Visualization for the Mind's Eye

Nelson Baloian[1], Wolfram Luther[2]

[1,2] Institut für Informatik und Interaktive Systeme der GMU Duisburg,
Lotharstraße 65, 47057 Duisburg, Germany
{Baloian, Luther}@informatik.uni-duisburg.de

Abstract. Software visualization has been almost exclusively tackled from the visual point of view; this means visualization occurs exclusively through the visual channel. This approach has its limitations. Considering previous work for blind people we propose that complementing usual approaches with those techniques used to develop interfaces for non-sighted people can enhance user awareness of logical structures or data types using different perception channels. To achieve better comprehension, we deal with new or augmented interfaces built on top of standard systems for data visualization and algorithm animation. The notion of specific concept keyboards is introduced. As a consequence, modern information and learning systems can be designed in such a way that not only sighted but also blind users can navigate within these systems.

1 Value and Problems of Software Visualization

Software visualization deals with the animation of algorithms, including numerical, geometric, graphic, and graph algorithms, as well as the visualization of data structures in information systems or in the computer's memory while certain complex processes are performed. One of its main goals is to achieve a better understanding of complex programs, processes, and data structures by means of showing complex digitized images displayed on a CRT- or LCD-monitor or a printing device. Through these images transmitted by the visual channel, users should generate planar or spatial structures with dynamic objects in their minds.

The visual channel permits a rapid overview of structures after an adequate abstraction process, a separation of important objects from less important ones whenever the former are distinguishable by graphic attributes, and a real time processing of dynamic process data output - furthermore, a strong data compression in the mind accompanies the high-band wide reception process. However, the visualization approach has several remarkable consequences: Multidimensional structures are projected to a plane which results in numerous design problems and higher data structures; parallel processes are serialized leading to nested screens or dialogues. Here, we will not contribute to the 'Layout Creation and Adjustment Problem' for graphs in visualization and animation systems [9].

Too often, the resulting graphic outcome is highly complex if we consider rasterized animated sequences and is strongly compressed during the reception process. Only a few details remain in memory. This fact is widely discussed in the literature. J. Norseen [18] deals with the conversion of 2D retinal visual sensory information signals into 3D semiotic mental representations and describes a visual semiotic language built from a finite alphabet of basic images: In constructing mental images certain left occipital areas of the human brain seem to perform the same neurological functions while the so-called Broca-Wernicke area does the conversion of aural sensory signals into a finite set of sound types producing neurolinguistic patterns. Norseen suggests that only about forty sound types and thirty images form the basis of expressed aural and visual languages.

Consequently, it would seem to be of interest to bypass, or better still, to complement standard graphic displays and search for alternative ways to provide logical structures using different perception channels to human minds provoking equivalent or alternative impressions and images.

The paper begins with a discussion of parallel reception modes and introduces the notion of concept keyboards. Then different ways for implementing an enhanced perception are discussed; this is followed by a case study. We formulate some ideas on the automatic generation of concept keyboards, look at recent work in the area and finish with some conclusions.

2 Parallel Reception Modes

Our working hypothesis will be that not only sighted but also blind users should be able to use and navigate within systems implementing new or augmented interfaces using enhanced perception tools to achieve software visualization for the mind's eye [16]. Thus, we do not intend to develop a system for blind people only, on the contrary, we claim that by proposing complementary perception channels and navigating facilities there will be a real enhancement for "normal users" when systems are navigable and usable for people with disabilities. It is reported in [17] that blind people develop special forms of navigating within an unknown environment and represent spatial structures with cognitive difficulty. This is true not only for the real world but also in virtual computer based environments. Certain evidence of this is given by experiences done according to the HOMER UIMS approach by Savidis and Stephanidis [23]; this approach consists of developing dual user interfaces for integrating blind and sighted people. To achieve this goal standard visualization elements like control element icons, tool menus, short cuts, logical structures with nodes and links, hypertext, images and animated sequences are enriched with acoustic elements or haptic interfaces, which allow direct interaction of the user with objects of the model used for the computer to represent the problem being explained or presented. To navigate independently from the graphic output we introduce the idea of a draft keyboard, which is realized by redefining keys on a traditional keyboard, by a matrix of small keys on a graphic tablet or by mapping them with the help of problem-specific hardware.

3 Ways to an Enhanced Perception

At present, virtual environments are basically built on visual displays, with some use of auditory and very little haptic information. The International Community for Auditory Display (ICAD) is a forum for presenting research on the use of sound to display data, monitor systems providing enhanced user interfaces for computers and virtual reality systems [12]. Research areas include the auditory exploration of data via sonification and audification, perceptual issues in Auditory Display systems and sound in immersive interfaces and virtual environments.

Mapping scientific data redundantly to visual and aural elements may increase the perception of the information and can lead to better insight and understanding. Conveying the same information using different channels to transmit it to the user becomes an important design element in systems where the network bandwidth is limited.

To complement the visual channel we have to design an acoustic interface which delivers at least the same information as the graphic one. So we have to develop a correspondence between visual and aural control elements as well as acoustic and graphic attributes. An earcon characterized by a typical melody can be added to any iconic control object. Earcons are abstract musical tones that can be used in structured combinations to create auditory messages. In 1989 Meera Blattner [7] introduced earcons as nonverbal audio messages to provide information to the user about computer objects, operations or interactions. Earcons are constructed from simple melodies also called motifs. A motif is a typical tonal pattern sufficiently distinct to represent an individual recognizable entity. The most important features of motifs are instrument, rhythm, and pitch. Earcons for such operations as ‘Play’, ‘Go left’, ‘Go right’, ‘Forward’, ‘Back’, ‘Jump’, ‘Start’ and ‘Stop’ could be created. It is possible to produce higher level earcons such as ‘Next problem’ or ‘Close program’ [8] and to create hierarchical structures. For modeling objects in our virtual world the following correspondences are possible:

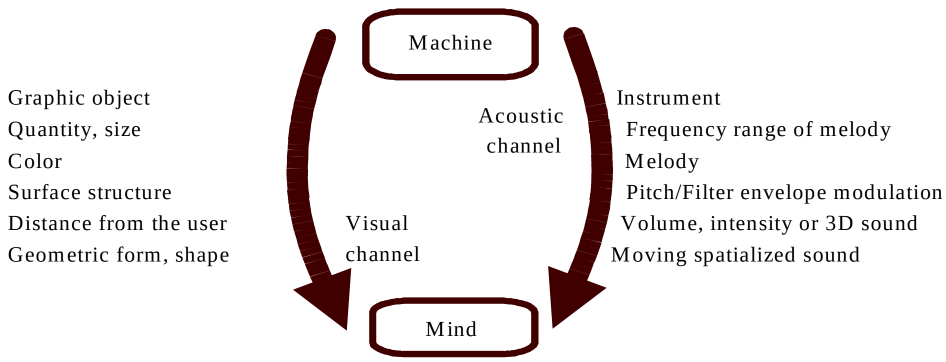

Fig. 1. Correspondences between graphic and aural attributes

According to Bissell [6] people’s association between tonal pitch and spatial position depends on the cultural context. Moreover, the correspondences proposed in

Figure 1 do not fully match the established psycho-physiological basis for Western culture. A tone, being of definite pitch, is the type of sound particularly relevant here. Tones represent a striking metaphor of location and motion, a change in pitch is perceived as being analogous to actual spatial motion and as taking place along the vertical dimension. Tonal brightness is associated with visual brightness, which is associated with visual or physical highness in space.

Nodes and links are the constituent components of Hypertext. While nodes can easily be complemented by text explaining the content in a straightforward way, there exist several possibilities for transforming links into sound. Neighboring nodes can be enriched by text hints or a pair of sounds; – a path can be illustrated by sequences of sounds. All these representations are volatile and must be activated after some time. This can also be done by using hotspots as particular restricted areas of larger graphics. Thus, moving into such an area or leaving it would trigger auto-narration or special wave files.

An interesting aspect of modern graphic user interfaces is they offer an easy and comfortable way to navigate and interact with software systems directly on visualized structures or through control elements like buttons, scrollbars, and dialogues.

– Users point directly at displayed interactive objects.
– They traverse the graphic representation of a logical or a hierarchical data structure with the aid of arrow keys or a pointing device.
– They manipulate or search for interactive objects by exploring a matrix of small areas called a concept keyboard with the aid of arrow keys or a pointing device and graphic tablet.

However, combining visual output with control elements is not appropriate for a blind person. Whereas the parallel use of icons and earcons enables blind people to control the system, the navigation on visual representations of internal data structures should be accompanied by appropriate concept keyboards custom-built for the application.

Currently, there are industries manufacturing touch devices called Concept Keyboards which can be connected to the computer through a serial port. For example, a touch pad consists of a flat touch-sensitive polycarbonate surface (A2, A3 or A4 size) made up of 128 or 256 cells set out in a rectangular array and allows the user to select the keyboard layout best-suited to the required application. Up to 256 programmable keys, defined individually or in groups with pre-designed overlays and blank templates can be customized using one of the existing overlay designer software packages. An overlay can have many different layers of information, pictures, video or sound. Paper sheets can be inserted on which objects are embossed and words are written in Braille. A blind person controls an application by pressing on the embossed pictures or Braille words. When particular areas of the concept keyboard are pressed, a digitized sound can be heard. A number of tactile overlays have been designed to be placed on the concept keyboard. Users can move their fingers along the tactile path, and, when they press certain 'nodes', the computer executes appropriate control commands or plays sound sequences [1].

4 Case Studies for Enhanced Perception

4.1 Visualization of Algorithms in Computer Graphics and Image Processing

ViACoBi [10, 11, 15] is interactive multimedia courseware visualizing Computer graphics and Image processing algorithms; it provides a new learning environment for undergraduate students to accompany classical lecture and lab courses. It proposes eight lessons, each one providing a certain interactive presentation mode with links to relevant technical facts and terms in a glossary and a context-sensitive help function. It also includes exercises, and a visualization and animation toolbox to construct relevant artifacts and to test algorithms in different scales and with various input parameters in a stepwise execution mode displaying all relevant variable contents. One aim of the system is the visualization of rastering algorithms to digitize straight lines, circles and ellipses.

We explain our approach with an example concerning 4-way and 8-way stepping in digital geometry. In raster graphics, there are four horizontal and vertical neighbors (x±1,y) and (x,y±1) of a pixel (x,y) or eight neighbors if we add the four diagonal neighbors (x+1,y±1) and (x-1,y±1). By stepping via 4- or 8- adjacent points we obtain 4- or 8-paths or so-called raster curves built from adjacent 4- or 8-neighbors. A concept keyboard should support 4-way or 8-way stepping from one position to the adjacent points in any of the 4 or 8 directions, respectively. This can be done via 4 or 8 arrow-keys which allow navigating stepwise on paths and shapes.

The keys are labeled with the directions east coded by the number 0, northeast by 1, north by 2 and so on. Each path can be described by a starting point (x_0,y_0) which can be reached by the focus-key and a sequence of directions, called the chain-code. There is an analogue three dimensional extension when we consider octrees and voxels with 26 neighbors. Here, we can use the layers of a 2D stepping keyboard. Starting from a *RasterCurve* class, its parameters and methods, and the sequence of method calls, it is possible to generate and to design the concept keyboard automatically. (This will be addressed later in point 5 of this paper.)

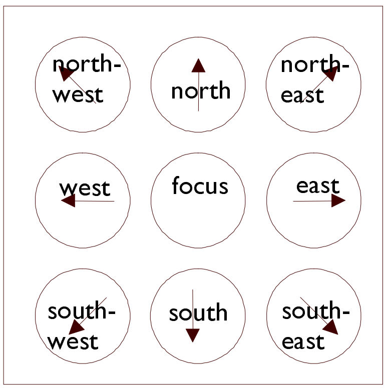

Fig. 2. Concept keyboard for 4-way or 8-way stepping

In the exercise part (see Figure 3) we ask for the number of the connected components of an 8-path in the case of the four neighbors topology where only horizontal and vertical steps coded by even numbers are allowed. It is more intuitive to use the aural representation and directly count the odd numbers within the chain-code enumeration than to derive the result from the visual representation of the path in the object area, which contains much more irrelevant information. Furthermore, the blind user can directly construct or modify raster curves such as lines, circles or ellipses by using the concept keyboard. The key in the center can be used to skip to the beginning of the curve. Correct and incorrect steps are signaled by different acoustic messages.

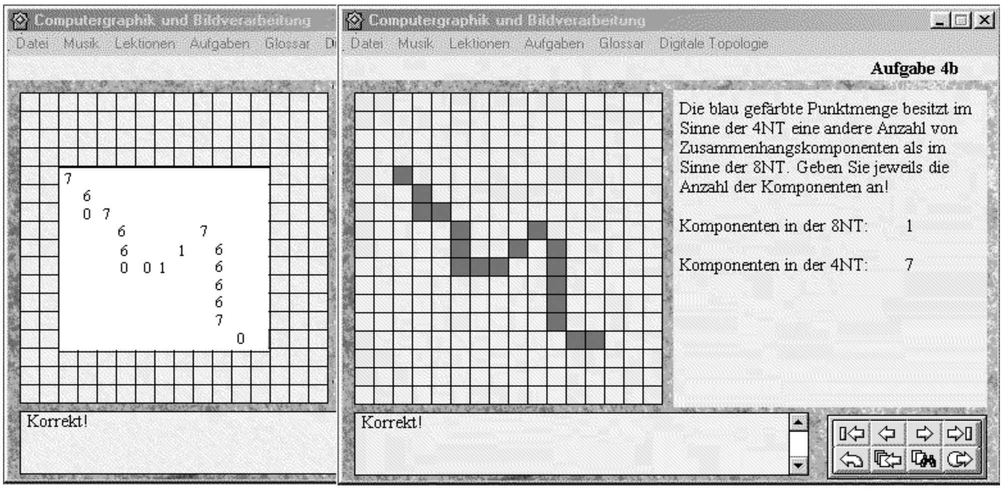

Fig. 3. Visualized aural and graphic representation of a raster curve

Now we will give further examples of how to animate standard graphic algorithms in two dimensional pixel geometry using different perception channels. There are basically two different ways to proceed. The common way is to show how the algorithm works using its logical representation, e.g., the Nassi-Shneiderman diagram or other control flow charts. Once, a pseudo code of the algorithm is given, a small window can be moved over the diagram lightening the current command line being executed. Thus, the control flow is animated and assignments, branches and loops are highlighted.

A completely different method is based on displaying the manipulated data. However, several points must be mentioned in regards to this method. First, an adapted visual representation of the data must be developed. Loop variables, flags, stacks and registers are interesting candidates for display. It is necessary to scale data in order to obtain coarser structures, to slow down the working algorithm, and to find an adequate representation of the output. We postulate an interactive user control allowing the user to step forward and backward within the algorithm, to redo steps and to modify the step width. This can be done with the concept keyboard.

Our second example is the Bresenham algorithm which digitizes lines, circles, and ellipses (see Figure 4). At the same time, the growing circle or ellipse altering the contents of typical variables can be seen. First, the growing curve is translated into appropriate intuitive sounds accompanied by spoken text concerning the contents of the variables. Another view presents the Nassi-Shneiderman diagram via text-to-speech. Finally, it is necessary to group the control elements at the bottom of the screen to simplify the navigation.

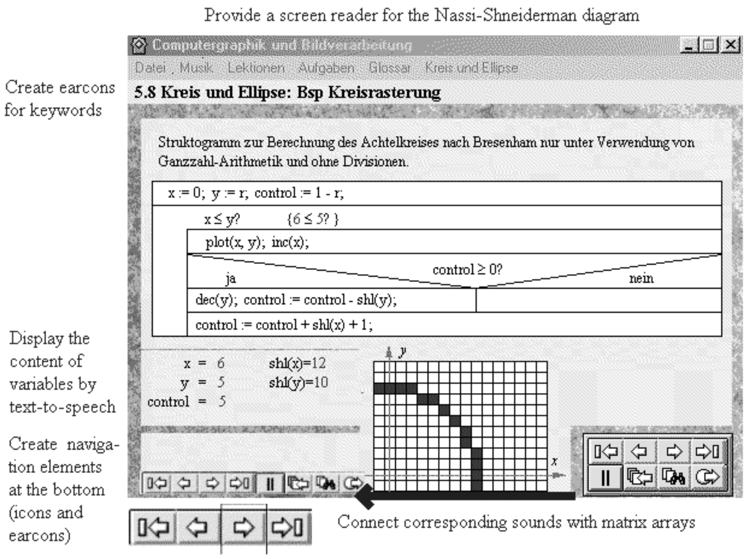

Fig. 4. Enhanced representation and interaction for the Bresenham algorithm

ViACoBi is complemented by another teaching system which presents comparable solutions for spatial data structures like bintrees, quadtrees or octrees; this system will be provided with a haptic interface in the future. These new features will be implemented in a new version of ViACoBi.

4.2 Sorting Algorithms

Usually algorithm animation maps elements on rectangles with heights proportional to sizes. Alternatively, a sorted array of numbers for a sorting algorithm can be represented by a sequence of sounds with ascending frequencies. For an example like Heapsort it is conceivable to replace the node contents with sounds and then explore the data structure using a concept keyboard, a pen or a joystick together with a tracking function and attracting nodes. The program code can be augmented by auditory messages concerning the current running procedure and its current parameter values, the values of control variables within the loops, the result of comparisons in conditions, the result of changing memory contents, and the progress in work. The last point is a very important one because the preceding ones only concern local actions without contributing to a generalized view on the algorithm or data structure. These messages can be generated automatically like some debugging information and can be displayed graphically or by text-to-speech. So our focus changes from algorithm visualization to algorithm explanation.

4.3 Graph Structures

In a graph each vertex can be complemented by an explanation concerning the content, the parent and child nodes and the global position within the graph. Outgoing and incoming edges are mentioned. There are many classical graph algorithms to construct shortest paths or spanning trees. When we want to go from point A to point B, a standard visualization tool proposes a two dimensional road map containing start and end point but also many further details. So the user must collect the relevant information, discard all roads which do not serve, and extract an adequate solution path. A better way is to preprocess the map and to use a spanning tree or only good paths near A and B. However, the simplest way is to announce through an acoustic channel a sequence of local landmarks and directions to reach point B.

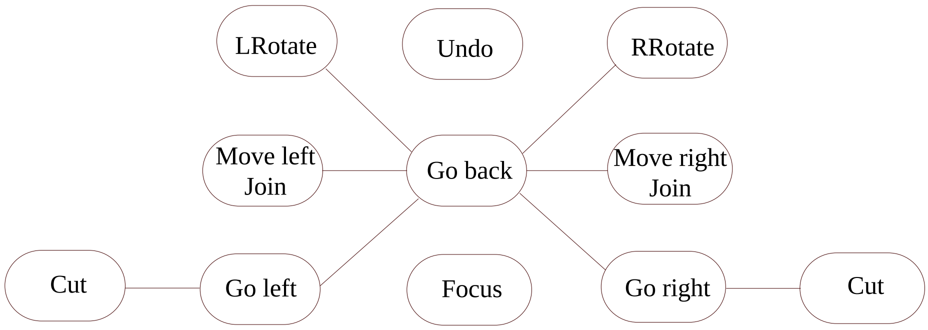

Fig.5. Concept keyboard for AVL-trees

Balancing an AVL-tree after the insertion or deletion of vertices is a problem which is addressed by several tools for algorithm animation. For instance, by using standard right or left rotations the tree can be rebalanced. To do this work and to navigate on the tree we propose the concept keyboard given in Figure 5.

4.4 Didactic Network

In their paper [2] N. Baloian et al. discuss the concept of didactic networks. In such a network, the teacher collects multimedia material which is organized as the vertices of a directed graph and presented to the students, while the teacher follows a certain instruction path. The network provides the following types of nodes: Graphic, Animation, Audio, Video, Text, Discussion, Individual and Group Work. Nodes are linked by different typed edges with labels corresponding to standard teaching methods, such as ’introduces to’, ‘refined by’, ‘explained by’, ‘exemplified by’ and ‘summarized by’. The lesson manager controls the visualization and presentation tools. Relevant parts of the network are constantly displayed to support the user. Different teaching methods lead to different weights of the edges. A spanning tree is calculated to overcome the very complex graph structure. The teacher steps through the tree visiting the most important nodes and presenting their contents. It is an interesting idea to develop appropriate concept keyboards for navigating within the network to explore the logical structures which brought to the teacher’s mind by different perception channels. The concept keyboard supports the teacher’s navigation through the learning material in a didactically coherent way according to a predefined teaching or learning strategy (see Baloian, Hoppe, Luther [3]).

4.5 Design Principles

Our considerations can be summarized in the following list of design principles which can be used to add other transmission/reception modes in parallel to classical visualization:

- Enable users to explore objects represented by digital shapes or three dimensional octree-models and neighborhoods by pointing to or grasping them in order to generate images in the mind.
- Transform graphs into acoustic structures; supply nodes with text, images, pictures, and links with captions which can be presented by text-to-speech.
- Use earcons and icons in parallel; use frequencies to represent coordinate values and different instruments for different axes.
- Provide for a graphic tablet and pen or special concept keyboards to traverse a graph structure.
- Develop for each application a suitable concept keyboard by redefining keys, creating special button schemes on a graphic tablet or introducing a new device custom-built for the application.
- Use commercial or free screen readers. They help visually impaired people navigate within information systems.
- For algorithm animation divide the screen into two parts: Place on the right side comments on the algorithm or a formal description which can be presented by text-to-speech and on the left side a visualization enhanced with acoustic motifs (directions or dimensions can be represented by different instruments and positions by tonal pitches). Thus, a natural slow down is obtained by spoken explanations.
- Introduce modern haptic interfaces like gloves or wireless ultrasonic joysticks. They provide a new feeling of three dimensional geometric data structures as was reported in [14, 21].

However, all these concepts lead to a stepwise or local processing of the model. Therefore, an important question must be answered: Is it possible to achieve a global mental image of a static or dynamic scene only from local exploration? Furthermore, we must develop evaluation strategies and best practice examples to prove that augmented interfaces and complementary perception channels provide better comprehension.

5 Ideas for Automatic Generation of the Concept Keyboard

A concept keyboard allows users to traverse a data structure and/or perform certain procedures or functions on it step by step. Ideally, the concept keyboard would be generated directly from the code implementing the data structures and algorithms over them. However, then we would too often be confronted with declarations of data structures defining fields and operations needed for implementing the whole system but making the understanding of important ideas and issues more difficult. The use of parameters in the implementation of methods for traversing or modifying the data structure may also complicate the generation of a simple but meaningful concept keyboard. To overcome these problems, we propose using an interface file which defines the data structures and procedures in a suitable way, making the generation of a keyboard easier and “cleaner”. Of course, how “clean” and meaningful (that is, how helpful for understanding the data structure and their algorithms) the resulting keyboard will be depends on how the interface file is written, but this seems to us to be unavoidable. The definition of the interface file should be made according to the following rules:

- Define the data structures containing only the relevant information which is needed to understand the general problem and its solution.
- Define a number of methods or operations having only the current node (or array element) as implicit parameter (like “this” in a Java class definition). Other nodes should not appear as parameters. Instead of this, more operations should be defined. (For example, instead of rotate(x) with x being the left or the right child, a rotateLeft() and a rotateRight() operation should be declared.)
- Associate with each operation the corresponding call to the method implementing it.

This last condition may cause the programmer to change the original code of the implementation but this should not demand too much work if the code has been reasonably written. The interface file definition language should be XML (eXtensible Markup Language), which is a meta-syntax used to declare Document Type Definitions (DTDs) for existing and new computer markup languages. The focused DTDs are intended for User Interface-oriented structural, textual, graphic, acoustic, or tactile renderings. XML promotes the creation of accessible documents which can be equally well understood by their target audience regardless of the standard or concept keyboard device used to access them. XML permits the simple yet flexible definition of structured documents. This feature makes the description of data structures and functions with this language very easy. This, added to the fact that XML is a standard, makes it the right candidate to be used as the interface definition language for concept keyboards.

6 Related Tools for Blind Users

A number of systems have been developed with the aim of being used by people with disabilities. While most systems targeted for the hearing impaired are oriented to train people by developing the necessary skills to overcome their disabilities, a considerable proportion of the systems for blind people aim to increase greatly the accessibility to current computing resources which are based on graphic user interfaces, such as games and web navigators.

For blind people screen reader software allows access to graphic user interfaces by providing navigation as well as a Braille display or speech synthesized reading of controls, text, and icons. The blind user typically uses the tab and arrow controls to move through menus, buttons, icons, text areas, and other parts of the graphic interface. As the input focus moves, the screen reader provides Braille, speech, or non-speech audio feedback to indicate the user's position. Blind users rarely use a pointing device, and as discussed above, typically depend on keyboard navigation. A problem of concern for blind users is the growing use of graphics and windowing systems [5].

The browser BrookesTalk reads out the Web page, using speech synthesis in word, sentence, and paragraph mode, and offers different views of the page to simulate ‘scanning’ [25]. Different views of the page take the form of keywords, summary or abstract and are derived using information retrieval and natural language processing techniques. BrookesTalk also offers a special search facility, a configurable text window for visually impaired users and a standard visual browser so that blind users can work alongside other people who can fully utilize a standard graphic interface.

The HOMER UIMS approach by Anthony Savidis and Constantine Stephanidis [22, 23] develops dual user interfaces for the integration of blind and sighted. HOMER supports the integration of visual and non-visual interaction objects and their relationships. In this context, a simple toolkit has been also implemented for building non-visual user interfaces and has been incorporated in the HOMER system. A Dual User Interface is characterized by the following properties: It is concurrently accessible by blind and sighted users and the visual and non-visual metaphors of interaction meet the specific needs of sighted and blind users, respectively. At any point in time, the same internal (semantic) functionality is made accessible to both user groups through different interactive channels.

Education for blind children uses special devices like a touch pad, speech synthesis or Braille displays. Another idea is to use concept keyboards which implement different actions to interact with applications. There are excellent tools for translating text to Braille; however, graphics production has problems creating tactile graphics on screen or printing it on special embossed or swell paper. A specialized thermal printer produces output on swell paper. Many modern Braille printers have a graphics mode where dots are embossed in a regular raster. It is even possible to leave the raster in order to create “smooth” curves. Other solutions print graphics directly from a program and handle the conversion to a relief image. We prefer a combination of touch, text-to-speech and sound landscapes, an access chosen in the European project TACIS (Tactile Acoustic Computer Interaction System). Together with Windows-based screen readers, Braille displays allow the visually impaired person to obtain a tactile layout of the desktop, applications, and windows [13].

AudioDoom [14, 21] allows blind children to explore and to interact with virtual worlds by using only spatialized sound. It was inspired by traditional Doom games where the player has to move inside a maze discovering the environment and solving problems posed by objects and entities that inhabit the virtual world. In doing so, it emphasizes the sound navigation throughout virtual spaces in order to develop cognitive tasks to enhance spatial orientation skills in blind children. AudioDoom is manipulated by using a wireless ultrasonic joystick or a labeled keyboard.

In the same context the sonic concentration game proposed by Roth et al. [20] contains several different levels offering pairs of geometrical shapes to be matched. To represent geometrical shapes it is necessary to build a two-dimensional sound space. Each dimension corresponds to a musical instrument and Raster points correspond to pairs of frequencies on a scale. Moving horizontally from left to right is equivalent to a frequency variation of the first instrument, and moving vertically to a frequency variation of the second one.

Another mode presents a shape by moving sound in a special plane. The third dimension can be represented by means of frequency. The Doppler effect can be used to enhance front and back differences.

Haptics is a technology that provides sensing and control through touch and gesture. It presents interface equipment for accessing and manipulating data normally available only through visual representations for seeing people. A haptic system must sense and analyze the forces applied by the user and deliver a physical sensation back to the person in real time. This kind of systems allows users to explore all sides of a virtual 3D object, to move and to turn the object freely in three orthogonal space coordinates and around three rotational axes, and to track force and to provide force-feedback sensations in multiple dimensions.

A haptic interface is a hand-held computer-controlled motorized device displaying information to human tactile and kinesthetic senses. It works bidirectionally, accepting input from the user and displaying output from the computer. Including haptics in our scenario offers a further important channel, parallel to visual and aural communication, which can substitute for the other sensorial streams, in particular graphic interfaces. For example, by producing forces on the user's hand depending on both the user's motions and properties of the icons under the cursor, touchable representations of the screen objects can be created [19]. Gloves with embedded sensors provide hand-grasping actions or whole-hand sensitivity. Realistic sounds of haptic interaction can be easily synthesized, enabling systems to convey many haptic perceptions, such as hardness, material, texture, and shape.

7 Conclusion

We have presented new concepts for enhancing standard visual interfaces with aural or haptic components to convey logical structures or data types to human minds. Optional dual or multiple interfaces enhance the human-machine interaction and support sensory-disabled people. It is important to separate control elements and visual objects by mean of an adequate concept keyboard which will be used to process data structures and geometric models. To achieve meaningful visualization, we focus on important information after a suitable abstraction process. The design guidelines given will be integrated into new versions of ViACoBi and a cooperative AudioDoom for the target groups mentioned above.

8 Acknowledgements

This research and common multi-media software development for disabled (AudioDoom for blind people and Whisper to enhance proficiency in speaking for postlingually deaf people) are being carried out by the authors in a current German and Chilean project funded by the German ministry BMBF and the Universities of Duisburg and (Santiago de) Chile [4, 14, 21, 24]. This cooperation gives us the opportunity to prove or disprove the notion that concept keyboards are useful for blind users. We thank Prof. Sánchez for valuable suggestions concerning the AudioDoom system and our collaborators Dr. W. Otten and C. Wans for their persistent valuable support during the development and evaluation of Whisper, discussing these ideas and making the material of their research available to us.

We would like to express our gratitude to our reviewers, who help us to reorganize the paper and to clarify our ideas and concepts.

References

1. Archambault, D., Burger, D.: TIM (Tactile Interactive Multimedia): Development and adaptation of computer games for young blind children. Workshop on Interactive Learning Environments for Children, Athens, Greece, March 1-3 (2000) http://ui4all.ics.forth.gr/i3SD2000/Archambault.PDF
2. Baloian, N., Pino, J., Hoppe, H. U.: Intelligent navigation support for lecturing in an electronic classroom. In Lajoie, S. und M. Vivet (Hrsg.). *Artificial Intelligence in Education – Open Learning Environments: New Computational Technologies to Support Learning, Exploration, and Collaboration.* Amsterdam et al.: IOS/Omsha (1999) 606-610
3. Baloian, N., Hoppe, H.U., Luther, W.: Structuring Lesson Material to Flexibly Teaching in a Computer Integrated Classroom. GI-Workshop der Fachgruppe "Intelligente Lehr-/Lernsysteme", Dortmund, Germany, October 8-11 (2001). Research Report 763, University of Dortmund, ISSN 0933-6192, 187-194
4. Baloian, N., Luther, W., Sánchez, J.: Modeling Educational Software for People with Disabilities: Theory and Practice. Submitted to ASSETS 2002 conference
5. Bergman, E., Johnson, E.: Towards Accessible Human-Computer Interaction, in "Advances in Human-Computer Interaction", Volume 5, Jakob Nielsen, Editor (1995)
6. Bissell, R. E.: Music and Perceptual Cognition[1]: Journal of Ayn Rand Studies, Vol. 1, No. 1, Sept. 1999. http://www.dailyobjectivist.com/AC/musicperceptualcognition6.asp

7. Blattner, M., Sumikawa, D., and Greenberg, R.: Earcons and icons : Their structure and common design principles. Human-Computer Interaction 4, (1) (1989) 11-44
8. Brewster, S.A.: Using earcons to improve the usability of a graphics package. HCI'98, People and Computers XIII, Sept. 1-4 (1998) Sheffield Hallam University, Sheffield, UK
9. Diehl, St., Görg, C., and Kerren, A.: Preserving the Mental Map using Foresighted Layout. Proc. Eurographics – IEEE TCVG Symposium on Visualization, VisSym, 28-20 May 2001, Ascona, Switzerland
10. Janser, A.: Ein interaktives Lehr-/Lernsystem für Algorithmen der Computergraphik. In: Schubert, Sigrid (Hrsg.): Innovative Konzepte für die Ausbildung, Informatik aktuell, Springer, Berlin (1995) 269-278
11. Janser, A.: Entwurf, Implementierung und Evaluierung des interaktiven Lehr- und Lernsystems VIACOBI für die Visualisierung von Algorithmen der Computergraphik und Bildverarbeitung. Logos, Berlin (1998) ISBN 3-89722-065-2
12. Kramer, G. (ed.): Auditory Display - Sonification, Audification, and Auditory Interfaces. Addison-Wesley (1994)
13. Lange, Max O.: Tactile Graphics - as easy as that. CSUN's 1999 Conference Proceedings. March 15-20, Los Angeles, USA. http://dinf.org/csun_99/
14. Lumbreras, M., Sánchez, J.: Interactive 3D Sound Hyperstories for Blind Children. *CHI '99*, Pittsburg PA, USA (1999) 318-325
15. Luther, W.: Algorithmus-Animation in Lehr- und Lernsystemen der Computergraphik. Diehl, St., Kerren, A. (Hrsg.). GI-Workshop SV2000. Dagstuhl, 11.-12.5.2000, 103-114
16. Mereu, S., Kazman, R.: Audio enhanced 3D interfaces for visually impaired users. Proc. ACM CHI 96 (1996) 72-78
17. Morley, S., Petrie, H., O'Neill A.-M., McNally, P.: The Use of Non-Speech Sounds in a Hypermedia Interface for Blind Users, in Edwards, A.D.N., Arato, A., and Zagler, W.L. (Eds.): 'Computers and Assistive Technology'. Proc. ICCHP'98. XV. IFIP World Computer Congress (1998) 205-214
18. Norseen, J.: Images of Mind: The Semiotic Alphabet (1996) http://www.acsa2000.net/john2.html
19. Ressler, S., Antonishek, B.: Integrating Active Tangible Devices with a Synthetic Environment for Collaborative Engineering. Proc. 2001 Web3D Symposium. Paderborn, Germany, Febr. 19-22 (2001) 93-100
20. Roth, P., Petrucci, L., Assimacopoulos, A., Pun, Th.: Concentration Game, an Audio Adaptation for the blind. CSUN 2000 Conference Proceedings, 20-25 March, Los Angeles, USA (2000) http://www.csun.edu/cod/conf2000/proceedings/gensess_proceedings.html .
21. Sánchez, J., Lumbreras, M: Usability and Cognitive Impact of the Interaction with 3D Virtual Interactive Acoustic Environments by Blind Children. 3rd Int. Conf. on Disability, VR and Assoc. Technologies Alghero, Sardinia, Italy, 23-25 Sep. 2000
22. Savidis, A., Stephanidis, C.: Developing Dual User Interfaces for Integrating Blind and Sighted Users : HOMER UIMS, Chi'95 conf. proceedings, Denver, CO (1995) 106-113
23. Savidis, A., Stephanidis, C., Korte, A., Crispie, K., Fellbaum, K.: A generic direct-manipulation 3D-auditory environment for hierarchical navigation in non-visual interaction. Proc. ACM ASSETS 96 (1996) 117-123
24. Wans, Cl.: Computer-supported hearing exercises and speech training for hearing impaired and postlingually deaf. Assistive Technology Res. Series, 6 (1). IOS Press (1999) 564-568
25. Zajicek M., Powell C., Reeves C.: A Web Navigation Tool for the Blind, ASSETS'98, 3rd ACM/SIGCAPH Conf. on Assistive Technologies, Los Angeles, USA (1998) 204-206