Collaborative Tactile Graphic Workstation for Touch-Sensitive Pin-Matrix Devices

Collaborative Tactile Graphic Workstation for Touch-Sensitive Pin-Matrix Devices

Jens Bornschein
Human-Computer Interaction Research Group
Technische Universität Dresden
Nöthnitzer Str. 46, 01187 Dresden, Germany
jens.bornschein@tu-dresden.de

Denise Prescher
Human-Computer Interaction Research Group
Technische Universität Dresden
Nöthnitzer Str. 46, 01187 Dresden, Germany
denise.prescher@tu-dresden.de

ABSTRACT

To allow for producing high quality tactile graphics, the involvement of blind users in the production process can be a very vital approach. In this paper, we describe a collaborative tactile graphic workstation. Thereby, a sighted user works with a conventional vector graphic production software while a blind user can edit the same graphic on a two-dimensional pin-matrix device at the same time. The software architecture as well as the tactile user interface is described. Furthermore, some conflicts that can occur in a synchronous collaboration task within the system are discussed.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*haptic I/O, input devices and strategies*; H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces—*collaborative computing, synchronous interaction*; K.4.2 [**Computers and Society**]: Social Issues—*Assistive technologies for persons with disabilities*

General Terms

Design, Human Factors

Keywords

Tactile graphic creation, blind users, collaboration, pin-matrix device, tactile display, conflicts, graphic object editing

1. INTRODUCTION

In today's society more and more information is presented visually. In order not to exclude blind and visually impaired users from these information, accessible alternatives such as descriptions or tactile graphics have to be prepared. The production of adequate material is time-consuming, but a lot of blind and visually impaired users take a great interest in working with tactile graphics [9]. Especially involving this user group in the production process can enhance the results.

TacTT '14, Nov 16 2014, Dresden, Germany.

Compared to static tactile print-outs, two-dimensional pin-matrix devices can allow for a dynamic exploration of tactile graphics. This enables blind users to check every modification to a drawing immediately without the need of laborious reprinting of the result. Besides, an autonomous editing of the graphic file by the blind user can be possible, too. For example, a touch-sensitive device allows for directly pointing and selecting tactile elements, which can lead to effective working strategies.

In the following, we present a graphic production system which allows for a cooperation of blind and sighted people in creating and modifying tactile graphics on a two-dimensional pin-matrix device. Using the *BrailleDis 7200* (see [8]) as an example, the user interface and interaction design of the tactile graphic workstation are described in detail. In conclusion, some conflicts which can occur as well as some future work will be discussed.

2. BLIND AND SIGHTED PEOPLE IN COOPERATION

In daily life, the cooperation and collaboration between blind and sighted people is common, for instance at work or also in a private environment. If only some cooperation is necessary, and therefore the participants "do their assigned parts separately and bring their results to the table" [4, p. 17], the requirements on a non-visual interface may not be as strict as in the case of collaboration. Considering the collaboration which "implies direct interaction among individuals to produce a product and involves negotiations, discussions, and accommodating others' perspectives" [4, p. 17], much more attention should be put on the consistence and synchronization of the visual and non-visual workspace.

There has already been done some work on the cooperation and collaboration of sighted and blind users. Especially scenarios in the field of education or support were considered. For example, a sighted user can get remote access to a workstation of a blind user. This is applied for instance by *Jaws Tandem*[1], which is used for technical support or training.

Another scenario can be some kind of the following: A blind user is guided by a sighted partner by using a haptic device, such as the *Phantom*[2]. Thereby, a pulling force drags the

[1] Jaws Tandem Quick Start Guide, see: http://www.freedomscientific.com/JAWSHq/JawsTandemQuickStart

[2] Phantom Omni haptic device from Sensable Technologies, see http://www.dentsable.com/haptic-phantom-omni.htm

pen of the *Phantom* to the position of the mouse cursor. For example, this is used to learn character shapes, handwriting and signatures (*McSig* [7]) or edit and explore virtual 2D relief drawings (*AHEAD* [11]).

BPLOT3 enables a collaborative production of tactile graphics because of the synchronization of a Character User Interface (CUI) for the blind and a Graphical User Interface (GUI) for the sighted user [2]. In this system the blind user can draw an image either by using the plotter control language or by touch input on a Tablet PC. However, without printing the result on a Braille printer, the user does not have any tactile feedback about his drawing. Particularly, he cannot track the input of the sighted user. For this reason, an effective collaboration seems to be hard to imagine.

To allow for the collaboration between blind and sighted users the communication about the different usage of application interfaces has to be possible [5]. This means, the non-visual interface not only has to be intuitive and usable, but it also has to express similar concepts as the graphical interface. Providing blind users with information about the spatial layout of the interface can enable a successful collaboration [6]. Especially in the field of tactile graphics this requirement is very important. Large dynamical Braille displays can help to achieve this goal.

Besides the requirements on the non-visual interface mapping, there are also some other challenges concerning collaborative editing tasks. Working on the same document is a problem that is well known in the field of groupware. Conflicts based on synchronous editing scenarios can be resolved by merging the different changes, preventing parallel editing by locking objects or whole areas and - which is the preferred solution - by making every editor aware of the other editors and their changes [3]. The detection of conflicts can be managed by the system, but the solution should be hand over to the users. Solving conflicts is a social process and, therefore, the workstation should not limit the cooperation as well as the communication between the editing partners. If necessary, it is helpful to protect very critical parts of the document from irreversible damages or information loss by locking them.

3. COLLABORATIVE CREATION & MODIFICATION OF TACTILE GRAPHICS

In the context of creative processes the exchange of ideas between several people can be very helpful. The production of graphics is such a process and, therefore, can benefit from a collaborative scenario. Normally, the draftsman of a tactile graphic is not part of the target user group. Especially in this case the involvement of a blind or visually impaired person can enhance the quality of the results.

In the following, we present our collaborative tactile graphic workstation (Tangram workstation), which is based on the open source vector graphic production software *OpenOffice Draw*. While the sighted user can work with the software in a conventional manner, a blind user can edit the same graphic simultaneously by using a two-dimensional tactile pin-matrix display, in our case the *BrailleDis 7200* (see Figure 1), as input and output device.

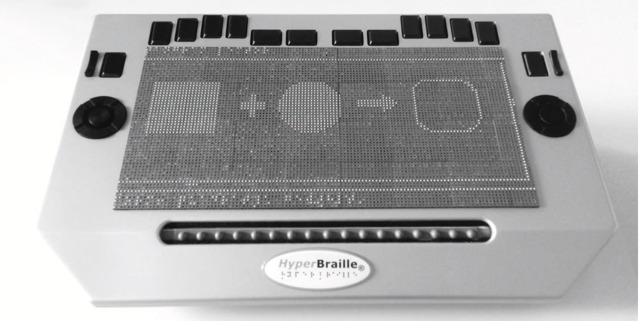

Figure 1. BrailleDis 7200 on which an example of the tactile user interface of the Tangram workstation is shown.

3.1 Architecture of the Tangram workstation

The Tangram workstation consists of a usual graphic production software as well as a tactile lector system allowing to use a pin-matrix device. Both users, the sighted and also the blind, work on the same computer system by using different input and output devices that are synchronized (see Figure 2). The tactile lector is completely event-driven, either triggered by user input or by the application. This saves performance, but has the disadvantage that if no event is fired or gets lost, no action or feedback can be activated.

3.1.1 Graphic Production Software and Access to its Graphic Objects

As mentioned above, *OpenOffice Draw*, which is an open source and free to use project, is used as graphic editor within the Tangram workstation. The main advantage is its big functional range for no costs. Furthermore, the open SDK and API allows for adapting the editor to special needs. Because of the active community a continuous development and bug fixing is in progress. Nevertheless, there are also some disadvantages. As *OpenOffice Draw* is no commercial tool, no support or warranty exists. Especially non-usual areas of the software are still full of little bugs. That means, while common GUI functions work pretty well and are tested sufficiently, internal functions can have serious errors. Because of limited resources in the developer community, the focus of developing and bug fixing is not on the elimination of internal problems, but on the user view.

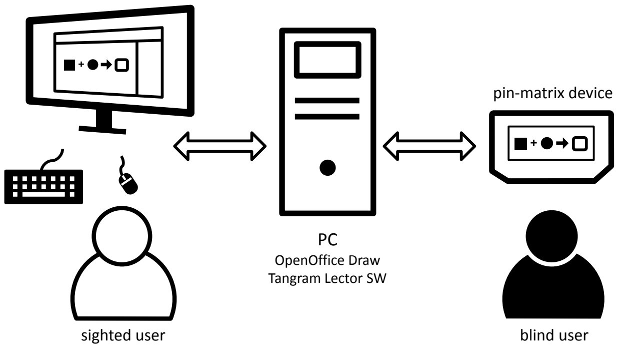

Figure 2. Schematic representation of the Tangram workstation.

There are two views on the graphical objects inside the *OpenOffice* document. At first, the accessible view giving access to the rendered screen objects. This includes information, such as screen position, rendered size in pixels as well as some further accessibility information (e.g. role, interaction state, parent/child structure, some kind of naming). Within the accessible view no manipulation of the object is possible. The second view is the Document Object Model (DOM) which is the real representation of the document's elements and, therefore, can be directly manipulated. The DOM view gives access to all parameters in relation to the document, but not in relation to the rendered objects.

There is no direct connection between these two views, which results in the following problem: It is not possible to directly get a manipulable DOM object from a screen object. Our solution to overcome this problem is to give every DOM object a unique identifier as name and try to link between the accessible and the DOM view via that unique name. However, this solution requires that the given unique ids are not changed by the user to not unique ones. Furthermore, searching and matching of these ids is very resource consuming and can lead to mismatches. For example, if the user selects an object on the pin device and a false DOM element (with the same id) is linked to that object, the false object would be manipulated by the user. Unfortunately, we cannot solve this problem as these two views on one object give no unique clue if they are from the same object or not.

3.1.2 *Tactile Lector for the Pin-Matrix Device*

The tactile lector part of the Tangram workstation is based upon the *BrailleIO* framework [1] which is a .NET based framework for abstracting different two-dimensional tactile displays. It allows for a fast generation of tactile user interfaces composed of a basic windowing system with Braille and image rendering possibilities, interaction mechanisms (including a gesture recognizer) and manipulation features, such as zooming and panning.

These basic functionalities are extended by application-specific features that are independent of the framework. To realize all in- and output functions of the tactile graphic workstation, the framework is extended by a window manager dealing with the user interactions, an audio renderer for giving auditory feedback and an interaction manager. The last admits of additional user interactions which are dependent on the concrete application. Thereto, different function proxies can be registered. These are used as context-sensitive input interpreters, for instance to realize a Braille keyboard in Braille input mode or the manipulation of DOM objects in graphic editing mode.

3.1.3 *Combining the Tactile Lector with the GUI*

The graphic editing mode is the core function of the tactile lector. If the blind user clicks on the accessible view of a graphic element on the pin device, the internal mapping gets the corresponding DOM object, which leads to the manipulation abilities of the user (see Figure 3). In contrast, the sighted user only works with the GUI objects of *OpenOffice Draw*. Therefore, working simultaneously is possible.

Nevertheless, a big problem arises because of the direct access of the blind editor's workstation into the DOM and the application. All properties of the DOM element can be manipulated without any restriction. Theoretically, this makes the blind user more powerful than the sighted user. But this also can lead to damages in the document or the application, which can cause the loss of the complete work.

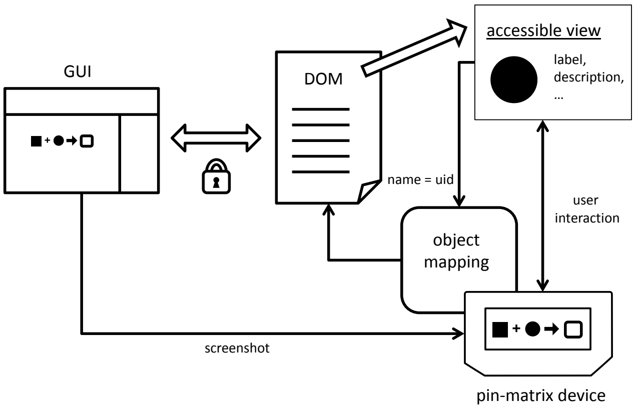

Figure 3. Mapping of DOM and GUI elements within the tactile lector system.

In conclusion, the Tangram workstation consists of a tactile and a visual interface. The sighted user works with the GUI of *OpenOffice Draw* and the blind user has direct DOM access. This allows for a synchronous editing within the same document. As the workflow of the sighted user is only the conventional usage of the *OpenOffice Draw* application and, therefore, is not influenced by our system, we only will describe the tactile user interface in detail.

3.2 Tactile User Interface

A consistent usage of different software applications on the *BrailleDis 7200* is very helpful to allow an efficient workflow for blind users. By and large, the Tangram workstation should be based on the concepts of the *HyperBraille* windowing system and its interaction techniques described in [10]. Therefore, a simplified window manager was included in the Tangram workstation. In the following the composition as well as the interaction design of the tactile interface are described.

3.2.1 *Composition of the Tactile Interface*

In the standard case, the output of the pin-matrix display is divided into three regions (see Figure 4). In the header region the title of the graphic is shown. In the body region the graphic is either shown as pixel-based screenshot or as text-based Braille description. Normally, in the detail region some information about the focused element or the current context menu of the Tangram workstation is shown. As the detail region was already rated by blind subjects as effective for receiving further information within the *HyperReader* [10], the tactile interface of the Tangram system will also include temporary status messages which can be shown for a predefined period of time, e.g. feedback about the last user input.

To allow an up-to-date output of the graphical content on the pin-matrix device, the screenshot view of the tactile lec-

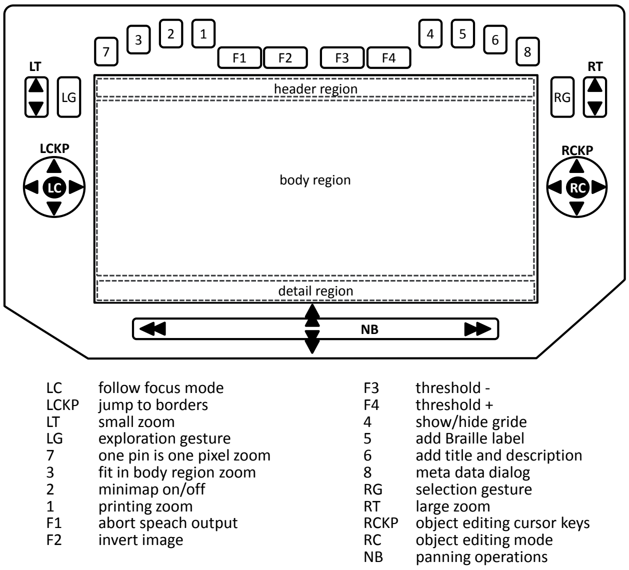

Figure 4. Schematic representation of the tactile interface and its basic button concept for BrailleDis 7200 (only single key commands are shown).

tor is continuously refreshed. Thereto, a screen observer generates every 100 milliseconds a screenshot either of the complete monitor, a part of the monitor, an application window or a part of an application window. Based on its color brightness and the defined threshold this image is transformed into a binary image. In consideration of the current zooming and panning values the binary image is presented in the body region. Using zooming and panning commands (see next section) enables the user to change the currently presented detail of the image. In this way, the complete graphic can by explored freely.

As the blind user should not be overloaded with too much information, the screenshot view only presents the drawing area of *OpenOffice Draw*. If necessary, toolbars and property windows will be shown in other views or regions in the future. Therefore, they have to be adapted for the tactile interface at first. Overlapping windows, such as the open file dialog, window explorer or software keyboard, are also ignored by the screenshot view. For this purpose every modification of the application window has to be followed and the image section has to be adapted. This is realized through interpreting window events.

As there is no interpretation of the presented content within the screenshot, no Braille can be shown within the tactile graphic. Indeed, the mapping of the pin device position to the associated screen position allows for getting the accessible object of the application. With it, auditory feedback to some basic properties of the object, such as name or title, can be given.

3.2.2 Interaction Design of the Tactile Interface

Besides the basic regions, the button commands are also based on that of the *HyperReader*. For example, the navigation bar and the left cursor key pad are used for panning operations, the toggle switches and the left Braille keys for zooming operations. This has the advantage that the user does not have to relearn all commands, if he already works with the *HyperBraille* system.

In addition to the existing zooming functionalities, such as zoom-in, zoom-out, fit in body region and one pin is one pixel, a printing zoom was added. This zoom level shows the content on the pin device in the same size as it is displayed on the monitor. Thereto, the relation of the monitor's and the pin device's resolution is removed. For instance, if the original *OpenOffice Draw* document is shown in 100% with the settings of an DIN A4 page, this allows for directly comparing a graphic in the same dimension on the pin device and an A4 swell paper printing.

For enabling an independent creation of tactile graphics on the pin-matrix device some special drawing operations are necessary. Especially if there is a collaboration with a sighted user, the keyboard could not be used for these tasks. Therefore, all commands have to be offered through only using the hardware keys of the pin device. The most important functions currently considered in the Tangram workstation are as follows:

1. manipulation operations
   (a) select object
   (b) edit object (move, scale, rotate, remove)
   (c) edit filling pattern and line style
2. annotation and document operations
   (a) add Braille label
   (b) edit title and description of objects
   (c) edit metadata of graphic file
   (d) open, save, export graphic
3. auxiliary operations
   (a) set threshold
   (b) invert image
   (c) show/hide grid or reference lines

Based on the button concept of the *HyperReader*, some keys or key combinations had to be adapted to these operations. In Figure 4 the concept for single key commands is shown. It is considered that functions which are very important or often used can be easily triggered. For example, the auxiliary functions for adjusting the tactile image output are put on the function keys and some annotation functions on the right Braille keys.

Especially, the object editing commands are important to enable blind users to adapt a graphic. However, as these direct manipulations demand more than one key code, the right cursor key pad is used to handle all these editing modes. For example, the center button is used for switching between moving, scaling and rotating the focused element. The four cursor buttons can be used to manipulate the object depending on the current editing mode (see Figure 5).

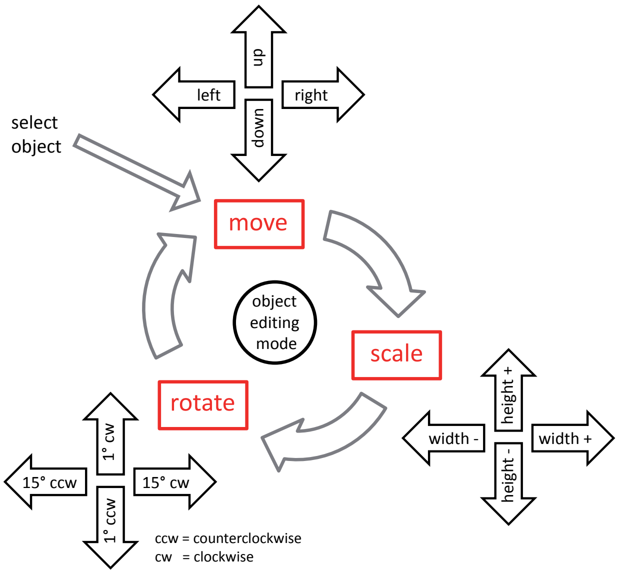

Figure 5. Usage of cursor keys in the three editing modes.

For instance, in moving mode the object is moved some pins in the chosen direction. To allow the user to recognize every manipulation on the pin device, at least one or two pins have to change. Therefore, the step which has to be considered is calculated depending on the current zoom level. That means, if a user has a small zoom level the step for moving an object in the chosen direction or for scaling it is much bigger than in a larger zoom level (see Figure 6).

This kind of mode concept allows for an easier and more robust interaction than the direct manipulation of modification handles which is used for mouse interactions. Compared to the mouse, the pointing at a handle with the finger can be very challenging as the area which has to be hit is very small. Furthermore, in the case of drag and drop operations blind users expect that dragging is done for them [12]. Therefore, the adaption of the direct manipulation of graphical shapes to a button input seems to be more reasonable for blind users.

A discussion with two blind people that have a lot of experience with the pin-matrix device and also with tactile graphics revealed that the proposed button concept can be applied intuitively.

3.3 Discussion of Possible Problems and User Conflicts

As the sighted and the blind user work on different perspectives of the document, namely the GUI and the DOM tree, no focus problems appear. Nevertheless, this leads to another conflict: Both users can edit the same graphic object at the same time, maybe without knowing from each other. This can result in data loss, because the first save will by overwritten by the last save.

Some of such cases can be reduced by giving the sighted person feedback about the object the blind user is currently editing. Therefore, the opacity of the DOM focused object can be changed rhythmically. This allows for simulating blinking pins on the pin-matrix device which can also highlight this object for the blind user. Until now, it is not clear how annoying this blinking is for the sighted and, especially, for the blind user.

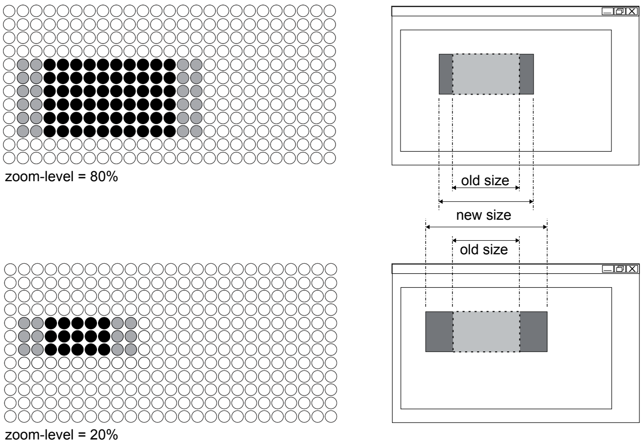

Figure 6. Schematic representation of scaling a graphic object depending on the chosen zoom level of the pin device.

Another problem can result from the direct manipulation of DOM elements. This can nullify security or validation mechanisms. For example, invalid names or attributes will not be checked by a dialog or the software application. Furthermore, the modification of properties or elements by one of the users may remain undetected by the other user who then relies on old and invalid information (e.g. name or title of an object).

As mentioned above, the output of the pin device shows an up-to-date screenshot of the drawing on the monitor. The synchronous collaboration on one workstation that only consists of one visual appearance of the drawing (GUI on the monitor) leads to the highly dependence of the blind user on the behavior of his sighted colleague. For example, if the sighted user scrolls or zooms, the relation system of the blind user changes and he can get lost. Therefore, we recommend to have a widescreen Full HD monitor with a resolution of at least 1920 x 1080 pixels to show a DIN A4 page (landscape mode) in 100% zoom level without any need of scrolling. A page in portrait mode will need a higher resolution.

In general, the blind reviewer can only see a part of the whole graphic on the pin device at once. If the sighted user modifies something out of this view range and wants to talk about it, the blind user could not know where to look. To allow such a collaboration between the sighted and the blind user, a follow focus function was added. Enabling this kind of special mode makes sure that the view range automatically shows the focused element in the middle of the body region. In addition, the name of the focused element is announced by voice feedback. To avoid auditory overload, the focused element (focus of the GUI and accordingly the sighted user) is spoken only in this mode. Elements which are lighter than the chosen threshold can be another reason not to be visible on the pin-matrix device. In this case the blind user could have no idea about the existence of such elements. At least he cannot recognize modifications to these objects. To minimize this problem there should be the possibility to go to the DOM focus. If there is no tactile feedback for the blind user, he could use that command to better follow his activities.

However, there are also DOM elements which cannot be focused at this time. For example, there is no chance to select an element by touch which is covered by another, because the overlaying element will be hit and selected. Therefore, a concept to sequentially navigate through the DOM tree is needed, too.

At this time, the blind user cannot create a new graphic object autonomously and, therefore, is dependent on the sighted user's favor. A discussion and agreement between the users is necessary. For the future, the question arises whether to give the blind person the power to create and delete elements or not. At the moment, it seems to be more reasonable to restrict the authority of the blind user because of the following reason. Manipulations of DOM elements are not registered in the undo-redo-history of *OpenOffice Draw* and, therefore, are not reversible currently. Hence, the error tolerance of the tactile interface is very low.

To provide an effective workstation for the collaborative production of tactile graphics, the loss of data should be avoided. Therefore, it also seems to be more reasonable to accept errors in readout and output of information to the blind user and the tactile interface respectively, instead of interrupting the application, which could cause a system crash.

4. CONCLUSION AND OUTLOOK

In this paper the Tangram workstation and, especially, its tactile user interface was described. The system enables blind and sighted users to highly collaborate in creating and editing tactile graphics in *OpenOffice Draw*. A touch-sensitive pin-matrix device, such as the *BrailleDis 7200*, enables the blind user not only to explore, but also to modify the graphic. Thereto, a button concept was designed which allows for doing some direct manipulation of graphical shapes on the tactile device. As collaborative tasks can cause interaction conflicts, possible problems were discussed in the paper, too.

In the future, appropriate menu structures and input dialogs for the tactile interface have to be implemented. In addition to the above-mentioned conflicts and problems some other aspects have to be solved, for instance the grouping of graphic elements on the pin device (multi selection) or the support in choosing a suitable threshold depending on the minimal color brightness of the graphic objects. Finally, a study with some blind users has to be conducted to allow for an evaluation of the workstation's usability.

5. ACKNOWLEDGMENTS

We thank our two blind participants for the rich discussion about the first concepts of our tactile drawing system. The *Tangram* project is sponsored by the Federal Ministry of Labour and Social Affairs (BMAS) under the grant number R/FO125423.

6. REFERENCES

[1] J. Bornschein. Brailleio - a tactile display abstraction framework. In *This Proceedings*, pages 36 – 41, 2014.

[2] M. Fujiyoshi, A. Fujiyoshi, A. Osawa, Y. Kuroda, and Y. Sasaki. Development of synchronized cui and gui for universal design tactile graphics production system bplot3. In K. Miesenberger, D. Fels, D. Archambault, P. Penaz, and W. Zagler, editors, *Computers Helping People with Special Needs*, volume 8548 of *Lecture Notes in Computer Science*, pages 18–25. Springer International Publishing, 2014.

[3] M. Koch. Design issues and model for a distributed multi-user editor. *Computer Supported Cooperative Work (CSCW)*, 3(3-4):359–378, 1994.

[4] O. Kozar. Towards better group work: Seeing the difference between cooperation and collaboration. In *English Teaching Forum*, volume 48, pages 16–23. ERIC, 2010.

[5] E. D. Mynatt. Transforming graphical interfaces into auditory interfaces for blind users. *Human-Computer Interaction*, 12(1-2):7–45, 1997.

[6] H. Petrie, S. Morley, and G. Weber. Tactile-based direct manipulation in guis for blind users. In *Conference Companion on Human Factors in Computing Systems*, CHI '95, pages 428–429, New York, NY, USA, 1995. ACM.

[7] B. Plimmer, A. Crossan, S. A. Brewster, and R. Blagojevic. Multimodal collaborative handwriting training for visually-impaired people. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 393–402, New York, NY, USA, 2008. ACM.

[8] D. Prescher. Redesigning input controls of a touch-sensitive pin-matrix device. In *This Proceedings*, pages 19 – 24, 2014.

[9] D. Prescher, J. Bornschein, and G. Weber. Production of accessible tactile graphics. In K. Miesenberger, D. Fels, D. Archambault, P. Penaz, P.z, and W. Zagler, editors, *Computers Helping People with Special Needs*, volume 8548 of *Lecture Notes in Computer Science*, pages 26–33. Springer International Publishing, 2014.

[10] D. Prescher, G. Weber, and M. Spindler. A tactile windowing system for blind users. In *Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '10, pages 91–98, New York, NY, USA, 2010. ACM.

[11] K. Rassmus-Grohn, C. Magnusson, and H. Eftring. Ahead - audio-haptic drawing editor and explorer for education. In *Haptic, Audio and Visual Environments and Games, 2007. HAVE 2007. IEEE International Workshop on Haptic Audio Visual Environments and their Applications*, pages 62–66, Oct 2007.

[12] G. Weber. Adapting direct manipulation for blind users. In *INTERACT '93 and CHI '93 Conference Companion on Human Factors in Computing Systems*, CHI '93, pages 21–22, New York, NY, USA, 1993. ACM.