修士論文

漢字構造を利用する漢字入力方式とそのデータ作成のための漢字構造自動判別の研究

平成 30 年 2 月 1 日

指導教員　相田仁教授

東京大学大学院工学系研究科

電気系工学専攻

37-166447　金子隆佐

概要

漢字は，その特性ゆえに文字数が厖大でまた限りなく増加し得るものであり，国際的文字集合の Unicode には現在 9 万字弱の漢字が登録されている．この漢字の大半は実のところ日本語の読みが設定されていない．漢字を入力する手段として一般的な読みを用いた入力手段のみでは，日本語の読みが存在しなかったり同音異字の多い文字だったりした場合に，入力が不可能であったり困難であったりする．そのためにますます多数の漢字が文字集合上で利用可能となってもアクセス手段が大変に限られてくる．

そこで本研究では，漢字がどのような部分文字から成っているかという構造の情報を利用可能な漢字入力方式を提案する．同時にそのシステムで利用するための構造データを漢字の画像から解析するためのシステムの構築を行う．

漢字構造解析では，画像入力から構造データの出力までを三つの段階に分けて，新規手法も検討しながら開発を行った．結果として全自動解析までは実現しなかったものの，漢字の知識があまり無くても使用できる漢字構造解析システムの開発に成功した．

漢字入力方式は，現在主流の読みによる方式の上で使用する補助手段を想定するが，この方式を有効に使用すれば一般に打鍵数は減少し，またよく知っている漢字などで構成されていれば，手書きパッドに対しても時間的優位性を持つことが示された．

目次

第 1 章

序論

1.1 研究目的

日本のパーソナルコンピュータ保有率は今や世帯数でみると 76.8 % に達し，スマートフォンやタブレットも普及が進み（平成 27 年通信利用動向調査より），情報機器において文字を入力する機会はますます増え，手書きの機会は更に減るとみられている．

文字の中で特に漢字の入力においては，ラテンアルファベットやアラビア文字などの表音文字の文字入力とは異なる問題が提起される．その原因は漢字の文字数の厖大さと，その数が原理的に無限であることにある．

現在主流の国際的な文字コードである Unicode には漢字が 9 万字弱含まれており，また漢字はその造字法ゆえにいくらでも数が増える可能性がある．これらをキーボードから入力する場合，主流の 106 キーボード（通称 JIS 配列，JIS X 6002）では，キーの数が漢字数に比べて圧倒的に少ないため，例えば日本語入力において通常は「かな入力」や「ローマ字入力」にによってひとまず仮名で漢字の読みを入力させそれを変換して所望の漢字を出力するものが多い．漢字を使用する中国や台湾においても基本的にはこの読みを用いたものが主流である．

しかし日本語では同音異字が中国，朝鮮に比べても特に多いため一字ずつを入力する場合に煩わしさを感じざるを得ない場面がある．また，日本語の読みが存在する漢字は，中国語の読みが存在する漢字より少ないために，漢字の固有名詞などを入力する際に読みではそもそも入力が不可能なことがある．

そのため本研究では，読みだけでは入力が困難，煩雑であったり，不可能であったりする漢字のために，読みだけに基かない漢字入力方式として，漢字の構造を使用できる入力方式を提案する．ここで漢字の構造というのは，一つの漢字が他の文字等から構成されている情報のことで，例えば「献」が南と犬からなっているというようなものである．ただし，いくつかの部品となる漢字をキーボードに割り当てて入力するというような，読みによる方法を排除したものではなく，あくまで読みによる入力の上で利用可能なものである．

また，このような入力方式に利用する構造のデータとして，既存の大規模に整備されたものも存在するが，一つの文字コードで閉じていない，手動作成であるといった問題を抱

えている．そこで本研究ではこの漢字構造データを漢字情報の具体的情報である漢字画像から自動的に作成するための，漢字構造自動判別についても検討を行う．自動的に判別が可能となれば，文字コードへの追加などでの扱うべき漢字の追加にも迅速に対応できる．

1.2　本論文の構成

本論文は六つの章と一つの附録から成る．

第 1 章 序論　研究目的と本論文の構成及び用語の説明を行う．

第 2 章 研究背景　漢字についての知識と，コンピュータで漢字を扱うためのこれまでの様々な試み（文字コードや入力方式）について紹介し，それらの問題点を指摘する．

第 3 章 提案システム　本研究において提案する漢字入力方式と，漢字構造の解析のためのシステムについて説明する．

第 4 章 漢字構造解析システム　漢字タイプの分類，漢字分割位置の決定，文字同定の 3 段階から成る漢字構造の解析システムについて，開発したものの評価，考察を行う．

第 5 章 漢字構造を利用する漢字入力方式　漢字入力方式の設計について述べ，いくつかの観点からこれを評価し，それを考察する．

第 6 章 結論　本研究の結論と課題を述べる．

附録 A 外国人名漢字リスト　第 5 章の評価において使用する漢字リストの作成方法とその一覧を記す．

1.3　用語と例示画像について

本論文においては以下の用語を多用するが，ここではそれぞれ以下の意味である．

漢字構造　ある漢字が他のどのような文字や筆画から成っていて，それがどのように配置されているかの情報．本研究においては筆画（はねやはらい，直線など）までは分解せず，ある漢字を分解した各々が文字等になる最小程度の分解に留める．なお漢字の構造の記述で部首は参考にするものの，これによる分類などには従わない．

文字コード　文字コードは符号化文字集合（例：Unicode）と文字符号化方式（例：UTF-8）に分けて理解されることが多いが，本論文においては特に断りのない場合，「文字コード」という語を符号化文字集合，またはそれによる具体的な各文字のコードポイントのこととして使用する．

グリフ　文字コードで定義される抽象的な文字 (Character) ではなく，実際の画像による字形表現のことである．たとえば日本語の始め鉤括弧は JIS X 0213 においては 1 面 1 区 54 点という符号位置が与えられているが，実際の表示では横書きでは < 「 > となり縦書きでは < ﹁ > と異なる画像となる．これは一つの Character に 2 つのグリフ (Glyph) が対応しているということである．

また本文において多数の文字を画像として示しているが，特に断りがない場合は Glyphwiki (https://glyphwiki.org/) のデータより作成したものである．Glyphwiki についての詳細は本文中第2.4.3節及び第4.1節に記す．

第 2 章

研究背景

本章では，まず漢字の理解のための基礎的な知識をまとめ，コンピュータにおいて漢字を取り扱うために行われたきた種々の試みのうち，特に文字コードと漢字入力法，漢字のデータベースについて紹介し，最後にその問題点を指摘する．

2.1　漢字について

漢字の簡単な歴史や地域的広がり，漢字の特性などについて記す．なお本節の記述は藤堂 [17]，金 [12]，齋藤 [13]を参考にした．

2.1.1　漢字の広がりと歴史

漢字は東アジアにおいて広く使われてきた文字である．漢字の使用されるこれら地域を文化圏として「漢字文化圏」あるいは漢文に注目して「漢文文化圏」とも呼ばれる．

紀元前 13 世紀に当時の中国周辺の言葉を表すために誕生した漢字は 3,000 年以上にわたってその命脈を保ち続けており，発祥の地は当然のこと，それ以外の地域においても文化に深く根差している．

現在でもそうであるが，各地様々に音声言語が存在する中国大陸では，いずれも甲骨文字や金文などを祖とする文字を使用していたものの，その書記体系は地域言語に結び付き多様であった．しかし紀元前 3 世紀に成立した秦によって統一され，現在まで続く漢字という文字体系が整備された．

その後 20 世紀以上，この基盤の上で中国文化が成立，発展していく．その直系といえる現代の中華圏の大陸中国，台湾，香港，その他世界各地の華僑，華人コミュニティは漢字を使用し続けている．

同時にその歴史の中で，大陸文化はその周辺に伝播していき，漢字もまたそれらへ伝えられることとなった．

日本は古代において大陸からの文化吸収と同時に漢字を受容し，いわゆる渡来人や遣隋使，遣唐使などによる積極的交流の結果漢字文化の根をおろしていった．漢字はその意味

のためにのみ使用されるのではなく，当時の日本語の音を表すためにも使用され万葉仮名が発展し，やがて今に続く平仮名や片仮名が生み出された．近世以降，日本においては漢字と仮名を両用する漢字仮名交じり文が日本語表記の標準的方法となった．ただし漢字制限や漢字廃止の声は絶えずあり，前者は 1946（昭和 21）年の当用漢字表制定に結びつくものの，漢字廃止は大衆の支持を得るには至っておらず，日本は中華圏に次いで漢字を多用する地域となっている．

朝鮮も日本と同様に中国大陸の文化と共に漢字を受容し，15 世紀に朝鮮語独自の表音文字である訓民正音（現在のハングル．諺文（オンムン）とも）を制定するものの，近代まで漢字が圧倒的な権威を持ち続けていた．19 世紀後半よりハングル・漢字混用文が公的にも普及した．戦後になると民族主義の高まりを受け，南北ともに漢字よりハングルを優先するようになり，現在では日常生活においては漢字を見かける機会はほとんど無くなっている．とはいえ語彙には多くの漢字語が残り，地名や人名などの固有名詞も漢字と結びついており，漢字を意識する機会は少ないながらも存在する．

ベトナム（越南）も周辺国として中国文化の多大な影響を受けた地域であり，漢字漢文の移入の後，ベトナム語を表記するためのチュノム（𡨸喃）と呼ばれる漢字を応用した文字[*1]もつくられ長い間漢字を使用してきた．しかしフランスの植民地であったベトナムは 1945 年以降，漢字及びチュノムを廃止し，現在は「チュー・クオックグー」（Chữ Quốc Ngữ，𡨸國語）というラテン文字を使用した書記法を採用している．ただし朝鮮語と同様にベトナム語の語彙のおよそ 60 % は漢字に由来するものである．

現代においてコンピュータにおける漢字を考慮する際には，これらの漢字文化圏とされる国それぞれの状況を理解する必要があり，これらの地域，さらにはそこで使用される文字である漢字を CJK = Chinese, Japanese, Korean 或いはベトナム (Vietnam) も加えて CJKV と称することが多い．

2.1.2　漢字の特性とその文字数

漢字は文字の分類上「表語文字」(logogram) とされる．仮名やアルファベットなどの音素を表す表音文字と違い，多くの場合意味のあり，また音を伴った一単語を一文字とするからである．ごく単純化すると「字」という漢字一文字は英語の character という数文字一語に対応するということである．なお他に古代エジプトのヒエログリフなどが表語文字であるが，現在広く使用されている表語文字は漢字がその唯一のものといってよい．

漢字の特徴の一つはその数が有限ではないというところにある．漢字の造字法には象形，指事，会意，形声の四つがある．前二者は具体的または抽象的な絵をもとにつくられた文字のことであり，日や月（象形），一や上（指事）などがそれである．これらの漢字を意味のうえで組み合わせたのが会意文字の信（人 + 言）などであり，意味を表す部首と音符を組み合わせたのが形声文字である．形声文字の「汗」（あせ，かん）は水を意味する氵と

[*1] 「𡨸」がチュノムの一種である．

「干」という漢字を組み合わせたものであるが，ここで干に意味はなく，単に音符としてのみしか機能しない．ただし会意と形声の別は排他的なものではなく，どちらの特徴も兼ねる会意兼形声という漢字が多数存在する[*2]．

とまれ組み合わせによって新たな漢字を創作できるという特性のため，古代から現代に至るまで漢字は様々に造字されてきた．前小節で紹介したチュノムを漢字の一種とみればこれはベトナムで創作された漢字であるし，日本においても国字と言われる日本で生まれた漢字[*3]が存在する．韓国にも同様に存在する．現代でも創作漢字コンテストが盛んに開催され，日々新たな漢字が生み出され続けている．「第 8 回創作漢字コンテスト」（2017 年，産経新聞社等主催）の最優秀賞は「フィギュアスケート」の意の漢字（第 2.1図，http://www.sankeisquare.com/event/kanjicontest_8th/index.htmlより）に与えられたように新たな事物の登場により漢字はいくらでも創作し得る．反対にかつて使用されたものの歴史の中に埋もれてしまった漢字も存在したであろう．

第2.1図　創作漢字「フィギュアスケート」

加えて漢字には音や意味は同一であるものの形が異なる異体字という関係にある漢字も多数ある．峯と峰 などという分かりやすいものもあれば，隆と隆 のように一見どこが異なるか分からないものまで存在する．なお一般に異体字としてよく例を挙げられる辺，邊などは後述の IPA 文字情報基盤においては 31 種のグリフが異体字として登録してある．この文字や異体字の多さは，近代前期まで漢字はふつう筆写と結び付いていたことに由来する．活版印刷は存在したものの現在ほどの普及率ではなく，漢字を含む文字は人の手により書写されるものであった．そのため意図的かどうかを問わずバリエーションが発生しやすく，それらが後代の主観により異なる異体字であると認識されて活字とするべき文字が増えていったのである．

以上の理由から歴史上存在した全ての漢字の総数は決して定まらず，また冒頭でも記したように漢字は表語文字であるから，英単語が時代に応じて増えるのと同様に漢字もまたいくらでも増え得るのである．

[*2] たとえば川を意味する「江」は水の意味を表す氵と音符工から成っているので形声文字として解釈できるが，他方で「上下の面に穴をあけて突き通す」という意味の指事文字「工」の突き通す，貫くという意味も含んでいるため，会意兼形声とされる．

[*3] 「峠」や「畑」が有名な例．

とはいえ一般的には様々な漢字字典や漢字表が作成され，社会で流通する漢字には限度がある．

日本で流通する小型の漢字字典「漢字源」には約 1.7 万字，世界最大の漢和辞典といわれる「大漢和辞典」（通称諸橋大漢和）には約 5 万字の親字が収録されている．

また各国で教育や産業での基準などのために策定された漢字表もある．

共産中国においては 2013 年にそれまでの現代漢語通用字表を廃止して，「通用規範漢字表」を制定した．これには 8,105 字が含まれており，一般の使用頻度などにより一級字 3,500 字，二級字 3,000 字，三級字 1,605 に分けられている．特に一級字は 9 年間の義務教育で学ぶ漢字と位置付けられている（辻田 [16]）．

台湾（中華民国）においては 1982 年及び翌年に教育部によって常用国字標準字体表（甲表），次常用国字標準字体表（乙表），罕用[*4]字体表（乙表）が定められ，それぞれ 4,808 字，6,341 字，18,480 字が示されている (Lunde [4, pp.81–82])．

日本では「法令，公用文書，新聞，雑誌，放送など，一般の社会生活において，現代の国語を書き表す場合の漢字使用の目安を示すもの」として「常用漢字表」（平成 22 年内閣告示）に 2,136 字が定められている．また子の名付けに仮名，常用漢字以外に使用できる漢字として「人名用漢字」（戸籍法施行規則別表第二）に 863 字を定めている．常用漢字のうち小学校で指導すべき漢字として学習指導要領別表として「学年別漢字配当表」が定められ 6 学年計 1,006 字が示され，これらは特に「教育漢字」と言われる．

韓国においては「漢文教育用基礎漢字」として 1,800 字が定められており，中等教育において習得することが期待されている．これとは別に日本の最高裁に当たる大法院が名付けに使用できる漢字として 2,964 字を定めている (Lunde [4, p.42])．

なお機能的非識字（社会生活を送る上で必要不可欠とされる読み書き能力を欠くこと）に陥らないために必要な漢字の数は Taylor et al. [7] によるとおおむね 3,500 字程度とされる．

2.1.3　漢字の三要素

ところで漢字には形・音・義の三要素がある．たとえば康熙字典 224 ページ中 26 番の漢字は「坂」という形であって，bǎn や판，さか，ばん というふうに発音し，「さか」という意味を持つといった具合である．

文字はふつう音声言語と結び付いているため，漢字をコンピュータで入力したり，他人に口で伝えたりする際には読み（音）が利用されることが多い．ところが一方で読みは他の 2 要素に比べて変化しやすいものであるため，似てはいるものの，受容地域，言語ごとに読みが異なる．形や意味においても多少の地域差はあるものの，読みほどではない．

単に変化するだけでなく，もともと声調言語である中国語を書き表すためのものだった漢字を，非声調言語である朝鮮語や日本語の中に受容していくと声調が失われ，同じ発音

[*4] 罕はまれ，すくないの意

にいわば縮退する．結果として日本語は中国語や朝鮮語に比しても同音異字が多いことになる（水野 [19]）．

また形と意味は言語を越えて容易に流入するのに対し，外国から来た新しい漢字には読みが当然にはすぐに定まらず，ローカライズのための時間が必要となる[*5]．そのためある地域においては読みが定まらない漢字というのが存在する．

後述の Unihan Database の読みデータを簡単に集計すると，Unicode 中の漢字で中国語の読みが記されているのが約 3.4 万字．一方日本語の読みが存在するのは約 1.3 万字である．

2.1.4 漢字の分類

漢字を部首によって分類することについて若干記述しておく．漢字字典においては 1615 年の「字彙」以来 214 の部首に分類し，各漢字をひとつの部首に割り振るのが伝統である．現代においても基本的にはこれを踏襲するものが多いが，筆記では通常 3 画で記す「阝」（こざとへん）は本来の形である 8 画の「阜」で引かなければならなかったり，「友」の部首は「又」であったり，「聞」の部首は「門」ではなく「耳」であったりと，その漢字の形を知っていたとしても直ちに理解が及ばないものも少なくないというのは注意を要する．そのため一部の辞典では部首の統合を行ったり，新たに部首をたてたりしている[*6]．

2.2 漢字を扱う大規模文字集合

コンピュータで漢字を使用するようになるにつれて漢字を使用する環境も整備されてきた．ここでは日本の事情が中心になってしまうが，文字コードについて多少その歴史にも触れながら，コンピュータにおいて多数の漢字を使用するためになされたいくつかの試みについて記す．

2.2.1 大規模集合以前

現代のパーソナルコンピュータで使用される多くの文字コードに繋がる始祖は ASCII (American Standard Code for Information Interchange) である．7 bit = 128 の空間を持つが，この中で図形文字として使用できる領域が 94 点であるため，ASCII との互換を図った後続規格の多くはこの 94 という大きさに縛られることとなった．

8 bit 目も使用するようになったのが多言語時代の始まりであった．ASCII 互換の 8 ビット規格の中でフランスやドイツは新たに使用できるようになった 94 個の GR 空間[*7]に

[*5] 中国から日本への流入が圧倒的なものの，国字「腺」に xiàn という読みが与えられ中国に輸入されたという事例もある．

[*6] たとえば岩波国語辞典第七版では，漢字のための索引において「匸」（はこがまえ）と「匚」（かくしがまえ）を統合したり，欺や碁の部首を「其」にしたりしている．

[*7] 8 ビット目がそれぞれ 0 と 1 の 2 つの 7 ビット空間を左右に並べたものとして理解することが多く，このうち右 (right) 半面（8 ビット目が 1 の空間）の図形文字 (graphic character) 用の空間のため GR と

ウムラウト (ü) やアクセント符号 (à) の付いた文字，エスツェット (ß) などを定義した（これらは ISO/IEC 8859 という国際規格になっている）．

日本では JIS X 0201（1969 年，旧 JIS C 6220）が策定され 94 の空間の中に片仮名が定義され[*8]，最低限片仮名と英数字は使える状況となった．

しかし東アジア諸国ではこれだけの空間では満足しきれなかった．そこで登場したのが 2 バイト文字であり，まず日本で策定されたのが JIS X 0208（1978 年，旧 JIS C 6226）である．いわゆる JIS 漢字と呼ばれるものである．ASCII との互換を保つ大きさ 94 の空間を使いながら 2 バイトのみ使用する場合，$94 \times 94 = 8,836$ 字が扱える文字数の上限となり[*9]，JIS X 0208 には英数字[*10]，平仮名，片仮名，漢字，その他記号などの合計 6,879 字（最新版）が含まれている．これらの文字には 94 区 94 点からなる空間における位置である区点コードが与えられた．漢字は第 1 水準と第 2 水準の 2 つに分けられ，常用漢字を含み日常で多く使用するものが第 1 水準，それ以外を第 2 水準とした．この JIS X 0208 を具体的にエンコーディングする方式としては Shift_JIS や EUC-JP，ISO-2022-JP が挙げられる．

ところがいざ使い始めると，特に固有名詞などが不足しているということで様々な追加要望があった．そのため次いで 1990 年に JIS X 0212「補助漢字集合」が策定された．ところが日本語 PC 環境で広く使用されていた Shift_JIS とは適合的でなかったためさほど普及しなかった（EUC-JP では使用できる）．

その後 JIS X 0208 を拡張して新たに漢字を第 3 水準，第 4 水準として追加したのが JIS X 0213（2000 年）である．合計 11,233 字を定義する．JIS X 0208 で定義された区点空間を第 1 面として新たに第 2 面を追加し，全ての文字には面区点コードが与えられる．こちらも Shift_JIS などの伝統的エンコーディング方式では使用できないが，Unicode が普及してきていた時代でもあり，また綿密な調査のもと採集された漢字集合であるため，直接 JIS X 0213 用のエンコーディング方式を使用することは少ないものの，追って Unicode に追加されておりしばしば言及される．

中華圏初の文字コードは大陸中国で 1981 年に策定された GB2312-80 （信息交換用漢字編碼字符号集[*11]）であり 7,445 字を含んでいた．中国ではその後様々な拡張集合などが策定されたが，今日の中国で最も重要なのは GB 18030-2005 （信息技術中文編碼字符集）である．中国国内で販売される全てのコンピュータソフトウェアはこれへの対応が義務付けられている．GB2312 や後続の様々な規格の上位互換であり，原理的には 100 万字以上の文字が追加可能であるが，独自での文字集合は行っておらず近年は専ら Unicode を取

いう．同様に ASCII の図形文字空間は GL に当たる．

[*8] 現在の使用場面ではいわゆる「半角片仮名」とよばれるものに当たる．

[*9] 8 ビットコードとして GL と GR 両方を使用し $(2 \times 94)^2 = 35,344$ まで扱うことも不可能ではないが，他の文字集合との共存が不可能となる．

[*10] Shift_JIS などで ASCII と併用する場合にいわゆる全角英数字とよばれるものであるが，規格として全角幅で表示しなければならないという規定はない．JIS X 0208 のみを使用するエンコーディング方式の場合は JIS X 0208 の英数字のみが使用できる唯一の英数字となる．

[*11] 碼はコードの意

り込んでいる．

なお台湾では 1984 年に約 1.3 万字を含んだいわゆる Big5 （電脳用中文字型与字碼対照表）と呼ばれる文字集合が発表された．公的規格ではなく有力民間企業が策定したものであるが台湾や香港などの事実上の標準エンコーディングである．政府規格としては追って 1986 年に CNS 11643 （中文標準交換碼）が発行され，現在では約 7 万字を収録している．

また韓国は KS X 1001 で 4,888 字の漢字を収録し，北朝鮮は KPS 9566-97 で 4,653 の漢字を収録している．

2.2.2 大規模漢字集合

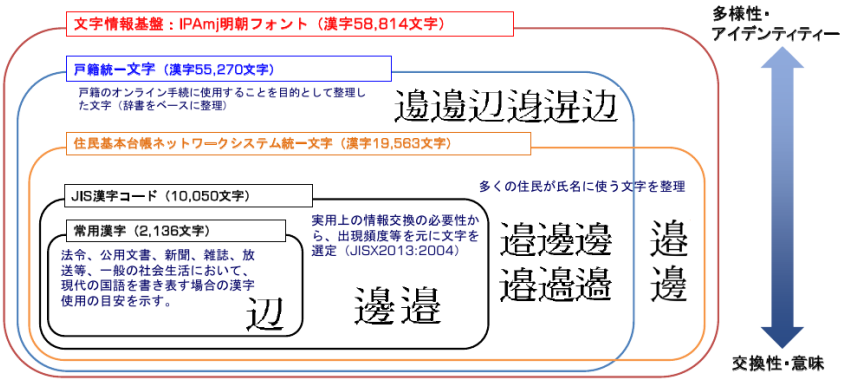
1990 年代の日本では公的規格としては JIS X 0208 の約 9 千字弱しか漢字が使用できなかったため，いくつかの大規模漢字環境が研究，開発されていた時代である．

東大の「マルティメディア通信システムにおける多国語処理の研究プロジェクト」がその最たるものである．様々な資料から詳細な漢字収集が行われ，最終的に約 6.6 万字を選定し，同時にそれを表示する GT 書体を作成し発表した（田村 [15]）．ただし TRON ではネイティブに使用できたものの，一般的な PC 用 OS におけるエンコーディングは規定しなかったため，Windows などで利用する際は Shift_JIS 等で入力したものをフォントを切り替えて出力し分けるという方法を採っていた．印刷前提であればこれで良いものの，世紀末から進展しつつあったインターネット時代にはディジタル化文書がそのまま流通していくため，GT プロジェクトはこの流れに取り残された感が否めず，広く使用されるには至っていない．

株式会社エーアイ・ネットから 1997 年に発売された「今昔文字鏡」は文字コードではなく印字ソフトウェアであるが[*12]，最新版では漢字を約 16 万字収録しており，細かな異体字の違いなどを印刷することができる．

現在日本において注目を受けているのが情報処理推進機構 (IPA) による「文字情報基盤整備事業」である．日本の行政事務において戸籍や住民基本台帳などを取り扱うための文字コードとしては「戸籍統一文字」（法務省）や「住民基本台帳ネットワークシステム統一文字」（総務省）などが存在する．日本人の名付けに使用できる漢字には 1948（昭和 23）年以降制限があるが，それ以前より存在する人名や氏，外国人の名はその対象外であるため，これらにおいて使用され符号化されるべき文字や既存の文字の異体字が JIS 漢字コードや Unicode の範囲外にまで広がっている．文字情報基盤整備事業においてはそれらを包含した文字情報基盤の整備を行っており，その成果物として MJ コードの付与と IPAmj 明朝の作成を行っている．また Unicode に含まれていない漢字については Unicode/UCS に追加提案を行い，異体字については後述の異体字セレクタデータベースへの登録提案を行う（平本 [18]）．

[*12] http://www.mojikyo.co.jp/software/mojikyo45/



第2.2図　文字情報基盤の概念図（平本 [18]より）

2.2.3　Unicode

現在最も普及している多言語文字集合が Unicode である．IBM や Xerox，Apple 等が開発を始め，Unicode Consortium が策定する業界標準規格であるものの，現在では国際規格である ISO/IEC 10646 “Universal Multiple-Octet Coded Character Set (UCS)” と双方協調することとなっているためその内容はほぼ同一であり，実質的に de jure 標準でもある．

Unicode は（2 バイトに限らない）マルチバイトを用いて文字を表現するという前提にたった符号化方式である．Unicode で定義される文字にはコードポイント[*13]が付されるが現在定義されている最大値は `U+10FFFF` であり理論上の最大収録文字数は約 111 万字ということになる．

2017 年に発行された Unicode 10.0 ではこのうち 136,690 の空間に文字を定義している．実に様々な書記系を収録しており，それらを列挙すると限りがないが，以下の書記系はしばしば注目される．

- アラビア文字（書字方向の制御メカニズムも規定あり）
- ヒエログリフ（必ずしも現代で使用される文字のみではない）
- 絵文字（Emoji，特に最近は絵文字の追加が活発である）

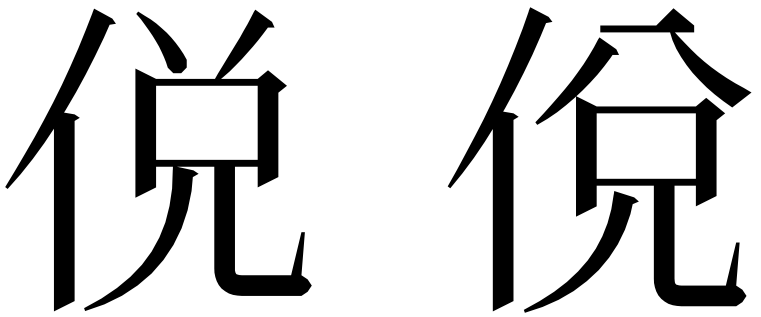
CJK 統合漢字

Unicode 内で最も大きな領域を占めているのは CJK 統合漢字 (CJK Unified Ideographs) である．

統合 (Unify) とは，上述したような各国が独自に定めたそれぞれの漢字集合のうち，字種として同一でありその差がデザイン差に留まるとされた漢字に同一のコードポイントを

[*13] ふつう `U+` に続けて 16 進数で表される．

割り当てた（包摂）ことの謂であり，包摂は CJK 統合漢字における原則の一つである．たとえば CJK 統合漢字基本ブロックの U+4FBB の漢字は第2.3図に示すように各国の標準規格における例示字形には差がある．しかし Unicode はこれらを区別することなく単一のコードポイントを付与している．つまり Unicode にとっては「八」の向きはフォントによって異なって良いということにしたのである[*14]．



大陸中国の規格における例示字形　　日本の規格における例示字形

第2.3図　倪 (U+4FBB) の例示字形の地域差

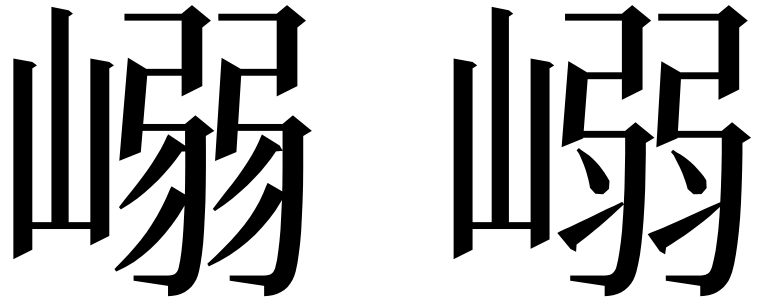
包摂しながらも各国の漢字規格を取り込み，その他にも追加提案を受けたものを少しずつ取り込んでいった結果，現在は基本ブロックと A から F までの拡張ブロックを全て合わせて 87,849 字に上る．Unicode 全体のおよそ 3 分の 2 を占める．

異体字セレクタ

また Unicode には異体字グリフを明示的に指定するための異体字セレクタ (Ideographic Variation Selector) という仕組みも存在する．異体字を区別したい場合にその字の直後に字形選択子 (Variation Selectors) を附加することでこれらを区別する．VS は U+FE00 – U+FE0F と U+E0100 – U+E01EF の計 256 字規定されている．嶋 (U+5D76) の例を第2.4図に示す．なおこれはもちろん IVS に対応したアプリケーションとフォントでなければ正しくは表示されないが，仮に IVS が無視されても標準字形は表示されることが担保される．

IVS を使用したグリフのデータベースである IVD には，Adobe による CID に準拠したものや文字情報基盤のものなど合計 15,264 字について 39,169 種が登録されている．すなわち Unicode をフル活用すれば基本漢字と IVS により 111,754 のグリフ（ただし重複は考慮外）を出力し分けることができるということになる．

[*14] ただし実際のところ，Unicode には CJK 互換漢字ブロックの U+2F806 に第2.3図の左の字形が規定してある．これは CJK 統合漢字の原則のまた一つである「原規格分離」，つまり基礎となる各国の規格において別のコードが与えられている場合は Unicode においても互換のためにコードポイントを与えるという規則のためである．



U+5D76 - U+E0101　　U+5D76 - U+E0102

第2.4図　嵺 (U+5D76) の IVS による異体字指定

その他の漢字関連文字

部首（CJK Radicals/Kangxi Radicals 分類は康熙字典[*15]による）が 224，補助用部首（CJK Radicals Supplement 部首の変化字）が 128. さらに分解した漢字の最小単位の筆画 (CJK Strokes) が 48 字，漢字の配置を示すための Ideographic Description Characters (IDC) が 12 字ある．

IDC の一覧

⿰, ⿱, ⿲, ⿳, ⿴, ⿵, ⿶, ⿷, ⿸, ⿹, ⿺, ⿻

IDC は書字方向を制御したりする制御文字や，仮名について濁点や半濁点つきの 1 つの仮名グリフを合成する合成用文字ではない可視文字であるが，漢字の記述に使用することができる．たとえば㹴 (U+24BC2) は⿰甘犬 と表すようにである．これを Ideographic Description Sequence という．ただし Unicode では文字としての IDC とごく簡単な例しか与えられておらず，表現方法の詳細な点は使用者の任意となっている．たとえば霊 (U+4259) は⿱雨⿱口云 とも⿱雨云 とも表現可能である．

漢字の追加について

2018 年 1 月現在は 2017 年の Unicode 10.0 で拡張 F を追加したばかりというのもあって，Unicode に対して漢字の追加提案はされていないが，前述したように漢字はいくらでも増える可能性があり，また Unicode はそれらをまだ十分受け入れる余地があるため，今後もおそらく増加していくと思われる．

[*15] 清の康熙帝の命により編纂された字典．1716 年完成．特にその配列が後世の字典の基準となり，デザインは明朝体の基準ともなった．

2.3 漢字の入力方法

この節では多少歴史的なものも含めて日本と中国，台湾などにおける漢字入力法を読みによるものとそれ以外に分けて紹介する．

2.3.1 読みを利用する入力法

日本においても中華圏においても現在は読みを利用する方法が優勢である．例えば台湾の調査[*16]によるとおよそ 7 割が読みを使用する注音輸入法を使用していた．

日本においては Microsoft が開発し Windows や Office に附属する Microsoft IME やジャストシステムの ATOK，最近は Google による Google 日本語入力などがよく使われている．他にも以前の Mac OS に搭載されていたことえりなどがある．基本的にはいずれもローマ字入力やかな入力によって仮名を入力してある程度の区切りにおいて変換モード（ふつうスペースキーか変換キーによって）に入ると，未変換文字列を形態素解析し，適当な部分ごとに漢字の候補を表示しユーザに選択させるというものである．なお他にも SKK (Simple Kana to Kanji conversion) という入力メソッドがあるが，これは形態素解析をせず，漢字変換の範囲をユーザが指定するというアプリケーションである．

中国語における読みに相当するのは大陸の拼音（ピンイン）と台湾の注音符号（ちゅういん）である．拼音はアルファベットと各種ダイアクリティカルマークを使用するもので，注音は独自の 37 個の符号を使用して表すもので歴史的にはこちらの方が古いが現在ではほぼ台湾でしか使用されない．

中国語の特に普通話，国語の漢字 1 文字に相当する発音は声母と韻母の組み合わせに分解でき，注音はこの構成要素に図形を割り当てたもので，声調を別にすれば漢字 1 文字を注音 2 文字もしくは 3 文字の組み合わせで表現できる．一方拼音は音に近いアルファベットを使用するため注音では 1 文字に相当するものをアルファベット 2 文字以上で表現することもある．たとえば「庄」という漢字の読みについては，声母がㄓ:zh，韻母がㄨ:u とㄤ:ang からなり，声調は第一声であるから，注音ではなにも附加せずㄓㄨㄤとなり，拼音ではマクロン (¯) を用いて zhuāng となる．

さて拼音はアルファベットを使用するために日本語のローマ字入力と同様にアルファベット配列キーボードをそのまま使用できる．拼音を表記通りそのまま入力するのが全拼 (Full Pinyin) と呼ばれる方法である．ただしダイアクリティカルマークで区別する声調の入力はできないので，同じ発音の漢字候補からひとつを選択する．また上述のように一つの発音要素を複数のアルファベットで表すことがあるため，打鍵数は大きくなる．この煩わしさを軽減するために簡拼 (Half Pinyin) では一つのキーに複数の発音要素を与え（結果として注音符号に対してアルファベット 1 文字が対応），双拼 (Double Pinyin) では

[*16] Pollster 波仕特線上市調網
http://www.pollster.com.tw/Aboutlook/lookview_item.aspx?ms_sn=1476

さらに進めて一部の韻母の組み合わせに一つのキーを与えている．当然一部は重複するものの，できるだけ声母要素と韻母要素を同時に割り当てるようにしており，結果として全拼よりは打鍵数は減少する．

注音を用いた入力は日本語における仮名入力と同様であり，注音用のキーボード配列があり，これに則って打鍵するのである．

2.3.2　読み以外を利用する入力法

読みを使用しない入力方法は更に，漢字の形，画像的特徴を利用するものとそうでないものに分類される．

形に基礎を置いた入力法

中国や台湾で使用されるものが主であるが，広く使用されるものでは以下が挙げられる．

■倉頡輸入法 (Cangjie input method)　独自に規定した部首をいくつか連続して入力してひとつの文字を形作るものである．ある漢字はいくつかの「字根」に分解してあり，ユーザはそれを入力していく．各々の字根や若干の特殊文字のためのキーが QWERTY 配列の上に割り当てられている．若干の例を挙げる．

- 車 = 十(J) 田(W) 十(J)
- 謝 = 卜(Y) 口(R) 竹(H) 竹(H) 戈(I)
- 谢 = 戈(I) 女(V) 竹(H) 竹(H) 戈(I)

もともと台湾で開発されたものであり，現在でも台湾でおよそ 1 割の人が使用している．流通最新版では 6 万字が入力可能である．

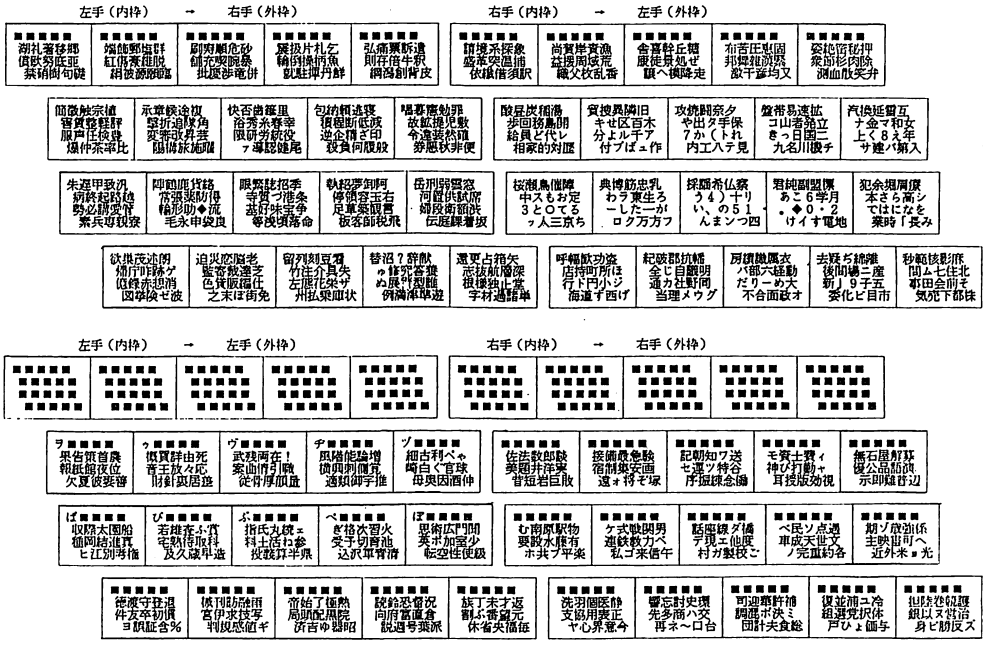
■五筆輸入法 (Wubi method)　大陸中国で開発されたものである．いくつかの字根を繋げて使用するという点では上記と同様だが，それらを最初の 5 つの基本筆画（一，丨，丿，丶，フ）に基いてグループ分けしてキーボード上に配置している点に特色がある．主に大陸で使用される．特に普通話に習熟していないために発音が分からず拼音が使用できない者に使用されている．

形を利用しない入力法

文字コード表の直接参照による入力法や画像認識による入力法もここに分類されるが，本節ではキーボードに何らかの割り当てを行って入力法とするものを以下に二つ挙げる．

■T コード　東大理学部の山田尚勇らのキー配列設計からソフトウェアまでのヒューマンインタフェースの一連の研究の中で 1980 年代前半に開発された入力法である．全ての文字は第2.5図に示した表に従って 2 打鍵で入力するが，この漢字などの配置に読みや形など漢字そのものの性質は全く関係なく，入力頻度などによって割り振られたものであり「無連想式」と呼ばれる．当然この表の暗記が必須であるが，訓練を積めば大変高速に入力

でき，また仮名漢字変換に比べて大脳への負担も小さいことが示され，日本語入力の途中で変換確定などのために思考を中断されることがないため創作などに向いていると主張されている．ただし T コードで直接入力できるのはここに示された約 1,260 字に限られる（山田 [22]）．



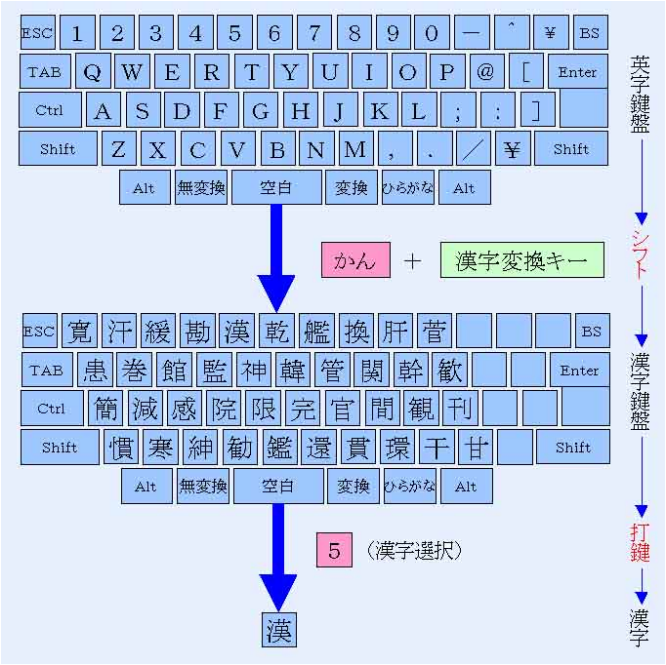
第2.5図　T コードの文字割振り（山田 [22, p.281]より）

■風（かぜ）（超多段シフト）　1988 年に発売された入力方法[*17]．「超多段シフト方式」という概念により漢字を入力する．かつての電算写植に用いられた「多段シフト」に由来するもので，風ではシフト操作をソフトウェアで漢字の読みによって実現する．たとえば「漢」という字を入力する時にはローマ字で「KAN」と入力し漢字変換キーを打ち，キーボードを「かん」読みをもつ漢字鍵盤に仮想的に切り替え，そこであるキーに割り当てられた「漢」を打鍵する（第2.6図）．また「間」という字を入力する場合に「かん」でシフトさせても「ま」でシフトさせても「間」の位置は変わらないという特徴をもつ．入力できる範囲は JIS X 0208（JIS 第一水準，第二水準）の 6,349 字．

2.3.3　既存の入力方式の問題点

以上いくつかの入力方式を概観したが，倉頡や五筆輸入法は現在でも需要はあるものの，やはり最も使用されるのは読みを利用する入力法である．読みを入力するものは，アルファベットやキーの数を越えない程度のキー一覧を覚えれば（或いは覚えずとも），音声言語に直結した読みというコンピュータ以前の知識で使用できるため習得のための訓練が短

[*17] 風のくに（開発者の web ページ）http://homepage3.nifty.com/togasi/



第2.6図　風における「漢」の入力例（開発者 web ページより）

い時間で済む．T コードなどは，慣れれば高速度を実現できるものの，習得までに時間がかかり，大衆化したコンピュータ社会で広く受け入れられるには至らなかったといえるだろう．

また読みを使用しない方式に共通するのはいずれも漢字の割当ては開発側で手動で行われることである．倉頡輸入法のようにかなり大規模な漢字集合にまで積極的に対応してあるものもある一方，JIS X 0208 時代に開発され，その後活発な開発は行われなかったものは小規模な範囲に留まっており，仮に現在でも利用者がいたとしても使用できる漢字に制限が加えられ，Unicode などの大規模文字集合の恩恵に浴せない．

2.4　漢字データベース

この章の最後に本研究で有用となりうる漢字のデータベースについて記述する．ここでいう漢字データベースとは，単なる漢字の一覧ではなく，漢字の読みや異体字などの漢字相互の関係，使用場面，構造など漢字の画像とコード以外にその漢字に人間が与える何らかの意味（漢字の義に留まらないより一般的なもの）についてのデータ集合を指す．

第2.1.2節で紹介した各国の標準漢字表も含まれるが，ここではディジタルデータとして利用可能なものを挙げる．

2.4.1 Unihan Database

CJK 統合漢字の制定主体である Unicode Consortium 自身によって整備されているのが Unihan Database である．https://unicode.org/charts/unihan.html から以下のデータが利用可能となっている．

- CJK 統合漢字の基礎となった各国国内規格における符号位置
- その他の文字集合における符号位置
- 漢字の意味（義）
- 康熙字典や諸橋大漢和などの著名な漢字辞典での位置
- 中国語（普通話，広東語），日本語，朝鮮語での読み（日本語は音訓とも）
- 部首（基本的に康熙字典による）
- 異体字（簡体字と繁体字の対応も含む）

テキストデータとして取得でき簡便に利用可能となっている．当然のことながら漢字の意味や発音などを細大漏らさず記しているわけではないものの，参考として利用する限りにおいては大変有用なものといえる．

2.4.2 CHISE

CHISE (CHinese Information Service Environment) プロジェクトは，単純に各文字にコードポイントを割り振る符号化ではなく，豊富な属性情報によって構成される情報によって文字を表すこと，「ポスト文字コード」を指向した研究プロジェクトである（守岡 [20, 6]）．

この CHISE の成果物の一つとして，Unicode の IDC を用いた漢字構造のデータがある．第2.2.3節で紹介した例もこれを引用したものである．

大変有用であるものの，手動で作成されているため，将来的に漢字の追加に追従できるか不安であり，また所々に間違いも見られる．加えて，元から決して Unicode をことさら重視するプロジェクトではないので，CJK 統合漢字の構造記述に Unicode 以外のコードが参照されることが多々ある．特に台湾中央研究院で策定された CDP コードが参照されることが多いが，これは多くが Unicode では符号化されておらず，またそもそも CDP 自体に新旧版が混在している．一般のコンピュータでは CDP はフォントもそれを表示するシステムも無いためこの部分については大変利用しづらくなっている．

2.4.3 パラメトリックフォント

直接構造を記述したものではないものの，間接的に漢字の構造データへの転用可能性を有するのが一部のフォント設計手法である．上述した GT フォントやインターネット上の wiki サイト Glyphwiki で設計される花園明朝（上地 [11]），大規模日本語フォント黎明

期に発表された和田研フォント（田中 [14]）などは漢字をいくつかの部品に分解して，それを合成することでフォント画像の生成を行っていた（Glyphwiki においてはこれは任意）．このためこのデータを利用すれば，ある程度の構造を利用可能と推測されるものの，Glyphwiki 以外は非公開であり，また Glyphwiki も部品に分解することが目的ではないので，これらのみに準拠して漢字構造を得るというのは不可能である．

2.5　本章のまとめ

20 世紀以上の長い歴史を持ち，中国のみならず東アジアの広い地域で使用され続けている漢字は，その一つずつが一単語に相当する「表語文字」であり，また強力な造字能力をもつために漢字の創作に限りはなく，実際に今日でも新たな漢字は次々と生み出されている．

また漢字は一般に形・音・義の三つの要素を持つとされるが，このうち読み（音）については全ての漢字について必ずしも自明なものでなく，読みの定まらない漢字も存在する．特に日本語で読みが明らかな漢字は中国語に比べて大変少ない．また日本語における漢字は，その言語的特徴から中国，朝鮮に比べて同音異字が多い．

現在最も広く使われている多言語文字集合である Unicode では積極的に漢字の追加が行われており，最新の Unicode 10.0 では 8 万字以上を収録しているが，上述の通り読みが全てに存在しているわけではないので，現在のところコードの利用はコード表からの指定などに限られる．容易にアクセス可能な仕組みが存在しなければ，「宝の持ち腐れ」になる可能性がある．

漢字をいかにして入力するかという問題は，ふつう日華双方とも読み，発音符号を利用した入力法が主流である．日本においてはかつて T コードなどの試みもあったものの，習得の負担が大きく，広く普及するには至らなかった．

第 3 章

提案システム

前章の内容を踏まえて，本研究で提案する手法について説明する．

3.1　漢字構造による漢字入力方式

前章において，漢字をコンピュータで入力する際に以下のような問題があることが分かった．

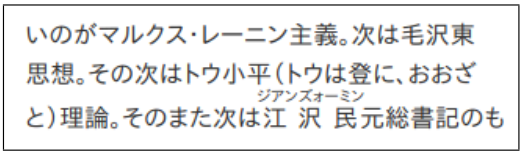
- 読みのない漢字や同音異字の入力には，読みのみの漢字入力では困難であったり不可能であったりすること
- 特に Unicode にはますます大量の漢字が登録され続けているにもかかわらず，その大多数は読みからは到達できないこと

そこで本研究では，漢字構造を利用可能な漢字入力方式を提案する．ここでいう漢字構造とは「ある漢字が他のどのような文字から成っていて，それがどのように配置されているか」という情報である．

一般に漢字の 3 要素といわれる形・音・義のうち，これまで大きく利用されてきた「音」だけでなく，「形」も使うということである．構造 = 形は，特定の漢字に関する読みや意味などの直接の知識がなくとも，その漢字の内部に含まれている他の文字等を知っていれば容易に理解でき，伝達できる．つまり「鄧」という字をどう読むかやどういう意味かを知らずとも，「山に登る，の『登』と部屋の『部』の右側」と表現できるということである．これは日常において漢字を口頭で伝える際によく用いる方法であるし，技術的制約から使用できる漢字が限られている場合（一部の新聞社の Web サイトなど）に使用する方法でもある（第3.1図）．

そのような漢字の構造を利用するために本研究で提案するシステムは，具体的には以下の特徴を有するものとする．

部品は文字　漢字は筆画（一，丨，丿，丶，フ）まで分解することも可能だが，ここでは会意や形声の原理でつくられた漢字を，その構成漢字に分解する程度に留める．一



第3.1図　「鄧」を説明する読売新聞 Web サイトの例
(http://www.yomiuri.co.jp/fukayomi/ichiran/20171031-OYT8T50136.html)

般的な漢字の説明と同様な感覚で使用可能にするためである．なお必ずしも漢字だけに留めないが，Unicode で使用できる画像文字に限定する．

配置は考慮しない　このシステムではあくまでどの漢字から成っているかだけに留め，それがどのように配置されているかという情報は使用しないこととする．つまり峰と峯の構造情報は同一となる．

補助手段　またあくまでこれは既存の文字入力法の補助に留める．現在は読みによる入力が大勢であり，かつて T コードが定着しなかったことから推し量るに，読みと全く異なる原理で入力する入力法は習得までの障壁が大きく，普及が阻害される．よって基盤としては読みによる入力法を使用しつつ，そこから呼び出して使用できる方法とする．

キーボードを使用　入力に際してはコンピュータキーボードでの入力を前提とする．近年はタブレット端末などタッチデバイスの普及も著しいが，本研究では標準的なオフィス環境における利用を考え，提案する漢字入力方式もキーボードで使用するものとし，比較もマウスで可能なものとする．

交換可能なデータ　基礎として使用する構造データは交換可能であること．

漢字の配置に関する情報は持たないために，入力するデータは`8cb8:4ee3,8c9d`（貸 = 代 + 貝の例）というシンプルな形とする．

この方針のもとで漢字入力方式を開発する．詳細は第5章に記す．

3.2　提案入力方式に必要な構造データ

前節の入力法を実現するために必要となる漢字構造データは「貸 = 代 + 貝」というようなデータである．

既存の類似データとして，第2.4.2節で紹介した CHISE の IDS によるデータが挙げられる．CHISE は以下のような Unicode IDC を用いた表現のテキストデータを提供している．

CHISE の IDS （抜粋）

```
U+8CB6 貶 ⿰貝乏
U+8CB7 買 ⿰&GT-36329;貝
U+8CB8 貸 ⿰代貝
```

この IDS には漢字のタイプを示す IDC が含まれているため，これを除去し，適切に処理すれば提案方式で使用できるデータとなる．

しかし，上のデータにもあるように，このデータセットは Unicode だけで閉じているわけではない．「買」の例では，同じ字形の罒 (U+7F52) が Unicode にも含まれているにもかかわらず，GT コード（第2.2.2節）を参照していて，そのままの形では利用できない．

そこで本研究では漢字構造データを漢字の一つの表現である文字画像から自動で解析するシステムも開発する．自動解析の枠組みが実現すれば，Unicode のみをテンプレートに指定することで，Unicode だけで閉じたデータベースにすることも可能となるし，テンプレートを変更すればそれ以外にも柔軟に対応可能となる．

また自動化することによって，新規に追加された漢字や創作漢字へも迅速に対応可能なことが期待される．

解析システムは，漢字画像を次の 3 段階の過程へ入力することによってその構造を解析することとする．

1. CNN による漢字タイプの分類
2. タイプに基づく分割位置の決定
3. 分割した部分文字の同定

前節にも記したように，最終的な出力は8cb8:4ee3,8c9d（貸 = 代 + 貝の例）というような配置情報は持たないデータとする．

その詳細を第4章に記す．

第 4 章

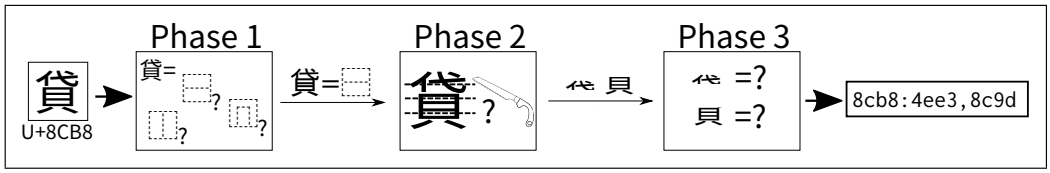
漢字構造解析システム

この章では，提案する漢字入力方式で利用可能な漢字構造のデータをコンピュータの解析により行うことを目的とする．

すなわち，漢字の画像データが入力されると，その漢字が

1. どのようなタイプ（⿰, ⿱, ⿴等）の漢字であるか = どのような方向に分割可能か
2. どの位置で分割すべきか
3. 分割した各部分文字は与えられたテンプレートの中のどれにあたるか

の過程を経て，最終的に`8cb8:4ee3,8c9d`（貸 = 代 + 貝の例）というような文字コード等による漢字の構造表現データが出力されるアプリケーションの開発を行う（第4.1図）．



第4.1図　漢字構造解析システムの概念図

以下の各節に上記の 3 段階のそれぞれに対応する手法の検討を行った結果を記す．

最終的にこれらの成果をまとめたシステムを作成したが，実現した正解率の関係上，全自動ではなく最終決定は人間が行う Computer-Aided なものに留まった．とはいえ部品としたい漢字が既に Unicode に存在するかどうかの知識などは必要なく，直感による切断線と同定の判断のみに労力を割けばよいプログラムとなった．

4.1　前提環境等について

研究開始時は Unicode 10.0 発行前だったため，本章で取り扱う CJK 統合漢字は基本的に拡張 E までとする．互換漢字などは必要に応じて使用する．

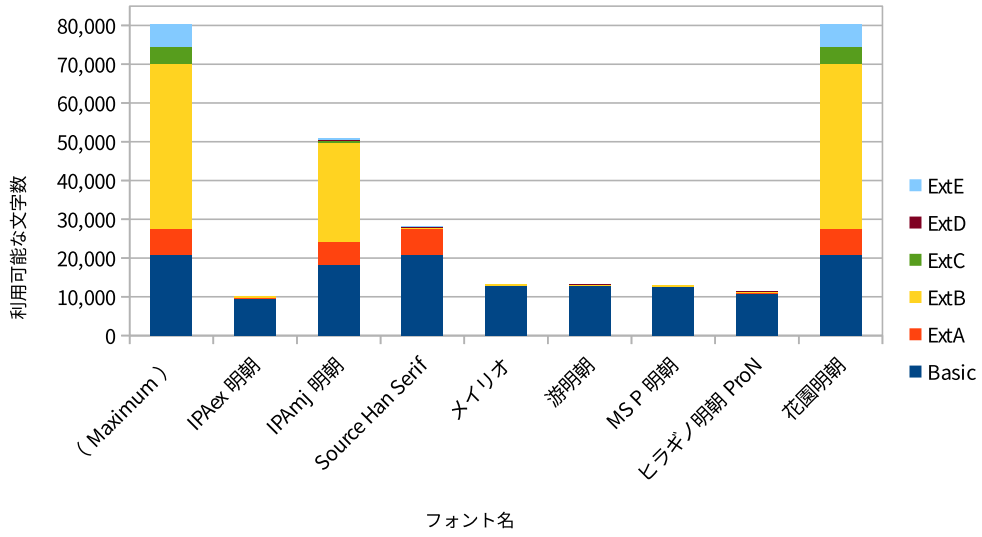
また以後では漢字の画像を使用していくが，本論文では Glyphwiki による漢字画像データ（花園明朝に基本的に等価，2017 年 4 月 17 日 19 時時点での Glyphwiki ダンプデー

タをもとにしたもの）を使用する．これは Glyphwiki によるフォントが唯一，CJK 統合漢字全てのグリフを収録していたからである．

CJK 統合漢字への対応状況をみるため，日本語でよく使用されるフォントのうち以下のものについて各フォントファイルの文字コードとグリフを結び付ける cmap テーブルを解析することによってそのフォントが対応する文字を集計した．

- IPAex 明朝
- IPAmj 明朝
- Source Han Serif （源ノ明朝）
- メイリオ
- 游明朝
- MS P 明朝
- ヒラギノ明朝 Pro N
- 花園明朝 (Glyphwiki)

その結果を第4.2図に示す．



第4.2図　フォント別の CJK 統合漢字への対応状況

一部のフォントを除いて基本的には CJK 統合漢字のうちほぼ基本ブロックしか対応していないものが多く，基本ブロックに対応とはいっても，基本ブロックを完全に網羅しているのは源ノ角ゴシックと花園明朝しか存在しない．

これはもちろんフォントデザインの労力の問題もあるが，大規模漢字集合固有の問題としてはフォントファイルに収録可能なグリフ数の制限も影響していると思われる．現在主流の OpenType フォント形式ではフォントフォーマットにおけるビット空間などの制限により，PostScript 互換方式では 64,000 グリフ，TrueType 互換形式などでは 65,536 グリフが収録できるグリフの最大値であり，いずれにせよおよそ 6.4 万字程度である．この

ため，Unicode 全体はおろか，CJK 統合漢字全体をカバー可能なフォントは存在し得ないのである (Lunde [4, p.367])．なお花園明朝は単一で第4.2図中においてあたかも全体をカバーしているように記しているが，実際は花園明朝 A と花園明朝 B という 2 つのフォントファイルに分割されている．システムプログラムによってはある文字がそのフォントでは表示できない際にフォント名の近い別のフォントで表示する機能があるため，こういった機能やユーザ自身の指定によって CJK 統合漢字全体を利用可能となっている．

なお第 1 段階のプログラムは Python + Tensorflow で，第 2 段階と第 3 段階のプログラムは C++ によって作成した．

Glyphwiki についての補足

Glyphwiki について補足しておく．第2.4.3節でも少し触れたが、Glyphwiki プロジェクト (`https://glyphwiki.org/`) は wiki によって，自由に字形を編集，登録することができるグリフ作成プロジェクトである．編集者は以下のような独自のフォーマットによって筆画ごとにグリフのスケルトン（骨格）を設計し，そのデータは肉付けプログラムも含む KAGE エンジンと呼ばれる独自のプログラムによってアウトライン表現に変換出力される．単に筆画を一つずつ入力するだけでなく，他の作成済みのグリフを参照して埋め込むことも可能となっている（上地 [11]）．

「子」(U+5B50) のソース

```
1:0:2:40:31:149:31
2:22:7:149:31:136:49:102:79
1:0:4:100:72:100:182
1:0:0:14:102:186:102
```

「疒」(U+24D33) を子 (U+5B50) と广 (U+7592) を引用して記述したソース

```
99:0:0:0:0:200:200:u7592
99:0:0:50:38:195:195:u5b50
```

4.2 CNN を用いた漢字タイプ判別

さて本節ではまず漢字を組み方のタイプに分類することを実現する．

ここでタイプというのは Unicode IDC に従った，漢字が左右二つに分割できるとか，上下に分割できるといったの類の情報であり，CHISE IDS による例を第4.1表に挙げる．なお以後このタイプを参照する場合はそれぞれのコードポイントから $(2FF0)_{16}$ を減じた値を使用することもある．この値については第4.1表に共に記した．

この 12 個の IDC のうち，合成を表す⿻については，仮に CNN による判定が成功したとしても後段の分割で他の IDC タイプに比べて大きな困難が予想されるので，省略することとした．

第4.1表　IDC を利用した漢字タイプ分類の例

タイプ	漢字例
⿰ (0)	乱仇惜
⿱ (1)	黎岩苣
⿲ (2)	徽澍斑
⿳ (3)	峦肇意
⿴ (4)	圓圖衡
⿵ (5)	閣鬩凩
⿶ (6)	凶輿函
⿷ (7)	區匿圡
⿸ (8)	庠雁屋
⿹ (9)	氛旬截
⿺ (10)	逑匏氾
⿻ (11)	匁卮包

すなわちここで作成されるのはある漢字画像を入力した際にそれがタイプ 0 ⿰からタイプ 10 ⿺の 11 個のラベルのいずれに当たるかを判別する分類器である．

またこの学習に用いる手法としては近年文字認識においては大きな成果を上げている (Lv et al. [5])，Convolutional Neural Network （畳み込みニューラルネットワーク）を用いることとし，その実行環境は Google によって開発されている Tensorflow を使用することとする．

4.2.1　訓練データの作成

訓練データの作成には何度も言及している CHISE の IDS データを用いた．構成字の記述には Unicode 以外のものも使用しているが，漢字構造のみに着目する場合は構成部品は不必要であり，また事実上 IDC 表現による唯一のデータであるからである．

CJK 統合漢字を，これをもとにタイプに分類した場合の出現頻度を第4.2表に示す．

約 8 万字のうち⿰ (0) である漢字が 5 万字強の一方，⿺ (6) の漢字は僅か 50 字と，千倍程度の差がある．訓練データの作成のためには，この偏りに留意しなければならない．

ここではこれへの対処として次の 2 つの方法を採った．

1. 不足している IDC タイプの漢字を創作して付け加える（創作法）TR1_SLF
2. 不足している IDC タイプの漢字を繰り返し複数回用いる（重複法）TR2_DUP

漢字の創作

IDS による漢字の構造表現は例えば以下のようになっている（いずれも CHISE IDS より）．

第4.2表 CJK 統合漢字の IDC 分布（CHISE によるデータ）

IDC	文字数
LEFT TO RIGHT ⿰ (0)	52,817
ABOVE TO BELOW ⿱ (1)	17,653
LEFT TO MIDDLE AND RIGHT ⿲ (2)	555
ABOVE TO MIDDLE AND BELOW ⿳ (3)	1,600
FULL SURROUND ⿴ (4)	342
SURROUND FROM ABOVE ⿵ (5)	815
SURROUND FROM BELOW ⿶ (6)	50
SURROUND FROM LEFT ⿷ (7)	157
SURROUND FROM UPPER LEFT ⿸ (8)	2,795
SURROUND FROM UPPER RIGHT ⿹ (9)	413
SURROUND FROM LOWER LEFT ⿺ (10)	2,833

IDS 表現の例

便 ⿰亻更
庁 ⿸广丁

以下では IDS 表現において各文字における左や上，外側の部分で示される漢字を第一要素，その次の部分として示される漢字を第二要素と呼称することにする．上の例では「便」の第一要素が亻，第二要素が更である．

漢字創作についてはこのうち第一要素を特に基準にして行う．興 (IDC6 ⿳) に対する第一要素「⾙」や閥 (IDC5 ⿵) に対する第一要素「門」といった例から，漢字においてその漢字を特徴づけるのは第一要素であるであるという直感的判断によってである．

漢字創作は IDC ごとに行い，その手順は以下のようにする．

1. この IDC タイプの第一要素の分布を得，その上位 90 % のうち出現回数が 6 以上のものを抽出する．ただしそれが IDC である場合，すなわち IDC による再帰的な構造を指示している場合は除き，Unicode 以外の外字コードは含む．
2. 第二要素の漢字タイプの分布を得る．
3. この IDC タイプで作成すべき文字数と第一要素分布から，第一要素の漢字を使用する回数を決定する．
4. 組み合わせる第二要素の漢字は，そのタイプ分布に従い，CJK 統合漢字基本ブロックからランダムに選択する．

ただしタイプ 2 ⿲は 4. の手順において，タイプ 0 ⿰の漢字からのみ選択することとする．

この手法で漢字を創作し，各ブロック最低 1,210 字を含むようにする．そのため，創作する漢字タイプは⿲ 2，⿴ 4，⿵ 5，⿶ 6，⿷ 7，⿹ 9 の 6 種である．

第4.3表から第4.8表に上記規則に従った各 IDC の第一要素分布を示す．なお表中で括弧書きして参照しているのは非 Unicode の CDP 外字である．

第4.3表　⿰ (2) の第一要素

第一要素	出現回数	累積出現頻度
氵	71	16%
亻	46	26%
木	42	35%
弓	26	41%
王	14	44%
口	13	46%
金	12	49%
言	11	52%
⿱	10	54%
女	10	56%
辛	10	58%
矢	10	60%
車	9	62%
糸	8	64%
米	6	65%

第4.4表　⿴ (4) の第一要素

第一要素	出現回数	累積出現頻度
口	228	68%
行	29	77%
衣	28	85%
辛	8	87%

第4.5表　⿸ (5) の第一要素

第一要素	出現回数	累積出現頻度
門	463	60%
冂	111	75%
冂	32	79%
鬥	31	83%
扁	23	86%
戊	17	88%
齊	14	90%
乃	6	91%

第4.6表　⿺ (6) の第一要素

第一要素	出現回数	累積出現頻度
凵	39	81%
貝(CDP-8BA8)	6	94%

また⿰ (2) を除く各 IDC の第二要素の IDC 分布を第4.9表に示す．

第4.9表中の各 IDC タイプの合計が必ずしも第4.2表の値と一致しないのは，第二要素として IDC や Unicode 以外の外字を選択しているものを除外しているためである．

以上の手順を具体的に記すと，まず IDC6 のうち第一要素が凵となって創作される漢字は 1,000 個であり，凵を含む漢字の Glyphwiki ソースを参考に

第4.7表　⿸ (7) の第一要素

第一要素	出現回数	累積出現頻度
⿸匚	120	78%
⿸亡	13	86%

第4.8表　⿰ (9) の第一要素

第一要素	出現回数	累積出現頻度
⿰勹	80	20%
⿰气	62	36%
⿰戈	45	48%
⿰戋	43	59%
⿰殳	30	66%
⿰⿱	19	71%
⿰弋	15	75%
⿰戔	12	78%
⿰弓	7	80%
⿰弋(CDP-88A7)	7	82%

第4.9表　第二要素の IDC 分布

第二要素の IDC	⿴ (4)	⿱ (5)	⿵ (6)	⿸ (7)	⿰ (9)
⿱ (0)	23	89	0	22	11
⿸ (1)	86	251	9	44	67
⿱ (2)	2	3	0	0	0
⿸ (3)	2	8	0	5	2
⿴ (4)	0	5	0	0	3
⿱ (5)	1	5	2	1	0
⿵ (6)	0	0	0	0	1
⿸ (7)	1	2	0	0	0
⿰ (8)	3	16	0	5	0
⿰ (9)	5	7	0	3	3
⿺ (10)	1	10	0	0	3
⿻ (11)	2	3	1	0	0
非合成	140	258	24	52	216

`99:0:0:0:0:200:200:u531a$99:0:0:34:34:172:169:xxxx` といった参照用コードを作成する．`xxxx` の部分に第二要素となる文字のグリフ名が入る．次に第4.9表に従ってランダムに第二要素の CJK 統合漢字を割り当てる．

こうして完成した一文字分のコードを KAGE エンジンにより SVG ファイルに変換する．また CNN に入力するために 56 × 56 の PBM 画像形式にラスタライズを行った．

以上により合計で 4,928 字の漢字を創作し，各タイプ最低 1,210 字の漢字を準備できた．`TR1_SLF` 訓練データは各タイプからランダムに 1,210 字選択した，合計 13,310 字分のデータとなった．

作成したうちからランダムに 100 字を抽出したものを第4.3図に示す。

なお重複法による訓練データ TR2_DUP は重複を厭わずにランダムに 1,210 字を選択して作成した.

4.2.2　CNN の構造

今回は Google が開発している Tensorflow というフレームワークを用いて学習を行う.

Tensorflow のチュートリアルに附属する MNIST の画像文字認識データに対して非常に高い精度を実現させたネットワークをほぼそのまま流用した．以下にそのネットワークのあらましを記す.

入力は 56×56 の二値画像であり，そこに 2 つの畳み込み層とそれぞれの直後のプーリング層，そして全結合層からなるネットワークがある.

第4.4図に 2 つの畳み込み層の模式図を示す．すなわちこの図の前段で漢字画像を 5×5 のパッチで畳み込みを行い，32 の特徴マップをもつ第一畳み込み層に接続する．ここでプーリングと再びの畳み込みとプーリングを行い，図の後段は 1024 のニューロンへ全結合し，ここから最終的な 11 のラベルの読み出しへと繋がる.

なおミニバッチのサイズを 15 とし，エポックを 1,000 とした.

4.2.3　テストデータと評価指標及び実行環境

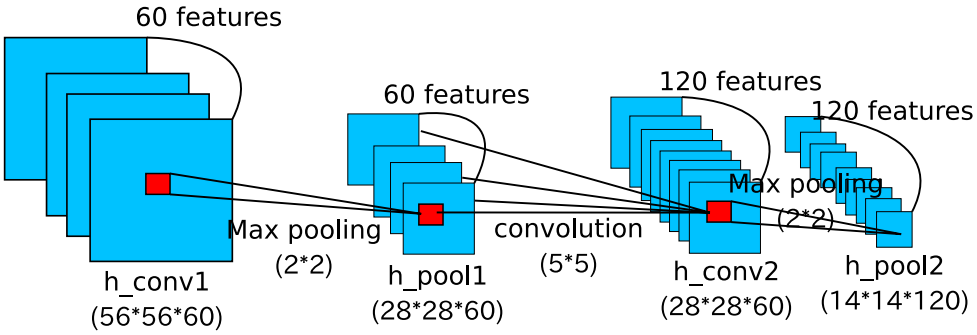
テストデータは以下のものを用意した.

1. 花園明朝のうち漢字全グリフ 80,029 字
2. インターネット上の創作漢字 84 字（Glyphwiki にてグリフ化されているもので以下を含む）[*1]
   - 産経新聞創作漢字コンテスト
   - セーラー万年筆創作漢字
3. 創作法で使用したのと同じ原理で新たに作成した創作漢字 3,000 字（IDC4，5，6，7，9 の 5 種類について）
4. その他のフォント それぞれ 3000 字
   - IPA 明朝
   - IPA ゴシック
   - Adobe Source Han Serif（源ノ明朝）

4. の一般フォントはあくまで参考のために使用したものであるが，これらに含まれる漢字の IDC の偏りは，まずは IDC タイプごとに均等になるように数を調整してランダムに選択し，もともとある IDC の漢字が少なくそれで不足する IDC タイプがあれば残りの IDC タイプの漢字をランダムに追加して 3,000 個とした.

[*1] https://glyphwiki.org/wiki/Group:%e5%89%b5%e4%bd%9c%e6%bc%a2%e5%ad%97

第4.3図 創作漢字の例



第4.4図　ネットワークの構造の模式図

なおテストデータの数を 3,000 個としたのは，今回の実験環境において一回のテストでメモリの観点から実用的に実行できるのがこの程度の大きさであったからである．

またその正解率についてであるが，特に一般フォントにおいては IDC タイプ間でその収録数に大きな差があるため，単純な正解率ではなく，ある漢字の真の IDC ごとの正解率の平均として計算する．

なお Tensorflow の実行環境は以下のようなものである．

- Python 3.5.2
- Tensorflow 1.1.0
- OS: Ubuntu 16.04
- CPU: Intel Xeon CPU E3-1246 v3 @ 3.50GHz

4.2.4　学習の結果

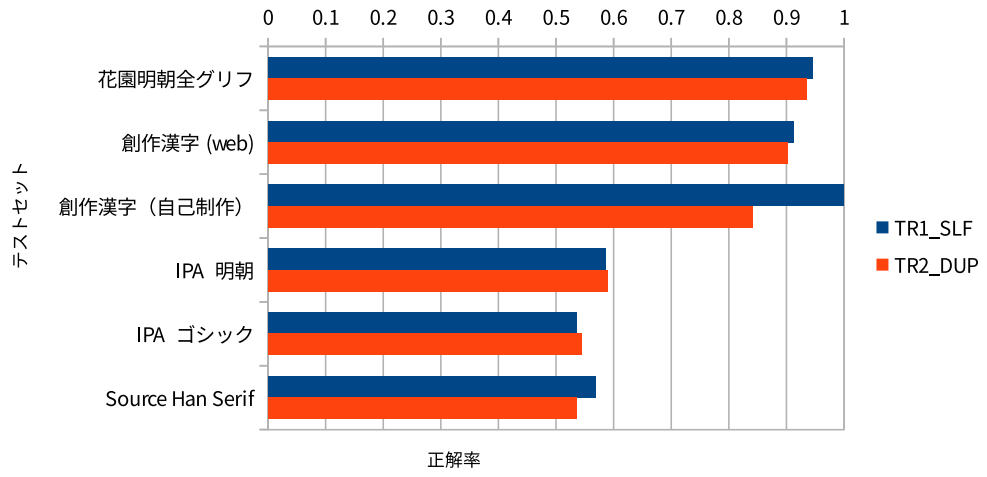
テストデータに対する正解率

それぞれのテストデータに対する，2 つの訓練データによる判定のブロック平均正解率を第4.5図に示す．

テストデータが花園明朝体であることから花園明朝と同じシェイプ，デザインでつくられたテストデータ（グラフの上 3 つのテストデータ）に対しては非常に高い正解率を記録した．

花園明朝とは異なるシェイプのフォントについてはいずれも正解率が 6 割を下回ったが，それでももともとがウェイトが細めの明朝体であるので，比較的それに似た IPA 明朝が相対的に正解率は高く，IPA ゴシックと，明朝体であるもののウェイトが太めである Adobe Source Han Serif がそれに続く結果となった．

なお TR2_DUP では不正解だったが創作漢字による TR1_SLF では正解となった例を 2 つ第4.10表に掲げる．



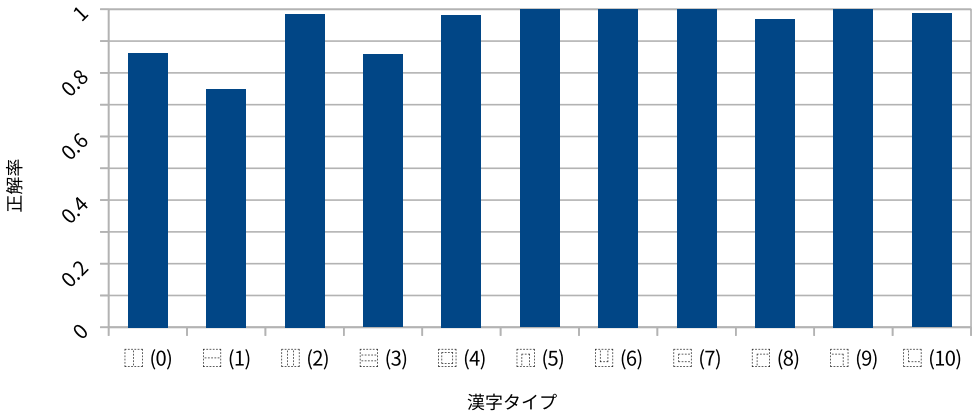
第4.5図　平均正解率

第4.10表　訓練データによる正解の差

文字画像	グリフ名（Glyphwiki における）	正解	TR2_DUP で誤認したラベル
虧	u2ff1-u624b-u98a8	□	□
鼬	u26932	□	□

IDC タイプごとの正解率

IDC ごとの正解率を，花園明朝全グリフに対する TR1_SLF による結果から得たものを第4.6図に示す．



第4.6図　花園明朝に対する漢字タイプ別正解率 (TR1_SLF)

他が軒並み 95 % を超える正解率を示すなか，IDC が⿰ (0)，⿱ (1)，⿲ (3) の 3 つの構造の漢字は一段低い 90 % 以下の正解率となっている．

4.2.5　考察

重複法と創作法

正解率について，IPA の 2 フォントを除いては，創作法による訓練の方が重複法による訓練より正解率が高くなっている．

正解率の差は自作創作漢字を除いてはいずれも 1 % 程度である（創作漢字：90.2 % に対して 91.3 %，全グリフ：93.5 % に対して 94.5 %）．

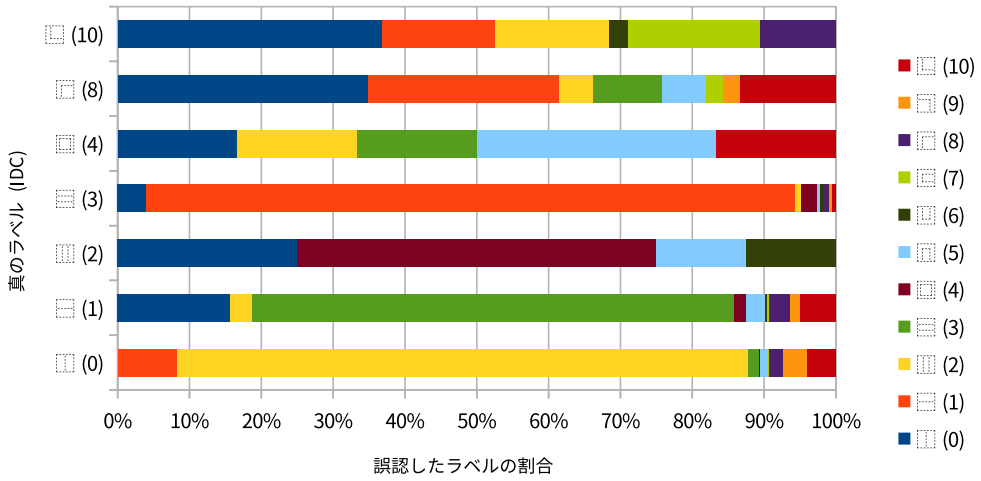
訓練データ作成と同じ方式で作成した創作漢字テストデータについては，文字デザインの癖などが作成方法に影響されて正解率が高くなったと推測される．

創作漢字による方法は，小規模ながらも改善が見込まれる方法であるということが言える．

漢字タイプの縮退

IDC タイプごとの正解率では，特定のタイプで不正解となる割合が大きかった．

そこで第4.7図に，不正解のなかった IDC6，7，9 と不正解が 1 つしかなかった IDC5 を除いた他の IDC タイプに真に属する漢字が他のどの IDC に誤認されたかを示す．



第4.7図　漢字タイプ別の誤認ラベル

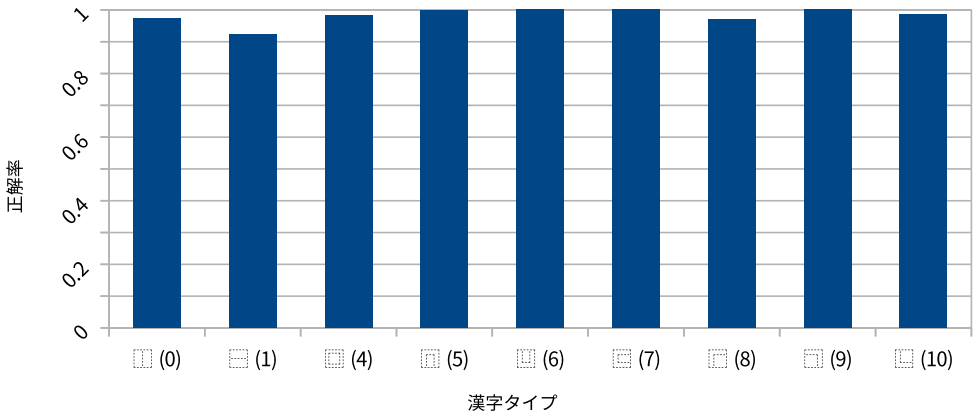
これによると IDC4，8，10 はどの誤認ラベルも単独で 40 % を超えないものの，上で注目した IDC0，1，3 はいずれもある一つの IDC に誤認する割合が 60 % を占めている．この対応関係を見てみると，

- ⿰ ↔ ⿰
- ⿱ ↔ ⿲

であることが分かる．これは興味深い結果である．先にも述べたように IDS の表現形式はその作成者によって任意であるため，□という漢字があったときに，これを□□と左右に分割できる漢字 2 つの組み合わせとしてもよいのである．

たとえばこの誤認の一例として，𥫱 (U+4259) という漢字は CHISE においては□のところ，この CNN では□と認識された．これは CHISE では□竹雲 としているが，一方で他にも□竹雨云 とも認識できるということとよく符合していると考えられる．

この観点から□ ↔ □，□ ↔ □をそれぞれ同じものとみなす縮退を行うと，IDC 認識の正解率は第4.8図のように平均で 98.2 % を達成する．



第4.8図　花園明朝に対する漢字タイプ別縮退正解率 (TR1_SLF)

ところで IDC2 □は IDC4 □に誤認されることが多いことが第4.7図から分かる．これは辦のような字を CHISE では□辛力辛 ではなく□辡力 としていることに影響された結果であると推測される．

4.2.6　本節のまとめ

今回の学習では訓練データの偏りを是正するために漢字を創作した訓練データを作成したが，これによる顕著な改善は見られなかった．とはいえおおむね正しく IDC による構造を判別することができた．しかし□ ↔ □，□ ↔ □の誤認のようなそもそもの IDS 表現の任意性に起因する誤判定は発生した．

しかし，この誤判定は大きな問題ではない．本研究の次の段階における漢字画像の認識を考えると，□と□はともに「縦に分割できる」という特徴でのみ認識できればよいと考える．このような「分割可能な方向」という情報をもとにある方向に走査していき，分割可能な線を発見するという手法を採るのであれば，□と□は区別する必要がないのである．

4.3　漢字画像の分割

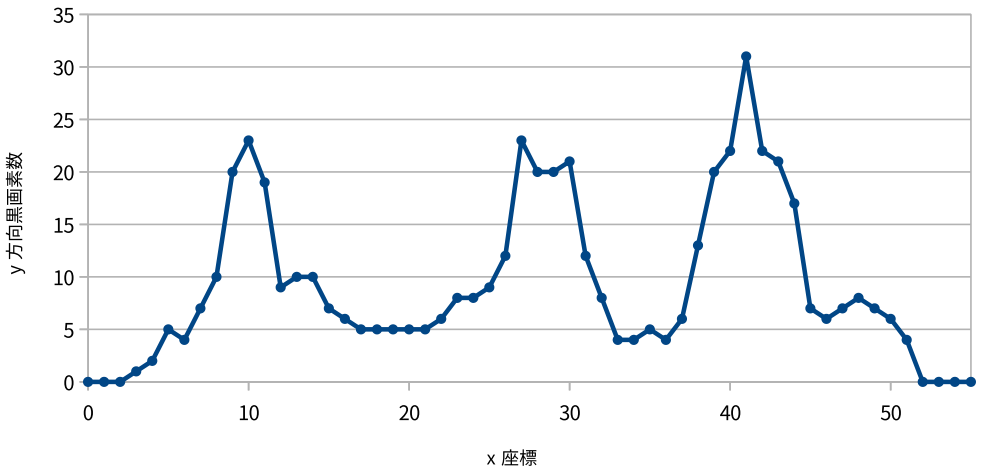
この節では漢字画像の分割，すなわち前小節の結果として推定した「漢字の切断方向」をもとに漢字画像を切断する位置を推定することを実現する．

文字画像の切り分け (segmentation) の手法については Casey and Lecolinet [1] が参考になるが，最も単純な方法はある方向に走査して，そのうち有色の画素の数が最も少ない部分において分割するというものである．本研究における部品分割は，漢字の整列した文章から漢字を切り出すのに比べて，部品の重なりがある点がこの単純な方法の適用を困難にしている．

ここでは単純な走査による分割位置の発見と，直交延長という概念の導入による分割位置発見手法の二つを比較する．

4.3.1　単純走査

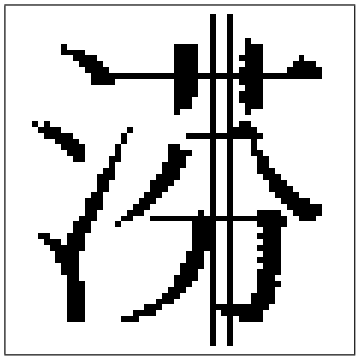
滞 (U+23DBC) を，縦方向の黒画素の数を集計した結果をグラフに表したものを第4.9図に示す．



第4.9図　滞 (U+23DBC) の単純な走査結果

この結果に基いて，画素数の極小値の中から最小値を選択し切断線とした場合，その切断線を示したのが第4.10図である．

その他のタイプについても単純走査による漢字分割を以下の規則に基いて行い検証する．



第4.10図　涜 (U+23DBC) の単純な走査による切断線

漢字分割の基本規則

- タイプ 0 から 3（  ，以後 A グループとする）の 4 種については，前節で記したようにと  ，  とは同種に扱い，x 方向もしくは y 方向に移動しながら走査し，その黒画素数の極小値のうち最小値（最小極小値）をとる x もしくは y 座標を結果とする．
- それ以外のタイプ（以後 B グループ）とするについては，各 IDC の形に基いて「折った」走査線に基き計数し，その黒画素数を走査線の長さで割った値（正規化）の最小極小値をとった座標群を結果とする（これは走査線で囲む領域が小さくなるほど当然に黒画素数も減少する影響を除くためである）．
- （当然のことながら）A グループの走査関数は x もしくは y の座標の値 1 つを返し，B グループの走査関数は，3 値以上の x もしくは y の座標を返す．
- B グループの走査範囲は，画像中最左の黒点から x 座標が 4 だけ右の点から右の範囲とする（ただし右上領域の第一要素があるタイプ 9  については左右を読み替える）．これは「風」のように第一要素 $<$ 几 $>$ が八字の如く広がっている場合に，几の先端の点のみを含んだ形で切断するのを避けるためである．

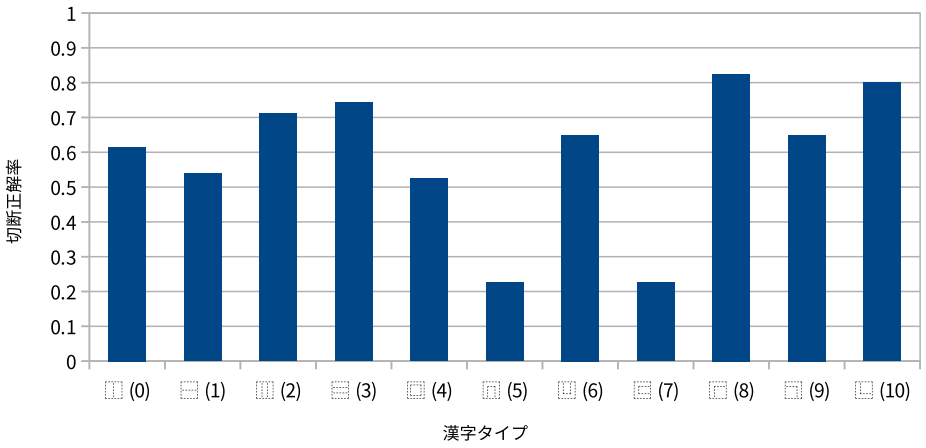
漢字 30 字ずつを分割し，その正誤を集計した結果を第4.11図に示す．平均正解率は 59.2% であった．

特にタイプ 5 と 7 の正解率が相対的に低いが，その典型的な誤認例として闇 (U+28D2A) と匝 (U+531D) を切断した結果を第4.12図と第4.13図に示す．

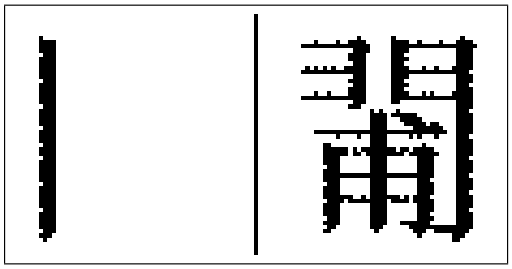
いずれの誤認結果も最も黒画素の少ない部分で切断した結果だが，漢字を知っている人であれば，横画などの線を跨ぐようには分割しない．このため単純走査ではなく，この点を考慮した走査法を考案した．

4.3.2　直交延長による走査

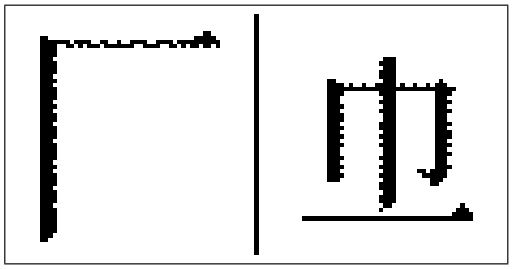
前小節で指摘したように，黒画素は少なくとも横や縦に伸びる走査線を切断線としないよう，走査線上の黒画素だけでなく，その黒画素から直交する方向に伸びる黒画素の量を



第4.11図　単純な走査による各漢字タイプ切断の正解率



第4.12図　鬧の単純走査による切断



第4.13図　匝の単純走査による切断

考慮することとする．この直交成分をここでは直交延長と呼ぶ．

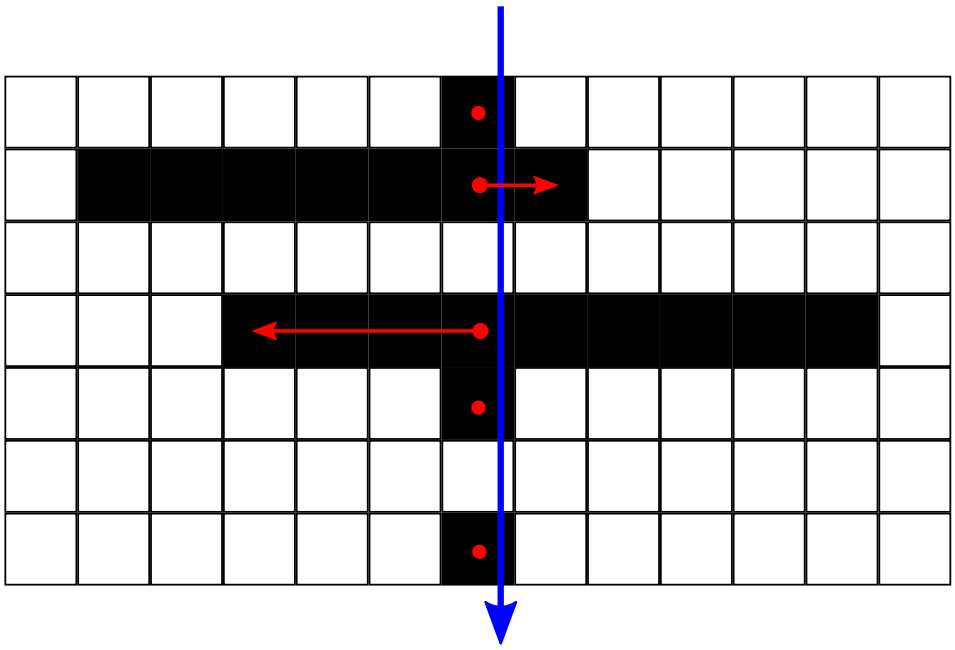
第4.14図に直交延長の概念を示す．ここでは青で示した線が走査線である．単純走査においては，この青線の通過する黒画素の数，すなわち図における赤点の数のみを数えていた．直交延長はこの赤点から，走査線と直交する方向へいくつ黒画素が続いているかを数えたものであり，図中の赤矢印線がそれである．このとき，直交する両方向へ画素を見ていくが，延長量の小さい方をとる．従って図の例でいえば，上の画素から直交延長量は 1, 2，0，4，1，0，1 となる．

この直交延長を再度, 澇 (U+23DBC) において走査, 計数した結果が第4.15図であり, これに基づいて直交延長量の極小値のうち最小値をとる x 座標を示したのが第4.16図である．

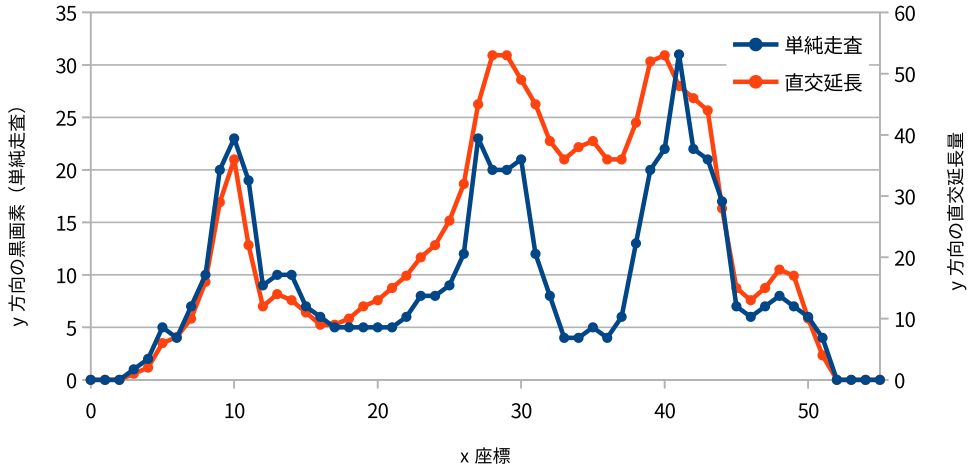
期待した通り，左右に長く伸びる直線を跨ぐ部分での量は落ちず，氵と芬に分割可能な点で最小極小値をとっている．

他のタイプについても，上記の「分割の基本規則」中の黒画素を直交延長と読み替え，ただし B グループに対する正規化作業を行わないとして，直交延長による走査を行い，それによる再度の分割を行った．その各タイプにおける正解率を示したのが第4.17図であり，平均正解率は 74.1 % となった．

また上で特に取り上げた鬧 (U+28D2A) と匝 (U+531D) についても，直交延長走査によって切断した結果を第4.18図と第4.19図に示す．正しく切断することに成功した．



第4.14図　直交延長の概念

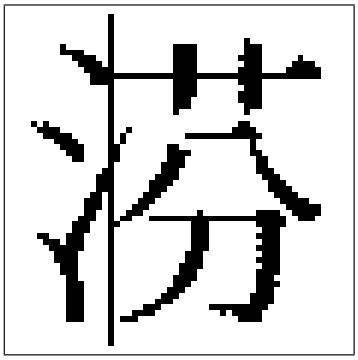


第4.15図　涔 (U+23DBC) の単純画素と直交延長による走査結果

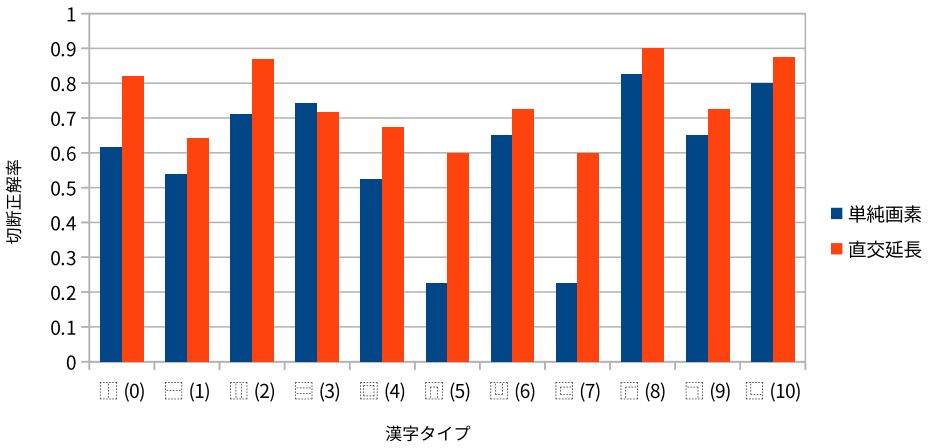
4.3.3　考察とまとめ

直交延長の導入によって，正解率は 59.2% から 74.1 % に向上した．

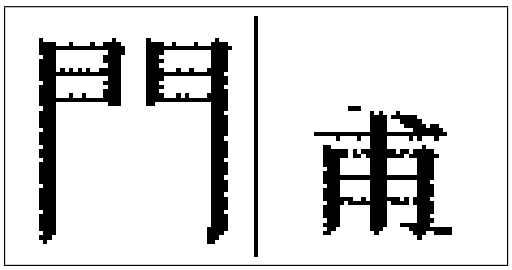
しかし完全に自動化できるほどには至っていない．またこのままの手法では慄 (U+6144) のように，忄（りっしんべん）などの部品内に切断線となりやすい白色部分の多い部分を含み，他の部品との重なりがあるような漢字については正しく判別することが期待でき



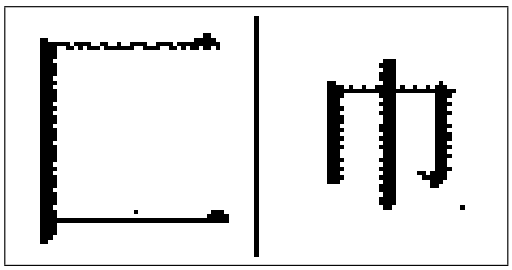
第4.16図　涝 (U+23DBC) の直交延長による切断線



第4.17図　直交延長走査による各漢字タイプ切断の正解率



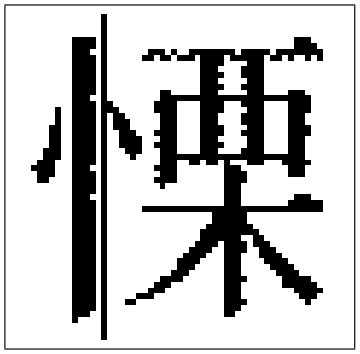
第4.18図　鬧の直交延長走査による切断



第4.19図　匝の直交延長走査による切断

ない.

そこで，この点については切断候補として提示するに留めるか，次節の部品同定において，分割してしまった部品だけでなく，特定のテンプレートを分割前の画像から探索可能にすることが望ましい.



第4.20図　慄の切断例

4.4 漢字部品の同定

最後に残された課題は，漢字の部品の同定（マッチング）である．基礎となる漢字部品のテンプレート群を作成し，それらとのマッチングを行うのである．

なおこの節のマッチングでは，前節の結果によって分割済みの部分文字の同定（例：「杯」を分割した「木」を木と同定する，以下「分割同定」）と，分割を施していない文字から特定の部首などを発見すること（例：「性」から「忄」を発見する，以下「探索同定」）の両方を想定した．後節で記す最終的なアプリケーションでは前者としてテンプレートマッチングを行っているが，開発途中においては後者を採用する可能性もあったため，以下の記述では後者の結果についても一部記している．

ある特定のテンプレートをターゲット画像の中に検出することをテンプレートマッチングといい，単純には，二つの画像とテンプレートの位置を入力とする類似度や相違度によって，その値が最も大きく（または小さく）なる位置を特定する．代表的な類似度関数 S としては相互相関関数 (cross-correlation) が，また相違度関数 D としては距離尺度が挙げられる（内田・石川 [10]）．

$$S_{\mathrm{CC}}(i,j) = \sum_{(x,y)\in\mathcal{T}} I(i+x, j+y)T(x,y) \tag{4.1}$$

$$D_p(i,j) = \left(\sum_{(x,y)\in\mathcal{T}} |I(i+x, j+y) - T(x,y)|^p \right)^{1/p} \tag{4.2}$$

ただし本研究で扱うような白黒 2 値画像の場合，類似度 S では I と T が共に 0 の場合も一方だけが 1 の場合も $I(i+x, j+y)T(x,y)$ が共に 0 になってしまう．Tubbs [8] では 2 値画像のための様々な類似度が比較評価されている．

本節ではこのテンプレートマッチングのための類似度関数として YULE の方法と，筆者が考案した最近傍距離による方法とを用いてその評価を行う．

4.4.1　類似度関数

以下で紹介するうち「最近傍距離関数」については相違度が大きいほど値が大きいため「相違度関数」と呼ぶべきだが，ここでは特に誤解の可能性もないため，両者とも類似度関数と呼び，その値も類似度と呼ぶこととする．

Yule 関数

2 値画像の類似度関数は Choi et al. [2] に紹介されているように種々あるが，Tubbs [8] でいくつかの類似度関数が比較されているうち，ここでは該論文において一般的に最も良い結果を残した Yule の手法を採ることとする．

まず次のようにデルタ関数を定義する．ただし X と Y は n 次元の 2 値画像のベクトルであり，x_m や y_m はそれぞれの成分である．

$$\delta_m(i,j) = \begin{cases} 1 & \text{if } x_m = i \ \&\ y_m = j \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

ここで n_{ij} を次のように定義する．2 値画像においては $(i,j) \in (0,1)$ である．

$$n_{ij} = \sum_{m=1}^{n} \delta_m(i,j) \tag{4.4}$$

これらを用いて Yule による類似度関数 S_{YULE} は以下のように定義される．

$$S_{\mathrm{YULE}} = \frac{n_{11}n_{00} - n_{10}n_{01}}{n_{11}n_{00} + n_{10}n_{01}} \tag{4.5}$$

S_{YULE} は 2 つの画像が完全に一致した時に最大値 1 をとり，最小値は -1 である．

最近傍距離関数

また本論文では類似度を測る関数としてさらに一つ,最近傍距離関数 (Nearest Neighbor Distance) を使用した．

最近傍距離 $d_{\mathrm{NN}}(x,y)$ を，テンプレート画像中の画素 $T(x,y)$ が黒であった場合，この画素に対応するターゲット画像の画素 $I(i+x, j+y)$ から x 方向もしくは y 方向に最も近傍にある画素との距離とする．画素は白であった場合は 0 で，またテンプレート画像に対応する範囲のターゲット画像中に最近傍画素が存在しない場合はその最近の端点までとする．すなわち d_{NN} の値は 0 以上の整数となる．

これを用いて次のように定義した類似度関数を使用する．

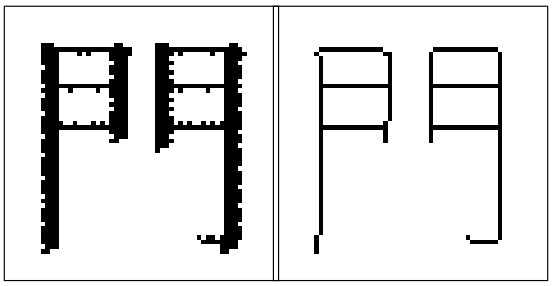
$$S_{\mathrm{NND}} = \frac{\sum_{(x,y)\in\mathrm{black}} \exp\left[\alpha d_{\mathrm{NN}}(x,y)\right]}{\sum_{(x,y)\in\mathrm{black}} 1} \tag{4.6}$$

線同士が離れているほどスコアは大きくなるが，この離間の影響を大きくするためにこ

こでは指数関数を用い，またその離間のペナルティ係数 α を使用する．分母の画素数は正規化のためのものである．

この関数は最小値 1 であり，線の離間が大きいほど，すなわち相違度が大きいほど値が大きくなるものである．

また本手法は性質としては漢字の線の近さを計量するものであるから，必ずしも筆画の太さを必要としないと考えられ，また計算時間の短縮も図るために，ターゲット画像，テンプレート画像共に細線化 (Skeletonization, Thinning) を行う．ここでは Zhang-Suen の細線化アルゴリズム [9] を用いる．第4.21図にその例を示す．

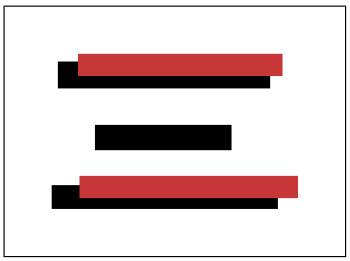


第4.21図　門 (U+9580) に対する細線化の例

加えて，本手法を使用する際，テンプレート画像中の線からターゲット画像の線がどれだけ離れているかと共に，ターゲット画像中の線からテンプレート画像の線がどれだけ離れているかも計算し，その合計をスコアとする．

例えばターゲット画像「三」とテンプレート画像「二」が第4.22図のように重なり合っているとする．このときテンプレート画像の線（赤）についてしか S_{NND} を計算しないとすると，ターゲット画像中の中央線は評価にまったく加えられず，仮にターゲット画像が「二」であっても値は変わらなくなる．実際は「三」の中央線のために相違度は増すことが望ましいため，テンプレート画像の大きさの範囲においてはターゲット画像の線についても，テンプレート画像の線とどれだけ離れているかを評価に入れる．

そのため $S_{\mathrm{NND}} = S_{\mathrm{template}} + S_{\mathrm{target}}$ の最小値は 2 となる．



第4.22図　ターゲット「三」（黒）とテンプレート「二」（赤）の模式図

4.4.2　テンプレートマッチングの手順

基本となるテンプレートマッチングの手順は以下の通りとする．類似度関数は任意である．

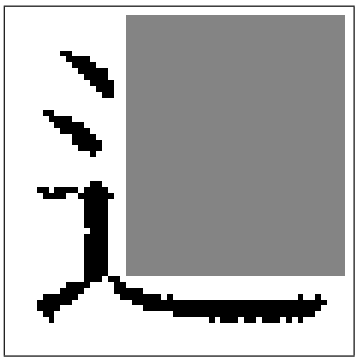
1. ターゲット画像の文字周囲の空白を削除する
2. テンプレート画像がターゲット画像より大きい場合は縮小する
3. テンプレート画像のマスクを行う（後述）．
4. テンプレート画像の縦横の大きさをそれぞれ 1.0 倍，0.9 倍，0.8 倍した画像（ただし一部の場合は別途定めた値，後述）を計 9 枚用意する．
5. テンプレートマッチングを行い，最大値（もしくは最小値）を得る．

この手順を検証するテンプレート全てについて行い，その序列を得る．

マスクについて

特に探索同定の場合に，漢字の形に由来する修正を画像に施す必要がある．ここではタイプ 10 □を例に挙げる．

□に属する漢字としては辻，颪，赴などが挙げられるが，この場合の第一要素はそれぞれ辶，風，走である．「辶」をテンプレートとしてそのまま使用した場合，この文字画像の右上部の白部分と，テンプレート文字画像のこの場合でいえば「十」の部分との照合が行われ，相違度が増加する．本来は左下の部分のみの特徴で照合を行いたい文字である．そのため，テンプレート画像に対して，その照合時に考慮しない範囲を定める．辶の空白部にあたる．例を第4.23図に示す．これを便宜上以後マスクと記す．



第4.23図　辶におけるマスクの例（灰色部がマスクされる部分）

縮小倍率の修正について

同じくタイプ 10 □の漢字について，上にも挙げた「颪」を前節までの手法により切り出した場合，風の変形である「凪」と「風」の照合を行うこととなる．しかしこの部首部品は風を特殊な形で変形したものであるので，上で記した倍率の縮小は適さない．そのた

め，タイプ 10 □の漢字の第一要素に対するマッチングの場合で，かつ前述のマスクをテンプレートに施した場合にマスクを行った領域が一定以下のものについては，横方向の縮小倍率を 0.4 倍，0.5 倍にすることとする．

4.4.3　2 手法の比較

上記の 2 つの手法によるテンプレートマッチングの比較評価を行う．特定のタイプについてと全体についてそれぞれ見ていく．

タイプ 10 □における部品推定（テンプレート制限）

最初にタイプ 10 □における両者の比較をするが，このタイプの漢字に特殊なのは，上述したように部首となる漢字が変形することが多いことである．「風」に対する「風」などである．

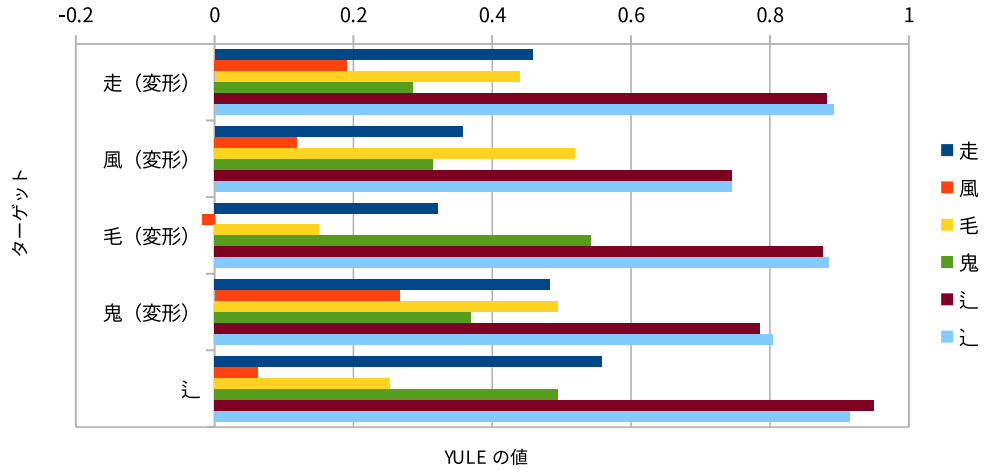
ここではこのようなタイプ 10 用の部首に変形した漢字 4 つと辶（之繞（しんにょう））に対して，そのもとの形をテンプレートとしてその推定が可能かをみる．

ターゲットとしては，走，風，毛，鬼，辶 を設定し，テンプレートはそれぞれの変形前の漢字である走，風，毛，鬼と辶，辶（いわゆる一点之繞と二点之繞）である．

なおテンプレート画像の横方向の縮小倍率については辶とそれ以外で半分程度の大きさとするか，1 倍程度の大きさとするかは上述の通り適切に処理している．

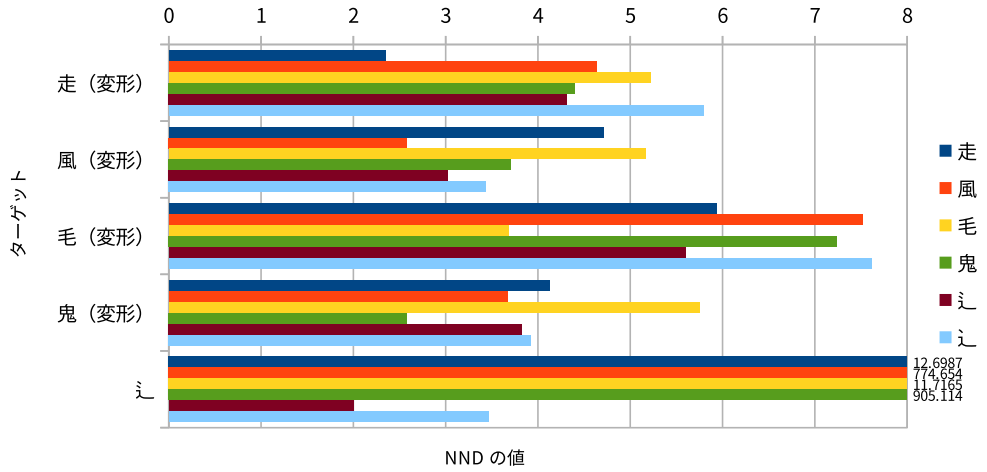
また第4.6式における離間の係数 α については $\alpha = 0.5$ としている．

第4.24図に Yule による類似度を，第4.25図に NND による類似度（相違度）をグラフに表す．第4.25図では比較のために数値軸を 8 までに留めたが，ターゲット「辶」に対するテンプレート走，風，毛，鬼の値はそれぞれ 12.7，774.7，11.7，905.1 となっている．



第4.24図　YULE によるタイプ 10 □における部品推定

YULE の手法においては辶に対してのみ正解となり，その他のターゲットは軒並み辶



第4.25図　NND によるタイプ 10 ⿴における部品推定

の値が大きくなっており，判定が辶に傾きがちである．

NND の手法（2 に近いほど類似している）においては，全てのターゲットに対して意図したテンプレートで値が最小となっていて正解している．

一般のテンプレートによる部品推定

次にテンプレートを拡大して，両手法を比較する．

テンプレート群としては，日本の常用漢字と人名用漢字，それらの構成部品である CJK 統合漢字（これは CHISE の IDS データに基く）の合計 3,245 字とする．ターゲットは前節の漢字分割によって分割した部品文字 30 字とする．

比較のための指標は，それぞれの手法によって作成したテンプレート群の序列中の正解文字の順位とする．なおこのうち 50 位以内に正解となる文字が存在しない場合は検出失敗（圏外）とする．

ここではタイプ 0 ⿴やタイプ 1 ⿱などで分割された長方形型の部品文字をターゲットとする．

結果をまとめたものを第4.11表に示す．

新手法である NND 法の方が順位が改善したものが YULE の 2 倍程度と多くなっている．

ただし YULE でのみでしか検出できなかったものが 2 字と，NND のみでしか検出できなかったものより多くなっている．

タイプ 10 ⿴の部首 10 種について同様にテンプレートマッチングを行った結果が第4.12表である．

NND の方が僅かに成績は良いものの，大きな差はない．

第4.11表　一般タイプにおける部品推定の正解順位の結果

種別		内訳　（括弧内の値は改善した順位の平均値）	
双方とも圏内	23	YULE 優位	6 (−9.8)
		NND 優位	13 (−4.2)
		同順位	4 (±0)
YULE のみ圏内	2		
NND のみ圏内	1		
双方とも圏外	2		
合計	30		

第4.12表　タイプ 10 □における部品推定の正解順位の結果

種別		内訳　（括弧内の値は改善した順位の平均値）	
双方とも圏内	7	YULE 優位	3 (−9.6)
		NND 優位	4 (−10.8)
		同順位	0 (±0)
YULE のみ圏内	2		
NND のみ圏内	0		
双方とも圏外	1		
合計	10		

4.4.4　考察

タイプ 10 □の走，風，毛，鬼，辶に対する実験では，YULE においてはいずれの部首も辶に誤認していることから，部首の存在する文字左側部分の形よりも辶と同じように右下部まで筆画が延びていることが評価されていると考えられる．一方 NND においてはあくまで線同士の近さの評価であるから筆画の集中する部首部分における近接度が評価されたものと考えられる．

テンプレート群が大きくなり，その他の漢字が多数使用されても，上記のような顕著な差こそ見られないものの，NND が優位であった．

なおここで漢字タイプごとに，その第一要素（「証」に対する「言」や，「颱」に対する「風」など IDS 表現において最初に表記される方の部分文字．一般的な認識の部首と近い）の分布について，HHI (Herfindahl-Hirschman Index) を計算した結果を第4.13表に示す．HHI は本来は経済学において寡占度を表す指標であり，全ての要素の占有率の二乗和で計算される．この値が 1 に近いほど寡占であり，0 に近いほど特定のものへの集中度合いが小さいことを表す．

いくつかのタイプでは第一要素が特定のものに限られていることが分かる．例えば□に

第4.13表　漢字タイプの第一要素の HHI

漢字タイプ	HHI
⿰ (0)	0.017
⿱ (1)	0.022
⿲ (2)	0.054
⿳ (3)	0.021
⿴ (4)	0.479
⿵ (5)	0.391
⿶ (6)	0.677
⿷ (7)	0.616
⿸ (8)	0.166
⿹ (9)	0.105
⿺ (10)	0.094

おいては凵，⿵においては門が第一要素の大きな位置を占める（この第一要素の一部は第4.3表から第4.8表に示してある）．

この事実から，マッチングのためのテンプレートは，漢字タイプによっては特定のものしか事実上登場しないため，最初から制限を加えたり，それ以外の漢字についても，第一要素になりやすい漢字から優先的にマッチングを行うなどの工夫を行うと，良い結果が得られるであろうと考えられる．

4.5　本章のまとめと作成した解析プログラム

本章では，漢字の構造を自動で判別するために，CNN による漢字タイプの推定，分割線の推定，部品の同定という 3 段階に分けて，それぞれの実現について論じた．

第 1 段階では 98.2 %，第 2 段階では 74.1 % の正解率を達成した．第 3 段階でも既存の単純なテンプレートマッチングより僅かに改善できる手法を実現した．

しかし各々の正解率が過半数であっても，それを単に接続しただけでは，一連の自動判別は実現できない．

そのため今回は，全自動で漢字の構造を判別するのではなく，これら実現した技術を用いて，人間の判断の補助をする Computer-Aided なプログラムを作成するのに留まった．

このプログラムでは，入力された漢字画像を上記の手段で解析した結果を初期値として表示するものとして利用し，その結果が誤っていたり，不満であれば，ユーザが他の値を入力できることとした．第 3 段階では，スコアの良いものを順に表示しそこから選択するものとした．

全自動でないものの，部品としたい漢字が既に Unicode に存在するかどうかの知識などは必要なく，直感による切断線と同定の判断のみに労力を割けばよいプログラムが実現

した.

なおこれを実際に作成する体制については検討すべき課題である.

第 5 章

漢字構造を利用する漢字入力方式

本章では，漢字甲は漢字乙と漢字丙の組合せであるというような漢字構造を利用可能な入力方式を提案，実装し，その評価を行う．

評価は打鍵数による評価と，手書きパッドとの入力時間による比較の二つを行った．

5.1 システム設計

第3章の再掲になるが，このシステムは次の特徴を持つこととする．

部品は文字 漢字は筆画（一，丨，丿，丶，フ）まで分解することも可能だが，ここでは会意や形声の原理でつくられた漢字を，その構成漢字に分解する程度に留める．一般的な漢字の説明と同様な感覚で使用可能にするためである．なお必ずしも漢字だけに留めないが，Unicode で使用できる画像文字に限定する．

配置は考慮しない このシステムではあくまでどの漢字から成っているかだけに留め，それがどのように配置されているかという情報は使用しないこととする．つまり峰と峯の構造情報は同一となる．

補助手段 またあくまでこれは既存の文字入力法の補助に留める．現在は読みによる入力が大勢であり，かつて T コードが定着しなかったことから推し量るに，読みと全く異なる原理で入力する入力法は習得までの障壁が大きく，普及が阻害される．よって基盤としては読みによる入力法を使用しつつ，そこから呼び出して使用できる方法とする．

キーボードを使用 入力に際してはコンピュータキーボードでの入力を前提とする．

交換可能なデータ 基礎として使用する構造データは交換可能であること．

この上でさらに留意すべきこととして以下を挙げる．

- 可能な限り打鍵数を小さくする
- 容易に理解，習得できる使用法であること

以下ではこのシステムのデータ管理方式と具体的な機能及びその使用方法について

記す.

5.1.1　データ管理方式

このシステムでは個々の漢字の情報を持った構造体の配列で文字情報を管理する．主配列の添字は文字の Unicode コードポイントとする.

各々の構造体は以下に掲げる 3 つの可変配列を持つ.

- この漢字の構成要素のコードポイント（親）
- この漢字を含む漢字（一世代派生漢字）のコードポイント（子）
- 同じとみなす漢字のコードポイント

なお以下ではある漢字をその一部に含む異なる漢字のことを派生漢字と呼ぶこととする．このデータ構造の中で直接指定されたものを一世代分の派生漢字とする．またデータ構造として構成要素や派生漢字を捉える場合はそれぞれ親や子とすることもある.

「應」(U+61C9) を例にとると，これの親は「广，隹，心」となり，子は「噟，嚛，曮，礒，蘟，蠅，鷹」となる．なお親の一つ「隹」はさらに「亻，隹」と分解可能なため,「應」の親として「广，亻，隹，心」としてもよいこととする．すなわち IDS 表現のうち分解性については任意とする.

また「同じとみなす文字のコードポイント」は以下の漢字のために使用する.

1. CJK 統合漢字の日本語例示字形と CJK 互換漢字の字形が一致しているもの
   第2.2.3節で記したように包摂と原規格分離の両立のために，ある地域のフォントでは同じグリフとなる文字が 2 つ以上登録されていることがあるため.
   （例）統合漢字の「林」(U+6797) と互換漢字の「林」(U+F9F4)[*1]
2. 本来の字体差が失われフォント上は同じグリフとなっているもの
   例えば，部首としての「月」は本来「にくづき」と「ふなづき」という二つの別のものであり,「肝」の部首は「にくづき」,「朔」の部首は「ふなづき」である[*2]．ところが日本においては現在この 2 つのデザイン差は消失し，同一視されている．そのため CJK 統合漢字の内部で腋 (U+6718) と腋 (U+8127) のように日本語フォントではグリフに違いのない文字が存在する.
3. 部首用の文字等と CJK 統合漢字との対応及び部首の統合
   Unicode には CJK 統合漢字や互換漢字以外にも漢字に類するものが存在する．CJK Radicals（部首）や CJK Radicals Supplement や漢文訓読用の文字（一，二点や上中下など）である.
   部首としての鬼は「鬼」(U+2EE4) として文字コード上は別の文字である．また之

[*1] この互換漢字は二つの「林」が韓国の文字集合において別の文字となっているために登録されている.

[*2] 余談であるがこの 2 つの区別のために，中国や台湾では「にくづき」を 月 としていて，また日本の旧来の活字では「ふなづき」を 月 としていた.

繞にあたる部首文字は部首ブロックには 3 つの異なる文字（⿺，⿹，⿻ U+2ECC – U+2ECE）として登録されている．

5.1.2 機能

この漢字入力システムでは主に以下の 2 つの機能を提供する．

1. ある漢字（単複両方）の派生漢字の提示
2. ある漢字の構成部品への分解

構成部品の提示

指定された漢字の漢字構造体の構成部品配列からその構成部品となっている漢字を得る．これを再帰的に繰り返し，一世代の親だけでなくすべての親の系統の集合「先祖集合」を得る．

派生漢字の提示

次の手順で派生漢字を取得する．

派生漢字集合の取得

指定された漢字が構成要素（親）を持たない場合 その派生漢字を一世代だけでなく再帰的に取得することとし，データツリーの上で，所定の漢字の「子孫」にあたる漢字を全て取得する．複数の漢字が指定された場合は，それぞれの漢字について子孫集合を得て，その集合の積を結果とする．

指定された漢字が構成要素（親）を持つ場合 前記した手順で構成部品を再帰的に使用し「先祖集合」を得る．以降の手順は上記の「親を持たない場合」と同様で，先祖集合それぞれの子孫集合からその積を得る．

親漢字を持つ場合に，一旦先祖集合まで求めるのは，構造表現の任意性の担保のためである．例えば，「衙」という漢字は「彳，吾，亍を左から並べたもの」とも「行の中に吾を入れたもの」とも表現できる[*3]．データを前者のように記述していた場合に，親漢字を顧みずに単純に派生漢字だけを得ていたのでは，後者のように解釈して行を含む漢字として入力を望む際に対応できない．

5.1.3 具体的な使用方法

今回の評価に使用するプログラムは，既存のインプットメソッドを拡張するものではなく，独立したプログラムである．

[*3] 実際，CHISE の IDS データでは後者のように記述してある．

ここでは「,」や「、」を区切り文字として，合成したい漢字（単数でも複数でも良い）を入力する．このとき，区切り文字で区切られた各文字列は 1 文字目のみ認識する．すなわち送り仮名をともに入力したり熟語として入力したりした場合でもそれらが 2 文字目以降にある限り無視するということである．

また漢字を分解したい場合には，分解したい漢字を含んだ文字列の 2 文字目以降に平仮名の「の」を含めることとする．

以下にいくつか操作の例を挙げる．

「杼」を「木」と「子」から入力する場合

```
> 木，子　　（⏎）
=====木 子 を含む漢字=====
=> あ：李 い：杼 う：栫 え：桲 お：桴　　（い⏎）
>>> 杼
```

「氓」を「瓜」と「失う」から入力する場合

```
> 瓜，失う　　（⏎）
=====瓜 失 を含む漢字=====
=> あ：氓　　（あ⏎）
>>> 氓
```

「杌」を「木」と「元」の部品から入力する場合

```
> 木，元気の　　（⏎）
----------「元」の分割----------
=> あ：元 い：一 う：兀　　（う⏎）
=====木 兀 を含む漢字=====
=> あ：杌 い：橈　　（あ⏎）
>>> 杌
```

5.2　評価環境

提案する漢字入力方式を,

- 読み主体の入力方式と打鍵数（ストローク数）において
- 手書きパッドと入力に要する時間において

比較した.

5.2.1　使用する構造データ

今回は漢字入力システムの構造データとして以下の 3 つを使用する.

1. 常用漢字，人名用漢字及びそれらの構成字（基礎リスト）
2. 日本以外の漢字文化圏の人名に使用されている漢字
3. JIS X 0213 の第 4 水準の漢字からランダムに抽出したもの

前章で作成した漢字構造解析プログラムによるデータ作成は未だコストフルであるため，今回は利用可能な範囲で CHISE の IDS データを用い，適宜これに修正を施した．

1 の基礎リストは，日本において初等，中等教育を受けていれば知っているであろうと仮定して基礎とするものであり，常に読み込む．もととなるのは常用漢字 2,136 字と人名用漢字のうち表一の 649 字[*4]と，CHISE の IDS データをもとにそれらの IDS 表現に使用されている文字のうち Unicode に含まれるものの，合計で 4,224 字．人名用漢字のうち表二の取り扱いについては後述する．

2 のリストは，日本人には馴染みが薄いが，海外の漢字文化圏で使用され，入力の需要が他の漢字に比べて大きいと思われるものを収集した．その作成方法と漢字一覧は巻末の附録 A に記すが，結果として 202 字からなる一覧を作成した

3 のリストは，本手法と Microsoft IME の手書きパッドとを比較するために JIS X 0213 の第 4 水準から，手書きパッドに対応している漢字をランダムに抽出した 16 字．

また同一と見做される文字についての情報も追加した．これは Glyphwiki の漢字データで，エイリアス機能を用いて他の 1 文字を参照しているものの一覧を基本とし，3 種の之繞などの部首の同一視データを適宜追加した．

異体字の取り扱いについて

漢字には異体字という概念が存在することは第2.1.2節に記したが，今回の評価段階では，日本語フォントにおいてグリフが同一またはほぼ同一であったり，部首の統合において必要である場合以外は，異体字は特に考慮しないこととした．異体字には「亜」に対する「亞」,「恵」に対する「惠」などのいわゆる「旧字体」も含む．

一口に異体字や旧字体といっても，上に挙げた「亞」の例は分かりやすいが，「野」と「埜」というふうに形が大きく変わっているもの，「著」と「着」のように現在では異なる字と認識されているもの，本来異なる字を一つにまとめた「弁」と「弁，辯，辨，瓣」などというようなものまで異体字という範疇に普通は含まれる．ここに挙げたのは全て常用漢字内の漢字の異体字であるが，一般の漢字使用者にその関係の知識を期待するのは難しいと判断したため上記のようになった．

そのため，常用漢字に対する異体字として名付けに使用されることが許される漢字である人名用漢字の表二については取り扱わないこととする．

5.2.2 漢字表示順

評価で用いる際の候補漢字の表示順は従来手法としては以下の通りとする．

[*4] データ生成時のもの．2018 年 2 月までに 2 字追加されている．

1. 常用漢字
2. 人名用漢字
3. 1，2 以外の JIS 漢字
4. Unicode CJK 統合漢字基本ブロック
5. Unicode CJK 統合漢字拡張ブロック（A ブロックから昇順）
6. その他（互換漢字など）

各項目内での順序は，1 から 3 までは JIS X 0213 の面区点コード順，4 以降は Unicode のコードポイント順である．

提案手法による漢字入力法では，常用漢字などの「身近」な漢字は提案手法では入力しないだろうという想定から，1 の常用漢字を最後尾に移動する．

5.3　打鍵数による評価

特定の漢字を入力するために必要な打鍵数（ストローク数）を，提案手法と一般的な読みのみで変換する漢字入力方式とで比較する．

5.3.1　この評価の環境

ここでは仮名の入力にローマ字入力を使用したとし，各々の漢字の読みは Unihan Database （第2.4.1節）に拠る．

計測する打鍵は仮名を入力するのに必要な打鍵と，変換モードに移行などの操作打鍵，変換候補の確定操作打鍵である．

従来手法としては日本語入力インプットメソッドで一般的な方式である，未確定の仮名を入力し，スペースキーもしくは変換キーで変換モードに移行し，スペースキーによって候補を繰っていき，所望の漢字でエンターキーを押下するものを想定する．

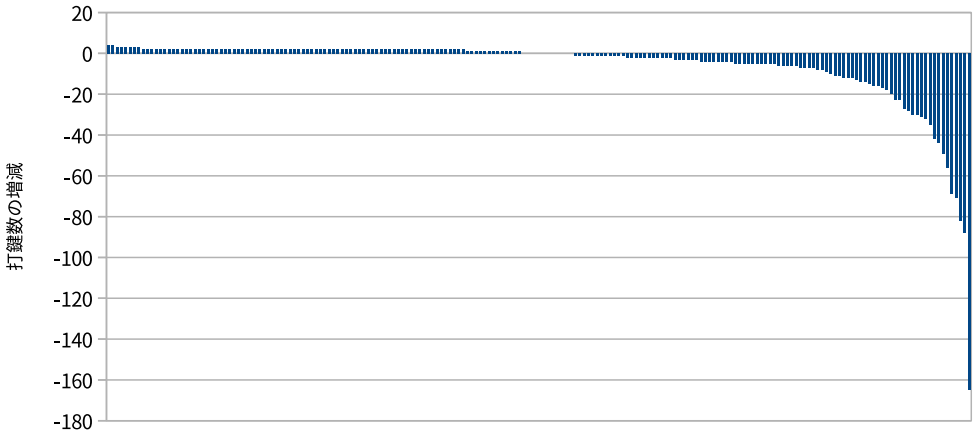
比較する提案手法の想定動作は，基本的には前節で紹介したシステム説明に準拠するものの，変換候補の送りと選択は従来手法と同様のスペースキー送りとエンターキー確定とする．また今回は合成操作のみを用いたこととし，分解して一部を使用してさらに合成する，前節で「杭」を入力する手法として紹介したような方法については考慮しないこととする．

比較対象となる漢字はリスト 2「漢字人名リスト」202 字を使用する．ただし 202 字のうち杤 (U+674B) と誹 (U+8ABD) の 2 字は Unihan Database において読みが設定されていないため，打鍵数増減比較の対象からは除く．

漢字の読み，到達方法が複数存在する場合はそのうち最小のものとする．

5.3.2　結果

結果としてまず漢字ごとの打鍵数の増減を整列させたグラフを第5.1図に示す．また第5.1表に増減数の分布を示す．対象の 200 字のうち，打鍵数が減少（グラフ中の負値）したものが 92 字，残りの 108 字の打鍵数は変わらないか増加した．



第5.1図　打鍵数の増減

第5.1表　打鍵数の増減の分布

区間	負値	0	1	2 〜 4	合計
数	92	12	13	83	200

まず 2 以上打鍵数が増加したものについて述べる．

提案手法では，従来手法に比べて漢字を特定するまでに必要な「確定動作」が 2 回多い．一つが構成要素の各々の漢字の確定で，複数の文字列を入力する場合になるべくエンターキーを押さないとしても最後の文字列の変換確定で最低 1 回は必要となる．さらにこれらの変換前の構成要素文字列を確定するためにもう 1 回エンターキーを押す必要がある．そのため同一の漢字を同一の読みで入力しても 2 回は打鍵数が増加する．

また 3 回以上の漢字は，その漢字を含む漢字一覧で所望の漢字が第 1 位で登場しないことによって増加している．例えば，「尹（おさ）」は打鍵数が 3 増加しているが，これを入力した場合の提案方式の挙動を下に示す．ただしこれは選択肢をキー入力で選択する方式であるので，比較評価に用いたスペースキーで送る方式とは異なるが，順番などは同一であり参考として掲載する．

参考：「尹」を入力した場合（打鍵数評価とは異なる提示方法）

```
> 尹
=====尹 を含む漢字=====
=> あ: 伊 い: 尹 う: 君 え: 群 お: 郡
>>> 尹
```

このように「尹」と入力すると，これを含む漢字一覧が表示され，第 1 位に人名用漢字の「伊」が来るため，打鍵数が増加するのである．

すなわちシステムの設計上，打鍵数が 2 増加することに鑑みると，実質的に打鍵数が減少しているのは打鍵数増減が 1 以下の漢字である．

ただし増減数 1 以下の 117 字のうち，漢字を 2 つ以上指定して合成入力したのは 8 文字に留まる．大多数は，その構成要素のうち 1 文字を短い読みで入力して，それを含む漢字から選択するものである．たとえば「芬」はその要素の「分」を入力して，これを含む漢字として選択している．

最も打鍵数が減少したのは「瑄」の −165 である．「せん」という音読みを持っているが，「せん」を読みに持つ漢字は CJK 統合漢字内に 378 字存在する．そのため従来手法では，その大半はスペースキーを押下する 178 回の打鍵が必要だったが，提案手法においては，その構成要素の「宣」を訓の「のる」で入力して，その派生漢字から選択する手法で，打鍵数は 13 に留まっている．

その他の例についても打鍵数の減少の大きい 10 種について第5.2表に示す．

第5.2表　打鍵数の増減（減少十傑）

入力した漢字	読みによる打鍵数	入力	提案手法による打鍵数	入力	打鍵数の増減
璐	47	ろ	12	路（ろ）	−35
洙	49	しゅ	7	朱（あか）	−42
阮	54	げん	10	元（こうべ）	−44
郝	58	しゃく	9	赤（あか）	−49
鄧	67	どう	11	豆（まめ）	−56
璨	78	さん	9	粲（いい）	−69
澎	88	ほう	17	壺（たてる）	−71
瑋	101	い	19	韋（なめしがわ）	−82
璜	94	おう	6	黄（き）	−88
瑄	178	せん	13	宣（のる）	−165

2 つ以上の漢字を用いた例としては「燮」が挙げられる．これは「火」と「言う」を用いて入力し，打鍵数が 13 から 9 に減少した．

また読みの存在しなかった 2 字についても，第5.3表のようにその構成要素を使用することで入力が可能となった．

第5.3表　読みの存在しない 2 字の入力

漢字	打鍵数	入力
杋	9	凡（およそ）
諗	8	臼（うす）

5.3.3　考察

この漢字入力方式では，同じ漢字を同じ読みで入力した場合に読みのみを用いる入力方式よりも最低でも 2 打鍵増加するものの，外国の固有名詞での比較において 46 % の漢字の入力に要する打鍵数を減少させることができた．

特に，同音の漢字が多数存在する漢字の入力においては，その漢字の簡単な一部品を入力することで，打鍵数が大幅に減少する顕著な効果が見られた．

減少十傑の漢字は，音読みは推測できたとしても，日本語の熟語で一般によく見るものでもなく，訓読みも無かったり直ちには分からなかったりするものが多い．これらの漢字は，この漢字自身ではなく，その一部部品に，よく親しんでいる漢字が入っており，これを利用することで大きく打鍵数を減少させることができたと解釈できる．

5.4　手書きパッドとの比較評価

ここでは Microsoft IME の「手書きパッド」と，主にその入力に要する時間を比較する．手書きパッドはキーボードではなくマウスを使用するが，想定している一般的なパーソナルコンピュータ環境において，手書きパッドは読み方の分からない漢字を入力するのに使用する手段として一般的に想起されるものであると考えられ，これとの比較するのは妥当でありかつ必要であると考えられる．

5.4.1　この評価の環境

比較に使用する文字はリスト 3 の以下の 16 字である．

リスト 3

痏，紌，皕，堹，皜，玅，壳，緐，彜，漸，叀，虷，隚，繿，耊，侲

いずれも JIS X 0213 の第 4 水準と JIS X 0212 補助漢字の双方にともに含まれる漢字で，かつ手書きパッドでの入力に対応している漢字である．

使用した手書きパッドは Microsoft Windows 10 上の Microsoft IME である．

評価実験では，各被検者にそれぞれの漢字入力システムの説明を行った後，必要に応じて援助を行いながら 4 文字から 6 文字の漢字を入力させ，ある程度慣れたところで，16 文字をそれぞれの方法で，合計 32 回入力してもらった．16 文字をそれぞれ 2 回，目にす

ることになるわけだが，16 文字のうち半数ずつをどちらかの入力法で先に出現するように調整し，出現順はランダムとした．また入力方法が不明であるとか，入力できないと被検者が判断した場合は放棄を宣言できることとする．

なお比較等のために通常のキーボードでのタイピング速度 (WPM: Word per minute) も測定する．

今回は 5 人の参加者を得た．

手書きパッドの対応範囲について補足

手書きパッドでの入力に対応している漢字の範囲について，公式な説明は見当たらないが，いくつかの漢字を抽出して試験をしてみたところ，JIS X 0208（第一，第二水準）と JIS X 0212 （補助漢字）に含まれているものにのみ対応しているようである．そのため JIS X 0213 で追加された第三，第四水準の漢字で JIS X 0212 に含まれていない漢字は手書きパッドで入力可能なものが見当たらなかった．JIS X 0212 の制定は 1990 年であるから，これ以降の漢字追加の動きに全く対応しておらず，30 年近く停滞していることになる．

なお，JIS X 0213 の漢字のうち第三水準と第四水準の漢字は合計で 3,695 字あるが，このうち JIS X 0212 補助漢字に含まれていないものは 952 字である．

また反対に補助漢字に含まれているが JIS X 0213 第三，第四水準に含まれていないものは 3,058 字存在する．

5.4.2　結果

実験で得られた，提案方式によって入力に要した「打鍵時間」と手書きパッドで入力に要した「手書き時間」，加えてそれらの入力時間の増減の集計を第5.4表に示す．また入力を放棄した字を第5.5表に記す．

第5.4表　提案手法と手書きパッドの比較

被験者	t1	t2	t3	t4	t5
平均打鍵時間 [s]	12.5	45.0	48.4	18.5	12.7
標準偏差（打鍵）[s]	5.63	46.1	91.1	11.4	17.3
変動係数	0.450	1.03	1.88	0.615	1.36
平均手書き時間 [s]	12.7	29.2	20.8	17.9	11.8
標準偏差（手書き）[s]	3.37	14.3	7.58	13.5	3.61
変動係数	0.266	0.490	0.364	0.750	0.304
入力時間減少 [字]	9	5	6	9	11
入力時間増加 [字]	6	9	9	6	4
スコア（減少 − 増加）	3	-4	-3	3	7
通常タイプ速度 [WPM]	45	22	15	26	30

第5.5表　入力を放棄した文字

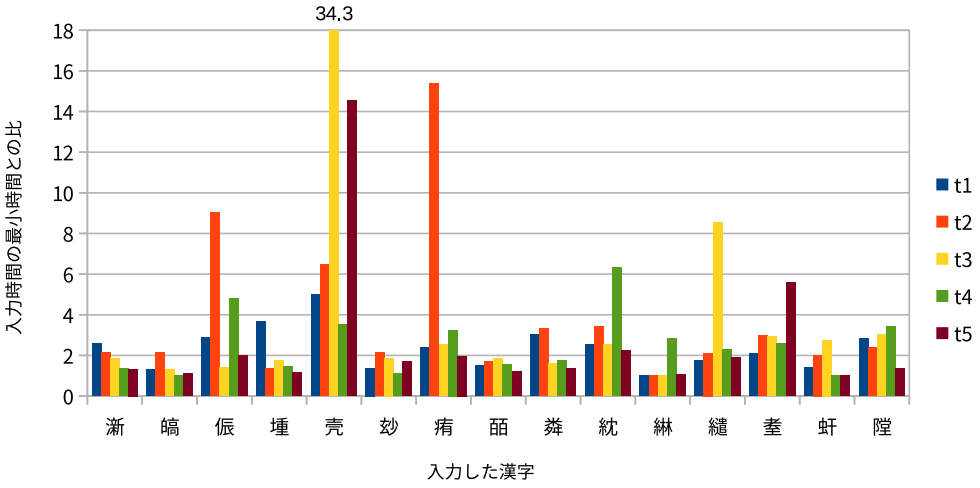
被験者	t1	t2	t3	t4	t5
提案手法	恵	恵	恵	恵	恵
手書きパッド	-	痕	-	-	-

打鍵時間と手書き時間の平均は被検者 t1 を除いて手書き時間の方が短くなっている．

ただし入力に要する時間のばらつき（標準偏差，変動係数）を見ると，全ての被検者で，提案手法の打鍵の方が手書きパッドよりも大きくなっている．

入力に要する時間が減少した文字の数から増加した文字の数を引いたスコアは被検者中 3 人が正になっている．

また各被検者が提案手法によって入力するのに要した時間の，それぞれの被検者の 15 字（16 字のうち恵は全員が放棄したので含まない）のうち時間が最小だったものに対する比を示したのが第5.2図である．



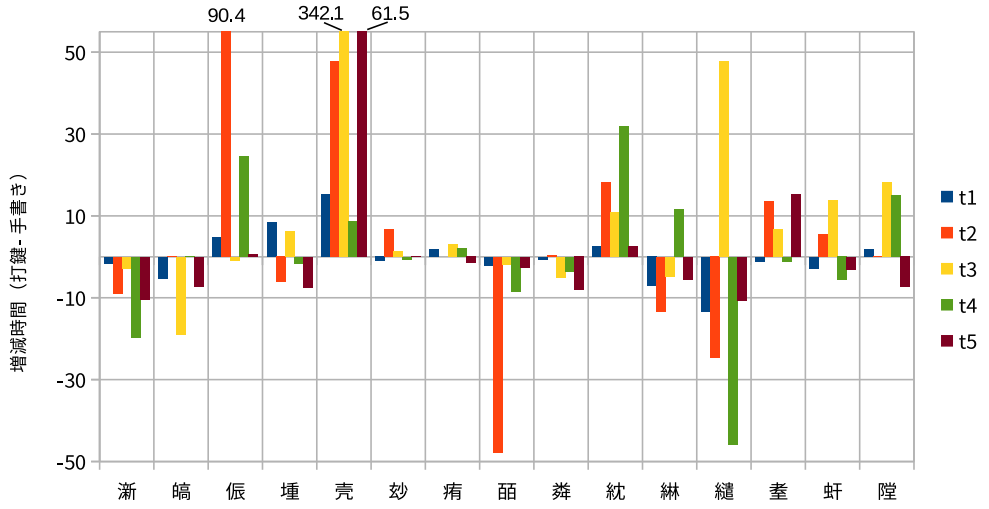
第5.2図　提案手法による入力時間の最小時間に対する比

被検者 t5 を例にすると，提案手法での入力では放棄した文字以外の 15 字のうち 13 字はいずれも 2.5 倍 (12 s) 以下で入力しているものの，売に 14.5 倍 (70.0 s)，耋に 5.6 倍 (20.1 s) を要している．

また漢字ごとに，打鍵時間と手書き時間の差を示したのが第5.3図のグラフである．

被検者からのコメントでは「知っている漢字が入っている場合は，提案手法の方が入力しやすい」，「キーボードからマウスへ手を移さなくて良い点は楽である」とあった．

なお，いずれの被検者も放棄している「恵」の字であるが，構成要素としては「厶，由」，派生漢字としては「惠」（恵の旧字体）がある．この字はリスト中でも特に普段見慣れないであろう字形であったために，いかに入力するかを発見できずに全員が入力を断念した．また片仮名の「ム」からの入力を試みた被検者もいたが，今回使用したデータでは漢字とそれに類似する片仮名を結び付けることは行っていなかったため，この方法も断念せざる



第5.3図　漢字ごとの打鍵時間と手書き時間の差

を得なかった．

5.4.3　考察

手書きパッドにおいては入力時間のばらつきが小さく，文字によってその入力に要する時間に大きな違いはないが，一方の提案手法においては，打鍵時間のばらつきが大きい．被検者 t5 は結果で言及したように，一部の文字だけ入力に時間がかかっていることが原因で平均打鍵時間が押し上げられている．そのために平均時間は手書きの方が小さいものの，スコアは正となっている．被検者 t4 も，手書き時間より打鍵時間の平均の方が大きいにもかかわらず，スコアは正になって，提案手法で入力時間が短くなったものの方が多い．

これは，入力（描画）に知識が不必要な手書きパッドに対して，提案手法ではまずどのように分解するか，どのような漢字を用いて入力するかということを考えるフェーズが必要だからであると考えられる．第5.2図に示されたように，相対的に最も時間を必要とした「売」は，どのような部分から入力すればよいかが一見しては摑みにくい．被検者 t4 にとっての「紘」も，その入力の際に，右側の要素をどのように入力するかについて考えることに時間を要していた．このような場合には事前知識が必要ない手書きパッドの方がスムーズに描画フェーズに入ることができ，入力時間も短くなっている．

とはいえ提案手法の方が常に劣位にあるわけではない．

漢字ごとに，どちらの入力方法が時間が短いかをまとめたのが第5.6表である．第5.3図にある内容から，被検者 4 人以上が，入力時間の差（打鍵時間 − 手書き時間）が減少したか，増加したかに偏った場合にそれらを傾向としてまとめた．

打鍵優位となった漢字は，雋は「隣」という字の部分文字であるし，その他の文字もその構成要素は漸の氵（さんずい）以外は全て常用漢字に入っている．氵は常用漢字ではないものの，最も親しまれている部首の一つであろう．他方で手書き優位の方は，その構成

第5.6表　入力時間の増減の傾向

入力時間の増減	漢字
減少（打鍵優位）	皕，皜，緲，斖，漸，繾
増加（手書き優位）	倀，売，痟，紘，隍

要素を他の文字で見たことがあってもそれ自体は常用漢字に入っていないものや，礻に比べれば一般的ではなかったり名前が分かりづらかったりする部首（阝（おおざと）と阝（こざとへん））が入っている．

この分類の傾向からも，直感的予想と一致するが，構成部品の入力方法が被検者にとって分かりやすいかそうでないかが，手法の優位性の差の一つの要因になると言える．

なおこの分解は習熟による時間減少が予想されるが，その検証は今後の課題とする．

さて，このように提案手法の時間的な優位性は文字によって差がある一方，コメントにもあったようにマウスに手を移さずに利用でき，キーボードだけで完結する点は提案手法の優位な点の一つであると思われる．今回の実験ではストレス度の測定などは行っていないが，この点はストレスの低減に大いに寄与するであろうと予想される．

なお平常時のタイピング速度と打鍵時間の平均は，被検者のタイピング速度に対して単調に増加という関係にはあったものの，それ以上の関連は今回の実験では見出せなかった．

5.5　本章のまとめ

漢字の構成要素などを格納するデータ構造を利用することで，漢字の合成や分解を利用できる漢字入力システムを開発した．

漢字文化圏の人名に使用できる漢字として収集した 202 字のうち半数近くの 92 字について，入力に必要な打鍵数が減少した．その大部分は漢字の構成部品の訓読みなどの同音の相対的に少ないものを経由して打鍵数の減少を実現させたものであった．また 2 字についてはそもそも日本語の読みが利用できないため，提案手法によりその部分文字を利用することで入力可能になった．

手書きパッドと比較した場合，構成要素がよく知っていたり簡単であったりする漢字であったり，知っている漢字の部分などである場合は，提案した入力方式が時間の面でも，手書きパッドに対して優位であった．一見して，構成要素が分かりにくい場合などは，事前知識の必要ない手書きパッドが時間的には優位であったが，コメントなどからも，キーボードのみで完結するという点などが，ストレスの低減などに繋がるのではないかと予想される．

第 6 章

結論

6.1 研究のまとめ

本研究では，使用できる漢字がますます増加しつつあるものの，それを入力可能な有力な手段がないことから，漢字がどのような部分から成っているかという漢字の構造を利用可能な入力方式の実現を目指した．

その前段階としてまず，この入力方式で利用できる構造データをコンピュータによって自動で解析するためのシステムを開発した．これは既存のデータベースが単一の文字コードで閉じていないことや手動で作成されているという問題を解決するためである．

この解析では漢字画像を入力として，三つの段階を経て，最終的な構造表現を得ることとした．

第一段階では Unicode の IDC に準拠した漢字のタイプを判別する．このために畳み込みニューラルネットワークによる分類器を作成した．このとき訓練データでの偏りを是正するために自動でランダムに部品を割り振って漢字の創作を行った．結果として創作漢字で僅かに正解率の向上が見られた．分類の正解率としては，  とのようなどちらでも表現が可能な文字のタイプで正解率が低かったために，と，とを同一視する縮退を施したところ，平均タイプ別正解率 98.2 % を達成した．

第二段階では，漢字のタイプに基いて切断位置の特定を行う．このために，走査線に沿って単純に黒い画素の多寡で決定すると，部品の重なりや，黒の量は少ないもののふつうは分割しない直線をまたぐようになるという問題があった．そのために走査線に垂直な方向へ黒画素がいくら続いているのかという直交延長という概念を導入し，これの最小極小値において分割する方法を採った．これによって正解率は 59.2% から 74.1 % に向上した．

第三段階では，分割した部分文字の同定を行う．漢字が線の集まりであることから線の離間の度合いを考慮する最近傍距離による類似度測定手法 (NND) を導入し，二値画像でよく使用される類似度関数である YULE の手法と比較した．全般的に NND の方が優位であったが，特に「颱」における「風」のように部首化する際に変形する部分文字についての正解率（テンプレートを制限）は，YULE の 20 % に対して NND は 100 % を達成するなど優位性が顕著に現れた．

これら三つのプログラムを総合して，最終的な決定は人間が行う Computer-Aided なアプリケーションを作成した．完全自動化には至っていないものの，対象の部分文字が Unicode に入っているかどうかなどの知識は必要ないシステムの完成を見た．

そして最終的な目的である漢字入力方式であるが，読みによる入力が現在大勢を占めていることに鑑み，読みによる入力法の上で補助的に使用できるようなプログラムを開発した．

外国人の漢字人名によく使用される漢字で打鍵数をみたところ，比較対象の半数近くの漢字で，目的の漢字の一部分を使用することで打鍵数を減少させることができた．

また読みの分からない漢字を入力する際によく使用される手書きパッドとの時間などの比較も行った．提案した入力方式では目的の漢字の分解の仕方や，各部分文字の入力方法（読みやどの漢字に含まれているか）を考えるという準備が必要なため，難しい漢字である場合にはここで時間を要したり，入力方法が分からなかったりする．そのような場合には事前知識の必要無い手書きパッドが優位になるものの，一方で構成要素がよく親しんだ漢字であれば，通常時のタイピング速度が低速な被検者 (15 WPM) であっても，提案する入力方式の方が時間的にも優位であった．

主に二つの部分から成る一連の研究の成果により，ますます増えつつある漢字をコンピュータで入力するという需要に対する一つの有用な手法を実現することができた．

6.2　今後の課題

解決すべき課題は山積している．

第 4 章の構造解析の第二段階に当たる漢字分割位置の決定において，本研究では走査方法を工夫したとはいえ，全て走査線は縦か横の直線あるいはそのごく簡単な長方形の組合せである．漢字分割の精度を向上させるには，必ずしも直線に限らず，曲線などの走査線による分割も考慮すべきである．

また同じく構造解析の第三段階に当たる文字同定については十分な精度が得られたとは言い難い．今回は画像同士の比較による類似度の評価をもとにして同定を行ったが，機械学習的手法などについても考慮を行い，また疎密探索などの画像認識におけるテクニックも実装すべきである．

加えて，今回第 4 章の解析においては実装の関係上，一貫してラスタ形式の PBM ファイルを使用したが，もととなるフォントファイルはアウトラインのベクタデータであり，これの性質をうまく利用することも考慮すべきである．

漢字入力法においては，異体字をどう扱うかという問題を今回は放棄したが，漢字入力に要請される需要の一つには漢字によっては多数存在する異体字を入力し分けたいというものがある（Unicode には異体字セレクタも存在する）．出来るだけ多くの漢字を簡単に入力できるようにするには，この課題についても解決しなければならない．画像認識関係の手法を本研究ではマッチングに使用したが，異体字選定においては，画像認識技術をどこの部分が異なるかということの解析に使用することなどがアイディアとしては考えら

れる．

なお一口に異体字といってもそれは様々なものを含むため，これらは例であるが，旧字体に対する新字体，繁体字に対する簡化字といった古い文字を簡略化した簡略異体字，部品の配置が変わっただけの配置異体字，一部の構成要素のみの表現が異なる小異異体字，複数の文字が一つになってしまった縮退異体字などタイプごとに分けて適切な処理を施すこととする必要があろう．異体字を提案した漢字入力方式で扱う場合には，同一とみなす漢字用のフィールドと同様なフィールドを追加すれば対応できる．

提案した入力方式自体についても，より被検者を増やして，訓練による習熟の度合いやマウスを使用する入力方法と比べてのストレスの度合いなどを計測することが求められる．

またユーザのタイピング速度との関連だけでなく，ユーザの漢字に関する知識（定量化は難しいが）との関連も明らかになれば，手法の訴求対象を明確にすることができる．

附録 A

外国人名漢字リスト

漢字入力法の評価において使用した，外国人名に使用される漢字のリストの作成方法とその一覧をここに示す．

A.1　作成方法

日本以外の漢字使用圏の人名で使用される漢字を収集するために，Wikipedia 日本語版の各国の有名人一覧を使用した．使用した記事とその版は以下の通りである．

- 「台湾の人物一覧」（2017 年 11 月 12 日 (日) 01:05 UTC; Tze Chiang Hao による版）https://ja.wikipedia.org/w/index.php?title=%E5%8F%B0%E6%B9%BE%E3%81%AE%E4%BA%BA%E7%89%A9%E4%B8%80%E8%A6%A7&oldid=66267161
- 「香港人の一覧」（2017 年 11 月 23 日 (木) 18:20 UTC; 218.252.68.12 による版）https://ja.wikipedia.org/w/index.php?title=%E9%A6%99%E6%B8%AF%E4%BA%BA%E3%81%AE%E4%B8%80%E8%A6%A7&oldid=66395209
- 「韓国の著名人一覧」（2018 年 1 月 6 日 (土) 04:19 UTC; NWG による版）https://ja.wikipedia.org/w/index.php?title=%E9%9F%93%E5%9B%BD%E3%81%AE%E8%91%97%E5%90%8D%E4%BA%BA%E4%B8%80%E8%A6%A7&oldid=66876888
- 「朝鮮民主主義人民共和国の著名人一覧」（2017 年 6 月 15 日 (木) 00:41 UTC; 2002:90d9:a7f0::1 による版）https://ja.wikipedia.org/w/index.php?title=%E6%9C%9D%E9%AE%AE%E6%B0%91%E4%B8%BB%E4%B8%BB%E7%BE%A9%E4%BA%BA%E6%B0%91%E5%85%B1%E5%92%8C%E5%9B%BD%E3%81%AE%E8%91%97%E5%90%8D%E4%BA%BA%E4%B8%80%E8%A6%A7&oldid=64450716

なお中華人民共和国の人物一覧に類するものは Wikipedia 日本語版，中国語版ともに存在せず，英語版のこれに類する項目は下部項目に細分化されており，また一覧記事には漢字を表記していないものが多かったため，今回は使用していない．

以上の人物一覧から人名に使用されている漢字を抽出し（一部はリンク先記事も適宜参照），ここから日本の常用漢字と人名用漢字（ただし表一のみ，常用漢字の異体字はここ

では人名用漢字には含まないこととする）にも含まれているものを除いた．

A.2　漢字リスト

以下が前節の手法により作成した 202 字のリストである．Unicode のコードポイントを付して記す．

なお本文でも指摘したように，このうち杌 (U+674B) と誨 (U+8ABD) には Unihan Database において日本語の読みが設定されておらず，日本の小型漢字辞典「漢字源」にも収録されていない．

于	U+4E8E	婷	U+5A77	曼	U+66FC
亞	U+4E9E	媚	U+5A9A	曾	U+66FE
佩	U+4F69	嫻	U+5AFB	杌	U+674B
佰	U+4F70	嬌	U+5B0C	杞	U+675E
倩	U+5029	寶	U+5BF6	杰	U+6770
傅	U+5085	尹	U+5C39	柯	U+67EF
傳	U+50B3	屏	U+5C4F	栩	U+6829
兒	U+5152	岱	U+5CB1	桓	U+6853
兪	U+516A	崔	U+5D14	槿	U+69FF
冰	U+51B0	嶽	U+5D34	樂	U+6A02
勛	U+52DB	嶽	U+5DBD	樵	U+6A35
勳	U+52F3	庾	U+5EBE	歐	U+6B50
厲	U+53B2	廖	U+5ED6	殷	U+6BB7
哈	U+54C8	廣	U+5EE3	汪	U+6C6A
喆	U+5586	彤	U+5F64	汾	U+6C7E
國	U+570B	彭	U+5F6D	沂	U+6C82
坤	U+5764	怡	U+6021	泓	U+6CD3
堃	U+5803	惠	U+60E0	泗	U+6CD7
增	U+589E	愷	U+6137	洙	U+6D19
壘	U+58D8	慷	U+6177	涂	U+6D82
壽	U+58FD	應	U+61C9	涵	U+6DB5
奕	U+5955	懷	U+61F7	淦	U+6DE6
妍	U+598D	扁	U+6241	渝	U+6E1D
姍	U+59CD	敖	U+6556	游	U+6E38
姚	U+59DA	斌	U+658C	潘	U+6F58
姜	U+59DC	旻	U+65FB	澎	U+6F8E
娟	U+5A1F	昱	U+6631	濤	U+6FE4
娣	U+5A23	曉	U+66C9	炅	U+7085

炘 U+7098
炫 U+70AB
炳 U+70B3
焜 U+711C
煇 U+7147
煊 U+714A
熙 U+7155
煜 U+715C
煥 U+7165
燁 U+71C1
燮 U+71EE
玟 U+739F
玥 U+73A5
玹 U+73B9
珉 U+73C9
琴 U+73E1
琛 U+741B
琦 U+7426
琪 U+742A
琬 U+742C
瑄 U+7444
瑋 U+744B
瑜 U+745C
瑤 U+7464
瑾 U+747E
璋 U+748B
璐 U+7490
璜 U+749C
瑽 U+74A8
璿 U+74BF
瓊 U+74CA
甄 U+7504
甯 U+752F
發 U+767C
盈 U+76C8
盧 U+76E7
眞 U+771E
睿 U+777F
瞿 U+77BF
祚 U+795A
祺 U+797A
禧 U+79A7
禹 U+79B9
秉 U+79C9
穎 U+7A4E
竇 U+7AC7
簇 U+7C07
絲 U+7D72
經 U+7D93
繆 U+7E46
羿 U+7FBF
翊 U+7FCA
翰 U+7FF0
耿 U+803F
聆 U+8046
聲 U+8072
臺 U+81FA
舒 U+8212
舫 U+822B
艾 U+827E
芬 U+82AC
芮 U+82AE
范 U+8303
茵 U+8335
茹 U+8339
荊 U+834A
莖 U+8396
菁 U+83C1
菘 U+83D8
菲 U+83F2
萬 U+842C
蒋 U+848B
蔚 U+851A
蔡 U+8521
蕙 U+8559
蕭 U+856D
薇 U+8587
袁 U+8881
覲 U+89B2
詹 U+8A79
誽 U+8ABD
譚 U+8B5A
赫 U+8D6B
趙 U+8D99
踐 U+8E10
辜 U+8F9C
逵 U+9035
邱 U+90B1
郝 U+90DD
鄒 U+9112
鄧 U+9127
鈕 U+9215
鈞 U+921E
鈺 U+923A
鉉 U+9249
錡 U+9321
鍾 U+937E
鎬 U+93AC
鎰 U+93B0
鏞 U+93DE
鑾 U+947E
閔 U+9594
閻 U+95BB
關 U+95DC
阮 U+962E
陞 U+965E
霄 U+9704
霆 U+9706
霽 U+9749
韋 U+97CB
韶 U+97F6

馥	U+99A5	驊	U+9A4A	龍	U+9F8D
馮	U+99AE	魏	U+9B4F		
騨	U+9A28	鹸	U+9EFB		

謝辞

まずは指導教員である相田仁教授に感謝申し上げます．振り返ると，卒論配属から数えて三年間という大学生活六年の半分もの間，相田研究室に席を与えられることとなりました．修士研究は，私の個人的趣味に大分寄ったテーマになってしまったにもかかわらず，日々様々な視点からご指導，ご助言いただきました．その他研究室生活，学生生活が充実したものになったのもひとえに先生の姿勢，人柄に依る所が大きいと感じています．本当にありがとうございました．

秘書の元岡みさ子氏にも研究室の日常生活において大変お世話になりました．同窓会企画は良い思い出です．研究室の生活が楽しく快適なものになったのは元岡さんのおかげだと思います．技術専門職員の千葉新吾氏には備品管理やコンピュータ管理などの日常生活の細かな点で色々と助けられました．元助教の古宇田フミ子氏にはしばしば研究手法についての有益なアドバイスをいただきました．本論文中にもそのアイディアが活かされています．

皆様ありがとうございました．

学部 4 年生の頃から研究室生活の多くを共にしてきた先輩である佐藤惟知氏にもこの場を借りて謝意を表したいと思います．修士になってからは同期がいない中，様々な面で良き相談相手になってくださいました．またこの一年間，色々な仕事を押しつけがちになった修士 1 年の長田知明君と山内智晴君をはじめ，その他研究室内外の先輩，同期，後輩の皆さんにも感謝したいと思います．

ありがとうございました．

最後に，時にくじけそうになった私を励まし支え，また頼ってくれた両親と妹に感謝します．ありがとうございました．

平成 30 年 2 月 1 日

金子隆佐

参考文献

[1] Richard G. Casey and Eric Lecolinet, A survey of methods and strategies in character segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 18, 1996, 690-706, July, DOI: http://dx.doi.org/10.1109/34.506792.

[2] Seung seok Choi, Sung hyuk Cha, and Charles C. Tappert, A survey of Binary similarity and distance measures, Journal of Systemics, Cybernetics and Informatics, 43–48, URL: http://www.iiisci.org/journal/CV$/sci/pdfs/GS315JG.pdf.

[3] Yannis Haralambous, Fonts & Encodings, P. Scott Horne 訳, O'Reilly Media, Inc., 2007.

[4] Ken Lunde, CJKV Information Processing, O'Reilly Media, Inc., 2nd edition, 2009.

[5] Yanping Lv, Feipeng Cai, Dazhen Lin, and Donglin Cao, Chinese character CAPTCHA recognition based on convolution neural network, in 2016 IEEE Congress on Evolutionary Computation (CEC) 4854-4859, July, 2016, DOI: http://dx.doi.org/10.1109/CEC.2016.7744412.

[6] Tomohiko Morioka, Multiple-policy Character Annotation based on CHISE, Journal of the Japanese Association for Digital Humanities, 1, 2015, 86-106, DOI: http://dx.doi.org/10.17928/jjadh.1.1_86.

[7] Insup Taylor, Martin M. Taylor, and Maurice Martin Taylor, Writing and Literacy in Chinese, Korean and Japanese, Studies in written language and literacy, John Benjamins Publishing Company, 1995.

[8] J.D. Tubbs, A note on binary template matching, Pattern Recognition, 22, 1989, 359–365, DOI: http://dx.doi.org/10.1016/0031-3203(89)90045-9.

[9] T. Y. Zhang and C. Y. Suen, A Fast Parallel Algorithm for Thinning Digital Patterns, Communications of the ACM, 27, 1984, 236–239, March, DOI: http://dx.doi.org/10.1145/357994.358023.

[10] 内田誠一・石川博「特徴照合」,『知識の森『パターン認識とビジョン』(2 群 2 編)』, 電子情報通信学会, 9–19 頁, 2009 年, URL：http://www.ieice-hbkb.org/portal/

doc_590.html.

[11] 上地宏一「漢字グリフ管理 Wiki システム (GlyphWiki) の構築」,『じんもんこん 2007 論文集』, 第 2017 巻, 2007 年, 237–244 頁, 12 月, URL：http://ci.nii.ac.jp/naid/170000083097/.

[12] 金文京『漢文と東アジア：訓読の文化圏』, 岩波新書, 岩波書店, 2010 年.

[13] 齋藤希史『漢字世界の地平：私たちにとって文字とは何か』, 新潮選書, 新潮社, 2014 年.

[14] 田中哲朗「部品合成による漢字スケルトンフォントの作成」, 博士論文, 東京大学大学院工学系研究科, 1992 年, URL：http://hdl.handle.net/2261/54326.

[15] 田村毅, 2001 年「日本学術振興会未来開拓学術研究推進事業研究成果報告書：マルチメディア通信システムにおける多国語処理の研究」.

[16] 辻田正雄「「通用規範漢字表」について」,『文学部論集』, 第 100 巻, 2016 年, 27–42 頁, 3 月, URL：http://archives.bukkyo-u.ac.jp/repository/baker/rid_BO0100000008079.

[17] 藤堂明保「中国の文字とことば」, 藤堂明保・松本昭・竹田晃・加納善光（編）『漢字源』, 学研教育出版, 改訂第五版, 1881–1896 頁, 2011 年.

[18] 平本健二「電子行政における文字環境の整備」,『情報管理』, 第 57 巻, 2015 年, 799-808 頁, DOI: http://dx.doi.org/10.1241/johokanri.57.799.

[19] 水野義明「朝鮮語における漢語の読み方について」,『明治大学教養論集』, 第 61 巻, 1970 年, 1–17 頁, URL：http://hdl.handle.net/10291/8780.

[20] 守岡知彦「ポスト文字コード時代の文書処理技術に関する展望」,『全国文献・情報センター人文社会科学学術セミナーシリーズ』, URL：http://www.chise.org/papers/dc2002.pdf.

[21] 矢野啓介『プログラマのための文字コード技術入門』, WEB+DB PRESS plus, 技術評論社, 2010 年.

[22] 山田尚勇『コンピュータ科学者がみた日本語の表記と入力 2：文字入力とテクノロジー』, くろしお出版, 2014 年.

発表文献

[1] 金子隆佐・相田仁「漢字構造の自動判別とそれを用いた漢字入力方式について」,『情報処理学会第 80 回全国大会講演論文集』, 2018 年, 3 月（予定）.