Proceedings of the 2007 IEEE
International Conference on Mechatronics and Automation
August 5 - 8, 2007, Harbin, China

Finger Braille Teaching System for People who Communicate with Deafblind People

Yasuhiro Matsuda and Tsuneshi Isomura
Department of Robotics and Mechatronics
Kanagawa Institute of Technology
Atsugi-shi,Kanagawa, Japan
yasuhiro@rm.kanagawa-it.ac.jp

Ichiro Sakuma and Etsuko Kobayashi
Graduate school of Engineering
The University of Tokyo
Bunkyo-ku, Tokyo, Japan

Yasuhiko Jimbo and Tatsuhiko Arafune
Graduate school of Frontier Sciences
The University of Tokyo
Kashiwa-shi, Chiba,Japan

Abstract **- Finger Braille is one of tactual communication media of deafblind people. In Finger Braille, index finger, middle finger and ring finger of both hands are likened to keys of a Braille typewriter. A sender dots Braille code on the fingers of a receiver like whether he/she does the type of the Braille typewriter. Then the receiver recognizes the Braille code. Deafblind people who are skilled in Finger Braille can catch up with speech conversation and express various emotions. Because there are small non-disabled people who are skilled in Finger Braille, deafblind people communicate only with interpreters. In this paper, we developed a Finger Braille Teaching System and designed a teaching interface which taught clauses explicitly. The Teaching System recognized non-disabled people's speech and converted to Braille code. By parsing the Braille code, the Teaching System retrieved clause information and segmented the Braille code into clauses. Then the dot pattern of the Braille code was displayed. By observing the dot pattern, non-disabled people dotted Finger Braille to deafblind people. An evaluation experiment between a blind person who was skilled in Finger Braille and two non-disabled people who were non-skilled in Finger Braille was carried out. The results showed that the fundamental functions (speech recognition, conversion to Braille code and clause segmentation) were practicable; the non-disabled senders could dot Finger Braille accurately and communicate with the blind receiver directly. Therefore it was considered that the Teaching System was effective.**

Index Terms - deafblind, Finger Braille, teaching system, communication aid.

I. INTRODUCTION

Recent surveys (The Deafblind Association of Japan, 2006) estimate that there are 16,354 deafblind people in Japan. Communication is one of the largest barriers to their independent living and participation. Deafblind people use many different communication media according to the age of onset and what resources are available to them. "*Yubi-Tenji*" or Finger Braille is one of tactual communication media of deafblind people (see Fig.1). In Finger Braille, index finger, middle finger and ring finger of both hands are likened to keys of a Braille typewriter. A sender dots Braille code on the fingers of a receiver like whether he/she does the type of the Braille typewriter. Then the receiver recognizes the Braille code. Deafblind people who are skilled in Finger Braille can catch up with speech conversation and express various emotions, because of prosody of Finger Braille [1]. Because there are small non-disabled people who are skilled in Finger Braille, deafblind people communicate only with interpreters. Thus the participation of deafblind people is greatly restricted.

Recently, some Braille input devices were developed [2], [3]. These devices let deafblind people wear gloves or an keyboard to input his/her Finger Braille and actuators to output non-disabled people's speech converted to Finger Braille. In such supporting devices, deafblind people are burdened with wearing the sensors and the actuators, and must master new communication system with the supporting devices.

Objective of this study is development of a Finger Braille supporting device which respected skin contact communication, because skin contact is only a non-verbal communication for deafblind people. Fig. 2 shows the concept of the Finger Braille supporting device. The largest features of this study are that deafblind people and non-disabled people who are non-skilled in Finger Braille are communicate by usual Finger Braille and all sensors are worn by non-disabled people. This supporting device includes two assistive systems; one is a Teaching System, the other is a Recognition System. The Teaching System recognizes non-disabled people's speech and displays the dot pattern of Finger Braille. Non-disabled people dot Finger Braille by observing the dot pattern [4]. The Recognition System senses deafblind people's Finger Braille and converts to speech [5]. Thus deafblind people don't have to aware of the supporting device and may communicate by usual Finger Braille without interpreters.

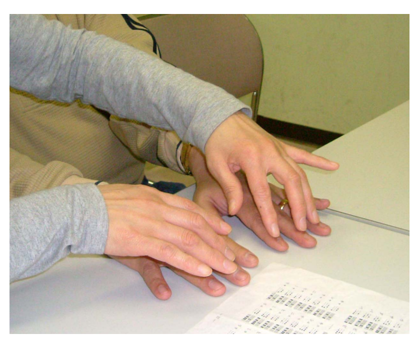


Fig. 1 Finger Braille

This study was supported by the Ministry of Education, Culture, Sports, Science and Technology of Japan under a Grant-in Aid for Scientific Research (No. 16700430).

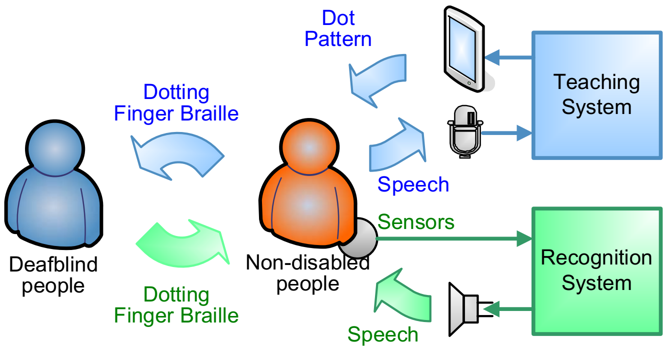


Fig. 2 Concept of Finger Braille supporting device

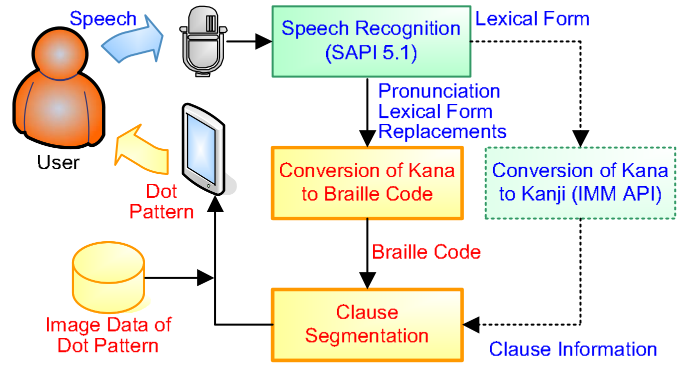


Fig. 3 Configuration of Finger Braille Teaching System

In this paper, we developed the Finger Braille Teaching System and designed a teaching interface which taught clauses explicitly. And an evaluation experiment between a blind person who was skilled in Finger Braille and two non-disabled people who were non-skilled in Finger Braille was carried out.

II. Materials and Methods

A. *Japanese Braille system and Finger Braille*

Features of Japanese script are: (1) the Japanese sentence is usually written with a combination of Kanji characters and Kana characters; (2) the Japanese sentence doesn't have space between words. Japanese Braille system is different from Japanese script in some points. The differences are: (1) Japanese Braille system consists only of Kana characters; (2) particles [ha] and [he] are described with their pronunciations [wa] and [e]; (3) long vowels [ū] and [ō] are described with their pronunciations [-]; (4) the Japanese Braille sentence has space between clauses (Bunsetsu unit).

In Finger Braille, a sender dots Braille codes directly on fingers of a receiver. A rule of Finger Braille is that the sender keeps on touching fingers of the receiver without dotting, because the receiver is uneasy without touching or any cues. Prosody of Finger Braille helps the receiver recognize dotted Braille codes. Features of prosody of Finger Braille are: (1) characters at the end of clauses are dotted long; (2) characters at the end of sentences are dotted long and strongly; (3) the sender pauses between clauses; (4) the sender must not pause during a clause.

B. *Configuration of the Teaching System*

Fig. 3 shows configuration of the Finger Braille Teaching System. First, speech recognition (SR) engine recognized sender's speech. Second, by using results of speech recognition, the Teaching System converted Kana characters to the Braille code. Third, by parsing Braille code, the Teaching System retrieved clause information and segmented the Braille code into clauses. Finally, the Teaching System displayed dot pattern of the Braille code. The development environments were as follows.

Tablet PC: HP TC1100 (CPU Pentium M 1.1GHz, RAM 1024MB, LCD 10.4 inches XGA)

OS: Microsoft Windows XP.

Programming language: Microsoft Visual Basic 6, LPA WIN-PROLOG 4.500

SR engine: Microsoft Speech SDK (SAPI5.1).

1) Speech recognition: Because the sender must keep on touching fingers of the receiver, speech recognition is suitable for input interface. First, the Teaching System created and loaded dictation grammar of SAPI5.1, and then SAPI5.1 was ready to recognition. When a sender spoke into a microphone, SAPI5.1 attempted to recognize it. Following successful speech recognition, the Teaching System retrieved results of speech recognition. Sender could train SAPI5.1 by a speech training wizard. After the speech training, SR engine could perform better and improve SAPI's personalization experience.

2) Conversion of Kana to Braille code: Table I shows an example of Braille code and results of speech recognition. Because Japanese Braille system consists only of Kana characters, pronunciation and lexical form of phrase elements are suitable for converting to the Braille code. The Teaching System checked each character of the pronunciation and the lexical form of the phrase elements and adopted the suitable character for Japanese Braille code. The converting rules were: (1) generally, the lexical forms of the phrase elements were adopted; (2) in particles [ha] and [he], the pronunciations of the phrase elements were adopted; (3) in long vowels [ū] and [ō], the pronunciations of the phrase elements were adopted.

TABLE I
Example of Braille Code and Results of Speech Recognition

Input Speech	お姉さんは学校へ行きました. (My sister went to school.)
Braille Code	おねえさんわ がっこーえ いきました. 0 nee san wa gak ko- e i kimashita.
Results of speech recognition	
Get Text	お姉さんは学校へ行きました
Pronunciation	お ねーさん わ がっこー え い き ました 0 ne- san wa gak ko- e i ki mashita
Lexical Form	お ねえさん は がっこう へ い き ました 0 nee san ha gak kou he i ki mashita
Display Text	お 姉さん は 学校 へ 行 き ました

3) Clause (Bunsetsu) segmentation: To realize non-disabled people's prosodic dotting, the dot pattern of clauses was displayed explicitly. SAPI5.1 can not retrieve clause information of result of speech recognition. Thus we developed a Braille code parser applying a technology of natural language processing (BUP system) [6]. The Braille code parser parsed the Braille code and segmented into clauses (insert space between the clauses). The Braille code parser consisted of dictionary, grammars of Braille code, BUP translator and control program. The dictionary and the grammars were described in definite clause grammars (DCG). The BUP translator translated the dictionary and the grammars into Prolog program. The control program controlled execution of parsing.

If the Braille code was not grammatical because of misrecognition of SR, the Braille code parser could not parse it. As a backup of the Braille code parser, we used Microsoft Global IME (Japanese) (IMM API). The Teaching System set the lexical form of the phrase elements as the reading string of the composition string of IMM API and directed IMM API to convert the composition string. Then the Teaching System retrieved the clause information of the lexical form of the phrase elements and inserted space between the clauses of the Braille code.

4) Design of teaching interface: Finally, the Teaching System displayed the dot pattern of the Braille code by reading from image database of dot pattern. Fig. 4 shows a teaching interface that we designed. The Braille code was displayed in upper text box. The dot pattern was displayed in fourteen picture boxes (two columns and seven rows). The first clause was displayed on the left column (from upper left to lower left) and the second clause was displayed on the right column (from upper right to lower right). The third clause was displayed on the next page. A clause not more than seven characters was displayed on one column and a clause more than seven characters was displayed on two columns. After a clause was displayed on the left column, the next clause was displayed on right column or the next page. Thus the dot pattern of clauses was displayed explicitly on columns.

Red pattern indicated left hand and blue pattern indicated right hand. We designed two kinds of presentation method for the sender's experience. Presentation method A only displayed the dot pattern and presentation method B displayed the dot pattern on illustration of fingers. Presentation B was more symbolic and easy to recognize dotting fingers for beginners. Presentation method A was the most simplified sign and suitable for experienced people.

5) Editing Braille code: Because the sender keeps on touching fingers of the receiver at least one hand, we installed a trackball (Kensington Expert Mouse USB/PS2) to operate the Teaching System by left hand and allocated six functions to keys of trackball (see Fig. 5). If the Braille code included any mistakes because of misrecognition of SR, sender could edit Braille code by using the track ball and a software keyboard (see Fig. 6).

Presentation method A Presentation method B

Fig. 4 Two kinds of presentation methods of teaching interface. Displayed dot pattern is "Oneesanwa / gakko-e {My sister / to school.}"

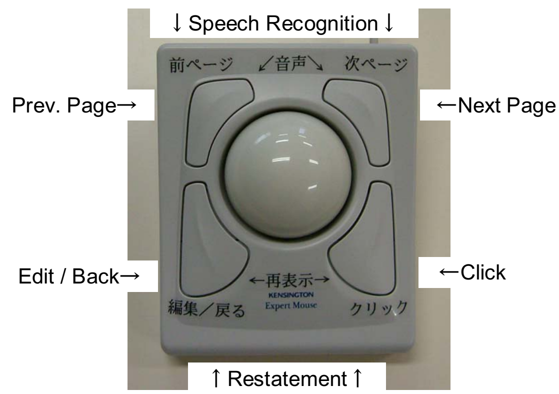
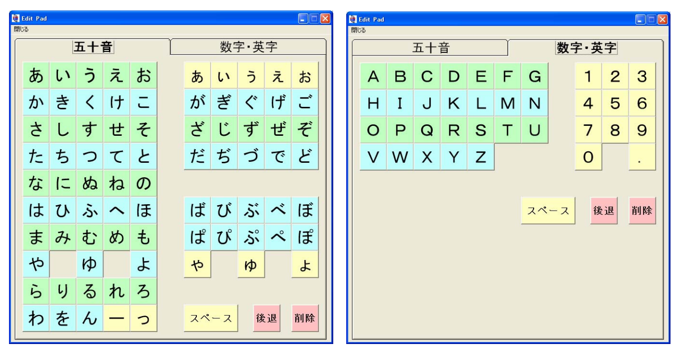


Fig. 5 Trackball to operate Teaching System



Japanese characters Alphabets and numerals

Fig. 6 Software keyboard to edit Braille code

C. *Evaluation experiment*

1) Objectives of evaluation experiment: Objectives of the evaluation experiment were: (1) to evaluate accuracies of fundamental functions (speech recognition, conversion to Braille code and clause segmentation); (2) to evaluate accuracy of dotting and recognition; (3) to evaluate operation time during communication.

2) Subjects: Receiver was a blind person who was skilled in Finger Braille (sending experiment: 20 years, receiving experiment: 8 years). Senders were two college students who were non-skilled in Finger Braille (M1/F1). All senders reported normal hearing and vision abilities and were native Japanese speakers. All subjects gave informed consent after hearing a description of the study.

3) Dialogues: Four daily conversations in Japanese textbook for foreign beginners [7]. (Total 51 sentences, 143 clauses, 288 words, 686 characters)

4) Procedure: Senders were instructed to operate the Teaching System, and trained SAPI5.1 by speech trainings "Introduction" and "Introduction of speech technology". Receiver wore earplugs and headphones and heard white noise to insulate with sound during experiment. In experiment, one sender pushed key of speech recognition and spoke one sentence of the dialogues. If result of speech recognition was correct, the sender dotted Finger Braille on fingers of the receiver by observing the teaching interface. If the result of speech recognition was not correct, the sender spoke the same sentence again or edited the Braille code and pushed key of restatement. Then the sender dotted Finger Braille on fingers of the receiver. The receiver spoke recognized sentence. If the receiver misrecognized, the sender dotted the same sentence again. Thus the sender repeated about the all dialogues.

In session 1, sender 1 (female) dotted conversation 1 displayed by presentation method B. In session 2, sender 1 dotted conversation 2 displayed by presentation method A. In session 3, sender 2 (male) dotted conversation 3 displayed by presentation method B. In session 4, sender 2 dotted conversation 4 displayed by presentation method A. The senders were directed: (1) to dot as accurately as possible; (2) to dot long in characters at the end of clauses and sentences; (3) to keep on touching the fingers of the receiver at least right hand during experiment.

All experiments were recorded by a digital video camera. History of operations by the senders and all recognized speeches were recorded in hard disk drive of the Teaching System. The receiver put fingers on pressure sensor sheets (Nitta Tactile Sensor System) which measured change of pressure by dotting.

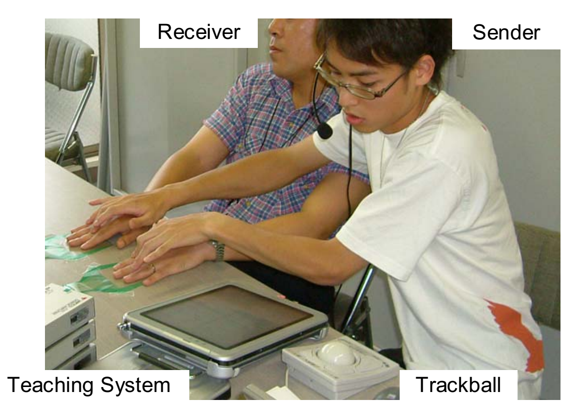


Fig. 7 An experiment

III. Results

A. Accuracies of fundamental functions

1) Accuracy of speech recognition: The words of dialogues were classified into interjections (15 words), proper nouns (14 words) and the other words (259 words). As the evaluation of accuracy of speech recognition, error number of substitutions in each part of speech and error number of deletions were counted. Fig. 8 shows error ratios of speech recognition. As the results, 7% of proper nouns (1 word) and 5% of the other words (13 words) were substituted and 0.7% of all words (2 words) were deleted. Because 47% of interjections (7 words) were substituted, there was a difficulty to recognize interjections by SAPI5.1 [4]. Then *Correct Ratio* of speech recognition was calculated. *Correct Ratio* is

$$Correct\ \ Ratio = \frac{N - sub - del}{N} \times 100\ \% \qquad (1)$$

where N is the number of words, sub is the error number of substitutions, and del is the error number of deletions. *Correct Ratio* was 92.0% of all words and 94.4% except substitution of interjection (see Fig.9).

Because of misrecognition of SR, the senders re-spoke 5 sentences (8 times) and edited 15 sentences (56 characters).

2) Accuracy of conversion to Braille code: All results of speech recognition were accurately converted to Braille codes. Because the senders edited Braille codes wrong, 0.4% of all dialogues (3 characters) were not correct Braille codes. Thus accuracy of conversion to Braille code was 99.6% (see Fig.9).

3) Accuracy of clause segmentation: When the results of speech recognition were correct, all grammatical Braille codes were accurately segmented into clauses by Braille code parser. When the results of speech recognition were not correct, 3.5% of clauses (5 clauses) were slipped their segmentation points by the backup of Braille code parser (IMM API). When the senders edited Braille codes, 1.4% of clauses (2 clauses) were deleted spaces. Thus accuracy of clause segmentation was 95.1% (see Fig.9).

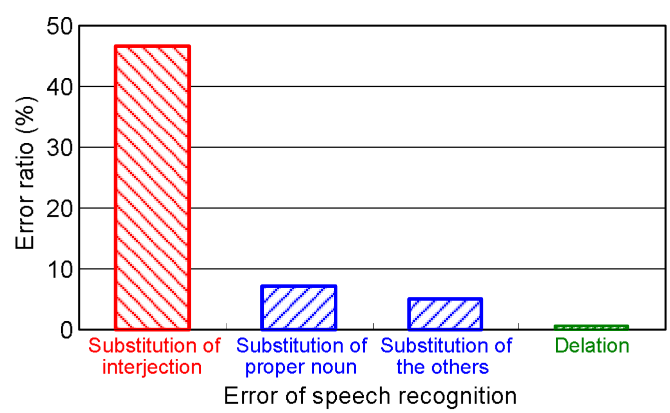


Fig. 8 Error ratios of speech recognition

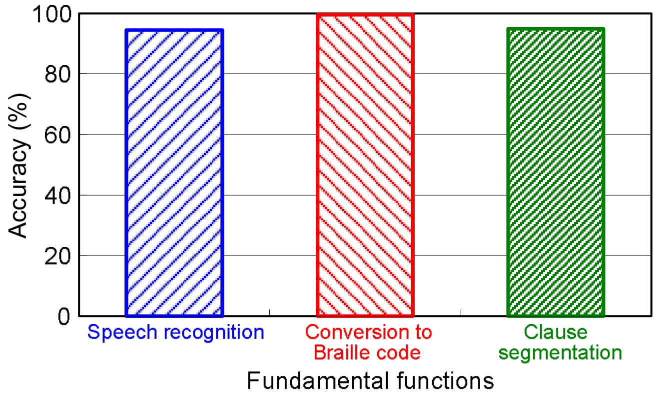


Fig. 9 Accuracies of fundamental functions

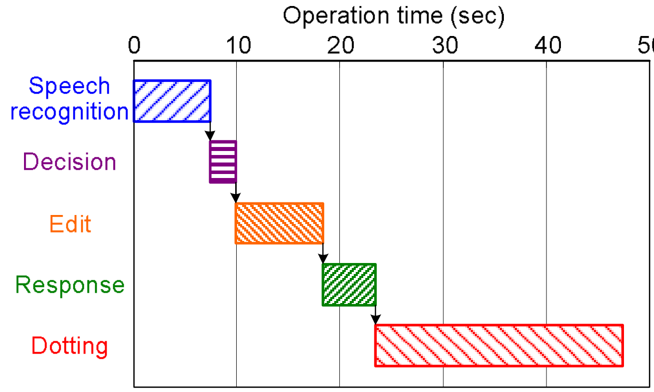


Fig. 11 Distribution of the operation times

B. *Accuracies of dotting and recognition*

Error ratio of dotting by the senders was only 1.2% of all characters (8 characters). The receiver could not recognize 7.8% of dialogues (4 sentences) because of error of dotting. As the senders re-dotted the same sentences, the receiver could recognize them. Thus accuracy of dotting by the senders was 98.8% and accuracy of recognition by the receiver was 92.2% in the first dotting and 100% in the re-dotting (see Fig.10).

C. *Operation time*

An operation time was divided into five times as follows. These operation times were calculated by the history of operations and the video images. Fig. 11 shows distribution of the operation times.

1) Speech recognition time: Speech recognition time meant the time until the dot pattern was displayed, after the sender pushed the key of speech recognition (including time of re-speech). Mean of speech recognition time was 7.4 sec (S.D.=8.2) and mean of speech recognition time per speech was 5.7 sec (S.D.=2.9).

2) Decision time: Decision time meant the time until the sender pushed the key of edit, after the dot pattern was displayed. Mean of decision time except non-edited 36 sentences was 8.5 sec (S.D.=2.8). (Mean of decision time of all sentences was 2.5 sec.)

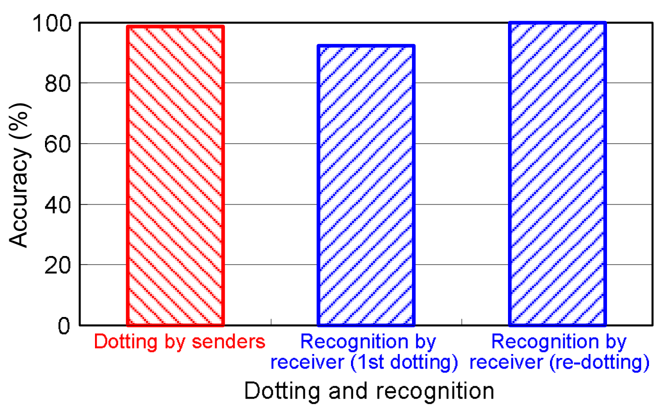


Fig. 10 Accuracies of dotting and recognition

3) Edit Time: Edit time meant the time until the sender pushed the key of restatement, after the sender pushed the key of edit. Mean of edit time except non-edited 36 sentences was 28.5 sec (S.D.=26.1) and mean of edit time per character was 8.1 sec (S.D.=4.3). (Mean of decision time of all sentences was 8.5 sec.)

4) Response Time: Response time meant the time until the sender started dotting, after the dot pattern was displayed. Mean of response time was 5.1 sec (S.D.=2.5).

5) Dotting time: Dotting time meant the time until the sender finished dotting, after started dotting (including time of changing page and re-dotting). Mean of dotting time was 23.9 sec (S.D.=19.3) and mean of dotting speed was 48.3 characters/min (S.D.=25.3).

6) Total communication time: Total communication time meant the time until the sender finished dotting, after the sender pushed the key of speech recognition (including time of re-speech, edit, changing page and re-dotting). Mean of total communication time was 47.7 sec (S.D.=39.8) and mean of total communication speed was 23.1 characters/min (S.D.= 10.3).

IV. Discussion

A. *Accuracies of fundamental functions*

As accuracies of fundamental functions, *Correct Ratio* was 94.4% except substitution of interjection. Accuracy of conversion to Braille code was 99.6%. Accuracy of clause segmentation was 95.1%. It was confirmed that these results corresponded to former study [4], and the fundamental functions were practicable. As for editing Braille code, the senders were puzzled at the first time but they could edit smoothly after the second time.

B. *Accuracy of dotting and dotting speed*

In this experiment, error ratio of dotting by the senders was only 1.2% of all characters and mean of dotting speed was 48.3 characters/min (S.D.=25.3).

For comparison, Amemiya et al. developed a Braille input device [3]. In an evaluation experiment, five non-disabled people who had no experience using Braille typewriter were given a sheet of paper with list of Braille codes and inputted as quickly and accurately as possible. As a result, error ratio was

8-6.76% (session 1-5) and input speed was 20-35.4 characters/min (session 1-5).

Thus, as for non-disabled people, dotting Finger Braille using the Teaching System was more accurate and quicker than inputting Braille code using the sheet of paper with list of Braille codes.

An et al. carried out an experiment that ten visually impaired people who just started to learn Braille codes inputted Braille code using a Braille typewriter [2]. As a result, error ratio was $2.8 \pm 2.3\%$ and input speed was 135.9 ± 37.0 characters/min.

Thus dotting Finger Braille using the Teaching System was more accurate than inputting Braille code using Braille keyboard by visually impaired people. Dotting speed by non-disabled people was one-third of input speed by visually impaired people.

Therefore the non-disabled senders who were non-skilled in Finger Braille could communicate with the blind receiver in Finger Braille directly, not but that total communication speed was limited to 23.1 characters/min. It was considered that the Teaching System was effective.

C. *Rule of communication with deafblind people*

In this experiment, we found out that not all the senders could understand and execute rule of communication with deafblind people. The senders were directed to keep on touching the fingers of the receiver at least right hand. But the senders got off both hands at the beginning of experiment, because they were preoccupied with operation of the Teaching System, especially when they were editing. Mean of edit time was 28.5 sec and maximum of edit time was 111 sec, the receiver spent uneasy moment without any explanations or communication cues.

Therefore it was the most important to keep on touching each other and to decide cues to edit or re-dotting (e.g. back-slapping) for supporting deafblind people.

V. CONCLUSIONS

In this paper, we developed the Finger Braille Teaching System and designed the teaching interface which taught clauses explicitly. And an evaluation experiment between a blind person who was skilled in Finger Braille and two non-disabled people who were non-skilled in Finger Braille was carried out. As accuracies of fundamental functions, *Correct Ratio* was 94.4% except substitution of interjection. Accuracy of conversion to Braille code was 99.6%. Accuracy of clause segmentation was 95.1%. Error ratio of dotting by the senders was only 1.2% of all characters. Dotting speed was 48.3 characters/min and total communication speed was 23.1 characters/min. The results showed that the fundamental functions were practicable; the non-disabled senders could dot Finger Braille accurately and communicate with the blind receiver directly. Therefore it was considered that the Teaching System was effective.

ACKNOWLEDGMENT

We greatly thank Ms. Satoko Mishina and Mr. Shinichi Hashima (interpreters of Finger Braille) for their support for the evaluation experiment.

REFERENCES

[1] S. Fukushima, *Person with Deafblind and Normalization*, Tokyo, Akashi Shoten, 1997.
[2] S.S. An, et al., “A Pair of Wireless Braille-Based Chording Gloves,” *Proc. of 9th International Conference on Computers Helping People with Special Needs*, France, pp.490-497, July 2004.
[3] T. Amemiaya, K. Hirota and M. Hirose, “OBOE: Oboe-Like Braille Interface for Outdoor Environment,” *Proc. of 9th International Conference on Computers Helping People with Special Needs*, France, pp.498-505, July 2004.
[4] Y. Matsuda, et al., “Finger Braille Teaching System for Non-disabled People who Communicate with Deafblind People,” *IFMBE Proc. vol. 8, 6th Asian-Pacific Conference on Medical & Biological Engineering*, Japan, PA-2-36, April 2005.
[5] Y. Matsuda, et al., “Finger Braille Recognition System for Non-disabled People who Communicate with Deafblind People,” *IFMBE Proc. vol. 14, World Congress on Medical Physics & Biomedical Engineering*, Korea, pp.2812-2815, August 2006.
[6] Y. Matsumoto, H. Tanaka, H. Hirakawa, H. Miyoshi and H. Yasukawa, “BUP: A Bottom-Up Parser Embedded in Prolog,” *New Generation Computing*, vol.1, no.2, pp.145-158, 1983.
[7] 3A Corporation, *Minna no Nihongo I Honsatsu (Main Textbook)*, Tokyo, 3A Corporation, 1998.