Terminal Operation Learning Application for the Screen Reader Users

Junji Onishi
Dept. of Computer Science
Tsukuba University of Technology
Tsukuba, Japan
ohnishi@cs.k.tsukuba-tech.ac.jp

Masatsugu Sakajiri
Dept. of Computer Science
Tsukuba University of Technology
Tsukuba, Japan
sakajiri@cs.k.tsukuba-tech.ac.jp

Takahiro Miura
Graduate School of Information Science and Technology
the University of Tokyo
Tokyo, Japan
miu@iog.u-tokyo.ac.jp

Tsukasa Ono
Dept. of Computer Science
Tsukuba University of Technology
Tsukuba, Japan 305-8521
ono@cs.k.tsukuba-tech.ac.jp

***Abstract*—The role of human sight becomes increasingly important as information technology progresses. Presentations based on multimedia help to explain many kinds of situations. However, this form of communication, so convenient for persons gifted with sight, is inaccessible to the visually impaired. Thus, persons lacking eyesight are eager for a method that can provide them with access to progress in technology. In education, the presentation of lecture materials involves a lecture focused on oral explanation with the use of compensating information through information technology; however, the lecturer is unable to confirm which part of the materials the students are accessing, and conversely it is not easy to direct the students to access the part on which the lecturer is currently focusing. In this paper, we propose terminal operation learning software which provides more effective to easy access many console messages under inclusive education system.**

***Index Terms*—visual impairment; educational assistive software; special education; education system development**

I. Introduction

The Convention on the Rights of Persons with Disabilities, adopted December 2006 by the General Assembly of the UN, made acknowledgement of the rights of persons with disabilities into a duty. To that end, the Assembly took the following as their goals, with the intent of ensuring the creation of educational systems inclusive of persons with disabilities at all levels and lifelong learning: [1]

- To fully develop human potential, sense of dignity, and self-worth, and to strengthen respect for human rights, fundamental freedoms, and human diversity.
- To develop the personality, talents, and creativity of persons with disabilities, as well as their mental and physical abilities, to their fullest potential.
- To enable persons with disabilities to participate effectively in a free society.

In order to realize this right, the following must be ensured:

- That persons with disabilities are not excluded from the general education system on the basis of disability, and that children with disabilities are not excluded from free and compulsory primary education, or from secondary education, on the basis of disability.
- That persons with disabilities can access an inclusive, quality and free primary education and secondary education on an equal basis with others in the communities in which they live.
- That reasonable accommodation of the individual's requirements is provided.
- That persons with disabilities receive the support required, within the general education system, to facilitate their effective education.
- That effective individualized support measures are provided in environments that maximize academic and social development, consistent with the goal of full inclusion.

Under these conditions, a policy that would promote a special needs education for the purpose of building an inclusive education system was put forward. Many problems arose that would have to be resolved in order to realize an inclusive education. [2]–[5] For example, systems would need to include an environment for those with visual disabilities that would ensure a form of communication and maximize academic development as well as integrate methods to support those persons and materials that they may use, which necessitates the presence of teachers and staff with the ability to understand braille. At the same time, there exists the issue of the difficulty of training these specialized personnel. Additionally, although the development of information technology has led to the creation of many tools to facilitate specialized education methods and materials, even in an inclusive educational environment there would be those who would not be able to use them or, depending on the situation, it would be possible that these tools are not adapted to use by others; potentially there would be some circumstances in which teachers would be required to have some specialized knowledge in order to make use of

978-1-4799-8697-2/15 $31.00 © 2015 IEEE
DOI 10.1109/SMC.2015.410

these tools. The education system of the future should require a minimum of specialized skill or knowledge while being able to present information that may be received by anyone.

That is why we began conducting research into the development of an education system that may be used by everyone, students and teachers alike, without the need for specialized knowledge or technical training. Fig.1 displays our research phase to this point. The primary factor inhibiting the visually impaired is visual information. In the field of education, visual information has the advantage of being the most effective way of getting students to comprehend material. In a presentation, slides present visual information to facilitate understanding in learners. These methods may not be used with persons with visual disabilities, however. In the majority of cases, information must be conveyed in written form [6]. This affects teaching methods in that if visual information cannot be used, education must make use of materials that are either tactile or written. Considering an inclusive education in which sighted persons are educated in the same environment, to those who are sighted, understanding the topics covered in class without the use of visual information is extremely difficult. Even if written and visual information are presented together, the visual information may be understood immediately, while the written information needs to be read in full before its contents are understood, which would lead to a gap in the amount of time it takes to convey information to different students. In particular, demonstratives such as "this" and "that" do nothing for students with visual disabilities, so the teacher often needs to adopt a specialized way of speaking. With these problems in mind, we began with the development a system that could, for instance, instantaneously convey the written information from a slide, and progressed to the development of a graphical display system that could make use of a device with a tactile display. The result of this was the realization of a method that allows the information shown by teachers to be represented in devices with braille and tactile displays held by the students, and also to transmit a simultaneous facsimile of the visual information displayed on slides. We observed that the use of this system allows students with visual disabilities to grasp information more quickly, and confirmed that it would be effective as a tool for educational support. We are currently researching remote support that would combine these methods with a guide using a kinesthetic device for the understanding of more complex visual information.

We must also expand the range of work in which persons with visual disabilities may make full use of their abilities so that they may lead more active lives in society. Although the development of information technology has led to an increase in the number of places where they may work, we still cannot say that this process is complete. We have ourselves helped in the education of persons with visual disabilities who were intent on entering the field of information technology and sent them to work at numerous IT companies, but the reality remains that they face limits even within the limited range of fields available to them. That is why we began developing educational support tools that would help persons with visual disabilities to become systems engineers and in particular to operate a console to set up network devices and manage servers. We came to develop software that would aid in the expansion of their range of work by allowing them to learn how to make use of the technology precisely. Looking into methods for persons with visual disabilities to operate a console, we discovered that there were already a number in existence.

One of these is to use an application called BRLTTY [7]–[11] that allows the output of a Linux console to be converted into output on a braille device. BRLTTY is a daemon that provides persons with visual disabilities access to a console through the use of a braille display. Running a braille display provides a function that reflects what is displayed on-screen. Making use of this function allows information to be grasped via audio or touch by sending the written information on a console screen to a screen reader or a braille display, and it is used by persons with visual disabilities who use Linux.

Other methods include the use of screen readers such as Jaws in conjunction with existing terminal software such as Tera Term. All of these methods provide a certain amount of control, but present the following issues:

- They require the use of a specific OS to run.
- Even if the console information displayed in a meaningful form as visual information can be easily read or output on a braille display, this does not always mean that its meaning will be easily grasped.

Most methods operate on the basis of simply using the current display as-is, and converting that into a form that may be output in a different medium. Words and speech constitute a dimension that is fundamentally separate from images, so the direct conversion of images cannot be said to work well at all. Our examination of terminal software for this paper revealed that while it is certainly possible to confirm the results of command input by having it read or using a braille display, when a variety of commands are used, it becomes difficult to discern where one output ends and the next begins, which is frequently indicated as a source of extreme difficulties in confirming the contents of a given screen.

For example, students point to the following disadvantages:

- All of the text is read, but it is only read a single time, making it difficult to grasp in its entirety.
- When information is missed, the arrow keys may not be used to go back in order to confirm it.
- When there is a great deal of output, it is extremely difficult to precisely grasp the desired information.

In order to resolve these issues, we developed an experimental terminal interface learning software program that would be tailored to those with visual disabilities while also being designed such that it would not lead to a gap in control for sighted people, and the results of our trial are described in this paper.

II. Designing the Software

A. System Configuration

The system configuration for our prototype is shown in Fig.2. Using normal terminal software, the application should

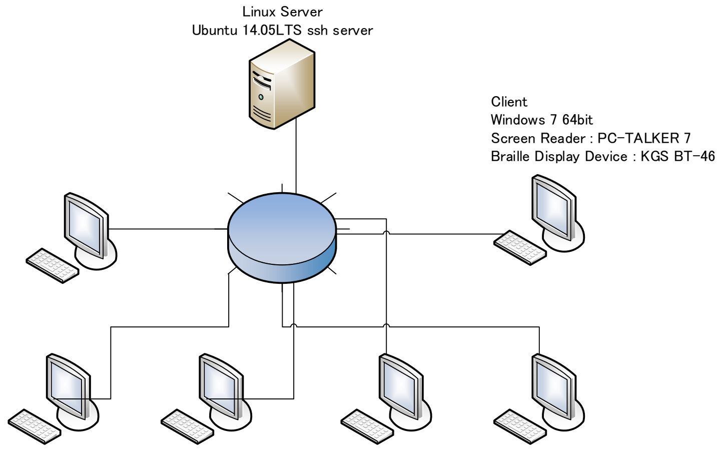

Linux Server
Ubuntu 14.05LTS ssh server
Client
Windows 7 64bit
Screen Reader : PC-TALKER 7
Braille Display Device : KGS BT-46

Fig. 2. Overview of our developed system components.

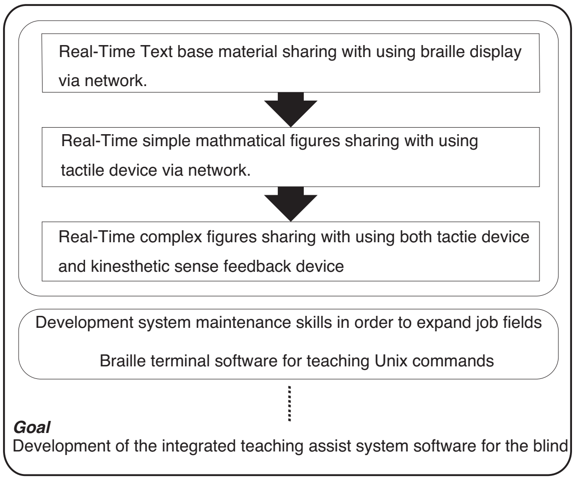

Real-Time Text base material sharing with using braille display via network.
Real-Time simple mathmatical figures sharing with using tactile device via network.
Real-Time complex figures sharing with using both tactie device and kinesthetic sense feedback device
Development system maintenance skills in order to expand job fields
Braille terminal software for teaching Unix commands
Goal
Development of the integrated teaching assist system software for the blind

Fig. 1. The strategy of our study for the development of teaching support software for the blind.

be connected to the same network segment as a server that has had a version of Linux OS prepared for the connection installed. To use the application, the student runs the program, which will automatically connect to the server using SSH protocol and switch to console operation. The experimental software has a dedicated command entry text box, and when a command is entered, the results are given in the form of a list, designed such that the message on each line may be confirmed either on a braille display or by screen reading.

The computers and braille displays used with the experimental software are as follows: Client Operating System
OS: Japanese Language Windows 7/8/8.1
Screen Reader: PC Talker 7/8, from Kochi System Development
Other Specifications: An environment capable of running Windows is required.
CPU: A minimum 1GHz processor that supports PAE, NX, and SSE2.
Memory: 1GB (32-bit) or 2GB (64-bit)
Free Hard Drive Space: 16GB (32-bit) or 20GB (64-bit)
Graphics Card: Microsoft DirectX 9 graphics device (with WDDM driver)
Framework: .Net Framework 4/4.5/4.5.1/4.5.2
Braille Display
From KGS: Braille Tender BT46
Main Specifications
Braille Display: 46 boxes with 8 protruding pins each
Connection: USB

B. Development Environment

Development of the software described in this paper took place in the following environment: Development Tool
Visual Studio 2013 C#

TABLE I
VISUAL STATUS OF EACH STUDENT INVOLVED IN THE SOFTWARE TRIAL LECTURE

Students	Visual status
A	Blind
B	Blind
C	Readable font size : 14pt.
D	Readable font size : 12pt.
E	Readable font size : 16pt.
F	Blind
G	Blind

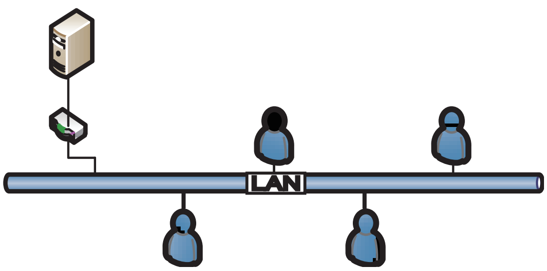


Fig. 3. Network environment

Development Environment OS
Windows 8 64-bit
We confirmed that the software we developed could run on the 64-bit version of either Windows 7 or 8. In order to use a braille display and screen reader, it is necessary to install PC Talker 7/8.
SSH Communication API Module
SharpSSH [13]

Provides an API that can communicate with the SSH server application running on the server which may be combined with any .Net application. We used it with the software we developed in order to establish and manage an encrypted communication channel between the server and the client.

C. Software Functions

The functions of our experimental software are as follows:

1) To convey the text information of commands and output on a braille display.
2) To include numbers for each line in a list in order to identify a given line so that any past output results may be easily accessed.
3) To allow the use of the following key combinations for basic console operation:
   - CTRL + C (Cancel)
   - CTRL + Z (Abort)

III. Results of Trial for Experimental Software

From our network programming classes, eight students with visual disabilities, ages 21-24, whose visual statuses are as printed in table I, were instructed in command operation in a Unix environment, using the experimental software we had developed, and we studied the effects. Fig.3 shows an overview of the network environment at the time of the experiment. The following commands were used as operations:

- File operation commands such as ls, cd, and touch
- Network operation commands such as ifconfig, tcpdump, ping, netstat, and traceroute
- Compiling and execution of C programs using console output using GCC

First, all of the students were able to run the program and log into the Unix server without any problems, and they were also all able to understand that the program was waiting for command input. Fig.4 shows a screenshot of the time immediately after the program started. All of the students correctly understood that line 22 of this screen indicated that the program was waiting for command input. All of the students also correctly understood the output messages from the login on lines 1-20. Next, as the area for inputting commands is, in contrast to conventional terminal software, set up separately, the students were able to clearly discern the command input and transmit commands successfully. All of the students were also able to comprehend the resulting output accurately. All of the students were also able to transmit commands precisely when directed by their instructors, and succeeded in carrying out all of the commands without any individual support. The first reactions of the students were positive; all of the students were able to grasp the material easily, and they commented that the impact of using the software for the first time was immense.

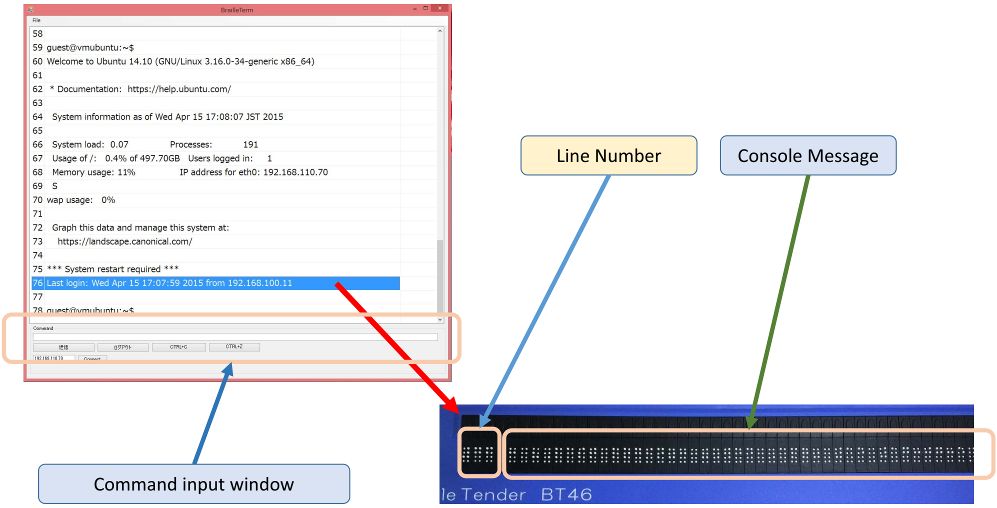
Following a brief trial operating the program, students were asked to comment on the experimental software, and the responses we collected were as follows:

(1) It was considered easy to use in that:

- Being able to read the numbering for each line was very convenient, allowing one to check the necessary output information quickly.
- It was very convenient to be able to check the output results as many times as necessary while following the log using the screen reader.
- The application is able to switch its focus to the output screen after a command has been carried out, which made confirming the result simple.
- Students often felt the various limitations of conventional terminal software, but this experimental software had none of those problems and the screen reader function worked effectively.

(2) Useful functions to add in the future

- A word completion function for the entry mode would be convenient.
- It would be better if messages would not be cut off midway when displayed in list form.
- A function that could display and check output results and command log by category.
- As the screen reader and the braille display were prioritized, the location of the control code transmit button is difficult to find, and we feel that there is no real need for the button itself.



```
58
59 guest@vmubuntu:~$
60 Welcome to Ubuntu 14.10 (GNU/Linux 3.16.0-34-generic x86_64)
61
62  * Documentation:  https://help.ubuntu.com/
63
64   System information as of Wed Apr 15 17:08:07 JST 2015
65
66   System load:  0.07              Processes:           191
67   Usage of /:   0.4% of 497.70GB   Users logged in:     1
68   Memory usage: 11%               IP address for eth0: 192.168.110.70
69   S
70 wap usage:   0%
71
72   Graph this data and manage this system at:
73     https://landscape.canonical.com/
74
75 *** System restart required ***
76 Last login: Wed Apr 15 17:07:59 2015 from 192.168.100.11
77
78 guest@vmubuntu:~$
```

Fig. 4. Screenshot of developed software

- Making it possible to copy error messages in the output results to the clipboard would make it easier to deal with errors.
- Assigning shortcut keys to the most frequently used commands would increase program usability.

We were able to collect the useful comments above for future improvements to the program. The overall opinion was that the application was easier to use than conventional terminal software. Because of the improvements to the operating environment, the blind students who had previously worked with a Unix terminal were able to confirm the effectiveness of switching from software such as Tera Term to our experimental software.

Existing terminal software that left one to completely depend upon the screen reader function gave rise to many problems, including being unable to easily check the status of a command operation or the status of a message on the output screen, which led to the necessity of helping students with their problems individually. Among the advantages found by instructors was the fact that with this experimental software, in addition to a dramatic reduction in the number of issues requiring individual help, the list of output messages is matched with a line number index made it possible for students to easily share the contents of a specific output message with instructors during class simply by pointing them to its line number.

IV. Conclusions and Future Work

For our study, we were able to create console terminal software that would respond to screen readers and braille displays which could be effectively employed in teaching students with visual disabilities how to operate a console command terminal. Moreover, the effectiveness of this software in student learning was confirmed through trials conducted in network programming courses. With the particularly dramatic improvement of the operating environment for completely blind students, the program received a good deal of praise for the fact that it made checking output messages a simple task.

At present, the output in response to commands is displayed only as a simple list, and at times it was found difficult to understand the connections between commands and output messages when checking past messages. This led us to determine that it would be necessary to improve the interface in the future to allow features such as the sorting of output messages by the command to which they responded. We also expect to be able to develop software that would feature improved command history functions, including being able to recall past commands by category or instantaneously check the command history together with the resulting output messages, in order to further reduce the gap in console operating time between sighted persons and those with visual disabilities. Also, in order to create an educational tool that would be effective when used by teachers who are not accustomed to students with visual disabilities or who are teaching remotely, we are considering the development of a network-linked educational support tool that would allow the teacher to check students' console screens remotely in order to provide appropriate instruction. In addition, we have to improve the user interface with investigating current user interface issues of our developed software.

In reality, it is difficult to rapidly prepare an educational environment with reasonable accommodation and it is impossible to simply train teachers who specialize in special education en masse. Our most important goal was to develop a compensatory tool that would limit the specialized measures necessary to provide reasonable accommodation and that could be easily used by anyone to provide an educational environment which does not burden students with visual disabilities. As the environment we envision comes into being, students with visual disabilities will be able to realize their talents in a variety of fields, widening the range of areas in which they may contribute to society. In order for a society to effectively foster its most highly gifted people, it is crucial that it provide an equally high quality education to everyone. In order to realize this goal, we hope to contribute to the development of a tool that will contribute in a small way to the development of information technology and help in providing reasonable accommodation in a number of ways.

Acknowledgment

This research was supported in both part by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (C), 15K04540, 2015, (B) 26285210, 2015, and Tsukuba University of Technology competitive research grants.

References

[1] Convention on the Rights of Persons with Disabilities
http://www.un.org/disabilities/convention/conventionfull.shtml
last viewed 04/13/2015.

[2] Ferrell Kay, Issues in the Field of Blindness and Low-Vision
http://www.unco.edu/ncssd/resources/issues_bvi.pdf
last viewed 04/12/2015.

[3] Konecki, M.; Lovrencic, A.; Kudelic, R., ”Making programming accessible to the blinds,” MIPRO, 2011 Proceedings of the 34th International Convention , vol., no., pp.820,824, 23-27 May 2011.

[4] Li Chunlian; Sun Yu, ”Study on speech supported test system for visual impairment college students,” Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on , vol., no., pp.987,990, Nov. 30 2010-Dec. 2 2010

[5] Teaching Students with Visual Impairments in Inclusive Classrooms
https://www.duo.uio.no/bitstream/handle/10852/36642/MasterxsxThesis.pdf?sequence=1 last viewed 06/30/2015.

[6] L Stearns, R Du, U Oh, Y Wang, L Findlater, R Chellappa, JE Froehlich, ”The Design and Preliminary Evaluation of a Finger-Mounted Camera and Feedback System to Enable Reading of Printed Text for the Blind” , European Conference on Computer Vision (ECCV), Workshop on Assistive Computer Vision and Robotics, Springer, pp.615-631, June 2014.

[7] BRLTTY
http://mielke.cc/brltty/
last viewed 04/12/2015.

[8] Mario Konecki, ”GUIDL as an Aiding Technology in Programming Education of Visually Impaired”, Journal of Computers, Vol 9, No 12 (2014), 2816-2821, Dec 2014

[9] S. Mealin, E. Murphy-Hill, ”An exploratory study of blind software developers”, in Proceedings of the 2012 IEEE Symposium on Visual Languages and Human-Centric Computing, 2012, pp. 71-74.

[10] J. Snchez, F. Aguayo, ”Blind learners programming through audio”, in CHI’05 Extended Abstracts on Human Factors in Computing Systems, 2005, pp. 1769-1772.

[11] S. Alexander, ”Blind Programmers Face An Uncertain Future”, ComputerWorld, vol. 32, no. 44, pp. 86-87, 1998.

[12] Promoting Special Needs Education Based on the Philosophy of the Convention on the Rights of Persons with Disabilities
http://www.mext.go.jp/b_menu/hakusho/html/hpab201001/detail/1326863.htm
last viewed 04/13/2015.

[13] SharpSSH
http://sourceforge.net/projects/sharpssh/
last viewed 04/12/2015.