

視聴覚複合障害者のための指点字コミュニケーションシステム (1)

5C-05

片山 滋友[†] 椋田 實[†] 長田 光平[†]阿部 正平[‡] 郡 千治[‡] 宮崎 卓弥[‡] 中山 克則[‡][†]日本工業大学 [‡]NECソフト株式会社 [‡]東北電子計算機株式会社

1. はじめに

視聴覚複合障害者 (以下、盲聾者) は、最新のデータで 1 万 7 千人ほどいると推定されている^[1]。人間の生活において、視覚と聴覚でそれぞれ 83%、11% の情報が処理されているというデータがあるが、盲聾者は 94% の情報が遮断されることになり、多くの場合、絶対的な孤独に置かれることになる。盲聾者のコミュニケーション方法として、盲聾に至る過程で得た知識や技術により様々な方法があるが、手書き文字や指文字、指点字などがよく用いられている。中でも指点字は、会話速度や内容の正確さなどの点で、非常に優れた方法であり、IT 時代に最も適した盲聾者のコミュニケーション方法として注目され、多くの研究がなされている。筆者らは、これまで携帯性よりも在宅での利用と、実際の指点字に近いタッチ感覚でコミュニケーションできる指点字会話支援システムを目指して研究開発してきた^{[2][3]}。

今回、インターネットを介して盲聾者と健常者あるいは盲聾者同士が、コミュニケーションできるプロトタイプの指点字会話システムを開発した。その概要を指点字入力装置とアプリケーションの二つに分けて報告する。本論文では、主に指点字入出力装置の基本原理とハード構成および指点字制御ソフトの構成について述べる。

Finger Braille Communication System for Deaf-Blind People -Part 1

Shigetomo Katayama, Minoru Mukuda, Kouhei Osada, Masatoshi Abe, Chiharu Kouri, Takuya Miyazaki, Katunori Nakayama

[†]Nippon Institute of Technology[‡]NEC Soft, Ltd. [‡]† Tohoku Computer Co. Ltd.

2. 指点字の原理と特徴

指点字とは、視覚障害者の使用する 6 点点字を人の両手の人差し指、中指、薬指に割り付けたもので、点字の凸側で見るパーキンスブレイラ式と、凹側で見るライトブレイラ式がある。図 1 に点字の「カ」を表示する例を示すが、凸では左列が 1, 2, 3 となり、凹では 4, 5, 6 となり左右が反転することになる。指点字のコミュニケーションは、隣合って指をタッチする場合は、2 方式のどちらかを指定すれば良く、向かい合って指をタッチする場合は、必ず相手方は逆の方式を使用する必要がある。

点字凸側						点字凹側					
1 ●	○	4				4 ○	●	1			
2 ○	○	5				5 ○	○	2			
3 ○	●	6				6 ●	○	3			
3	2	1	4	5	6	6	5	4	1	2	3
○	○	●	○	○	●	●	○	○	●	○	○
左手			右手			左手			右手		
パーキンスブレイラ式						ライトブレイラ式					

図 1 文字「カ」の点字と指点字との対応

3. 指点字入出力装置のハードウェア構成

指点字入出力装置は、図 2 に示すように右手用と左手用の入出力デバイスとそのドライブ回路から構成されている。この装置は、デジタル入出力ボード (DIO) を介してパソコンに接続される。入力デバイスは、指点字の入力部であり左右に 3 個のスイッチと、両側面にモード切換え用波型スイッチ (4 個) やプッシュスイッチ (2 個) が配置されている。出力デバイスは、指点字の出力部であり左右に 3 個のフィンガー

付ロータリーソレノイドと、応答確認用振動モータ 1 個配が置されている。

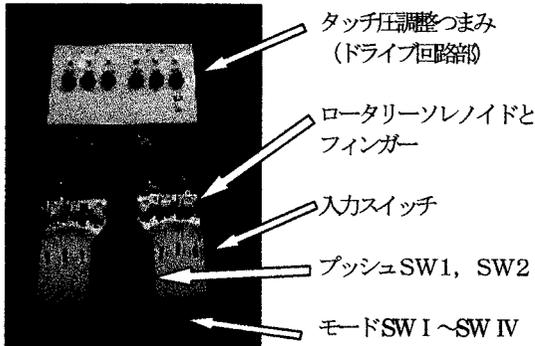


図2 指点字入出力装置の概観図

4. 入力スイッチの基礎特性と選定

正確に早く疲労感無く入力するためには、入力スイッチの選定が重要になる。市販されているスイッチは、通常、複数同時に押す設計となっていないため、複数同時に押すと非常に重くなる。選定に当たりコンピュータのキーボードなどに使われるプッシュスイッチ（以下、プッシュ SW）と制御システムなどで良く使われるマイクロスイッチ（以下、マイクロ SW）を使って 6 個同時に入力した時の時間差を測定した。回路には RS-FF によるチャタリング防止回路を組み込んだ。目隠した状態で 6 個のスイッチを同時に押し、入力したときの ON のタイミング時間差をロジックアナライザーで測定した結果を表 1 に示す。これはスイッチの種類を変えて左

表1 スwitchの違による最大時間差

		最大時間差の平均値 (ミリ秒)			
		プッシュ SW		マイクロ SW	
右手	薬指	15.2	9.9	6.6	6.9
	中指	8.6		8.3	
	人差指	5.8		5.8	
左手	人差指	6.3	9.3	4.0	4.6
	中指	4.7		5.4	
	薬指	16.8		4.3	
総平均		9.6		5.8	

右の三本の指について 10 回測定した ON 時間差の平均値で示してある。左右の手について比較すると、左手側の平均値が小さく、スイッチの種類を変えても同様であった。

タッチ入力時の最も早い指と遅い指との時間差平均は、4.0~16.8 ミリ秒であった。左右の手を比較すると左手の方が小さく、スイッチの種類を変えても同様であった。スイッチの種類では、マイクロ SW の方が 4 割ほどプッシュ SW に比べ時間差が少なく、指によるバラツキも小さい。この結果より試作装置にはマイクロ SW を使用することにした。

5. 出力フィンガーの基礎特性と選定

これまでの報告で、指点字出力のアクチュエータとして用いられているのは、主に振動モータ¹⁴⁾とソレノイドである。この二種類についてかな文字 50 音の指点字の認識特性を測定した。なお、実験時のタッチ時間は 500 ミリ秒である。表 2 に 2 名の被験者（右利き）による 5 回の平均値を示す。これを見ると左右の手とも中指の認識率が低くなっている。左右の手では各アクチュエータとも差異が認められなかった。認識率はソレノイドの方が振動モータよりも 2 割ほど良好である。振動型は携帯性の面からは優れているが、実験結果からも分かるように 6 点の振動を識別するには感覚の麻痺が生じやすく認識率が悪くなる。こららの結果や在宅での使用

表2 フィンガータッチの認識特性

		タッチ認識率の平均値 (%)			
		振動モータ		ソレノイド	
右手	薬指	85.5	84.9	99.8	97.6
	中指	77.1		95.7	
	人差指	92.0		97.2	
左手	人差指	82.8	82.5	99.4	97.9
	中指	73.3		95.0	
	薬指	91.5		99.4	
総平均		83.7		97.8	

に主眼を置いたシステムでは、電源の問題をあまり気にする必要が無いので、ソレノイドを使用することにした。また、実際のタッチ感覚とするには振動では無理と思われる。次に人の指にタッチするフィンガー先端部の形状や材質について評価を行ったが、先端部を種々変えても認識率には殆ど影響が見られず、タッチの感覚では硬めのスポンジが良いという結果になった。

6. 指点字制御ソフトの構成

次に、指点字制御ソフトについて説明する。図 3 に指点字装置とアプリケーション間をつなぐ指点字制御ソフトのブロック図を示す。装置を直接制御するには 16 点入出力の DIO ボードを使用している。このボードへの指示は C++ の関数である DLL プログラムである。Java と C++ 関数の接続には、Java Native Interface (JNI) で作成した JNI 形式の Pci2727J クラスを使用している。このクラスは生成時に C++ 関数の DLL をロードして使用する。

電子メールやニュースなどのアプリケーションソフトとのインターフェース (指点字装置用 API) は、interface FingerBraille として定義し

ている。機能は次のとおりである。

- ①指点字装置からの入力 String gets();
- ②指点字装置への出力
boolean puts(String data);
- ③点字速度を読む int getSpeed();
- ④点字コードの種類を読む (P/R)
int getCode();
- ⑤指点字装置のモードを読む
int getMode();
- ⑥指点字装置を振動する void bell();

アプリケーションとの入出力に用いる文字コードは UNICODE である。

FingerBrailleDevice クラスは、Pci2727J を経由して指点字装置を駆動する。start メソッドで読み込み開始、10m 秒ごとに装置からデータ (点字ボタン, SW1, SW2, SW I, SW II, SW III, SWIV) をサンプリングする。このデータを基に文字入力やコード設定, 速度設定, 連続・ステップ出力設定の機能を実現している。

点字の出力には、連続出力とステップ出力があり、連続出力は文字を指定速度で出力する。ステップ出力は、1 文字ごとに停止し、SW1 が押されたら次の文字を出力する。

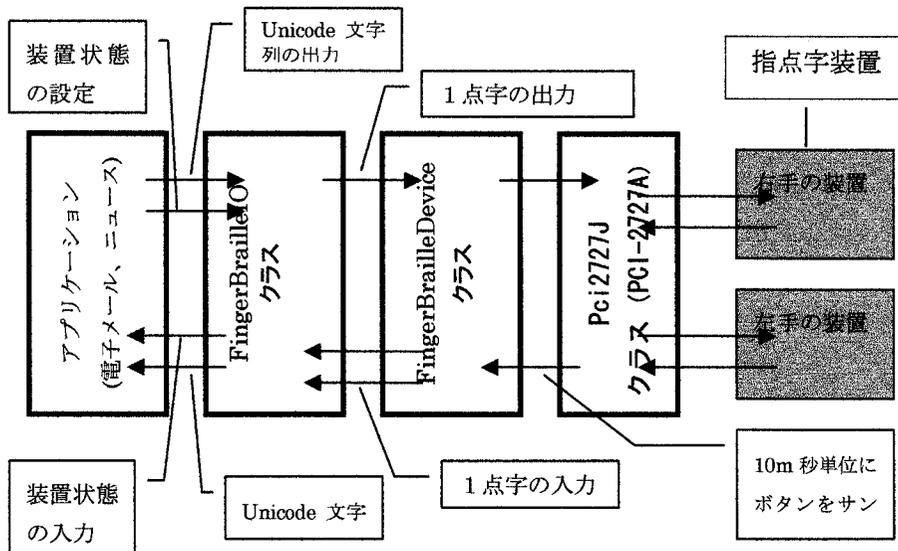


図 3 指点字装置の構成

FingerBrailleIO クラスは、Unicode 文字列を点字に変換して1文字ごとに出力する。点字を読み込み、Unicode 文字に変換して文字列で返す。常に入力可能状態ではあるが、文字列を点字装置に出力中は入力機能を一時停止する。

7. 指点字の多点入力アルゴリズム

指点字では6個のボタンを左右の指で押して、文字コードを入力する。人が指で押す場合、先の実験結果からも分かるように、各指が揃って押されることはなく、少しずつずれる。多数の点を押す場合、点数が多くなるほどそのバラツキは大きくなる。これらのことを踏まえて2つの多点入力のアルゴリズムをテストした結果、ロジカル OR 方式を採用した。

① ヒストグラム方式

一定時間内のサンプリングデータを頻度表にして、最多値のデータを入力データと確定する。⇒機械入力的な規則的遅れを伴うものに適する。

② ロジカル OR 方式

一定時間内のサンプリングデータを各点ごとに時間軸方向に論理和を行い、これを入力データと確定する。⇒人間的な不確定な遅れを伴うものに適する。一定時間とは1文字の有効な入力範囲を意味する。今回は指によるキータッチ入力なので空データ (b5~b0 が 0) から空データ間を文字のサンプリング範囲としている。

これらの2つの方法は指を揃えて押す場合、1点押しから6点押しまで、ほぼ100%の識別率である。しかし、左右の手に20ms~50msのバラツキがある場合は、ヒストグラム方式は誤って識別(表3)する。ロジカル OR 方式は正しく識別する。

表3は「め(6点のすべてを押す)」を入力する場合で、サンプリング周期を約10ms、左右の手でバラツキが起きた例である。正しくは6ビットが1(No.4)でなければならない。このような場合にロジカル OR 方式は効力を発揮す

る。理由は時間方向の遅れを無視するアルゴリズムが効果的に働くからである。なお、「め」の入力時間は約200msであった。

表3 ヒストグラム方式の頻度表

No	回数	ビットパターン
1	11
2	5111
3	1	..1.1111
4	2	..111111
5	1	..1111..
6	5	..111...
7	2	..1.1...
8	31...
確定した文字	111

8. まとめ

盲聾者がネットワークを介して健常者あるいは盲聾者同士でコミュニケーションできるシステムの指点字入出力装置の開発を行った。この装置はタッチ圧やタッチ時間などの調整ができる他、ビーブー音役の振動モータや多数の多様なスイッチを配置し、ハードウェアで必要なヒューマンインタフェース上の様々な工夫が加えられている。基本的にハード上はソフトウェアにより様々な対応が可能な設計となっているが、今回は、指点字をマスターしてすることを前提としてソフトの設計を行った。

最後に、本研究開発は情報処理振興事業協会の委託を受けて開発されたものである。

参考文献

- [1]全国盲ろう者協会だより,11号,pp14-15,2001
- [2]長田、片山、王、北島：盲聾者のための指点字ロボットを用いたコミュニケーションシステムの開発、ヒューマンインタフェースシンポジウム2000、1212,pp41-44,2000.
- [3]長田、片山：指点字会話支援装置の指点字認識特性の改善方法、ヒューマンインタフェースシンポジウム2001、1411,pp93-96,2001.
- [4]坂尻、岡田、竹内、富田、和田：振動を用いた指点字提示方式による盲ろう者用コミュニケーション支援機器の開発、信学技法 HCS99-2,pp11-17,1999