制限のない日本語文字表現手法の有用性証明

代表研究者　　Antoine BOSSARD　　神奈川大学　理学部　准教授
共同研究者　　金　子　敬　一　　東京農工大学　工学府　教授

1 はじめに

コンピュータで日本語の書記系における字形を表現することは，長年に亘って研究されてきた難しい問題である．さらに，文字表現だけでなく，文字入力も重要な問題である．後者の問題に対する解決策については，例えば BOSSARD (2016)に詳しい．この研究プロジェクトでは，前者の問題，すなわち，日本語の書記系が用いる字形の，コンピュータシステムにおける表現に焦点を当てた．日本工業規格(JIS)による JIS コード(JAPANESE INDUSTRIAL STANDARDS COMMITTEE 1969)と国際文字コードである Unicode (THE UNICODE CONSORTIUM 2007)は，日本語に対する 2 つの主要な文字コードである．いずれも複数の観点から利点を持つ一方，重大な欠点を持つ．JIS コードに対しては，コード化された文字数は高々10,000 字形ほどで，実際に存在する字形の数は，この数倍となる．Unicode に関しては，このコード化手法は，「統一化」に依存し，数多くの書記系の字形をカバーする．例えば，我々の場合，中国語や日本語に対する文字が混在する．このアプローチは，単一のコード化で複数言語の文書を扱うことができるとった利点を持つ一方，Unicode の 1 つの重大な問題は，ユーザビリティである．すなわち，このコードに特定の字形を配置することは難しい．

この研究プロジェクトの目的の 1 つは，日本語に対する無制限の文字コード化の提案である．これは，文字の制限や上述の現存する解決策の問題点を解消する．厳密に言うと，提案コードは，無制限の文字数を可能にし，コードへの新しい文字の追加に関する柔軟性を保持し，コード体系における文字検索を簡単にしなければならない．本研究は，日本語が用いる漢字に焦点を当てる．したがって，本研究は，中国語（簡体字および繁体字）やハンジャ（한자，韓国語における漢字），チュノム（Chữ Nôm，ベトナム語を表記するために漢字から派生して作られた文字）などに適用可能である．

2 準備

本節では，漢字のいくつかの重要な特徴について振り返る．この主題に関する詳細な議論は，BOSSARD (March 2018) によってなされている．

まず，漢字は，その部首によって分類可能であり，実際にそうされている．現在の分類体系では，214 の部首がある．そして，ときには，部首の割り当てに関して曖昧さが存在するけれども，各文字には固有の部首が割り当てられている．例えば，「砂」という文字の部首は，「石」であり，その「石」自体も「石」自身を部首とする文字である．

第二に，文字は，1 つ以上の筆画からなり，これからの筆画は特定の順序で描かれる．文字の筆画は，有限集合として表すことができる．例えば，文字「木」は，横画，縦画，斜画（左），斜画（右）の 4 つの筆画からなり，この順に描かれる．

第三に，同じ文字は異なるように書くことができ，これらの字形を，異体字と呼ぶ．例えば，20 世紀に政府が実施した新字体の導入によって，いくつかの文字は，単純化された．これにより，いくつかの文字に対して，新旧の異体字が存在するようになった．例えば，文字「体」には，古い異体字「體」が存在する．この場合，発音や意味は不変で，単純化によって文字の形だけが影響を受けている．異体字には，いくつかの種類が存在する．詳しくは BOSSARD (March 2018) を参照．

3 制限のない日本語文字表現手法(UCEJ)

本節は，BOSSARD と KANEKO による論文(July 2018, January 2019, March 2019)の注釈つき概要である．

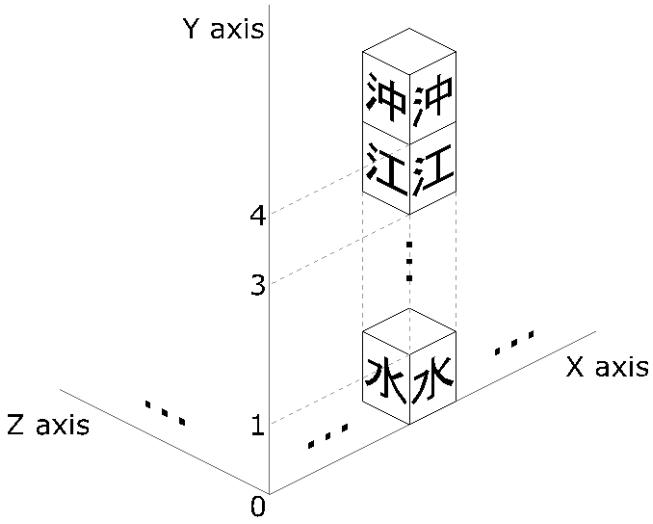

図 3-1.1: 3 次元コード体系の例示.

3-1 定義

の場合，発音や意味は不変で，単純化によって文字の形だけが影響を受けている．異体字には，いくつかの種類が存在する．詳しくは BOSSARD (March 2018) を参照．提案した制限のない日本語文字表現手法(Unrestricted Character Encoding for Japanese, UCEJ)は，3 軸で定義される．したがって，各文字は，それぞれが 1 軸に対応する 3 つの座標でコード化される．UCEJ は，以下の論理的な体系を持つ:

- X 軸の座標は，文字の部首を表現するために用いられる．
- Y 軸の座標は，部首の画数を除いた，文字の画数を表現するために用いられる．
- Z 軸の座標は，異体字を区別するために用いられる．

「水」を部首にもつ 2 つの文字を例に，この体系を図 3-1.1 に例示する．

上述したように，現在の分類体系では文字の部首は 214 ある．したがって，ある文字の X 座標は，[0, 213] の範囲に入る．各文字に対して，その異体字の 1 つが規範的な文字と指定されている．この規範的な文字を

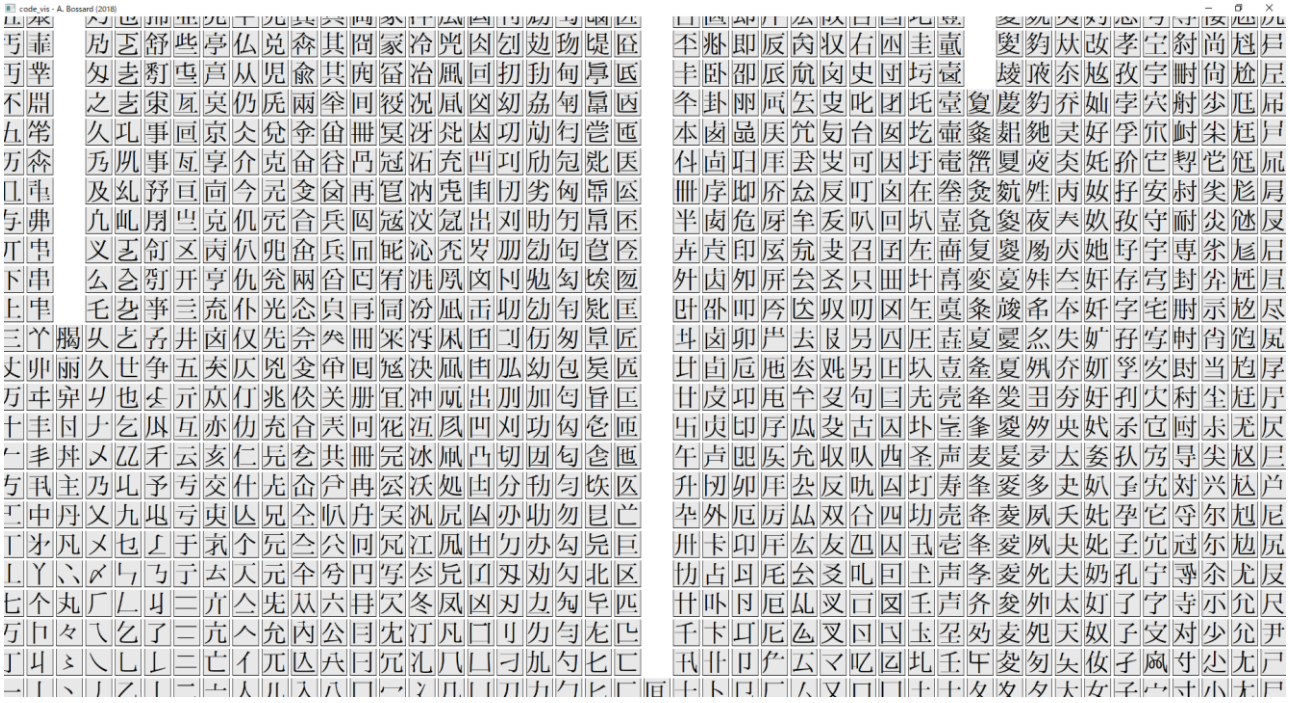

図 3-2.1: XY 平面上のレンダリングしたコードの抜粋.

正字体と呼ぶ．これは，例えば，一連の政府の単純化によって導入された新たに単純化された異体字である．さらに，ある同一の文字の異体字は，同じ部首を持つことを仮定している．

Y 軸に関しては，明らかに，同じ部首，同じ画数を持つ複数の文字が存在する(例えば，「杉」と「村」は，いずれも同じ部首「木」と同じ画数 3 を持つ．部首の画数は除外することに注意)ので，より詳細に説明をする．各文字は，異なるコードポイント(すなわち，3 次元座標)に写像しなければならないので，そのような文字を区別するためには，付加的な情報が必要とされる．このため，小数点以下 6 桁の小数を用いる．この数は，0 から出発して，新しい文字を登録する際に増やしていく．

3-2 実験

コード体系が定義できたので，提案した 3 次元ノードレイアウトを実現する文字データベースを実際に構築した．この目的のため，我々は，独立行政法人情報処理推進機構(IPA)が提供する文字データベース「文字情報基盤データベース」に依存した．具体的には，IPA のデータベースにある文字の属性(例えば，文字の部首や画数)を利用して，我々は，IPA のデータから UCEJ コードのメモリ表現を自動的に生成し，56,875 の漢字を写像した．この実験に用いたコンピュータ(Intel i7-6700 CPU, 16 Gb RAM, Windows 10 64-bit OS)上で，IPA のデータベースを元に実施した我々のデータベース構築には，約 2 時間 30 分かかった．

次に，我々は，構築したデータベースの 3 次元可視化を実現した．可視化ソフトウェアは，OpenGL グラフィックスライブラリを用いて実現し，各文字を，字形を持った立方体へ写像した．XY 平面上のレンダリングしたコードの抜粋および Z 軸上の文字の抜粋を，それぞれ図 3-2.1，3-2.2 に示す．

この実験には，複数の目的があった．まず，この実現により，提案コード体系の実行可能性を示したかった．第二に，コードの中身に関して，文字の分布に関する定量的情報，すなわち，データベースには，いく

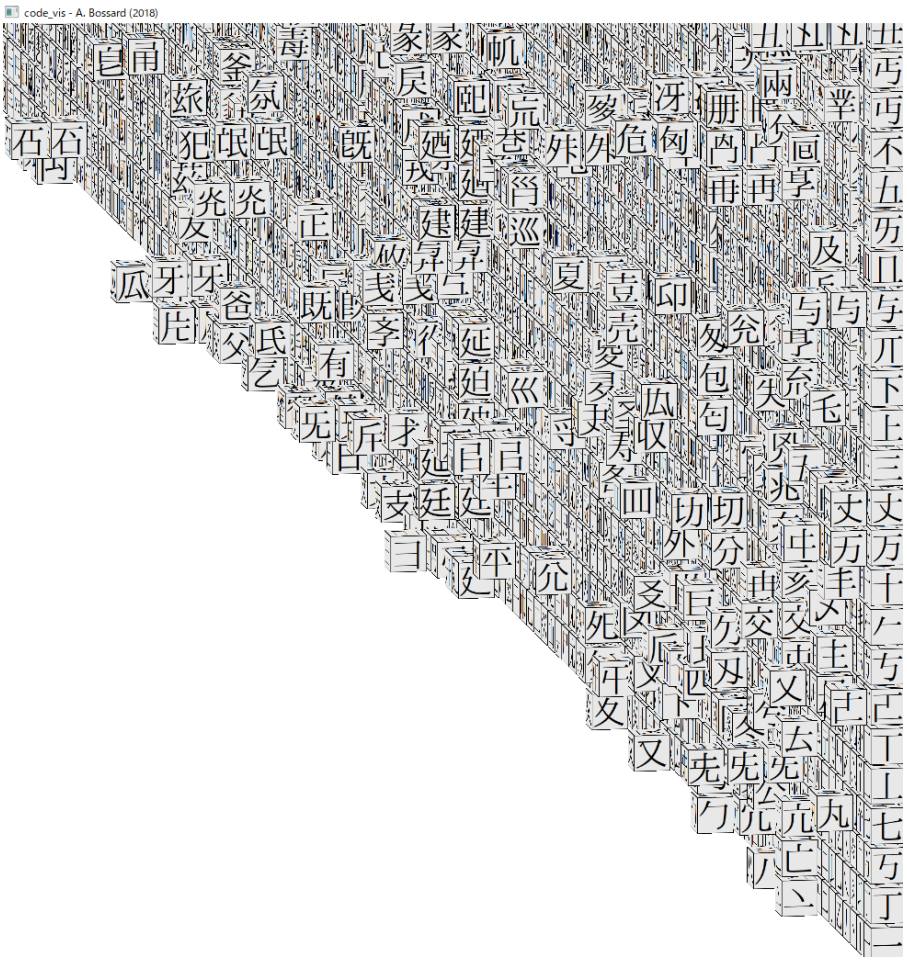

図 3-2.2: Z 軸上の文字の抜粋.

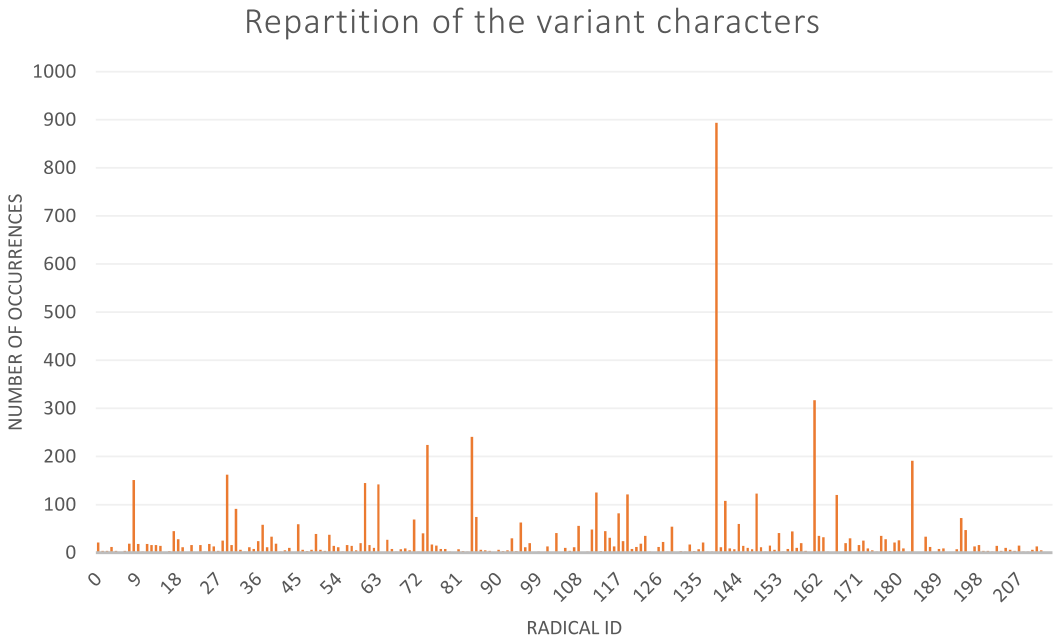

図 3-3. 2：実験結果：異体字の分布.

つ正字体があり，異体字があるか，といったことを収集することが重要であった．第三に，構築したデータベースの 3D グラフィックス可視化を実現することで，コード内の文字探索といった，システムの適用可能性とユーザビリティを証明することを目的とした.

3-3 結果と議論

本節では，得られた結果を示す．まず，図 3-3. 1 と 3-3. 2 は，それぞれ，規範となる文字と異体字の分布を示す. 後者の図によれば，異体字の分布が一様でないことが分かる. 最も多くの異体字を持つ部首は，「艸」であり，894 の規範的でない文字をもつ.

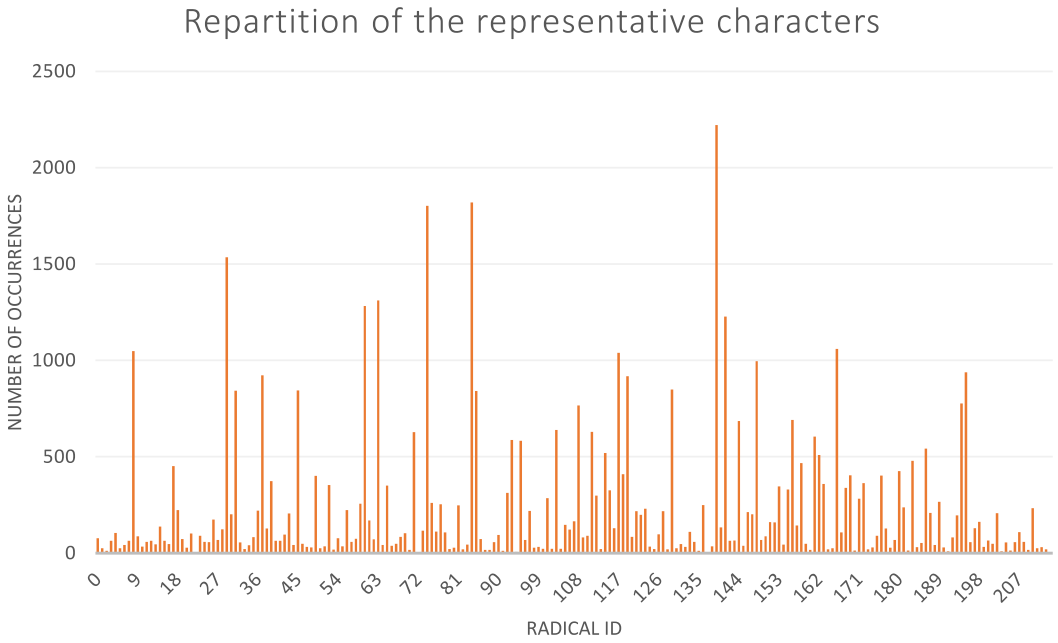

図 3-3. 1：実験結果：規範となる文字の分布.

4 将来の UCEJ の展開に向けた TeX における日本語支援の調査

提案した UCEJ コードの形式的な記述は，第一段階であった．第二段階として，その評価が，次に重要な目標であった．最後の段階として，UCEJ コードの適用可能性を示すことも不可欠である．そのため，我々は，文書整形システムである TeX に焦点を当て，その日本語文書処理能力の評価から始めた．この主題は幅広いため，我々は，TeX の索引システムを考察することから出発した．これは，TeX のオリジナルの検索システムが，国際化のための支援を欠いていることが良く知られているためである．本節は，Bossard と Kaneko (October 2018) の注釈つき概要である．

4-1 TeX による索引付けについて

従来，TeX (LaTeX) システムでは，文書の索引は，*makeindex* プログラムとパッケージの組合せで，半自動的に生成される (Lamport 1987)．TeX のソース文書の著者が文書全体を通して索引の項目を手動で宣言する必要があるため，完全に自動化されてはいない．索引生成過程を図 4-1.1 にまとめる．LaTeX プログラムは，ユーザが宣言した索引項目から，IDX ファイルを生成し，次に，これを *makeindex* プログラムへの入力として使用する．LaTeX プログラムは，その結果として得られる IND ファイルをそのまま TeX 文書のソースコードへ挿入し，実際の文書索引を必要とされている場所に表示させる．

本研究では，我々は，日本語文書支援，具体的には，UCEJ による支援に関連しているので，まず，標準に索引付け機構である *makeindex* の多言語能力を調査した．*makeindex* は，ASCII コード用に設計されたため，Unicode に対応していなかった．しかし，我々の調査から，いくつかの技術的工夫から，*makeindex* が日本語文書や多言語文書に対して，部分的な支援を提供するということが示された．

例えば，¥index{てんわ@電話}のように仮名文字に依存して日本語の索引項目を宣言することができる．我々は，平仮名を用いた．濁点や半濁点，拗音といった仮名文字の派生は，無視する必要があった．これが，先の例で，索引のキーとして，「でんわ」ではなく，「てんわ」を用いた理由である．このように，生成された索引項目は，正しく順序付け（整列）され，文書の索引に正しく表示される．仮名文字に依存する，この工夫は，XeTeX (Robertson と Hosny 2017) のように，Unicode を利用可能なように拡張された TeX システム上で動作する．

4-2 *kameindex* の導出

単純な索引生成の場合，上述の国際化技法で満足することができる．しかし，索引項目グループの生成に関連して，いくつかの重大な問題がある．例えば，頭文字による索引項目の分割である．*makeindex* は，1 文字が 1 バイト（すなわち，ASCII コード）であると考えるので，ASCII コードでない索引項目のグループ化や索引項目グループの文字の検出に失敗する．

こういった重大な問題を解決するため，TeX に対する新しい索引手法を提案した．この手法は，Unicode を扱うことができ，したがって，日本語を完全に扱うことができる．重要なこととして，この新しい索引システムである *kameindex* は，もとの *makeindex* システムと完全互換であり，したがって，ユーザは，LaTeX が生成した IDX ファイルの入力先として，単に，*makeindex* プログラムの代わりに，*kameindex* プログラムを指定すればよい．その結果，IND ファイルが生成され，TeX システム（および LaTeX の *makeindex* パッケージ）とは完全に透過である．

索引項目のグループ化は，*makeindex* にスタイルファイルを指定することによって実現される (Chen と Harrison 1988, Chen 1991)．我々は，*makeindex* が対応するスタイルの特徴のサブセット，厳密にいえば，

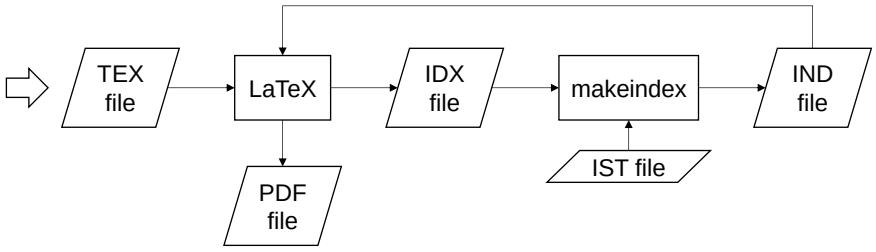

図 4-1.1: *makeindex* ユーティリティによる索引生成．

headings_flag, heading_prefix, heading_suffix, プリアンブル, およびポストアンブルのディレクティブを *kameindex* に実現した. これらの規定値は, *makeindex* プログラムによって設定されているものに設定した. *kameindex* プログラムは, Racket 言語(FLATT 2012)を用いて実現し, オープンソースソフトウェアとして公開した(https://www.sci.kanagawa-u.ac.jp/info/abossard/kameindex/). そのソースとバイナリは, 公開後約 6 カ月の間に既に併せて 100 回以上ダウンロードされている.

4-3 結果と議論

例として, *makeindex* と *kameindex* のそれぞれによって生成された索引の比較を図 4-3.1 に示す. 日本語だけでなく, ラテン, キリル, ギリシア, ハングルといった他の文字も示している. *kameindex* を用いると, *makeindex* とは異なり, 索引項目が正しくグループ化されており, グループ名も正しく検出されていることが分かる. さらに, 2 つのグループが'A'と名付けられていることに気づく. 1 つは, ギリシア文字の'α'で始まる要素であり, もう一つは, キリル文字の'а'で始まる項目である.

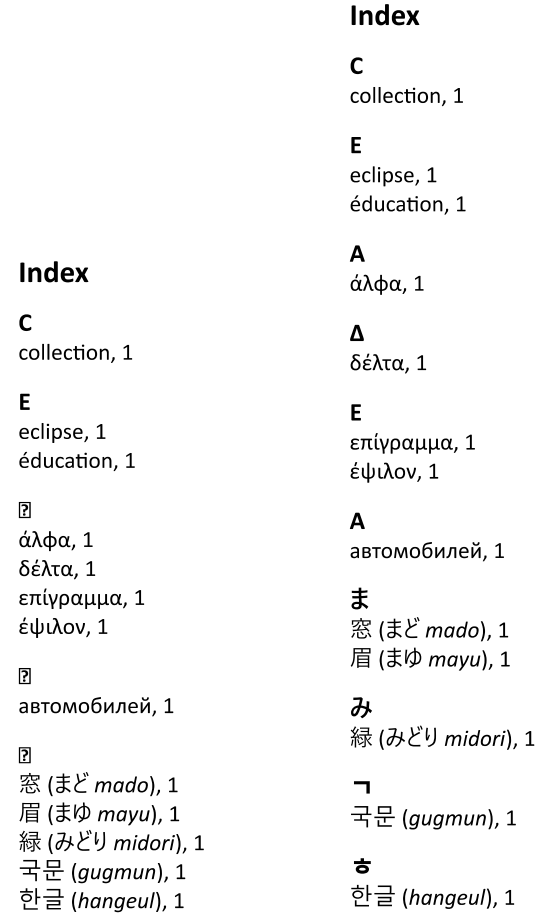

図 4-3.1: *makeindex*(左)と *kameindex*(右)を用いた索引生成.

kameindex と *makeindex* の比較に加えて, 我々は, Unicode を扱うことのできる索引手法である texindy (SCHROD 2004)と *kameindex* を比較した. 結果として, 主に以下の四つのことが分かった:

- *kameindex* とは異なり, *texindy* は, 主流の相互参照システムである hyperref に対応していない.
- *kameindex* とは異なり, *texindy* は, 日本語, 中国語, 韓国語に対応していない.
- *texindy* の使用法は, kameindex の使用法に比べて複雑である.
- *kameindex* とは異なり, *texindy* は, ユーザによる手動の言語選択を必要とする.

もちろん, *texindy* は, *kameindex* が持たない特徴を持つ.

5 結論

Unicode を利用可能なように拡張された TeX システム上で動作する．我々は，本研究プロジェクトの初年度に日本語に対する制限のない文字コードの形式的な提案，およびその評価，有用性に焦点を当てた．前者については，すでに何度かの改善を済ませた一方，後者の二つの問題については，いくつかの理論的および実験的評価を実施したにもかかわらず，依然として現在進行中の研究である．有用性については，将来の UCEJ の応用として文書処理，具体的には文書整形システムである TeX の問題を考え始めたところである．国際化の支援を欠くことで良く知られているため，我々は，LaTeX の索引システムに焦点を当てた．

最初の一年で，我々は，UCEJ コードの定義を記述し，改善することができるようになった．その一方，その適用可能性の証明を含めた評価は始まったばかりであり，UCEJ 文書の可視化，編集，TeX の支援といった重要な作業は，今後の課題として残っている．提案した UCEJ コードのユーザが実際に参加する形で，さらなる評価実験を行うことも重要であろう．さらに，実現した文字データベースを要約して，3 次元辞書を作ることも非常に有望なアイデアである．

謝辞

著者らは，公益財団法人電気通信普及財団より受けました本研究プロジェクトに対する助成について，心から感謝致します．

【参考文献】

Antoine Bossard, aIME: a new input method based on Chinese characters algebra, *Proceedings of the 15th IEEE/ACIS International Conference on Computer and Information Science (ICIS), Studies in Computational Intelligence (Springer)*, Vol. 656, pp. 167-179, Okayama, Japan, June 2016.

Antoine Bossard, *Chinese Characters, Deciphered*, Kanagawa University Press, Yokohama, Japan, March 2018.

Antoine Bossard and Keiichi Kaneko, Proposal of an unrestricted character encoding for Japanese, *Proceedings of the 13th International Baltic Conference on Databases and Information Systems (Baltic DB&IS), Communications in Computer and Information Science (Springer)*, Vol. 838, pp. 189-201, Trakai, Lithuania, July 2018.

Antoine Bossard and Keiichi Kaneko, Experimenting with makeindex and Unicode, and deriving kameindex, *Proceedings of the GuIT meeting 2018, ArsTeXnica*, Vol. 26, pp. 55-61, Rome, Italy, October 2018.

Antoine Bossard and Keiichi Kaneko, Unrestricted Character Encoding for Japanese, *Databases and Information Systems X (in Frontiers in Artificial Intelligence and Applications series)*, Vol. 315, pp. 161-175, IOS Press, Amsterdam, Netherlands, January 2019.

Antoine Bossard and Keiichi Kaneko, Refining the unrestricted character encoding for Japanese, *Proceedings of the 34th ISCA International Conference on Computers and Their Applications (CATA)*, Vol. 58, pp. 292-300, Honolulu, HI, USA, March 2019.

Pehong Chen and Michael A. Harrison, Index Preparation and Processing, *Software: Practice and Experience*, Vol. 19, No. 9, pp. 897-915, 1988.

Pehong Chen, makeindex(1): makeindex – a general purpose, formatter-independent index processor, Unix man page (https://linux.die.net/man/1/makeindex), December 1991. Last accessed August 2018.

Matthew Flatt, Creating languages in Racket, *Communications of the ACM*, Vol. 55, No. 1, pp. 48-56, 2012.

Japanese Industrial Standards Committee (JISC): 7-bit and 8-bit Coded Character Sets for Information Interchange (7ビット及び8ビットの情報交換用符号化文字集合, in Japanese), 1969.

Leslie Lamport, *MakeIndex*: an index processor for LaTeX, Package documentation (https://ctan.org/pkg/makeindex), February 1987. Last accessed August 2018.

Will Robertson and Khaled Hosny, The XeTeX reference guide (https://ctan.org/pkg/xetexref), September 2017. Last accessed August 2018.

Joachim Schrod, texindy(1): texindy – create sorted and tagged index from raw LaTeX index, Unix man page (http://xindy.sourceforge.net/doc/texindy-man.pdf), May 2004. Last accessed August 2018.

The Unicode Consortium, *The Unicode Standard 5.0*. Addison-Wesley, Boston, MA, USA, 2007. More recent versions accessible online at http://www.unicode.org/versions/latest/. Last accessed April 2019.

〈発　表　資　料〉

題　　名	掲載誌・学会名等	発表年月
Proposal of an unrestricted character encoding for Japanese	Proceedings of the 13th International Baltic Conference on Databases and Information Systems (Baltic DB&IS), Communications in Computer and Information Science (Springer), Vol. 838, pp. 189-201	July 2018
Experimenting with makeindex and Unicode, and deriving kameindex	Proceedings of the GuIT meeting 2018, ArsTeXnica, Vol. 26, pp. 55-61	October 2018
Unrestricted Character Encoding for Japanese	Databases and Information Systems X (in Frontiers in Artificial Intelligence and Applications series), Vol. 315, pp. 161-175, IOS Press, Amsterdam, Netherlands	January 2019
Refining the unrestricted character encoding for Japanese	Proceedings of the 34th ISCA International Conference on Computers and Their Applications (CATA), Vol. 58, pp. 292-300	March 2019