

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Finger Braille Teaching System

Yasuhiro Matsuda and Tsuneshi Isomura
*Kanagawa Institute of Technology
Japan*

1. Introduction

Deafblindness is a condition that combines varying degrees of both hearing and visual impairment. All deafblind people experience problems with communication, access to information, and mobility. Deafblind people use many different communication media, depending on the age of onset of deafness and blindness and the available resources. For example, "deafblind manual alphabet" is a method of spelling out words onto a deafblind person's hand. Each letter is denoted by a particular sign or place on the hand. "Block" is a manual form of communication where words are spelled out on the palm of the deafblind person's hand. "Hands on signing" is based on sign language. With this system, the deafblind person follows the signs by placing his hands over those of the signer and feeling the signs formed. "Yubi-Tenji" (Finger Braille) is one of the tactual communication media developed by Satoshi Fukushima in Japan (see Fig. 1). In Finger Braille, the index finger, middle finger and ring finger of both hands function like the keys of a Braille typewriter. A sender dots Braille code on the fingers of a receiver as if typing on a Braille typewriter. The receiver is assumed to be able to recognize the Braille code. Deafblind people who are skilled in Finger Braille can understand speech conversation and express various emotions because of the prosody (intonation) of Finger Braille (Fukushima, 1997). Because there is such a small number of non-disabled people who are skilled in Finger Braille, deafblind people communicate only through an interpreter.



Fig. 1. Finger Braille

Based on "Finger Braille Teaching System for People who Communicate with Deafblind People", by Yasuhiro Matsuda, Tsuneshi Isomura, Ichiro Sakuma, Etsuko Kobayashi, Yasuhiko Jimbo and Tatsuhiko Arafune which appeared in Proceedings of 2007 IEEE International Conference on Mechatronics and Automation (ICMA 2007). © 2007 IEEE.

Various Braille input devices have recently been developed (Amemiya et al., 2004; An et al., 2004), but they require deafblind people to wear gloves or type on a keyboard to input the Finger Braille, or to use actuators to output and convert the speech of non-disabled people to Finger Braille. With these devices, deafblind people are burdened with wearing sensors and actuators, and they must master a new communication system with these support devices.

The objective of this study is the development of a Finger Braille support device which employs the skin-contact communication of deafblind people, because skin contact is the only form of nonverbal communication for deafblind people. The concept of the Finger Braille support device is shown in Fig. 2. The advantages of this device are as follows: both deafblind people and non-disabled people unskilled in Finger Braille can communicate using conventional Finger Braille, and deafblind people are not encumbered by a support device because the non-disabled people operate the support device and wear all of the sensors. Our support device consists of a Finger Braille teaching system and a Finger Braille recognition system. The teaching system recognizes the speech of a non-disabled person and displays the associated dot pattern of Finger Braille. The non-disabled person can then dot Finger Braille on the fingers of the deafblind person by observing the displayed dot pattern (Matsuda et al., 2007). The recognition system recognizes the dotting of Finger Braille by the deafblind person and synthesizes this tactile communication into speech for the non-disabled person (Matsuda et al., 2010a).

In this chapter, we describe the Finger Braille teaching system and present experimental results. We first developed the Finger Braille teaching system and designed the teaching interface, which taught clauses explicitly. Then, an evaluation experiment between a blind person who was skilled in Finger Braille and two non-disabled people who were unskilled in Finger Braille was conducted.

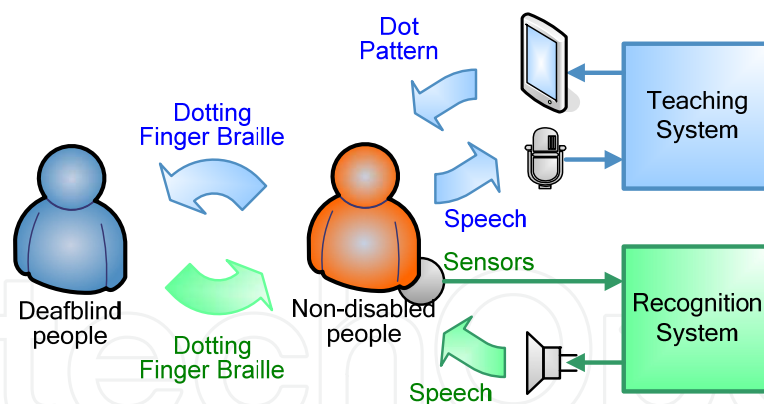


Fig. 2. Concept of Finger Braille support device

2. Japanese Braille system and Finger Braille

In Japanese script, Kanji (the Chinese ideographic script), Hiragana and Katakana (syllabic script) are the three kinds of Japanese writing symbols used (The Association for Overseas Technical Scholarship, 1975). Hiragana and Katakana are collectively called Kana characters because Japanese sentences are usually written with a combination of these two scripts. Foreign names and words of foreign derivation are usually written in Katakana. In addition to the abovementioned scripts, Romaji (Roman letters) is used. The Japanese sentence does not include spaces between words.

The Japanese Braille system was created by Kuraji Ishikura in 1890 (see Fig. 3). The Japanese Braille system is different from Japanese script in the following ways: (1) Japanese Braille consists only of Kana; (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 symbols marked voiced sound, semivoiced sound and diphthong are used as prefixes to modify the consonants; (5) the Japanese Braille sentence has a space between clauses (Bunsetsu unit).

Voiceless sound 五十音	Voiced and semivoiced sound 濁音・半濁音
⠠ ⠠ ⠠ ⠠ ⠠ ア イ ウ エ オ	⠠ ⠠ ⠠ ⠠ ⠠ ガ ギ グ ゲ ゴ
⠠ ⠠ ⠠ ⠠ ⠠ カ キ ク ケ コ	⠠ ⠠ ⠠ ⠠ ⠠ ザ ジ ズ ゼ ゾ
⠠ ⠠ ⠠ ⠠ ⠠ サ シ ス セ ソ	⠠ ⠠ ⠠ ⠠ ⠠ ダ デ ツ テ ト
⠠ ⠠ ⠠ ⠠ ⠠ タ チ ツ テ ト	⠠ ⠠ ⠠ ⠠ ⠠ バ ビ ブ ベ ボ
⠠ ⠠ ⠠ ⠠ ⠠ ナ ニ ヌ ネ ノ	⠠ ⠠ ⠠ ⠠ ⠠ パ ピ プ ペ ポ
⠠ ⠠ ⠠ ⠠ ⠠ ハ ヒ フ ヘ ホ	⠠ ⠠ ⠠ ⠠ ⠠ マ ミ ム メ モ
⠠ ⠠ ⠠ ⠠ ⠠ ヤ ユ ヨ	⠠ ⠠ ⠠ ⠠ ⠠ シャ シュ ショ ジャ ジュ ジョ
⠠ ⠠ ⠠ ⠠ ⠠ ラ リ ル レ ロ	⠠ ⠠ ⠠ ⠠ ⠠ チャ チュ チョ チャ チュ チョ
⠠ ⠠ ⠠ ⠠ ⠠ ワ ウ エ ヲ	⠠ ⠠ ⠠ ⠠ ⠠ ニャ ニュ ニョ ビャ ビュ ビョ
⠠ ⠠ ⠠ 撥音符 促音符 長音符 (ン) (ッ) (ー)	⠠ ⠠ ⠠ ⠠ ⠠ ヒャ ヒュ ヒョ ピャ ピュ ピョ
	⠠ ⠠ ⠠ ⠠ ⠠ ミャ ミュ ミョ ⠠ ⠠ ⠠ ⠠ ⠠ リャ リュ リョ

Fig. 3. Japanese Braille code (National Association of Information Service for Visually Impaired Persons, 2002)

In Finger Braille, the sender dots the Braille codes directly on the fingers of the receiver as if typing on a Braille typewriter. The receiver is assumed to be able to recognize the Braille code. A rule of Finger Braille is that the sender keeps touching the fingers of the receiver even when not dotting, because receivers feel uneasy in the absence of touching or tactile cues. Prosody (intonation) of Finger Braille helps the receiver recognize the dotted Braille

code. The features of prosody of Finger Braille are as follows: (1) the sender dots long at the end of clauses; (2) the sender dots long and strongly at the end of sentences; (3) the sender dots short and strongly at the double consonants; (4) the sender dots short and strongly at the symbols of voiced sound, semivoiced sound and diphthong; (5) the sender pauses between clauses; (6) the sender must not pause during a clause (Miyagi et al., 2007; Matsuda et al., 2010b).

3. Development of the teaching system

3.1 Configuration of the teaching system

Fig. 4 shows the configuration of the Finger Braille teaching system. First, a speech recognition (SR) engine recognizes the speech by the sender. Second, by using the results of the speech recognition, the teaching system converts the Kana to the Braille code. Third, by parsing the Braille code, the teaching system retrieves the clause information and segments the Braille code into clauses. Finally, the teaching system displays the associated dot pattern of the Braille code. The teaching system was developed on a tablet PC (HP TC1100, CPU Pentium M 1.1 GHz, RAM 1024 MB, 10.4 inches XGA LCD). The operating system was Microsoft Windows XP. The programming languages were Microsoft Visual Basic 6 and LPA WIN-PROLOG 4.500. The speech recognition engine was Microsoft Speech SDK (SAPI5.1).

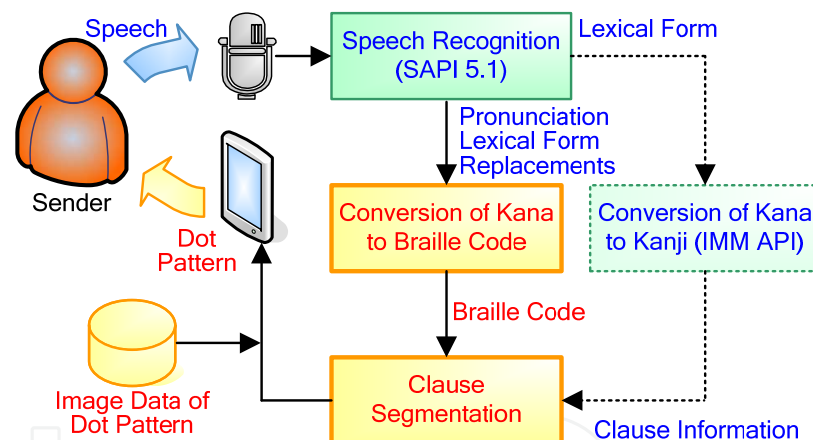


Fig. 4. Configuration of the teaching system

3.2 Speech recognition

Because the sender must keep touching the fingers of the receiver, speech recognition is suitable for the input interface of the teaching system. First, the teaching system was created and the dictation grammar of SAPI5.1 was loaded, and then SAPI5.1 was ready for recognition. When the sender spoke into a microphone, SAPI5.1 attempted to recognize it. Following successful speech recognition, the teaching system retrieved the results of the speech recognition.

The sender could train SAPI5.1 by a speech training wizard. After speech training, the SR engine could perform better and improve SAPI's personalization experience.

3.3 Conversion of Kana to Braille code

Table 1 shows an example of the Braille code and the results of the speech recognition. Because the Japanese Braille system consists only of Kana, the pronunciation and lexical form of the phrase elements are suitable for converting to the Braille code. The teaching system checked each character of the pronunciation and lexical form of the phrase elements and adopted the suitable character for the Japanese Braille code. Fig. 5 shows the flow chart of the conversion of Kana to Braille code. The following were the rules for conversion: (1) generally, the lexical forms of the phrase elements were adopted; (2) in the particles [ha] and [he], the pronunciations of the phrase elements were adopted; (3) in the long vowels [ū] and [ō], the pronunciations of the phrase elements were adopted.

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 O ne e san wa / ga k ko - e / i ki ma shi ta.
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 O / ne - sa n / wa / ga k ko - / e / i / ki / ma shi ta
Lexical Form	お ねえさん は がっこう へ い き ま した O / ne e sa n / ha / ga k ko u / he / i / ki / ma shi ta
Display Text	お 姉さん は 学校 へ 行 き ま した

Table 1. Example of the Braille code and results of speech recognition

3.4 Clause (Bunsetsu) segmentation

To realize the non-disabled sender's prosodic dotting, the dot pattern of clauses was displayed explicitly. SAPI5.1 cannot retrieve the clause information of the result of speech recognition. Thus, we developed a Braille code parser that applies natural language processing (BUP system) (Matsumoto et al., 1983). The Braille code parser parsed the Braille code and segmented it into clauses by inserting a space between the clauses. The Braille code parser consisted of a dictionary, grammar, BUP translator and control program. The dictionary and the grammar were described in the definite clause grammars (DCG). The BUP translator translated the dictionary and grammar into a Prolog program. The control program controlled the execution of parsing.

If the Braille code was not grammatically 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 a space between the clauses of the Braille code.

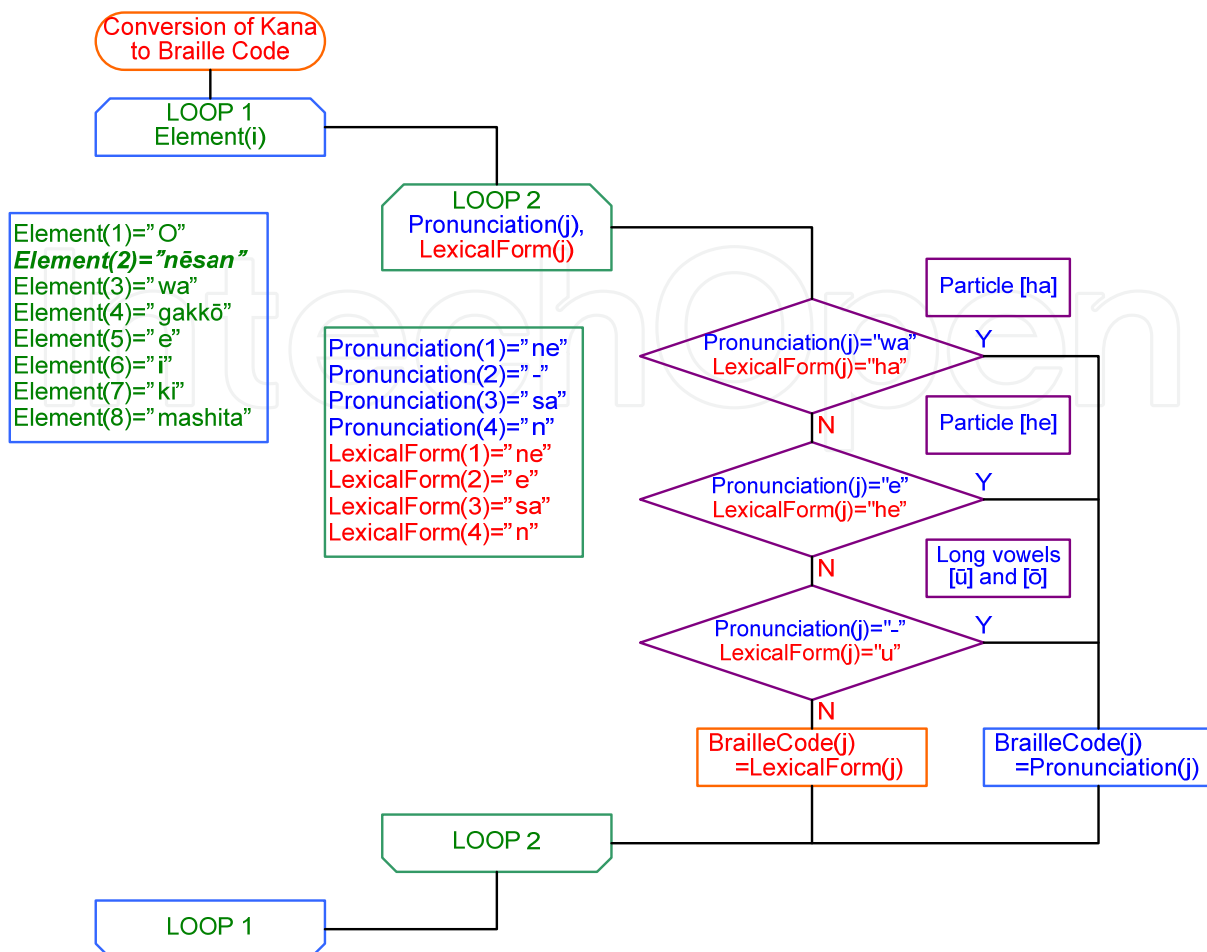


Fig. 5. Flow chart of conversion of Kana to Braille code using “ne - sa n” as an example

3.5 Design of the teaching interface

Finally, the teaching system displayed the dot pattern of the Braille code by reading from the image database of dot patterns. Fig. 6 shows the teaching interfaces that we designed. The Braille code was displayed in an upper text box. The dot pattern was displayed in fourteen picture boxes (two columns and seven rows). The first clause was displayed in the left column (from the upper left to the lower left) and the second clause was displayed in the right column (from the upper right to the lower right). The third clause was displayed on the next page. The clause that did not consist of more than seven characters was displayed in one column and the clause that consisted of more than seven characters was displayed in two columns. After one clause was displayed in the left column, the next clause was displayed in the right column or the next page. Thus, the dot pattern of the clauses was displayed explicitly in the columns.

The red pattern indicated the left hand and the blue pattern indicated the right hand. We designed two kinds of presentation methods for the sender. Presentation method A only displayed the dot pattern and presentation method B displayed the dot pattern on the illustration of the fingers. Presentation method B was more symbolic and easier for beginners to recognize the dotting fingers. Presentation method A had the most simplified signing and was suitable for the experienced senders (Matsuda et al., 2005).

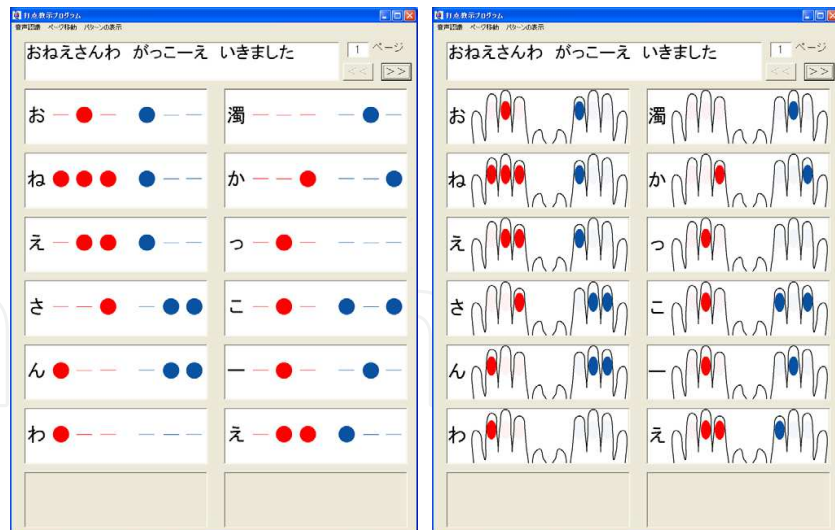


Fig. 6. Two kinds of presentation methods for the teaching interface, presentation method A (left) and B (right). The displayed dot pattern is “Onesasanwa / gakko-e {My sister / to school.}”

3.6 Editing Braille code

Because the sender keeps touching the fingers of the receiver with at least one hand, we installed a trackball (Kensington Expert Mouse USB/PS2) to operate the teaching system by the left hand of the sender. We allocated six functions to the keys of the trackball (see Fig. 7). If the Braille code included any mistakes because of misrecognition of SR, the sender could edit the Braille code by using the track ball and a software keyboard (see Fig. 8).

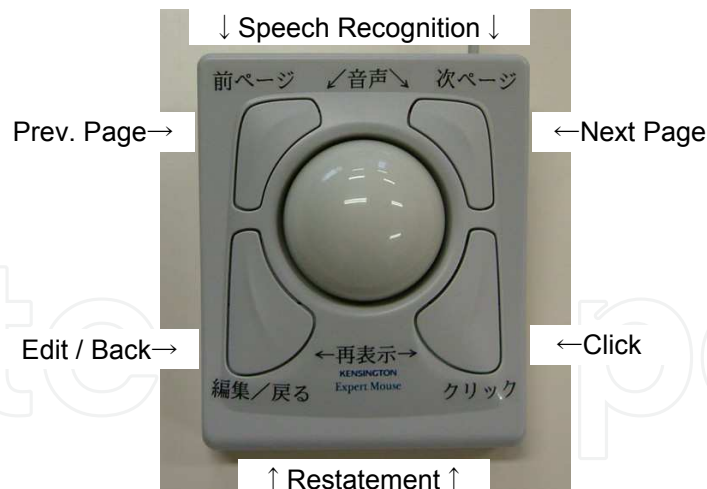


Fig. 7. Trackball to operate the teaching system

To start speech recognition, the sender pushed the upper two keys at the same time. To display the next page of the dot pattern, the sender pushed the upper right key. To display the previous page of the dot pattern, the sender pushed the upper left key. To start editing the Braille code, the sender pushed the lower left key. To select the character of the software keyboard, the sender pointed to the character with the trackball and pushed the lower right

key. After editing, to restate the dot pattern of the edited Braille code, the sender pushed the lower two keys at the same time.

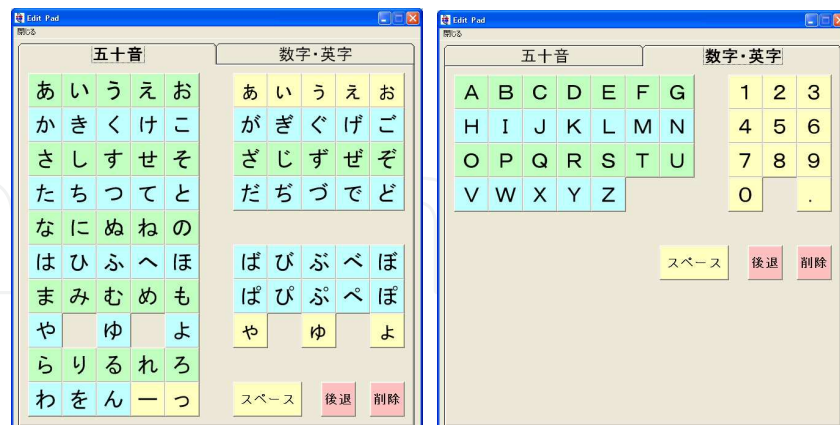


Fig. 8. Software keyboard to edit Braille code. Japanese characters (left), alphabets and numerals (right)

4. Evaluation experiment

4.1 Method

The objectives of the evaluation experiment were as follows: (1) evaluate the accuracies of the fundamental functions (speech recognition, conversion to Braille code and clause segmentation); (2) evaluate the accuracies of the dotting and recognition; (3) evaluate the communication speed.

The receiver was a blind person who was skilled in Finger Braille with 20 years of sending experience and 8 years of receiving experience. The senders were two college students who were unskilled in Finger Braille (one male and one female). Both senders reported normal hearing and vision abilities and were native Japanese speakers. All subjects gave informed consent after hearing a description of the study.

The dialogues (total: 51 sentences, 143 clauses, 288 words, 686 characters) comprised four daily conversations in a Japanese textbook for foreign beginners (3A Corporation, 1998).

The senders were instructed to operate the teaching system, and trained SAPI5.1 by the speech training wizards "Introduction" and "Introduction of Speech Technology." To simulate deafness, the receiver wore earplugs and headphones that played white noise.

The experimental flow is shown in Fig. 9. In the experiment, one sender and the receiver sat side by side (see Fig. 10). The sender pushed the key to begin the speech recognition and spoke one sentence of the dialogues. If the result of the speech recognition was correct, the sender dotted Finger Braille on the fingers of the receiver by observing the teaching interface. If the result of the speech recognition was not correct, the sender spoke the same sentence again or edited the Braille code and pushed the restatement key. Then the sender dotted Finger Braille on the fingers of the receiver. The receiver responded to the recognized sentence. If the receiver misrecognized the sentence, the sender dotted the same sentence again. In the experiments, the sender repeated almost all of the sentences of the dialogues.

The experiment included four experimental sessions. In session 1, sender 1 (female) dotted conversation 1; the dot patterns were displayed by presentation method B. In session 2, sender 1 dotted conversation 2; the dot patterns were displayed by presentation method A.

In session 3, sender 2 (male) dotted conversation 3; the dot patterns were displayed by presentation method B. In session 4, sender 2 dotted conversation 4; the dot patterns were displayed by presentation method A. The senders were instructed as follows: to dot as accurately as possible; to dot long in the characters at the end of the clauses and sentences; to keep touching the fingers of the receiver, at least with their right hand, even when not dotting.

All of the sessions were recorded by a digital video camera. The log of operations by the senders and the recognized speech were recorded in the hard disk drive of the teaching system. The receiver put his fingers on the pressure sensor sheets (Nitta Tactile Sensor System), which measured the change of pressure as a result of dotting, during the experiment.

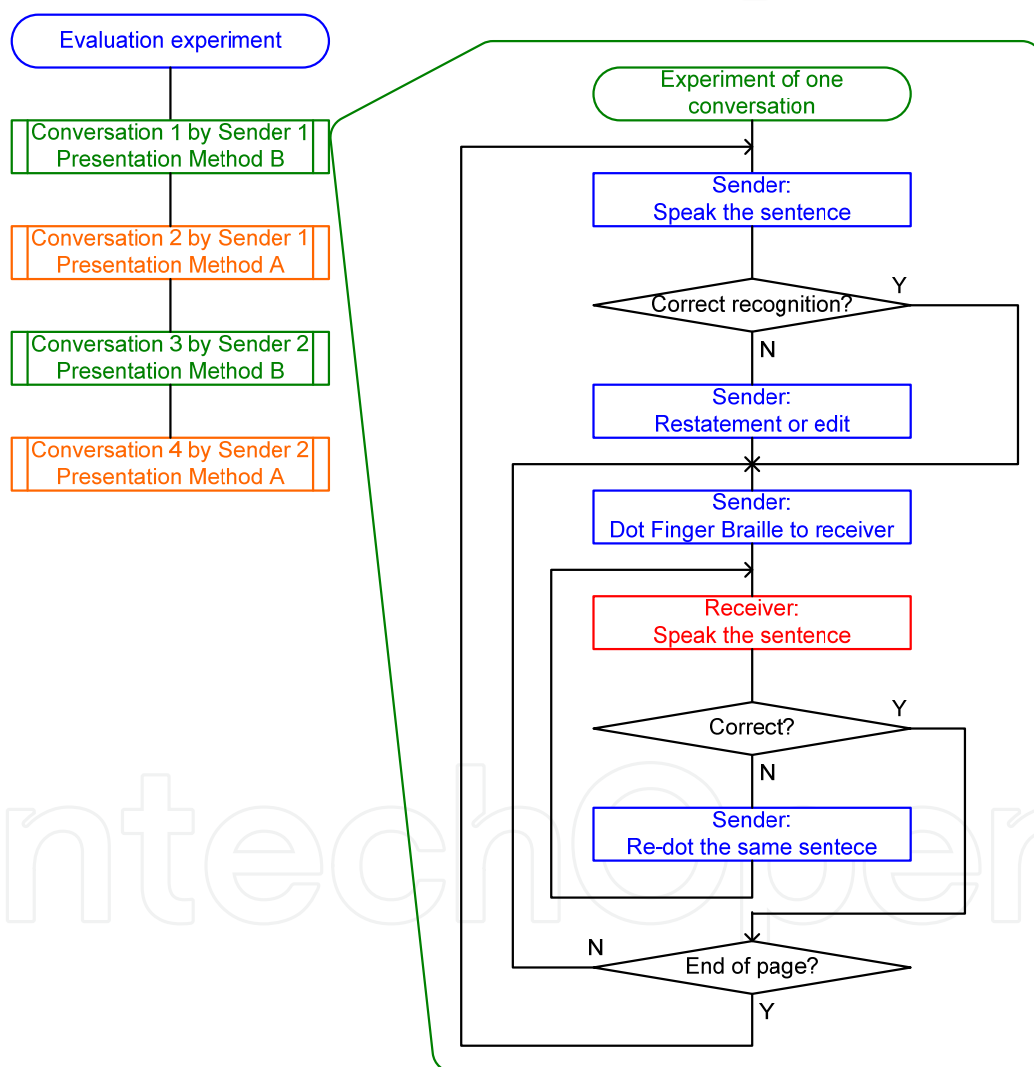


Fig. 9. Experimental flow

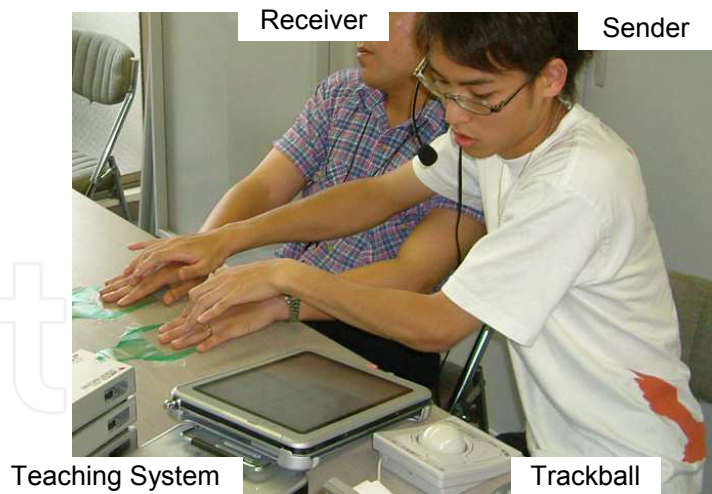


Fig. 10. Experiment in progress

4.2 Results

4.2.1 Accuracy of fundamental functions

We classified the words of the dialogues into interjections (15 words), proper nouns (14 words) and other words (259 words). To evaluate the accuracy of the speech recognition, the error number of the substitutions in each part of speech and the error number of the deletions were counted. Fig. 11 shows the error ratios of the speech recognition. The results showed that 7% of the proper nouns (1 word) and 5% of other words (13 words) were substituted and 0.7% of the words (2 words) were deleted. Because 47% of the interjections (7 words) were substituted, SAPI5.1 had difficulty in recognizing the interjections (Matsuda et al., 2007). Then, the *Correct Ratio* of speech recognition was calculated, as follows.

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

where N is the number of the words, sub is the number of substitution errors, and del is the number of deletion errors. *Correct Ratio* was 92.0% for all of the words and 94.4% without substitution of the interjections (see Fig. 12).

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

The results of the speech recognition were accurately converted to the Braille codes. Because the senders edited the Braille codes incorrectly, 0.4% of the dialogues (3 characters) were not correct Braille codes. Thus, the accuracy of the conversion to the Braille code was 99.6% (see Fig. 12).

When the results of the speech recognition were correct, the grammatical Braille codes were accurately segmented into clauses by the Braille code parser. When the results of the speech recognition were not correct, 3.5% of the clauses (5 clauses) were slipped their segmentation points by the backup of the Braille code parser (IMM API). When the senders edited the Braille codes, 1.4% of the clauses (2 clauses) deleted the spaces between the clauses. Thus, the accuracy of the clause segmentation was 95.1% (see Fig. 12).

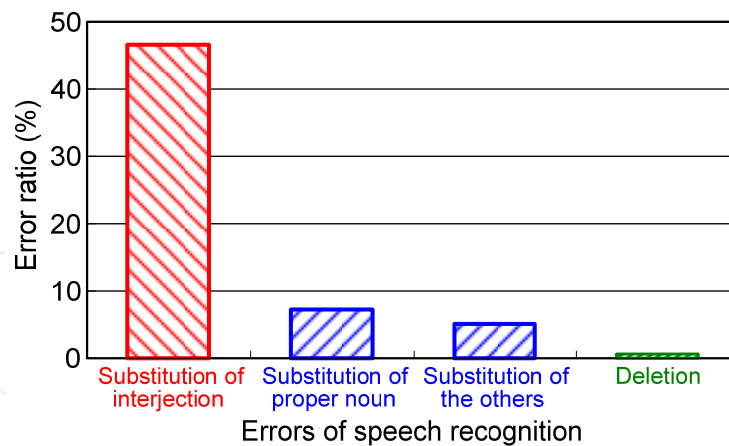


Fig. 11. Error ratios of speech recognition

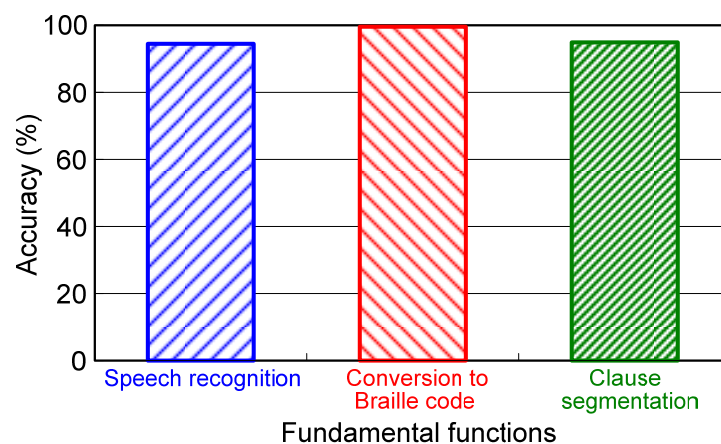


Fig. 12. Accuracy of the fundamental functions

4.2.2 Accuracy of dotting and recognition

The error ratio of dotting by the senders was only 1.2% of the characters (8 characters). The receiver could not recognize 7.8% of the dialogues (4 sentences) because of the dotting errors of the senders. As the senders re-dotted the same sentences, the receiver could recognize them. Thus, the accuracy of dotting by the senders was 98.8% and the accuracy of recognition by the receiver was 92.2% in the first dotting and 100% in the re-dotting (see Fig. 13).

4.2.3 Operation time

An operation time was divided into five sections. The operation times were calculated in the log of the operations and the video images. Fig. 14 shows the distribution of operation times. The speech recognition time was the time until the dot pattern was displayed, after the sender pushed the key for the speech recognition (including the time of re-speech). The mean of the speech recognition time was 7.4 sec (S.D.=8.2) and the mean of the speech recognition time per speech was 5.7 sec (S.D.=2.9).

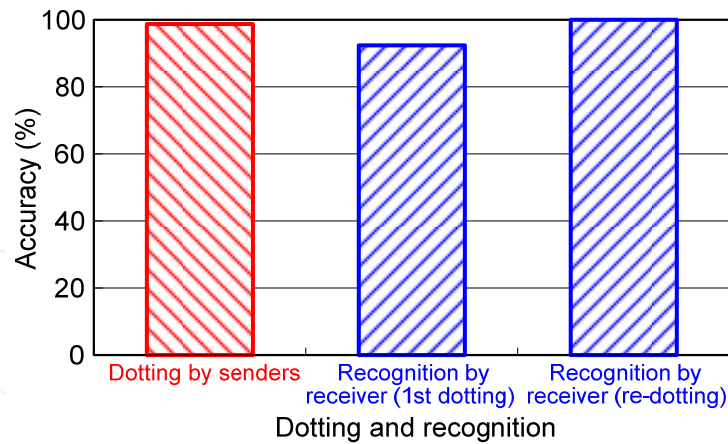


Fig. 13. Accuracy of dotting and recognition

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

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

The response time meant the time until the sender started dotting, after the dot pattern was displayed. The mean of the response time was 5.1 sec (S.D.=2.5).

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

The total communication time was the time until the sender finished dotting, after the sender pushed the speech recognition key (including the time of re-speech, editing, changing pages and re-dotting). The mean of the total communication time was 47.7 sec (S.D.=39.8) and the mean of the total communication speed was 23.1 characters/min (S.D.=10.3).

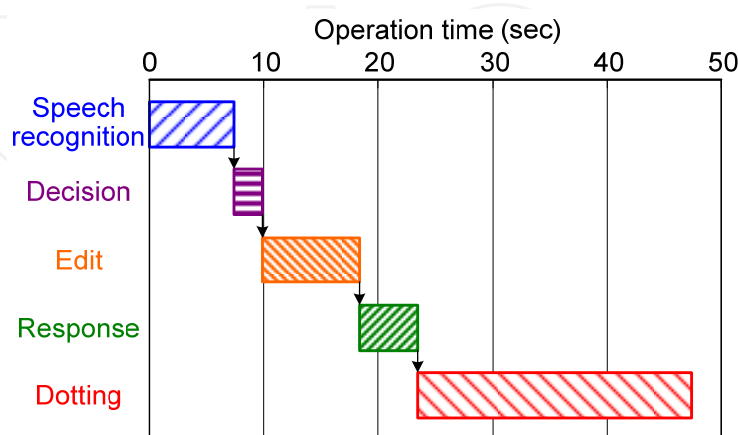


Fig. 14. Distribution of the operation times

4.3 Discussion

4.3.1 Accuracy of fundamental functions

As the accuracy of the fundamental functions, *Correct Ratio* was 94.4% except for substitution of the interjections. The accuracy of the conversion to Braille code was 99.6%. The accuracy of clause segmentation was 95.1%. It was confirmed that these results corresponded to those in our previous study (Matsuda et al., 2007), and the fundamental functions were practicable. As for editing the Braille code, the senders were puzzled the first time, but they could edit smoothly the second time.

4.3.2 Accuracy of dotting and dotting speed

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

As previously mentioned, Amemiya et al. (2004) developed a Braille input device. In an evaluation experiment, five non-disabled people who had no experience using a Braille typewriter were given a sheet of paper with a list of Braille codes and instructed to input the codes as quickly and accurately as possible. As a result of this test, the error ratio was 8-6.76% (sessions 1-5) and the input speed was 20-35.4 characters/min (sessions 1-5).

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

As mentioned above, An et al. (2004) carried out an experiment in which ten visually impaired people who had just started to learn Braille codes inputted Braille code using a Braille typewriter. As a result, the error ratio was $2.8 \pm 2.3\%$ and the input speed was 135.9 ± 37.0 characters/min.

For the visually impaired people, dotting Finger Braille using the teaching system was more accurate than inputting Braille code using the Braille keyboard. The dotting speed by the non-disabled senders in our experiment was one-third of the input speed by the visually impaired subjects.

The non-disabled senders who were unskilled in Finger Braille could communicate with the blind receiver in Finger Braille directly, but the total communication speed was limited to 23.1 characters/min. Therefore, it was considered that the teaching system was effective.

4.3.3 Rule of communication with deafblind people

In this experiment, we found that both senders could not understand and execute the rule of communication with deafblind people. The senders were directed to keep touching the fingers of the receiver with at least their right hand. But the senders removed both hands at the beginning of the experiment, because they were preoccupied with the operation of the teaching system, especially when they were editing. Then, the receiver felt uncomfortable in the absence of touching or tactile cues.

Therefore, the non-disabled sender must constantly touch the fingers of the deafblind receiver even when not dotting and must decide the cues to edit or re-dot (e.g., back-slapping).

5. Future plans

5.1 Improvement of interfaces

The present teaching system consists of the tablet PC, microphone and trackball. To improve the input interface for operating the teaching system, we adopted a touch panel on the LCD. To expand the teaching interface, we adopted a WXGA display. Fig. 15 shows the expanded teaching interfaces that were developed on another tablet PC (Dell Latitude XT, CPU Core 2 Duo 1.33 GHz, RAM 2 GB, 12.1 inch WXGA LCD with touch panel). The Braille code is displayed in the upper text box. The dot pattern is displayed in sixteen picture boxes (two columns and eight rows). The buttons of speech recognition, edit, restatement, previous page and next page are located on the lower part. The sender can touch the LCD directly to operate the teaching system and edit the Braille code. The teaching interface is expanded and the number of pages to display the Braille code of one sentence can be reduced. We have been evaluating the improved teaching system.



Fig. 15. Expanded teaching interface, presentation method A (left) and B (right). Displayed dot pattern is “Onesesanwa / gakko-e {My sister / to school.}”

5.2 Teaching of prosody and emotional expression

For the prosody of Finger Braille, the sender dots long at the end of clauses and sentences, and dots short at double consonants and voiced sound. To realize the non-disabled sender's prosodic dotting, the dot pattern of clauses was displayed explicitly in the present teaching system. We designed the dot pattern with long and short arrows to indicate the duration of dotting (see Fig. 16) (Matsuda et al., 2009).

In Finger Braille, the sender can express various emotions by changing the duration and strength of dotting (Fukushima, 1997; Matsuda et al., 2010b). The intent of our support device is to assist not only verbal communication but also nonverbal (emotional) communication. To assist in emotional communication, we have been developing an emotion teaching system and an emotion recognition system (Matsuda et al., 2010c).

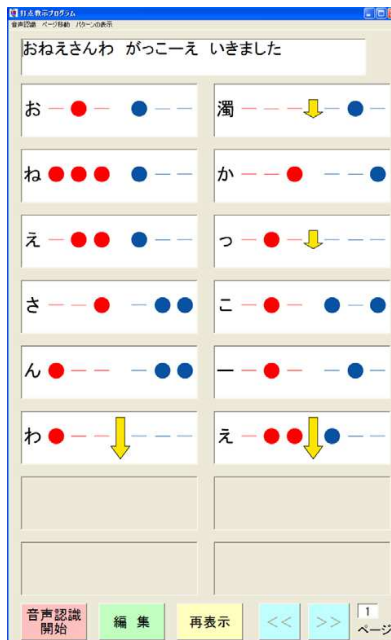


Fig. 16. Teaching interface for teaching of prosody. The displayed dot pattern is “Oneesanwa / gakko-e {My sister / to school.}”

6. Conclusion

In this chapter, we developed the Finger Braille teaching system and designed a teaching interface that teaches clauses explicitly. An evaluation experiment between a blind receiver who was skilled in Finger Braille and two non-disabled senders who were unskilled in Finger Braille was conducted. For the accuracy of fundamental functions, *Correct Ratio* was 94.4%, except for the substitution of interjections. The accuracy of the conversion to Braille code was 99.6%. The accuracy of clause segmentation was 95.1%. The error ratio of dotting by the senders was only 1.2% of all of the characters. The dotting speed was 48.3 characters/min and the total communication speed was 23.1 characters/min. The results show that the fundamental functions are 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.

7. Acknowledgment

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

This study was supported by the Japan Society for the Promotion of Science under a Grant-in-Aid for Scientific Research (No. 21500522) and the Ministry of Education, Culture, Sports, Science and Technology of Japan under a Grant-in-Aid for Scientific Research (No. 16700430). This study was partly supported by Kanagawa Academy of Science and Technology (KAST) under a research grant.

8. References

- Amemiya, T.; Hirota, K. & Hirose, M. (2004). OBOE: Oboe-Like Braille Interface for Outdoor Environment, *Proceedings of 9th International Conference on Computers Helping People with Special Needs*, pp. 498-505, ISBN 978-3-540-22334-4, Paris, France, July 2004, Springer, Berlin
- An, S.S.; Jeon, J.W.; Lee, S.; Choi, H. & Choi, H.G. (2004). A Pair of Wireless Braille-Based Chording Gloves, *Proceedings of 9th International Conference on Computers Helping People with Special Needs*, pp. 490-497, ISBN 978-3-540-22334-4, Paris, France, July 2004, Springer, Berlin
- Fukushima S. (1997). *Person with Deafblind and Normalization*, Akashi Shoten, ISBN 4-7503-0982-6, Tokyo, Japan
- National Association of Information Service for Visually Impaired persons (2002). *Handbook of translation into Braille*, Daikatsuji, ISBN 4-86055-013-7, Tokyo, Japan
- Matsuda, Y.; Sakuma, I.; Jimbo, Y.; Kobayashi, E.; Arafune T. & Isomura, T. (2005). Study on Teaching of the Way to Dot of Finger Braille – Teaching of dotting finger and position of monosyllable –, *Transaction of Human Interface Society*, Vol. 7, No. 3, pp. 379-390, ISSN 1344-7262
- Matsuda, Y.; Sakuma, I.; Jimbo, Y.; Kobayashi, E.; Arafune T. & Isomura, T. (2007). Development of Finger Braille Teaching System – Teaching of dotting finger and position using speech recognition –, *Journal of the Society of Life Support Technology*, Vol. 19, No. 3, pp. 105-116, ISSN 1341-9455
- Matsuda, Y. & Isomura, T. (2009). Finger Braille teaching system – teaching of prosody of finger Braille, *Journal of Communication and Computer*, Vol. 6, No. 2, pp. 55-64, ISSN1548-7709
- Matsuda, Y.; Sakuma, I.; Jimbo, Y.; Kobayashi, E.; Arafune T. & Isomura, T. (2010a). Development of Finger Braille Recognition System, *Journal of Biomechanical Science and Engineering*, Vol. 5, No. 1, pp. 54-65, ISSN 1880-9863
- Matsuda, Y.; Sakuma, I.; Jimbo, Y.; Kobayashi, E.; Arafune T. & Isomura, T. (2010b). Emotional Communication in Finger Braille, *Advances in Human-Computer Interaction*, Vol. 2010, Article ID 830759, 23 pages, ISSN 1687-5893
- Matsuda, Y.; Sakuma, I.; Jimbo, Y.; Kobayashi, E.; Arafune T. & Isomura, T. (2010c). Emotion Recognition of Finger Braille, *International Journal of Innovative Computing, Information and Control*, Vol. 6, No. 3(B), pp. 1363-1377, ISSN 1349-4198
- Matsumoto, Y.; Tanaka, H.; Hirakawa, H.; Miyoshi, H. & Yasukawa, H. (1983). BUP: A Bottom-Up Parser Embedded in Prolog, *New Generation Computing*, Vol. 1, No. 2, pp. 145-158, ISSN 0288-3635
- Miyagi, M.; Miyazawa, K.; Ueno, A.; Nishida, M.; Horiguchi, Y.; Ichikawa, A. & Noshiro, M. (2007). Analysis of Prosody in Strength and Time Structure of Finger Braille, *IEICE Technical Report (WIT2007-1~15)*, Vol. 107, No. 61, pp. 25-28, ISSN 0913-5685
- The Association for Overseas Technical Scholarship (1975). *Nihongo no Kiso I (Grammatical Note)*, 3A Corporation, ISBN 4-906224-13, Tokyo, Japan
- 3A Corporation (1998). *Minna no Nihongo I Honsatsu (Main Textbook)*, 3A Corporation, ISBN 4-88319-102-8, Tokyo, Japan



Character Recognition

Edited by Minoru Mori

ISBN 978-953-307-105-3

Hard cover, 188 pages

Publisher Sciyo

Published online 17, August, 2010

Published in print edition August, 2010

Character recognition is one of the pattern recognition technologies that are most widely used in practical applications. This book presents recent advances that are relevant to character recognition, from technical topics such as image processing, feature extraction or classification, to new applications including human-computer interfaces. The goal of this book is to provide a reference source for academic research and for professionals working in the character recognition field.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Yasuhiro Matsuda and Tsuneshi Isomura (2010). Finger Braille Teaching System, Character Recognition, Minoru Mori (Ed.), ISBN: 978-953-307-105-3, InTech, Available from:

<http://www.intechopen.com/books/character-recognition/finger-braille-teaching-system>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen