

## 点字の翻訳システムについて

### On the translation system of Braille points

○塚越雅斗, 早坂匡

○Masato Tsukagoshi, Tadashi Hayasaka

秋田大学

Akita University

キーワード: 点字 (Braille) 墨字 (Ink-point) 点訳 (transcribe in braille), 墨訳 (transcribe in ink-point)

連絡先: 〒010-8502 秋田市手形学園町1番1号 秋田大学大学院 通信・制御講座 佐々木研究室  
塚越雅斗, E-mail: tukakosi@uws15.phys.akita-u.ac.jp

#### 1. はじめに

現在, 点訳及び墨訳の補佐をするソフトウェアは多数存在するが, われわれの調べた限りではUNIX系OS上で動作するものは存在しない. 視覚障害者やボランティアの方々の利便性を考えた場合, UNIX系OSで動作するプログラムを開発することにも価値があると思われる. このような観点から我々はLinux(X Window System)上において開発を行い, 基本的な動作を確認した. ここではそのデータ構造やプログラム構成及び使用法について報告する.

今回使用した開発環境とそのバージョンはTable 1に示す.

Linux kernel	2.5.25
X(XFree86)	3.3.6
gcc	2.95.2
Tcl/Tk	8.2

#### 2. 点字について

##### 2.1 点字の構成

今日, 世界中の盲人の常用文字となっている点字は1825年フランスのパリ王立盲学校の教師であったルイ・ブライユによって考案された. 点字のことを「ブレイル(Braille)」というのは彼の名前によるものである. 日本では1890年, 官立東京盲学校教員石川倉次がブライユの6点点字を日本語の仮名文字に翻案した. 現在使われている点字は文章記号などが加えられ改良されたものの, 基本的には岩倉倉次の翻案とは異なっていない. (1, 2)

点字はFig. 1のように縦3, 横2の6点を一つの単位として構成され(一マスと呼ぶ), この一マスが原則として一文字を表す. この六つの点にはそれぞれ名称があり前のマスに近い側の上から, ①の点・②の点・③の点と呼ばれ, 後ろのマスに近い側の上から, ④の点・⑤の点・⑥の点と呼ばれる. これらの点を凸面にすることにより, 指先で読むことができる.

- ① ④
- ② ⑤
- ③ ⑥

Fig. 1 点字の構成と各点の名称

なお、本稿では1990年版「日本点字表記法」<sup>3)</sup>に従い、凸点を‘●’、平らな点を‘-’で表す。

Table 2 清音・撥音・促音・長音

●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
あ	い	う	え	お
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
か	き	く	け	こ
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
さ	し	す	せ	そ
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
た	ち	つ	て	と
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
な	に	ぬ	ね	の
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
は	ひ	ふ	へ	ほ
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
ま	み	む	め	も
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
や	ゆ			よ
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
ら	り	る	れ	ろ
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
わ	ゐ	ゑ	を	

Table 3 濁音・拗音・特殊音などの一部

●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
が	ぎ	ぐ	げ	ご
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
ば	び	ぶ	べ	ぼ
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
きゃ	きゅ	きょ		
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
ぎゃ	ぎゅ	ぎょ		
●●	●●	●●	●●	●●
●●	●●	●●	●●	●●
ぴゃ	ぴゅ	ぴょ		

## 2.2 点字仮名の構成

点字の清音・撥音・促音・長音はTable 2のように一マスで表される。しかし濁音・拗音・拗濁音・拗半濁音などはTable 3のように一マス前置して二マスで表される。当然ながら濁音などで前置される点字は清音などでは使われていないものとなっている。

数字は、数符を前置して表す。一つの数には数符を一つだけ用いて、位取り記数法で記す。アルファベットを日本語の文中で用いる場合、文字には外字符を前置し語句や文は、外国語引用符でくる。

Table 5に示された句読符は、それ自身は一マスで表されるが、その後続く空白部が異なっている。すなわち句点・疑問符・感嘆符は二マスあけ、読点・中点は一マス開ける。

日本の点字には以上の他に外国語引用符・括弧・矢印などがあるが、本稿では未対応部分である。

Table 4 数字・アルファベット

●●					
●●	数符				
●●	●●	●●	●●	●●	●●
●●	●●	●●	●●	●●	●●
1	2	3	4	5	
●●	●●	●●	●●	●●	●●
●●	●●	●●	●●	●●	●●
6	7	8	9	0	
●●	外字符				
●●	●●	●●	●●	●●	●●
●●	●●	●●	●●	●●	●●
a	b	c	d	e	
●●	●●	●●	●●	●●	●●
●●	●●	●●	●●	●●	●●
f	g	h	i	j	
●●	●●	●●	●●	●●	●●
●●	●●	●●	●●	●●	●●
k	l	m	n	o	
●●	●●	●●	●●	●●	●●
●●	●●	●●	●●	●●	●●
p	q	r	s	t	
●●	●●	●●	●●	●●	●●
●●	●●	●●	●●	●●	●●
u	v	w	x	y	
●●	●●	●●	●●	●●	●●
●●	●●	●●	●●	●●	●●
z					

Table 5 句読符

●●----- ●●----- ●●-----	句点 (。)	-----●● -----●● -----●●	読点 (，)
●●----- ●●----- ●●-----	疑問符 (？)	-----●● -----●● -----●●	中点 (・)
●●----- ●●----- ●●-----	感嘆符 (！)		

### 2.3 点字による仮名遣い

点字の基本的な仮名遣いは「現代仮名遣い」にほぼ対応しているが、次の2点が異なっている。第1点は「現代仮名遣い」では「は」「へ」と書き表している助詞を、点字の点字の場合は発音のとおり「わ」「え」と書き表すことである。第2点はう列とお列の長音のうち「現代仮名遣い」で「う」と書き表す長音部分を、長音符を用いて書き表すことである。

### 2.4 墨字と点字

墨字とは点字に対して、普通に書いたり印刷したりした文字の事をいう。

日本語(墨字)はひらがな・カタカナ・漢字・算用数字・アルファベット・の5種類の文字を用いて書き表されているが、一方日本の点字は漢字を用いず、ひらがなとカタカナの区別もない(区別する必要がない)。そのため点字には語の区切り目を明らかにするための分かち書きという書き方の規則がある。

したがって、墨字を点字に訳す場合、まず漢字が混じっている場合それらを全て平仮名に変換し、きちんと分かち書きをした後に点訳をする必要がある。

### 2.5 点字の分かち書き

本節では分かち書きの基本的な説明とそれによる問題点について述べる。

元来分かち書きは、意味を理解しながら速く読むことができるようにするために行うものである。それ故、文の単位ごとに区切ることによって分かち書きを手がかりとして、それぞれの語の意味の理解はもとよりのこと、文節と文節の関係を明らかにし、文全体の意味を速く理解できるようにするといった目的を持っ

ている。したがってその規則のかなりの部分は個人的、または地域的なものであった。

前述の「日本点字表記法」では、これを統一するため、分かち書きの詳細な規則を定めている。ここにその一部を抜粋する。『文の単位は、意味を表す自立語と、役割や立場を表す助詞や助動詞から成っており、一般に文節と呼ばれている。そこで自立語は、文の単位となるから前を区切り、助詞や助動詞は、単独で文の単位とはなれないので自立語に続けることを分かち書きの第1の原則とする。』

すなわち分かち書きとは基本的には自立語の前に区切り目を入れればよいということである。Table 6にその例を示す。

ただしTable中の‘□’は、区切り目を表す(Table 7も同様)。

Table 6 分かち書きの例1

- 涼しい風が木の葉を揺らせた  
→ スズシイ □ カゼガ □ コノハラ □ ユラセタ
- 我輩わ猫である  
→ ワガハイワ □ ネコデ □ アル

分かち書きがこの原則のみですむとすれば、自立語を全て登録しておき処理をすることで対処が可能である。しかしTable 7に示すようにかなり微妙な表現もある。

Table 7 分かち書きの例2

- 彼わ一人寂しく亡くなった  
→ カレワ □ ヒトリ □ サビシク □ ナクナッタ
- 君のせいで寂しくなくなった  
→ キミノ □ セイデ □ サビシク □ ナク □ ナッタ
- 全力投球する  
→ ゼンリョク □ トーキュー □ スル
- 断固反対する  
→ ダンコ □ ハンタイスル

ここに挙げたものはその一部にすぎない、現時点ではこの対処方法の方針はできておらず、したがって本稿では分かち書きの処理に関して基本的に未対応である。当座の対応策としては、墨字を入力する時点で分かち書きにしておくと言う方法により対処することとした。

### 3. データ構造と処理の概要

#### 3.1 点字の8進数コードへの変換

いままでで見たように点字は通常6点で表される。本稿ではこれを左右3点づつにわけ3ビットの8進数としてとらえることにより点字を8進数2文字のコードとして取り扱う。以後、本稿ではこれを点字コードと呼ぶ。たとえば、“あ”を表す一文字の点字は点字コードでは‘10’、“が”を表す二文字の点字は点字コードでは‘0214’となる。

この方式の利点は点字と点字コードの変換が計算により容易に行えることである。

#### 3.2 構造体を用いた墨字と点字コードの相互変換

点字コードと墨字との変換は、構造体に対応する点字コード及び墨字を格納しそれらを検索することにより行った。構造体とは操作しやすいように一つの名前までまとめられた、一つまたはそれ以上のおそらくは異なった変数の集まりである。(Pascal等では構造体はレコードと呼ばれる。)

#### 3.3 濁音などの変換

一マスで墨字一文字を表す文字(静音、撥音、促音、長音)の変換は非常に容易である。しかし濁音や半濁音は二マスで墨字一文字、拗音などは二マスで墨字二文字を表す。これらの場合は特殊な変換を行う必要がある。まず濁音など一文字を点字二マスの点字コードに変換する場合、この場合は通常の処理で問題はない。次に点字二マスの点字コードを墨字に変換する場合、この場合はまず始めの一マス分の点字コードを検索し該当するものが無かった場合、次の一マスを足し

た計二マス分の点字コードを検索する。最後に拗音などの墨字二文字を点字二マスに変換する場合であるが、この場合は最初の一文字をみただけでは清音などであるか拗音などであるかの判断はできない。そこで検索前に常に次の文字を確認し、その文字が“ゃ”などの拗音となる文字であるならば二文字を含め検索するという方法をとった。

### 4. プログラムの構成と使用例

点字を墨訳、及び墨字を点訳する場合、一度点字データに変換してから翻訳を行うわけだが、前章でみたように点字↔点字データの変換と墨字↔点字データの変換は処理方法が全く異なる。そこで本稿では点字↔点字データの変換を行うプログラムと墨字↔点字データの変換を行うプログラムの2種類をプログラミング言語C<sup>4)</sup>により作成した。

以下ではこれら二つのプログラムの概略と使用法について説明する。なおプログラム中の表示では点字の凸面は‘o’で、平らな面は‘-’で表している。

#### 4.1 墨字と点字コードの変換(実行名は j2c)

本プログラムは墨字もしくは点字コードを入力としてうけとり、前章で述べた構造体による辞書を検索して点字コードもしくは墨字として出力する。この際入力はファイル及び標準入力(stdin)のどちらからでも可能であり、出力もまたファイル及び標準出力のどちらからでも可能である。入出力の切替え及び変換の切替えは引数によるオプションで行う。

Table 8 オプション表

-J	点字コードを墨字に変換
-C	墨字を点字コードに変換
-O	ファイルに出力をする

ただし、-J と -Cのオプションはお互い排他的に存在し、またどちらのオプションも無い場合は-Cのオプションがついているものとして振舞う引数にオプション以外ない場合(入力ファイルが指定されていない場合)標準入力から読み込み処理をする。Fig. 2は標準入力から墨字を入力し点字コードを標準出力に出

力したときの“j2c”の実際の動作である。

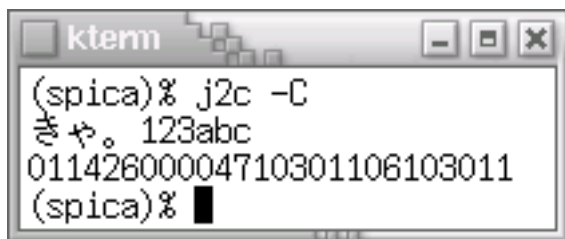


Fig. 2 実行例(j2c)

#### 4.2 点字と点字コードの変換(実行名は t2c)

本プログラムは点字もしくは点字コードのデータを入力として受け取り、‘o’と‘-’のパターンと8進数点字コードとを相互に変換するものである。このプログラムも前節の“j2c. o”と同様に標準入出力及びファイルのどちらからでも処理が可能になっている。このプログラムも入出力の切替え及び変換の切替えは引数によるオプションで行う。

Table 9 オプション表

-T	点字コードを点字に変換
-C	点字を点字コードに変換
-O	ファイルに出力をする

ただし、-T と -Cのオプションはお互い排他的に存在し、またどちらのオプションもない場合は-Tのオプションがついているものとして振舞う。引数にオプション以外ない場合(入力ファイルが指定されていない場合)標準入力から読み込み処理をする。

#### 4.3 一連の処理及び漢字混じりの文章の問題

プログラムを二つに分けた性質上「墨字 → 点字」の処理や「点字 → 墨字」の処理を一度に行うことはできない。そこでLinuxコマンドライン上でパイプを使い“j2c”の出力を直接“t2c”の入力(あるいはその逆)にすることによって処理をする。その例をFig 3に示す。

漢字混じりの文の場合、漢字を平仮名に直す必要がある。Linuxには“kakasi”というツールがあり、これを使用することにより、漢字混じりの文の点訳が可能である。

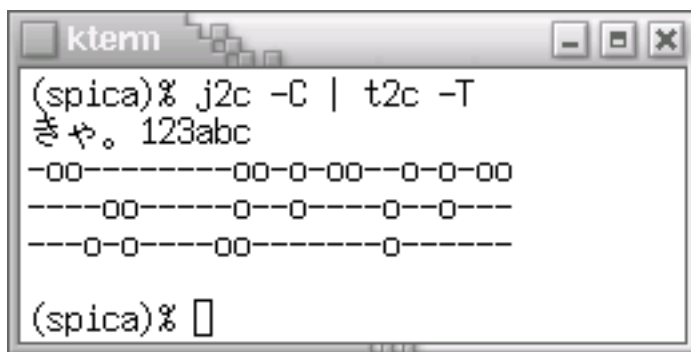


Fig. 3 実行例

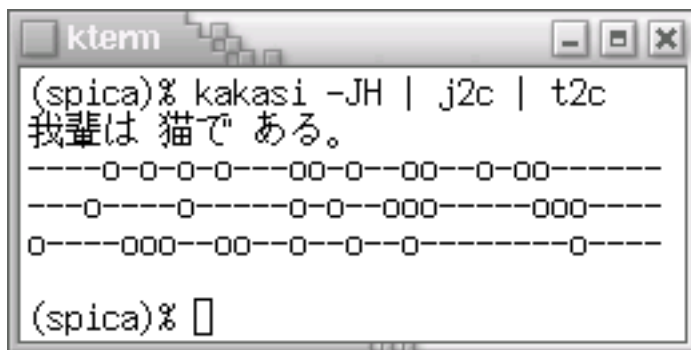


Fig. 4 実行例(kakasiを使用)

### 5. GUIを用いた使用例

前章の使用例は、実行する際、全てコマンドラインから行う必要がある。しかしこの場合Linuxに慣れていないと使いにくい面がある。そこで本章ではコマンドを知らなくても使うことができるようなTcl/TkによるGUIを用いた例を紹介する。

Fig. 6 は実際の動作の画面である。2つあるウィンドウの上部が入力ウィンドウ下部が出力ウィンドウである。上部メニューの[ファイル]には[開く]及び[保存]、そして[終了]の3つのコマンドがあり、ファイルから読み込む場合[開く]コマンドをマウスで選択し、ファイル選択ウィンドウから読み込むファイルを選ぶことにより、ファイルの内容が入力ウィンドウ(Fig. 5)に表示される。なお入力ウィンドウは編集可能であり読み込んだファイルを編集した後点訳及び墨訳をすることが可能である。

墨訳もしくは点訳をする場合、上部メニューの[変換]から[墨字を点字へ]を選ぶと点訳、[点字を墨字へ]を選ぶと墨訳が実行され、実行結果は下部出力ウィンドウに表示される。結果を保存する場合は[ファイル]メ

ニューの[保存]コマンドを選択することにより、ファイルに保存することができる。

なお、ここで使用したXのウィンドウマネージャはSawfish、ターミナルタイプはktermである。

## 6. 問題点及び今後の課題

ここまでで基本的な点訳及び墨訳については完成したが未だ問題点も多い、以下それら問題点について述べる。

### 6.1 未対応の符号及び変換の効率化

本稿では外国語引用符や矢印などいくつかの点字に未だ未対応である。これらは基本的に辞書に登録するだけで対応可能であるが、変換の効率や速度などの面において辞書の作成方法に再考の余地があると考えられる。

### 6.2 分かち書きについて

2. 5章でも述べたが現時点では分かち書きに対する対処は全く行っていない。Table 6の様な文章の処理は比較的単純なため、対処することは可能であると思われるが、Table 7の様な文書に対する処理に関しては見通しすら立っていない状態である。

### 6.3 GUIの強化

5章で使用したアプリケーションを使用することでLinuxにあまり詳しくない人でも容易に点訳及び墨訳が可能になる。しかし、本稿で紹介したこのアプリケーションの完成度はまだまだ低いといわざるをえない。

たとえば、このアプリケーションではファイルからのみでなく、直接墨字や点字を入力して点訳及び墨訳を行うことができるが、墨字を入力するときは容易であるが点字を入力する場合非常に労力を要するという問題がある。これについては、ある特定の6個のキーを点字の各点に割り当て、キーボードを利用して入力することができるようにすることも考えられる。



Fig. 5 ファイル選択画面

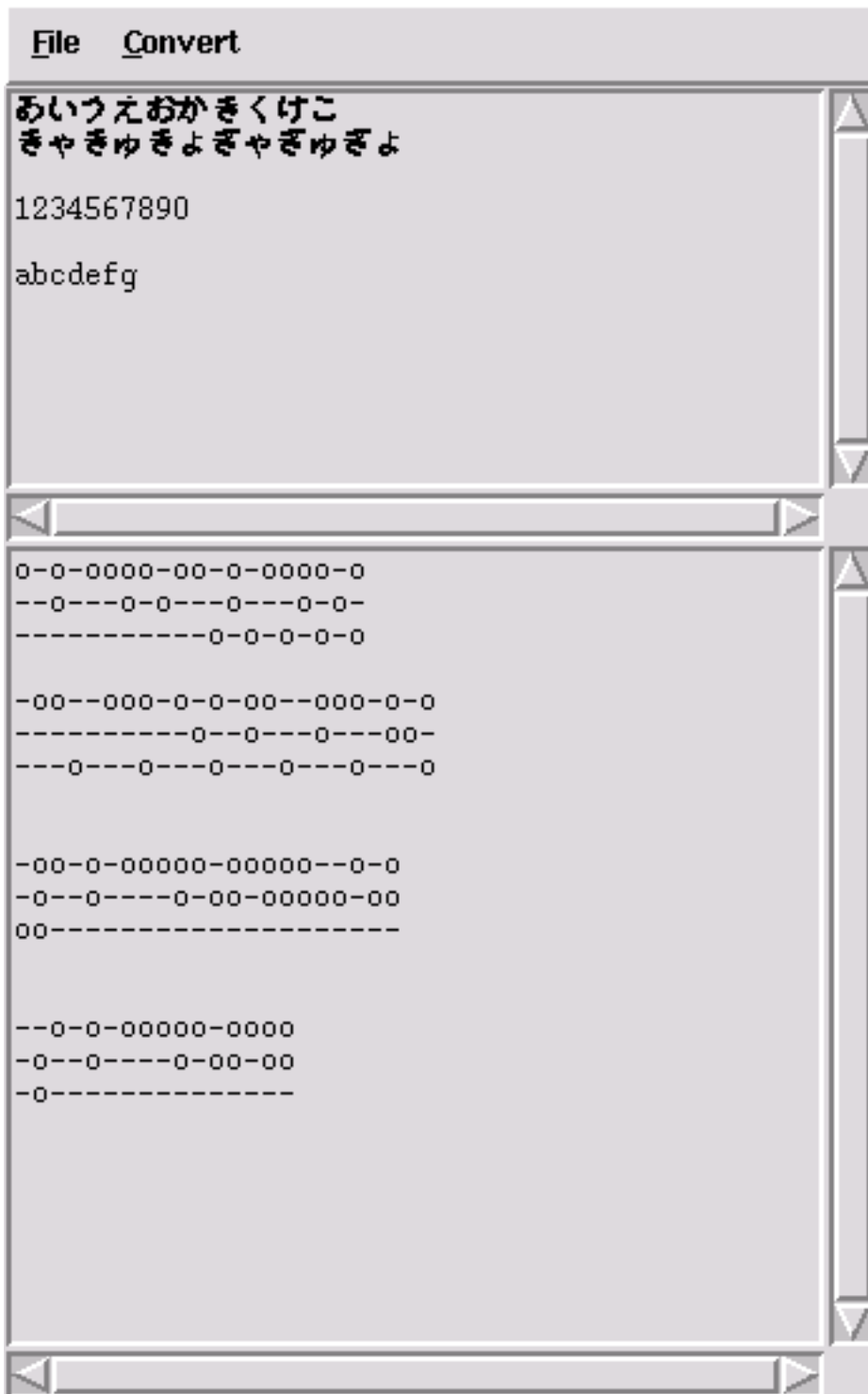


Fig. 6 実行例(GUI使用)

以上の問題を解決していくことが今後の課題となるが、その際実際に点訳墨訳を行っている方々に実際に利用していただき、御意見を聞くことは欠かすことができないと考えている。

#### 謝 辞

本研究をすすめるにあたり、教育文化学部障害児教育講座 大城英名助教授には、点字に関する資料を貸していただいた。ここに記して感謝の意を表する次第である。

#### 参考文献

- 1) 遠藤謙一:点字・点訳入門, 廣濟堂出版(1993).
- 2) 本間一夫, 岩橋明子, 田中農男(編):点字と朗読を学ぼう, 福村出版(1991).
- 3) 日本点字委員会:日本点字表記法1990年版, 日本点字委員会.
- 4) B.W.カーニハン, D.M.リッチー(石田晴久訳):プログラミング言語C第2版, 共立出版株式会社(1999).
- 5) B.B.ウェルチ(西中芳幸, 石曾根信訳):Tel/Tk入門第2版, 株式会社ピアソン・エデュケーション(1999).