ACM Press, pp.265–6.

Geißler, J. (to appear), Design and Implementation of Pen-centric User Interfaces, PhD thesis, Technical University of Darmstadt.

Grudin, J. (1989), "The Case Against User Interface Consistency", *Communications of the ACM* **32**(10), 1164–73.

Kurtenbach, G. & Buxton, W. (1991), Issues in Combining Marking and Direct Manipulation Techniques, *in* ***EDITOR*** (ed.), *Proceedings of the ACM Symposium on User Interface Software and Technology, UIST'91*, ACM Press, pp.137–44.

Kurtenbach, G. & Buxton, W. (1993), The Limits of Expert Performance Using Hierarchic Marking Menus, *in* S. Ashlund, K. Mullet, A. Henderson, E. Hollnagel & T. White (eds.), *Proceedings of INTERCHI'93*, ACM Press/IOS Press, pp.482–7.

Meyer, A. (1995), "Pen Computing. A Technology Overview and a Vision", *ACM SIGCHI Bulletin* **27**(3), 46–90.

Nielsen, J. (1993), *Usability Engineering*, Academic Press.

Shneiderman, B. (1997), *Designing the User Interface: Strategies for Effective Human–Computer Interaction*, third edition, Addison–Wesley.

Streitz, N. & Geißler, J.and Holmer, T. (1998), Roomware for Cooperative Buildings: Integrated Design of Architectural Spaces and Information Spaces, *in* N. Streitz, S. Konomi & H.-J. Burkhardt (eds.), *Cooperative Buildings. Integrating Information, Organization, and Architecture*, Vol. 1370 of *Lecture Notes in Computer Science*, Springer-Verlag, pp.4–21.

Streitz, N., Geißler, J., Holmer, T., Konomi, S., Müller-Tomfelde, C., Reischl, W., Rexroth, P., Seitz, P. & Steinmetz, R. (1999), i-LAND: An Interactive Landscape for Creativitiy and Innovation., *in* ***EDITOR*** (ed.), *Proceedings of CHI'99: Human Factors in Computing Systems*, ACM Press, p.***PAGES***.

Wolf, C. (1992), "A Comparative Study of Gestural, Keyboard, and Mouse Interfaces", *Behaviour & Information Technology* **11**(1), 13–23.

Wolf, C., Rhyne, J. & Ellozy, H. (1989), The Paper-like Interface, *in* G. Salvendy & M. Smith (eds.), *Designing and Using Human–Computer Interfaces and Knowledge-based Systems*, Edward Arnold, pp.494–501.

Human–Computer Interaction — INTERACT'99
Angela Sasse and Chris Johnson (Editors)
Published by IOS Press, © IFIP TC.13, 1999

Author Index

Human–Computer Interaction — INTERACT'99
Angela Sasse and Chris Johnson (Editors)
Published by IOS Press, © IFIP TC.13, 1999

Keyword Index