

岐 阜 大 学 博 士 論 文

事例ベースに基づく
フォントの個性化に関する研究

学位論文：博士(工学)甲90号

平成10年1月

安本 護

Building Fonts with Individual Features Based on a Handwriting Character Database

by

Mamoru YASUMOTO

Abstract

This paper describes two kinds of methods for building a font resembling the user's handwriting style.

It consists of 5 chapters. Chapter 1 describes the background and an overview of this study.

Chapter 2 describes an outline of the font technology and a survey of former studies in font generation. These studies are divided into three groups according to their goals:

1. Typical typefaces like Mincho or Gothic
2. Script types like Brush or Pen
3. Font which resemble the user's individual handwriting style

This research belongs to the latter group.

Chapter 3 describes a method for generating a handwriting style font by assembling the radical parts. It is very hard for an ordinary user to input 1,945 Jouyou-Kanji which were selected to be sufficient for daily use of Japanese. For the purpose of cutting down the amount of input, our proposed method extracts the radical parts from a small number of characters written by a user. Using this kind of example based method, all character

elements needed to generate Jouyou-Kanji can be extracted from 388 characters. Thus, using only 19.9% of this standard set, high quality output characters that retain the user's individual feature can be created.

Chapter 4 describes another method for generating handwriting style fonts. This method is based on the global and local features in an Individual's handwriting. We use three kinds of parameters to generate a font. The first represents unique global aspects. The next parameter represents local features. The last parameter controls the degree of global and local uniqueness respectively. User's individual features are extracted from a few characters written by the user. In our model, global features are expressed by a nonlinear deformation of each base pattern in the handwriting database, and local features are expressed by a linear combination of the deformation. Experiments show that our proposed method can generate various styles of fonts according to each user's individual handwriting style.

Chapter 5 summarizes the study and discuss on the future work.

目次

第1章	序論	1
1.1	本研究の背景および目的	2
1.2	本論文の構成と概要	4
第2章	フォントとその生成法の現状	7
2.1	緒言	8
2.2	フォントの概要	9
2.2.1	印刷技術の変遷	9
2.2.2	フォント方式	12
2.2.3	フォントの制作工程	15
2.2.4	フォント制作の問題	19
2.3	フォント生成に関する従来の研究	21
2.3.1	印刷用書体生成に関する研究	21
2.3.2	手書き風書体生成に関する研究	22
2.3.3	筆者の個人性を反映した手書き文字生成に関する研究	23
2.4	結言	24
第3章	部分字形組み合わせを用いた手書き風フォント生成	27
3.1	緒言	28
3.2	システムの概要	28
3.3	字形データベースの構築	29
3.3.1	漢字の構造	29
3.3.2	部分字形	30
3.3.3	構成パターン	30
3.3.4	出現位置属性	32
3.3.5	字形データベースの構築結果	32
3.4	フォント生成	34

3.4.1	文字データの表現	34
3.4.2	生成規則	34
3.4.3	基本部分字形の抽出	35
3.4.4	部分字形の配置	36
3.4.5	仮名文字の扱い	38
3.5	生成実験	38
3.5.1	手書き文字データの収集	38
3.5.2	出現位置属性の効果	38
3.5.3	個性化	39
3.6	結言	40
第 4 章	大域的個人性と局所的個人性に基づく手書き風フォント生成	43
4.1	緒言	44
4.2	システムの概要	44
4.3	手書き文字パターン	45
4.3.1	文字データの入力	45
4.3.2	文字間距離	46
4.4	個人性を表すパラメータ	47
4.4.1	大域的個人性パラメータ	47
4.4.2	局所的個人性パラメータ	50
4.4.3	大域的個人性および局所的個人性を抑制するパラメータ	50
4.5	生成実験	52
4.5.1	生成パラメータの決定	53
4.5.2	実験内容	53
4.5.3	実験結果と評価	54
4.6	結言	64
第 5 章	結論	67
5.1	研究結果の概要	68
5.2	今後の課題	69
	謝辞	71
	参考文献	73
	研究業績	79

第 1 章

序論

1.1 本研究の背景および目的

1873年にタイプライタが商品化されて以来、欧米ではタイプライタによる文書作成が一般的であった。ところが、日本では長年に渡り手書きによる文書作成が行われていた。これは、数千もの漢字を扱わなければならない日本語の宿命であった。このため明治時代以降、非効率な漢字を全廃し、日本語の表記をカナ文字やローマ字に統一しようという運動まで起こっている。このような日本語処理の効率化を目的として、カナタイプ（1923年）や漢字を含む約3,000字を備えた邦文タイプライタ（1915年）が発明された。その後、邦文タイプライタは、企業や学校などに普及していったが、数千の文字から目的とする文字を選び出す操作には、特別な訓練を受けることが必要であった。この問題を解決し、一般のユーザにも日本語文書の能率的な作成を可能としたのは、1978年に発表された日本語ワードプロセッサとそこに採用されたかな漢字変換日本語入力方式である。[1]

しかし、当時の日本語ワードプロセッサの印刷品質は邦文タイプライタや一般の印刷物に比べ著しく見劣りするものであった。これは、縦横24個の点から構成された24ドットフォントを印刷に使用していたためであった。

このように低解像度のフォントを使用したのには、以下の理由があった。

- 高解像度の普通紙プリンタが無かった。
- 高解像度データを扱えるマイクロプロセッサがなかった。
- 高解像度データの保持に必要な大容量メモリが非常に高価であった。
- 高品位なフォントデザインは、印刷業界が独占しており、電子機器メーカーは入手が困難であった。

その後、プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、更に電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパーソナルコンピュータ用のプリンタで邦文タイプライタに匹敵する美しいフォントを用いた印刷が可能となっていった。この結果、邦文タイプライタはその使命を完全に終えることとなった。

表現力豊かな文書の作成を実現するには、多様な書体を使い分けることが有効である。また、さまざまなシステムにおいて高品質の印刷結果を得るには、システムの特性に合わせてフォントを開発する必要がある。

1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザビームプリンタは、アウトラインフォント技術に基づく高品位フォントを標準で備え、高品質な印刷を可能とした。しかし、フォントデータのフォーマットが非公開であったため、利用者がフォントを追加することは困難であった。

ところが、GUI（グラフィカル・ユーザ・インタフェース）ベースのパーソナルコンピュータ用OSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サード・パーティーのフォントベンダーが独自に開発したフォントも流通するようになった。この結果、今では美しく多様なフォントデザインの中から好みのフォントを選択し、使用できる環境が整っている。[2]

しかしながら、従来のフォントは、大量に生産され、不特定のユーザが利用するものである。すなわち、特定の個人向けにデザインされるものではなく、画一的であり没個性である。一方、今後の情報社会の成熟、文書作成環境の変化を考えると、個人向けにデザインされた個性的なフォントに対する要求も高まってくるものと予測される。ワードプロセッサを使用して年賀状を作成する人の多くが、毛筆体やペン字体といった手書き風フォントを使っているのも、自分だけの個性的な書体を使いたいという要求を示しているといえよう。手紙の類に限らず、各種のレポートやさまざまな個人的文書あるいは文学作品等がその作者の個人性を反映した固有の書体で印刷されるとすれば、より豊かな文書表現の実現に寄与することになる。

また、インターネットの発展によって個人でもWWW（World Wide Web）や電子メールを利用して、広く情報発信することが可能となっている。画面上のフォントから発信者の個性を感じられるようにすれば、いっそう楽しく親近感にあふれたコミュニケーションの促進に寄与できると考えられる。

ユーザの個人性を反映した固有のフォントを作成する素朴な方法の一つとして、ユーザの筆跡そのものをデータ化することが考えられる。しかし、日本語文書の作成には数千もの文字種が必要であり、これを筆記することはユーザにとって大変な負担となる。

また、ユーザの個人性を反映した手書き文字パターンを自動的に生成しようとする研究も行われているが、生成対象文字カテゴリーすべての筆記を要したり、複雑なルールを導入して個人性を記述しなければならないものであった。すなわち、ユーザの個人性を反映した自分専用のフォントを容易に作成する方法は、まだ実現されていないのである。

本論文では、従来のような大量の文字筆記や複雑なルール作成を必要としない事例ベースに基づいた手書き風フォント生成方法を提案する。

1.2 本論文の構成と概要

本論文は、次の5つの章から構成されている。

第1章は、本章であり、本研究の背景と目的および概要を述べている。

第2章では、本研究に関連の深いフォント技術を概観し、フォント生成に関する従来の研究について述べる。

フォント技術は、印刷技術の変遷にしたがって、活字や写植文字盤といったアナログフォントから電子的に文字形状を記憶するデジタルフォントへと変わってきた。当初は、16～24ドット程度と低解像度であったデジタルフォントも、アウトラインフォントに代表されるベクトルフォントの開発によって従来のアナログフォントを超える表現力を得るまでに至った。

従来のフォント自動生成に関する研究は、ほとんどがフォント提供者に効率的なフォント作成手段を与えることを目的とし、エンドユーザに個人専用のフォントを提供しようという研究は、まだ少ない。

この他、フォント作成を目的とするものではないが、手書き文字認識辞書の効率的構築や識別性能の評価を目的とする変形文字パターン生成や筆者認識において個人性を表す多変量解析パラメータと文字形状との関連を視覚的に表現するための文字パターン生成に関する研究がある。

第3章では、漢字の構造的特徴を利用して、個性的フォントを生成する手法を提案する。

漢字の成立には、象形、指事、形声、会意、転注、仮借の6通りが知られている。この中で、意味を表す要素と音を表す要素を組み合わせる「形声」と複数の漢字を組み合わせる「会意」の占める割合は非常に大きい。更に、他の漢字も誕生から長い年月を経る間に字体の整理が行われ、多くが規則的なパターンの組み合わせでできている。本手法は、このような漢字の構造的特徴に着目し、ヘンやツクリなどの基本部分字形を生成規則に記述した大きさおよび位置に配置し、フォントを生成する。この手法では、フォント生成結果に個人性を現すために手書き文字に含まれる個人性を解析するのではなく、事例ベースすなわちユーザの筆記した基本部分字形そのものをを用いる。ここで課題となるのは、基本部分字形を収集する際にユーザの筆記労力をできるだけ軽くしつつ、品質の高い生成結果を得ることである。そこで、できるだけ少ない漢字から必要な基本部分字形を抽出するアルゴリズムを提案した。常用漢字1,945文字を生成対象とする場合、全体の19.9%に相当する388文字から必要な基本部分字形を抽出可能であった。また、文字の生成品質を向上させるた

めに部分字形が漢字の中に出現する位置を分析し、これを新たな属性として付加することを行った。生成実験の結果、同一の生成規則を用いても基本部分字形を取り替えれば、出力結果は大きく変化し、個性的な手書き風文字を生成することができることが分かった。

第4章では、第3章の手法より更に少量の手書き文字入力から個性的フォントを生成可能な手法を提案する。

この手法は、事例ベースすなわち複数の人の筆記による手書き文字データベースを基準パターンとし、非線形の幾何学的変形を加えた後、線形結合して手書き風文字フォントを生成するものである。フォントの生成結果に現れる個人性は、次の3種類のパラメータを使用して制御する。第1のパラメータは、文字枠正方形を任意の四辺形に変形する非線形の幾何学的変換係数であり、文字の大きさ、傾き、縦横比などの概略形状を決定づける大域的個人性を表すものである。第2のパラメータは、局所的個人性に対応し、システムに蓄積されたさまざまな個性をもつ手書き文字データ、すなわち事例ベースを線形結合してフォントを生成する際の重み係数である。これら2つの個人性パラメータは、ユーザの入力した手書き文字とフォント生成結果との距離が最小となることを条件に決定する。最後のパラメータは、前記2つの個人性を個別に調整するパラメータである。このパラメータの導入により、ユーザによる個人性制御も可能とした。文字間距離に基づく客観的評価と19名の被験者による主観的評価の結果、ユーザの個人性を反映した手書き風文字が生成できることが確認できた。

第5章では、本論文の結論と今後の課題について述べる。

第2章

フォントとその生成法の現状

2.1 緒言

本研究のテーマは、ユーザの個性を反映したユーザ独自のフォント、すなわちユーザの筆跡に類似した手書き風フォントを自動生成することである。本章では、本研究で扱うフォントとは、どのようなものであるかについて概説した後、フォント生成に関する従来の研究について述べる。

計算機技術の発展により、従来は数値・記号処理を主としていた計算機の応用は、動画、図形、音声、文字などのさまざまなデータを統合し、インタラクティブな操作を可能とするマルチメディア処理へと移ってきた。文字は、他のメディアに比べ、地味で目立たない存在ではあるが、もっとも基本的な必須の情報伝達手段である。その重要性は、他のメディアに劣るものではない。

計算機処理で用いるデジタル化された文字パターンデータは、フォントと呼ばれている。これは、印刷技術のデジタル化に由来するものである。

印刷技術の変遷にともなって、活字や写植文字盤などのアナログフォント方式は、ほとんど使われなくなり、現在では、ビットマップフォントやアウトラインフォントなどのデジタルフォントが主流となっている。

フォント制作には、特別な技能を持つ専門家が必要である。しかも、文字数の多い日本語フォントは、開発に膨大な労力を要する。このため、従来のフォント生成に関する研究はフォント制作の効率化を目指すもので、生成対象のフォントは、明朝体やゴシック体のようにデザイン手法の確立された印刷用書体が主流であった。楷書体などの手書き風フォントを生成対象とする研究もあるが、筆記者の個性を生成結果に反映しようとするものは、わずかである。

このほか、文字認識や筆者認識の研究にも筆記者の個人性を文字パターン生成結果に反映させようとする試みがある。しかし、生成対象とする文字すべての入力が必要としたり、生成ルールの作成に特別な知識を必要とするものであり、本研究の目的を満たすものではない。

以下、2.2節ではフォント処理技術に関する概説、2.3節ではフォント生成に関する従来の研究について述べる。2.4節は、本章で得られた結論である。

2.2 フォントの概要

フォントとは、元来、「同じ大きさ、同じ書体の活字一揃い」を意味する印刷用語であった。やがて、活字を用いる活版印刷は写真技術の応用による写真植字（phototypesetting）に取って代わられ、現在では、電子的に記憶した字形データを用いる電算写植（computer typesetting）が主流となっている。

また、パーソナルコンピュータや高精細プリンタの急速な普及を背景に、ワードプロセッサによる文書作成が一般化し、商業印刷並みの出力品質を実現する DTP（Desk Top Publishing）が個人でも利用可能になっている。このような文書作成・印刷のデジタル化がエンドユーザにまで及んだ結果、フォントという用語は、本来の意味から離れ、デジタルデータ化された印刷用書体すなわちデジタルフォントという意味で広く使われるようになった。[3],[11]

計算機において、自由にフォントを扱えるようになった結果、テレビ番組の制作[13],[14]や、テレビゲーム[15]、通信カラオケ[12]等の新しいメディアにおいても高品質なフォントの利用が可能になっている。

本節では、印刷技術の変遷と、それに伴いアナログ処理からデジタル処理へと変わっていったフォント方式を概観し、現在の主流であるデジタルフォントの制作法について述べる。

2.2.1 印刷技術の変遷

印刷技術の発達は、従来特権階級だけが所持することを許されていた書物を広く一般大衆へ広めることを可能とし、文化の発展に大きく寄与してきた。印刷には大量・高速処理と多彩で高品質な出力が要求され、これを実現するために、さまざまな印刷機器の開発が行われた。また、印刷技術変遷の歴史は、フォント技術の歴史そのものでもある。印刷機器の特性やユーザからの要求に応えるため、フォント制作技術やデザイン技法が確立されていったのである。[4],[6],[7],[8],[10]

（1） 活版印刷

法隆寺に伝わる百万塔陀羅尼經（770 年）は、現存する最古の印刷物とされている。このような古代の印刷は、1 枚の版ごとに文章を彫り付けるものであった。このため、その生産性は高いものではなかった。ところが、1 文字ずつがばらばらになった活字を組み合

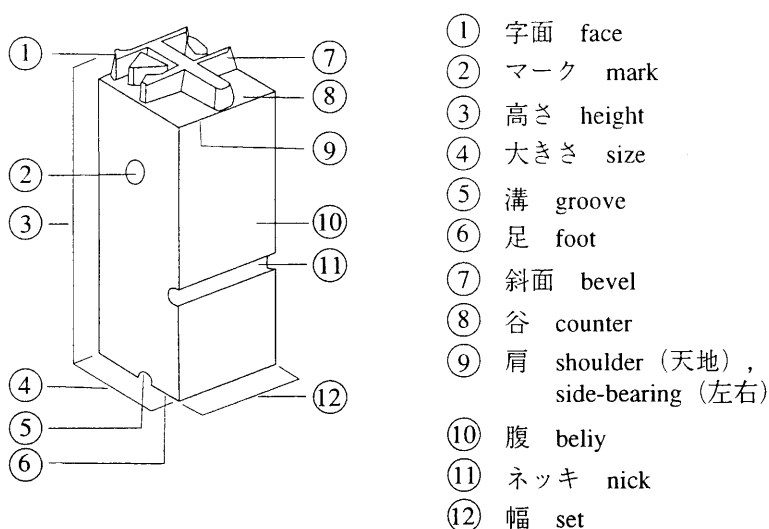


図 2.1 活字

Fig. 2.1 Typeface.

わせて任意の文章を印刷する活版印刷技術が発明され、印刷の生産性は飛躍的に高まった。

活字とは、凸形の文字面を持った版（図 2.1）で、ごく初期には木製の活字も用いられたようであるが、耐久性や必要な本数を複製する必要から、実用的な活字は鉛合金を素材に鋳造で製作された。この文字面の鋳型を母型と呼んでいる。母型にはその製法により、活字と同一サイズの文字を手彫りした木片をメッキ加工して作る電胎母型、紙にデザインした原字をもとに縮小彫刻して作る彫刻母型、活字と同じように文字面が凸になったパンチ父型を金属材料に打ち込んで作るパンチ母型の 3 種類がある。

活版印刷では、凸型の活字に圧力を加え紙にインクを転写させるため、文字線が太くなる特性がある。このため、活字母型は、あらかじめ線を細くデザインしてあった。

金属活字を用いる近代的な活版印刷技術は、ドイツ人のグーテンベルク（Johannes Gutenberg, 1397～1468）によって発明されたものである。また、日本語の金属活字の鋳造とそれを用いた活版印刷は、1870 年頃、本木昌造によって始められたものである。

活字組版は、同一文字に多くの活字が必要であり、しかも、文字サイズが変われば、別の活字が必要である。このため、活字の格納には広いスペースが必要である。また、組版速度も決して速くなかった。更に、組み上がった活字面の高さをそろえて、ムラのない印刷を行うには熟練した技術者が必要であった。このため、現在では活版印刷はほとんど用いられなくなっている。

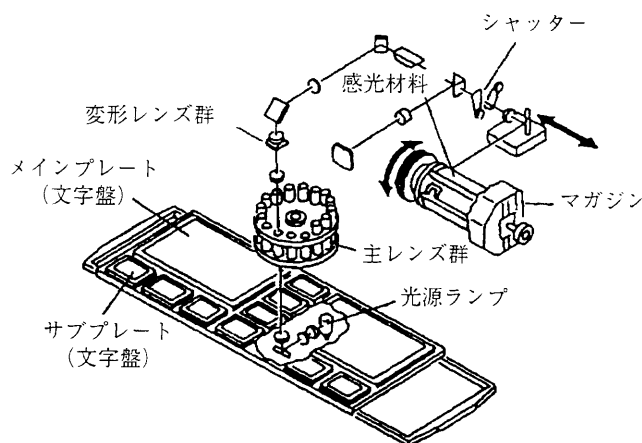


図 2.2 手動写植機

Fig.2.2 Manual phototypesetting system.

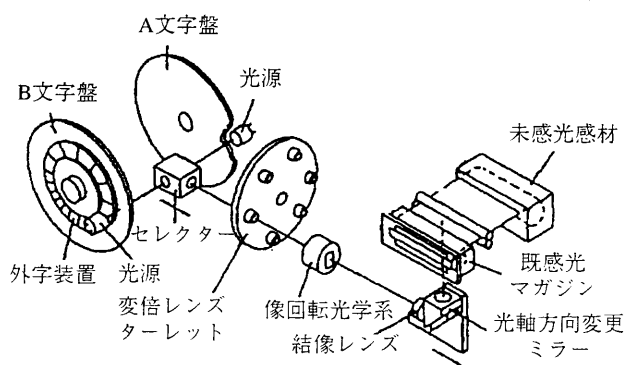


図 2.3 全自動写植機

Fig.2.3 Automatic phototypesetting system.

(2) 写植

写植とは、複数の文字をネガフィルム上に収容した文字盤を用い、写真技術の応用によって印画紙やフィルム上に印刷版下を形成する手法である。1950年代から本格的に普及した第1世代の写植機は、オペレータが文字を1つずつ選字するもので、手動写植機と呼ばれた。(図 2.2) その後1960年代後半からは、選字を自動化した回転文字盤方式の第2世代機も普及した。(図 2.3)

写植の最大の特長は、活字のように文字サイズが固定ではなく、光学処理により文字の拡大、縮小や斜体などの変形が行えることである。このため、写植は活字のように広い収

納場所を必要としない。また、文字盤の交換により、簡単に書体を切り替えることが可能であるため、豊富な書体が登場することになった。

写植機の開発当初は、活字デザインの流用が考えられたが、結局、写植専用書体がデザインされることとなった。これは、活字のデザインそのままでは、文字線が細すぎてかすれてしまったり、小さな文字ではストローク交差部がつぶれ気味になったためである。

(3) CTS

CTS (Computer Typesetting System) は、半導体メモリに蓄積したビットマップフォントやベクトルフォントを用いて、写植機と同様に印画紙やフィルム上に版下を形成するシステムで、電算写植機とも呼ばれる。1970年代に登場したCTSは、CRTを出力デバイスに用いるもので、文字出力を主体に高速自動処理を可能とするものであった。現在では、RIP(Raster Image Processor)を使用して、文字と画像を同時に処理し、レーザーを用いて一括出力することが可能となっている。

(4) DTP

DTP(Desk Top Publishing)は、GUI(Graphical User Interface)をもったパーソナルコンピュータ、高品質なベクトルフォントを備えたレーザービームプリンタ、高度な編集を可能とした組版ソフトウェア、この3つの登場によって生まれたものである。図や画像入りの複雑な組体裁を可能とするDTPは、それまでの文字出力を主としていたワードプロセッサ文書とは、一線を画するものであった。[16]

当初、DTPは、企業内文書やミニコミ誌のように少部数の出力に対して、専門の印刷業者の手を経ずにレーザービームプリンタの出力をそのまま使用するものと定義されていた。しかし、プロのデザイナーや印刷業者にも次第にパーソナルコンピュータを用いた制作手法が普及していったため、今では、版下作成までのいわゆるプリプレス工程をDTPで制作することが珍しくなくなっている。また、一般ユーザ向けのワードプロセッサも図や画像を扱うことが可能となっており、DTPとの機能的な差は小さくなっている。

2.2.2 フォント方式

フォント方式は、図2.4に示すように活字や写植文字盤に代表されるアナログフォントと計算機処理を前提としたデジタルフォントに分類できる。ここでは、本論文に関わりの深いデジタルフォントについて述べる。デジタルフォントは、出力デバイスの解像

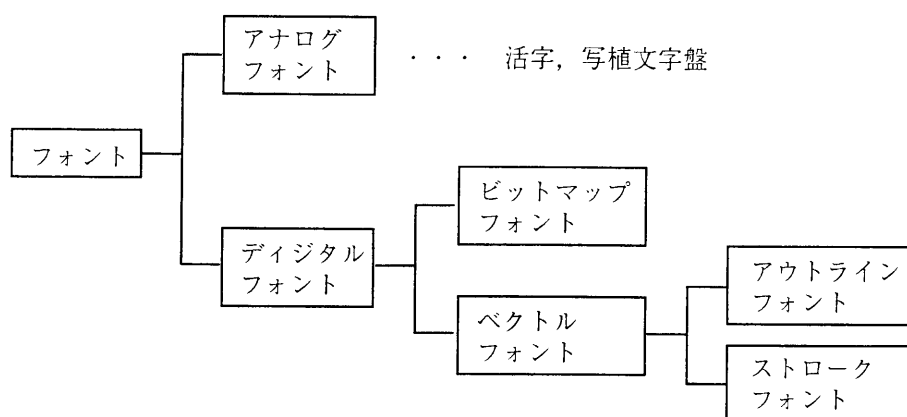


図 2.4 フォント方式
Fig.2.4 Font methods.

度と文字サイズに応じて個別のデータを用いるビットマップフォント方式と出力デバイスの解像度や文字サイズに依存しないベクトルフォント方式の2つに大別できる。これらは、その使用目的の違いから制作法も異なるものとなっている。

(1) ビットマップフォント

ビットマップフォントは、図 2.5に示すように文字を縦横の細かなメッシュに区切ってその中が黒であるか白であるかを1または0で表したものである。あらゆる情報を1と0にデジタル化して処理する計算機では、CRT 表示やプリンタ印刷の際に文字を小さなドットの集まりとして表現する。したがって、ビットマップフォントは記憶形式と表示・印刷の形式が一致し、出力デバイスの特性に合わせた高品質の出力を高速に得られることが特長である。

しかしながら、ビットマップフォントを単純に拡大するとドットがそのまま大きくなって、文字の輪郭にぎざぎざが目立ってしまう。このように、ビットマップフォントは、拡大縮小が難しいため、文字のサイズごとにデータが必要となる。ある文字サイズにおけるデバイスの解像度とビットマップフォントのドット数との間には、図 2.6に示す関係 [17] がある。大きなサイズの文字や高解像度のデバイスに対応するには膨大な記憶容量を必要とすることが、ビットマップフォントの欠点である。

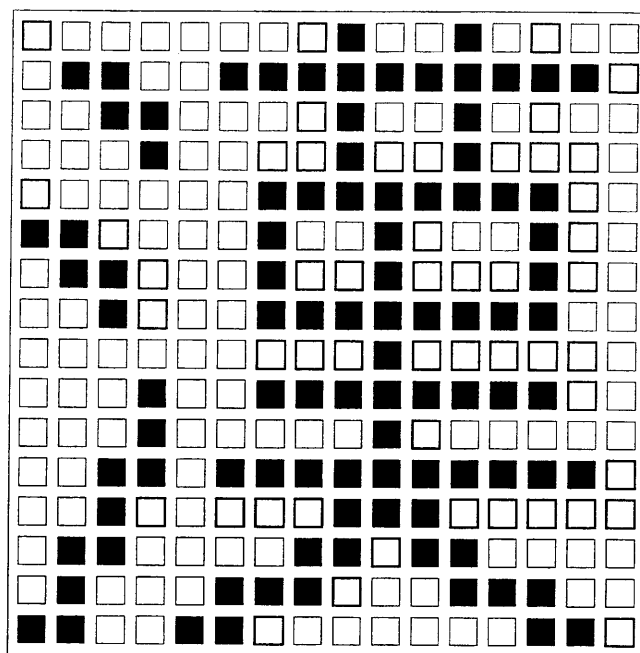


図 2.5 ビットマップフォント

Fig. 2.5 Bitmap font.

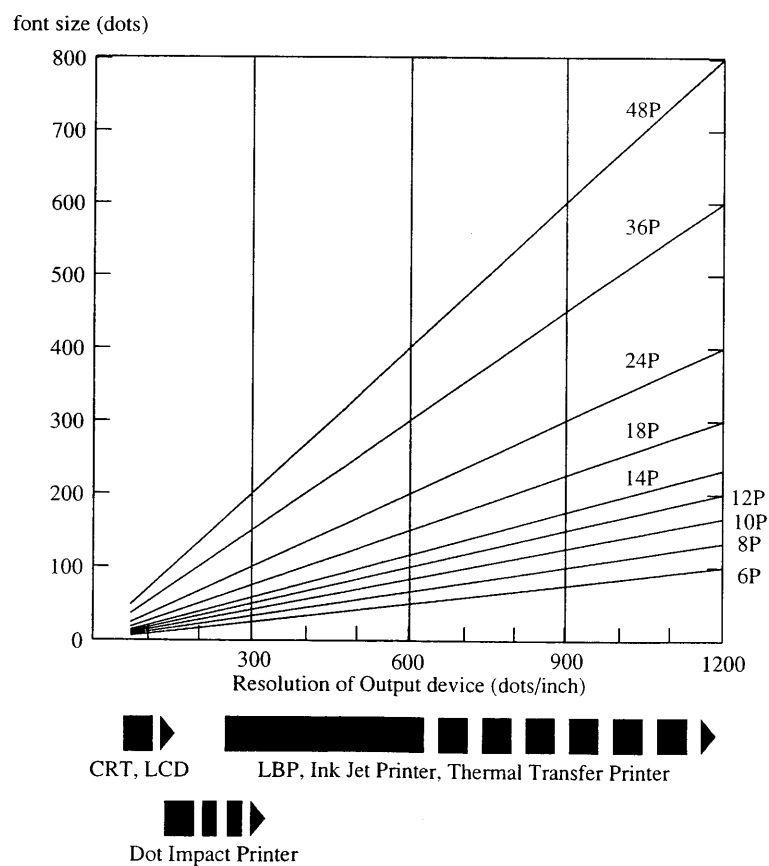


図 2.6 各文字サイズにおける出力装置の解像度とフォントのドット数

Fig. 2.6 Font size to device resolution.

(2) ベクトルフォント

ベクトルフォントは、文字形状をベクトルすなわち座標値として記憶しておき、出力の際に文字サイズに応じて変倍処理した後、ビットマップデータに変換するものである。ベクトルフォントは、更にアウトラインフォントとストロークフォントに分類できる。

アウトラインフォントは、図 2.7に示すように文字の輪郭を抽出し、それを特徴づける点の座標を記憶する方式である。座標間の補間にはスプライン曲線やベジェ曲線を用いるため、大きく拡大してもぎざぎざのない高品質の文字を出力することが可能である。アウトラインフォントが、DTP の普及に果たした役割は非常に大きく、現在では、パーソナルコンピュータの OS や個人向けのワードプロセッサにも標準搭載されている。

ストロークフォントは、図 2.8に示すように文字ストロークの中心線座標を記憶する方式である。本来のストロークフォントは、文字骨格だけでストロークの太さ変化を表現できないため、ペンプロッタや機械彫刻が主な用途である。

しかし、ストロークにアウトラインデータを張り付けたり、ストロークに沿って移動する筆触形状を制御して肉付けすることによって多彩な文字表現を行うことも可能である。ストロークフォントに基づくフォントデータ圧縮やフォント制作の効率化を目指した研究が報告されている。[23],[26],[30], [33]

デジタル時代のフォントには、活字とも写植とも異なるデザイン技法が求められる。特にベクトルフォントは、さまざまな解像度、文字サイズで出力されるため、いろいろな条件下において、一定の品質が得られるようにデザインすることが必要である。[8]

2.2.3 フォントの制作工程

図 2.9は、フォント制作フローの典型的な例を示すものである。以下、この例にしたがって、原字デザイン (Artwork Design)、ディジタル化工程 (Digitization)、品質補正データの付加 (Hinting)、データ圧縮 (Data Compression)、フォーマット変換 (Format Conversion) の順に各工程について述べる。[9],[10]

(1) 原字デザイン

原字デザインに要求されることは、1 文字 1 文字が美しく、しかもそれを組み合わせて文章としたときに統一感があって読みやすいことである。このため、フォントデザインには、専門のデザイナーが単独あるいは小人数のグループで当たることが多い。この作業は、

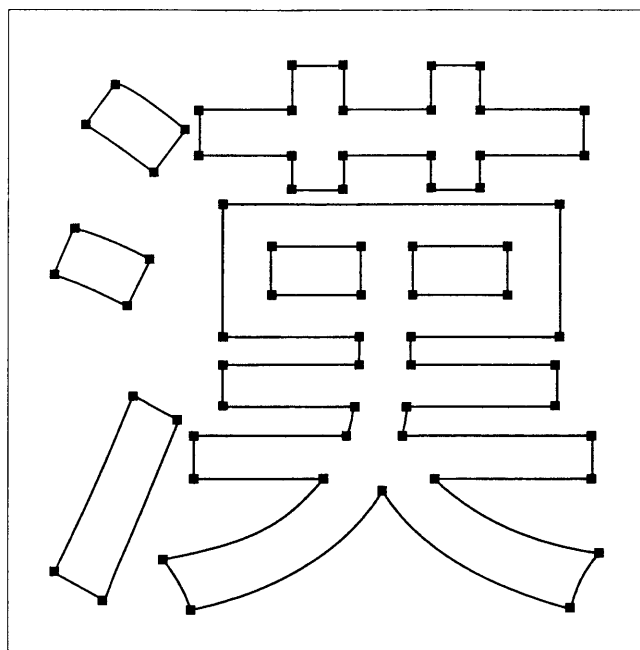


図 2.7 アウトラインフォント

Fig. 2.7 Outline font.

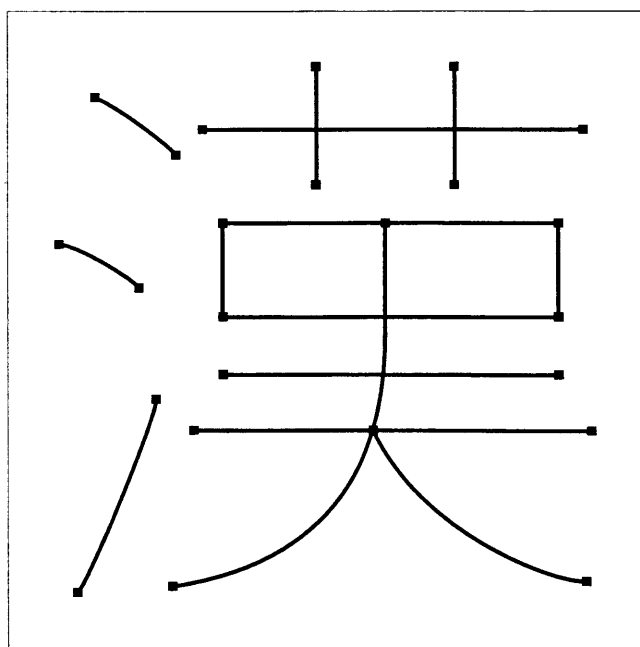


図 2.8 ストロークフォント

Fig. 2.8 Stroke font.

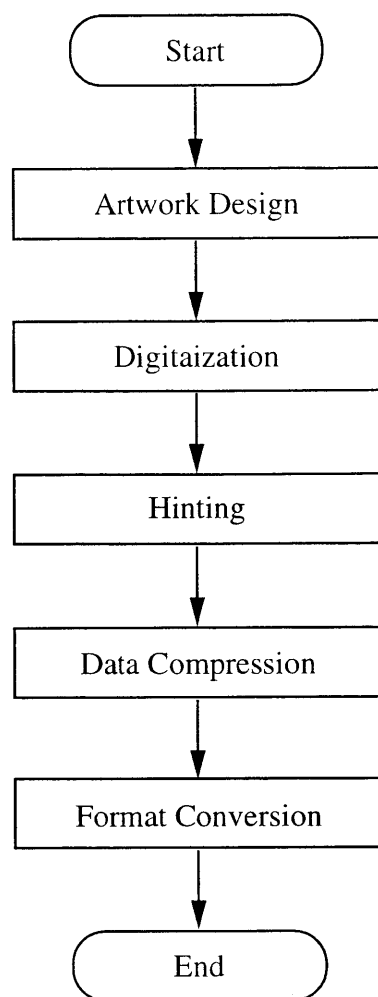


図 2.9 フォント制作フロー
Fig.2.9 Font production flow.

従来、デザイナーが紙上で行っていたが、最近では、コンピュータに直接入力する場合もある。

(2) デジタルイズ工程

デジタルイズ工程は、原字をビットマップデータあるいはベクトルデータに置き換える工程である。高品質なフォントを制作するには、原字デザインに忠実であることだけでなく、時には出力特性を考慮してフォント形状を変更することも要求される。例えば、600dpi以下の出力デバイスでは、1ドットの変化が文字品質に重大な影響を及ぼすため、線幅を統一したり、ゆるやかな曲線を直線に変更することが、出力品質の向上に有効である。また、32ドット以下のフォントでは、時として、画数の多い複雑な漢字を正確に表現するこ

とが不可能となるため、字形の簡略化が必要となる。このような処理を行うため、この工程には経験豊富な専任技術者が必要となる。

(3) 品質補正データの付加

品質補正データの付加は、ベクトルフォントの品質を向上させるために必要となる工程である。ビットマップフォントが、ディジタル化されたデータをそのまま使用するのに対し、ベクトルフォントは、サイズ変更を行って使用することが前提となっている。このため、ベクトルフォントでは、使用時の演算誤差による品質劣化の発生する可能性がある。そこで、ベクトルフォントを用いて高品質の出力を得るために、使用時にフォント形状を動的に変更して文字品質を補償する処理が行われる。品質補正データは、この品質補償処理を行うために、あらかじめフォントデータに付加しておくもので、ヒントとも呼ばれる。[2],[3]したがって、品質補正データの付加は、必ずしもフォント制作に必須な処理ではない。

(4) データ圧縮

欧文フォントが、記号等を含めても 200 文字程度であるのに対し、漢字処理に必要な日本語では、1 フォントあたり約 7,000 文字が必要となる。しかも、漢字は欧文に比べて形状が複雑である。このため、日本語フォントは、データサイズがかなり大きくなる。実際にアウトラインフォントでデータサイズを比較すると、欧文が数十 KB であるのに対して、日本語のデータは数 MB にも及ぶ。そこで、パーソナルワードプロセッサや携帯情報端末のように大容量の外部記憶装置を使用できない機器では、フォントデータ圧縮が必要となる。

フォントデータ圧縮においては、FAX に用いられる MH 符号化や MR 符号化といった一般的な画像圧縮手法よりも文字の構造的特徴を利用した圧縮手法が有効である。例えば、漢字を形成するストロークには同じ形が繰り返し出現するので、これを図 2.10 に示すように文字に依存しない共通の部品とし、文字ごとに保持するデータを部品番号とその配置座標のみとすれば、大幅なデータ圧縮が実現できる。[20]

(5) フォーマット変換

最後に、出力機器の OS やアプリケーションに対応したフォーマット [18] にフォントデータを変換して、フォント制作は完了する。

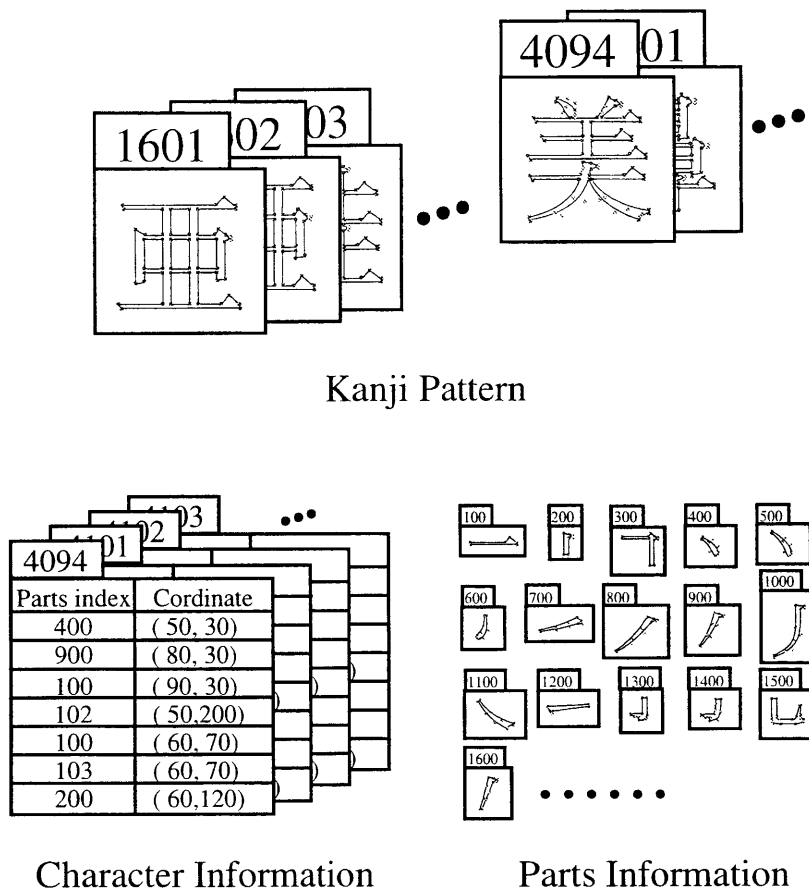


図 2.10 フォントデータ圧縮
Fig.2.10 Font data compression.

2.2.4 フォント制作の問題

前節で述べたフォント制作工程には多くの人手が必要である。特に日本語の場合には、何千もの漢字データを開発しなければならないため、1書体の制作に数年を要することも珍しくない。数多くの書体デザインを所有している印刷機器メーカーの中には、これを門外不出とし、他社へ提供しないところもある。このため、電子機器メーカーが、日本語ワードプロセッサや日本語 DTP の開発を始めた 1980 年頃には、高品質なフォントを自由に使えるという状況があった。そこで、この問題を解決し、日本語フォントの健全な普及を目指すため、標準となる日本語フォントを共同開発しようとする試みも行われた。[19]

その後、フォントを取り巻く状況は徐々に改善されてきた。パーソナルコンピュータ用 OS は、高品質アウトラインフォントを標準搭載するようになった。フォント提供者の絶え間ない努力によって多彩な日本語フォントも市場に流通するようになった(図 2.11)。

■明朝系書体



■ゴシック系書体



■POP系書体



■筆書系書体



図 2.11 日本語フォントの例

Fig. 2.11 Examples of Japanese font.

しかしながら，ユーザは，より豊かな文書表現を可能とするために，新たなフォントを求め続けている．自分だけの個性的な書体を使いたいという要求も強いであろう．ユーザの個人性を反映した固有のフォントを生成する手法，あるいはフォントを個性化する手法が求められているのである．

2.3 フォント生成に関する従来の研究

本研究は，DTP やワードプロセッサなどの文書作成において，筆記者の個人性を反映した手書き文字の利用を目指すものである．この目的を実現するための素朴な方法の一つとして，ユーザの筆跡そのものをデータ化して個人専用のフォントを作成することが考えられる．しかし，日本語文書に必要とされる数千もの文字を筆記するのはユーザに大変な負担を課すこととなる．しかも，従来のフォント制作システム [21],[22] は，専門家の使用を前提としたものであり，一般ユーザが使用することは容易でなかった．これを解決するには，効率的なフォント自動生成の手法が必要である．

これまでもフォントあるいは文字パターンの生成に関する研究は，広く行われてきた．これらの生成対象を分類すると以下のようなになる．

1. 明朝体やゴシック体のような典型的な印刷用書体
2. 筆文字やペン字のような手書き風の書体
3. 筆記者の個人性を反映した手書き文字

以下，上記分類にしたがって，従来の文字パターン生成に関する研究を概観する．

2.3.1 印刷用書体生成に関する研究

日本語処理では，膨大な数の漢字を効率よく扱うことが，常に問題となる．文書の作成・出力において，記号等を含めて1 書体あたり 200 文字もあれば足りる欧文に比べ，数千文字の漢字を扱わなければならない日本語は，フォントの制作コストが非常に高い．そこで，文字パターンの自動生成により，日本語フォントの制作を効率化しようとする種々の研究が行われてきた．

陳 [23] は，文字を構成する 9 種類の基本ストロークを 21 種類の手本ストロークに分類して作成した輪郭線パターンデータベースと，使用するストロークのコード，大きさおよび文字内の相対位置からなる構造データを用いて，明朝体文字の規則的な生成を行ってい

る。この研究は、文字数の多い日本語フォントにおいて問題となるフォントデータ圧縮を目的としており、1文字あたり50バイトの構造データでフォントを表現可能であると報告している。

上原 [24],[25],[26] は、骨格ベクトルに肉付けする方式を用いた漢字フォントの生成を行っている。この研究では、漢字のストロークを基本エレメント種別、ストロークの骨格点の位置、ストローク形状を表すパラメータという3種類の情報で表現し、JIS 漢字約6,000字を試作している。

長橋 [27] は、手書き文字認識の辞書作成に用いた漢字符号化法 [28],[29] を応用して、漢字の骨格パターンを生成する研究を報告している。この漢字符号化法は、部分パターンの相対的な位置関係を座標を含まない近接関係とクロス関係に分析し、記号列で表現するものである。

田中 [30],[31] は、漢字を具体的な座標を含まない抽象的な組み合わせ情報で定義しておき、プログラムによって部品を組み合わせでフォント生成する方法を提案している。この方法は、線の太さを自由に変えられるスケルトンデータで部品を表現しており、部品デザインの変更によって複数書体に対応するものである。

奥村 [32] は、文字要素を組み合わせで漢字フォントを再構成する方法において、新たにフォントをデザインするのではなく、既存のアナログフォントの画像データから部品データを識別し、その位置や大きさの情報を抽出・収集して自動的にフォント化する手法を提案している。

以上の研究は、デザイン技法の確立した印刷書体を対象として自動生成を行うものであって、ユーザの個性を生成結果に反映させることは目的としていない。

2.3.2 手書き風書体生成に関する研究

手書き系の文字に関しては、張による一連の毛筆文字生成に関する研究 [33],[34],[35] がある。この研究は、漢字のストロークを分類し、それらを始点座標、方向、長さ、太さ、曲度からなるパラメータの入力で生成するプログラムを作成しておき、このストローク発生プログラムの組み合わせで毛筆漢字パターンを生成するものである。この方法は、良好な品質の生成結果を得るものであるが、一文字ごとに多くのパラメータ設定を必要とし、エンドユーザが手軽に扱えるものではない。

また、塩野 [36] は、パソコン内蔵のビットマップフォントからベクトルパターンを生成し、これに非線形な幾何学的ひずみを印加して手書き風文字パターンを得る手法を提案し

ている。しかし、この方法も筆記者を特定して、その人の手書き文字に似た文字パターンを生成するための係数取得方法は課題としている。

2.3.3 筆者の個人性を反映した手書き文字生成に関する研究

ユーザの個性を生成文字に反映させようとする研究は、その応用目的によって更に2つに分類できる。

1. DTP やワードプロセッサなどの文書作成

2. 文字認識や筆者識別の精度向上

(1) 文書作成への応用

木島 [37] は、複数の標準文字パターンから入力文字パターンに最も類似した標準パターンを選出し、入力文字パターンと標準文字パターンの対応する特徴点間の内分点を算出することにより、ユーザの手書き文字の特徴を残しながら、標準的なきれいな文字を出力しようとする手法を考案している。

山崎 [38] は、毛筆筆記における始筆、送筆、終筆などの情報を書字知識ベースに格納しておき、これをユーザがペンタブレットから筆記入力した文字骨格に適用して毛筆書体を生成する手法を提案している。

中村 [39],[40] は、ユーザが電子筆圧ペンを用いて描いた一画を認識部のニューラルネットワークに入力し、毛筆ストロークをもつデータベースと照合して入力ストロークに相当する毛筆ストロークを生成している。

しかし、これらの方法では、生成対象文字ごとにユーザの手書き文字入力が必要であるので、文字カテゴリーの多い日本語では、ユーザへの負担が大きすぎる。

(2) 文字認識や筆者識別への応用

文字認識や筆者識別 [41] の分野においては、多様な手書き変形に対応できる認識辞書の開発や認識システムの性能評価、更に文字パターンの統計的特徴量の図示を目的として、文字パターン生成の研究が行われている。

石井 [42],[43] は、標準字形に振幅と平滑化の2つのパラメータで記述できる不規則な摂動を加えて変形文字を発生させる手法を提案し、この手法によって発生させた文字パター

ンを文字認識の識別辞書設計の学習パターンに適用している。しかし、この生成結果は特定の筆記者の個人性を反映させたものではない。

倉掛 [44] は、手書き文字の学習ベクトルをクラスタリングして得た平均ベクトルと標準ベクトルを結ぶ直線を求め、この直線を中心とした正規乱数により変形文字パターンを発生させる手法を提案している。そして、この変形パターンを手書き文字認識系の文字品質依存特性、安定性、リジェクト機能の評価データに用いている。

また、吉村 [45] は、高次元の特徴量を主成分分析して得た特徴量と文字パターンの変形傾向との関係を示すことを目的に、特徴パラメータから円弧近似を用いた文字パターン生成を行っている。

これら二つの生成結果は、筆記者の個人性をかなり反映しているが、生成対象の文字カテゴリごとに個人性の抽出が必要となるため、本論文で目的としているフォント生成に応用することは難しい。

喜多 [46] は、個人性を見本字型からの変形として文字カテゴリに依存しないようにルール化し、少数のユーザ手書き文字をもとに任意カテゴリの文字を生成する方法を提案している。しかし、このルールはかなり高度な知識を必要とする複雑なものであり、一般ユーザに記述させるのは難しい。

更に、手書き過程を確率モデルととらえる研究もある。松尾 [47],[48] や Attachoo[49] は、文字パターンをストロークの線分長と角度変化でモデル化し、標準パターンを正規分布にしたがって変形する手法を提案している。しかし、これらのモデルは、複雑な字形を有する漢字には適用されていない。

2.4 結言

本章では、まず最初に活版印刷から計算機を用いた CTS や DTP へ至る印刷技術の変遷をたどることにより、アナログからデジタルへと変わっていったフォント技術およびその制作手法について述べた。

フォント制作には、既に確立された手法が存在するものの、すべての文字について均質で読みやすいフォントを制作するには、相当な経験が必要である。特に文字数の多い日本語フォントの場合、制作に要する労力は並大抵ではない。一方、表現力豊かな文書作成を行うために、今後も継続的なフォント開発が求められている。

以上の理由から、自動生成により、フォント制作工程の効率化をはかろうとするさまざまな研究が行われてきた。この中には、本研究と同様に筆記者の個人性を持った文字パター

ンを自動生成しようとする研究が含まれている。

しかし、従来の研究は、文字カテゴリーに依存した入力を要したり、カテゴリーへの依存を回避するために高度なルールの導入を必要とするため、一般ユーザが利用することは難しかった。本研究は、この原因が、ただ1組の基準パターンをもとに生成を行うところにあるのではないかと考える。

次章以降では、個人性を反映した手書き風フォントを一般ユーザにも利用可能とすることを目指し、むしろ多くの基準パターンに依拠する方法、つまり事例ベースに基づくフォントの個性化を提案する。

第3章

部分字形組み合わせを用いた 手書き風フォント生成

3.1 緒言

本章では、漢字の構造的特徴を利用し、エンドユーザの筆記した漢字からヘンやツクリなどの部分字形を抽出して組み合わせることにより個性的フォントを生成する手法について述べる。[50],[51],[52]

漢字の成立は、象形、指事、形声、会意、転注、仮借の6通りに分類されている。[4],[5]この中で「形声」すなわち意味を表す要素と音を表す要素の組み合わせ、および複数の漢字の組み合わせからなる「会意」の占める割合は非常に大きい。更に、他の漢字も誕生から長い年月を経る間に字体の整理が行われ、多くが規則的なパターンの組み合わせでできている。

本手法は、手書き文字に含まれる個人性を解析して文字パターンを生成するのではなく、事例ベースに基づいてフォント生成結果に個人性を反映しようとするものである。すなわち、漢字の構造的特徴に着目し、ユーザの筆記した漢字から直接抽出したヘンやツクリなどの基本部分字形を生成規則にしたがって変形、配置してフォントを生成するものである。

ここで課題となるのは、基本部分字形を収集する際にユーザの筆記労力をできるだけ軽くし、かつ品質の高い生成結果を得ることである。そこで、できるだけ少ない漢字から必要な基本部分字形を抽出するアルゴリズムを提案する。また、文字の生成品質を向上させるために部分字形が漢字の中に出現する位置を分析し、これを新たな属性として付加する。生成実験によって、同一の生成規則を用いても基本部分字形を取り替えれば、出力結果は大きく変化し、個性的な手書き風文字を生成できることを示す。

3.2 システムの概要

図3.1に本手法による手書き風文字生成システムのブロック図を示す。生成規則格納部 (General generation rule) は、あらかじめ生成対象文字すべてから抽出した生成規則を格納している。次にユーザが、基本部分字形抽出に必要な特定の漢字を手書き入力すると、部分字形抽出部 (Character element extraction) は基本部分字形の抽出を行う。文字生成処理部 (Character generation) は、これらの生成規則と基本部分字形を用いて、入力された文字コードに対応する手書き風フォントを生成する。

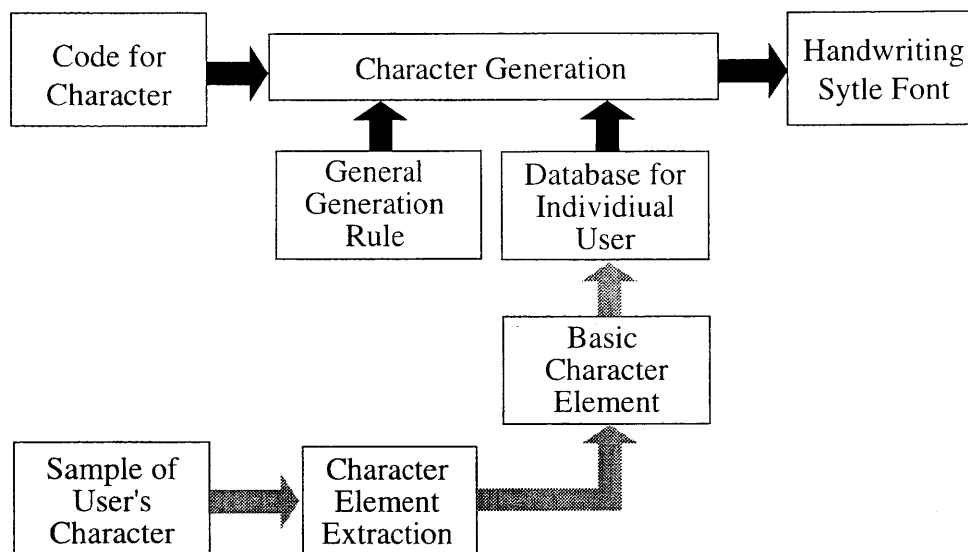


図 3.1 システムの構成
Fig.3.1 System diagram.

3.3 字形データベースの構築

3.3.1 漢字の構造

漢字の成立には次の6通りがあるといわれている [4],[5].

- 象形 物体の形状を描いたもの
例：日，月，山，川
- 指事 線や点で事物の性質等を表したもの
例：一，二，上，下
- 会意 2つ以上の文字から別の意味をつくったもの
例：日+月=明
- 形声 形状と音声を組み合わせてつくったもの
例：水+可=河
- 転注 ある漢字の意味を別の意味に転用するもの
例：悪=「わるい」から「にくむ」への転用
- 仮借 音だけあって文字のない言葉に同じ音の別の意味の漢字をあてたもの
例：豆=もともと食物を盛る器



図 3.2 部分字形の例
Fig. 3.2 Examples of radical parts.

教育漢字 996 文字に含まれる会意および形声の割合が、約 86%と非常に高いことが示すように多くの漢字は、会意または形声によって成立したものである [5]。すなわち、大部分の漢字はヘンやツクリなどの構成要素を一定の規則に従って組み合わせたものとして表すことができる。構成要素を更に小さく分解すれば、会意や形声以外の漢字にも構成要素の組み合わせで表現できるものがある。本研究では漢字の構成要素を部分字形、部分字形の組み合わせ方を構成パターンと呼ぶ。

3.3.2 部分字形

部分字形には、漢和辞典で用いられる部首の他に生成対象漢字の中に 2 回以上出現する構成要素を加えている。例えば、「葉」のクサカンムリから下の構成要素は、漢和辞典で用いられる部首ではないが、JIS X0208 規格に収録された 6,355 個の漢字中、図 3.2 に示す 9 個に出現する。そこで、この「葉」のクサカンムリから下の部分は、部分字形とする。

3.3.3 構成パターン

構成パターンは、漢字を部分字形に分割する領域を示すものである。図 3.3 に基本的な構成パターンとその構成パターンをもつ漢字の例を示す。基本的な構成パターンの種類は、表 3.1 に定義する 4 種類である。

















Unique	 十 (u x)
Pararelle	 岐 (p x x)  衛 (p x x x)  淵 (p x x x x)
Cascade	 阜 (c x x)  莫 (c x x x)  寧 (c x x x x)
Fence	 国 (f1 x x)  間 (f2 x x)  ヨ (f3 x x)  凶 (f4 x x)  匡 (f5 x x)  辻 (f6 x x)  屈 (f7 x x)  司 (f8 x x)  斗 (f9 x x)

図 3.3 構成パターン

Fig. 3.3 Constitution pattern.

図 3.3は、構成パターンの図的表現と記号表現の2種類を示している。図的表現は、文字枠を水平・垂直の線分で分割して図示するものである。記号表現は、分割タイプを表す記号 u, p, c, f1~f9 と部分字形を表す記号 x を用いるものである。図的表現は直観的で分かりやすいことが特長であり、記号表現は計算機上での扱いが容易なことが特長である。

記号表現において、部分字形を示す記号 x が左から右へ並ぶ順と図的表現における部分字形領域の並びは、一意に対応づけられている。すなわち、記号 x が左から右へ並ぶ順は左右分割タイプ p では左から右の部分字形領域、上下分割タイプ c では上から下の部分字

表 3.1 構成パターンの種類

Table 3.1 Type of constitution patterns.

種 類	記 号	意 味
Unique	u	分割不可タイプ
Parallel	p	左右分割タイプ
Cascade	c	上下分割タイプ
Fense	f1~f9	周辺分割タイプ

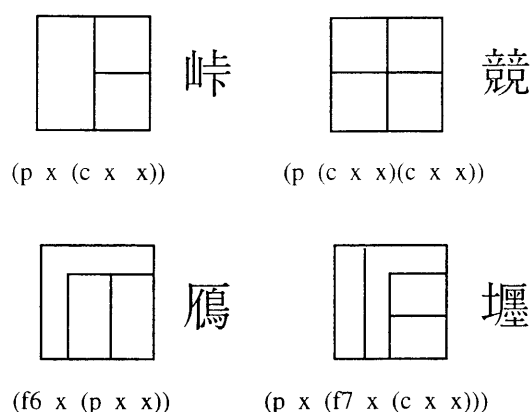


図 3.4 複合構成パターン

Fig.3.4 Compound constitution pattern.

形領域，周辺分割タイプ f では周辺から内部の部分字形領域へそれぞれ対応する．また，基本構成パターンで表せない複雑な構造をもつ漢字は，複数個の分割タイプを組み合わせた複合構成パターンによって表現できる．複合構成パターンの記号表現は基本構成パターンの入れ子表現となる．図 3.4 に複合構成パターンの例を示す．

3.3.4 出現位置属性

字形データベースでは，同一概念をもつ部分字形に原則として 1 つの部分字形コードを与える．したがって，図 3.5 に示す「禾」のようにヘンとなる時とカンムリとなる時で形状の変化する部分字形でもコードは 1 つである．ところが，文字生成では出現位置による部分字形の形状変化を正しく表現する必要がある．そこで，図 3.6 に示すように部分字形が大きく形状変化する位置を四つに分類し，ヘンの位置に来る場合は P 属性，カンムリの位置に来る場合は C 属性，カマエなどの位置に来る場合は F 属性，その他の位置に来る場合は E 属性を付与した．ここで，例外的なのは構成パターン F8 の内部に来る場合を E 属性とせず，P 属性としたことである．このような文字には，「武」や「式」がある．

3.3.5 字形データベースの構築結果

JIS X0208 規格に収録された 6,355 の漢字を対象に字形データベースを構築した結果，基本部分字形が 1,012 個，その組み合わせを示す構成パターンが 105 通り，および出現位置属性が 4 種類となった．

次節では，この字形データベースを利用したフォント生成手法について述べる．

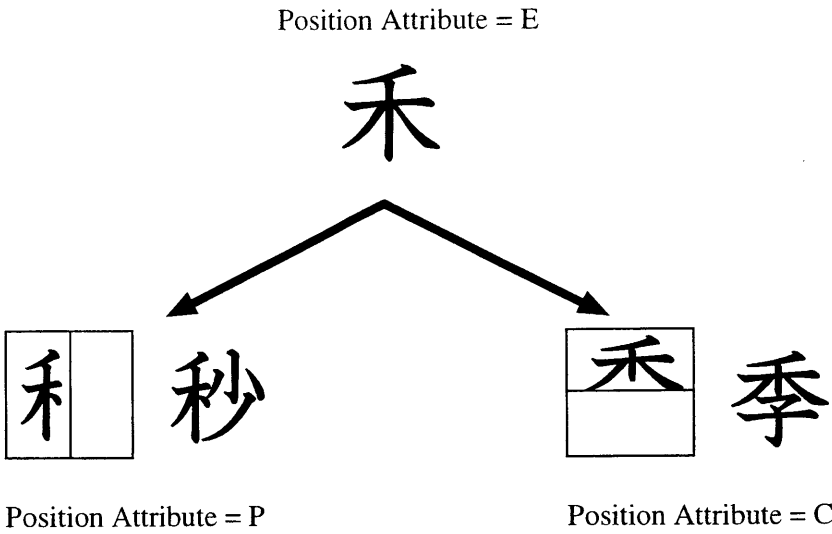


図 3.5 出現位置による形状変化
Fig.3.5 Variation according to the position.

P(Parallel)	
C(Cascade)	
F(Fence)	
E(Else)	

図 3.6 出現位置属性
Fig.3.6 Position attributes.

3.4 フォント生成

3.4.1 文字データの表現

手書き文字を形成する要素は、文字の骨格と筆触形状の2つである。筆触形状とは、筆記具と記録素材の接触によって現れるものである。本研究では、ペンタブレットを用いた筆記動作によって文字データを収集するため、文字データとして扱うのは文字骨格のみとなる。フォント生成時には、一定の半径をもった円で筆触を近似してボールペン風の出力結果を得る。

文字データは、第 s 番目のストロークの第 i 番目の点を $P_s^{(i)}$ 、その座標を $(x_s^{(i)}, y_s^{(i)})$ と表記する。すなわち、点 $P_s^{(i)}$ をストローク s ごとに i の順に直線で結んだものが文字骨格である。また、ストロークを区別せずに全ストローク上の点を連続して扱い、第 i 番目の点を $P^{(i)}$ 、その座標を $(x^{(i)}, y^{(i)})$ と表記することもある。なお、ここでは x 座標の正の向きを右、 y 座標の正の向きを下に取っている。

3.4.2 生成規則

文字生成を行うには、ユーザの手書き文字から抽出し字形データベースに収録した基本部分字形から

- どれを選択するか
- どのように変倍するか
- どこに配置するか

を決定しなければならない。本研究では、これらの項目を生成規則と呼ぶ。

字形データベースには、漢字を階層的に部分字形に分解し登録している。すなわち、字形データベースの構造をツリー表現すると、ルートに漢字、リーフにそれ以上分解不可能な部分字形が来る。また、ルートからリーフへ至る経路の中間ノードには、リーフの部分字形で構成される部分字形あるいは漢字が存在する。リーフの部分字形は、入力された手書き文字中に必ず含まれている。

今回用いた生成規則では、ルートからリーフに向かってたどり、中間ノードの部分字形が、字形データベースに登録されていれば、それより下のレベルへは降りずに、その部分字形を使用する。これは、ユーザが入力した文字中に適切な組み合わせの部分字形が含まれていれば、それを用い、不要な分解を避けるほうが、良い生成品質を得られると考えたか

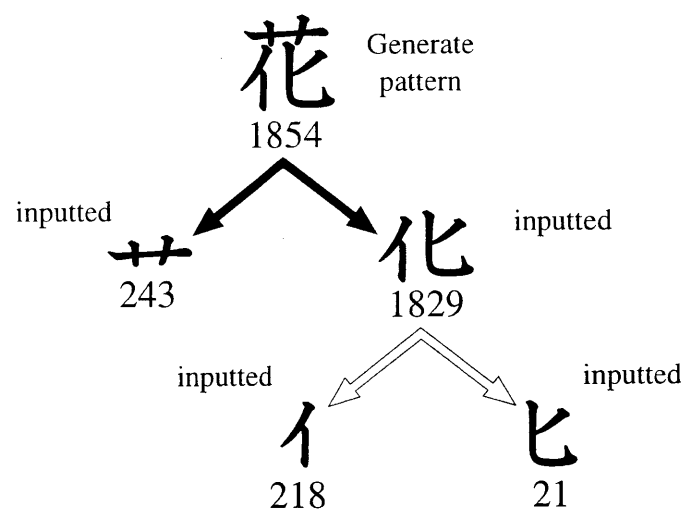


図 3.7 部分字形選択の例
Fig. 3.7 Example of selecting radical part.

らである。図 3.7は，ツリーをたどったところ，「花」という漢字がクサカンムリと「化」から生成できたことを示す例である。

3.4.3 基本部分字形の抽出

基本部分字形データの作成においては，部分字形を独立に筆記して入力するよりも漢字として筆記された中から抽出するほうが，より自然な形状のデータを得られるであろう。この時，筆記対象漢字の数は少ないほどユーザの負担を軽くできて好ましい。そこで，基本部分字形の抽出に用いる漢字をできるだけ少なくするために，部分字形を多く含む漢字から順に筆記対象に加えるアルゴリズムを提案する。このアルゴリズムを常用漢字 1,945 文字，JIS 第 1 水準漢字 2,965 文字並びに JIS 第 1・第 2 水準漢字 6,355 文字に適用した結果得られた筆記対象漢字の数を表 3.2に示す。

表 3.2 部分字形抽出に必要な文字数
Table 3.2 Chinese characters needed to extract the basic radical part.

生成対象文字	出現位置属性なし	出現位置属性あり
常用漢字	257 (13.2)	388 (19.9)
JIS 第 1	282 (9.5)	454 (15.3)
JIS 第 1 第 2	345 (5.4)	650 (10.2)

カッコ内は，生成対象文字集合全体に対する割合 (%)

<入力文字絞り込みアルゴリズム>

- (i) 文字ごとに含まれる相異なる部分字形を数えあげ、その数を N とする
- (ii) N が最大の文字をサーチし、 C とする
- (iii) *if* $N \neq 0$
 then 文字 C を基本部分字形抽出の対象に加える
 else 終了
- (iv) $N > 0$ の文字について、文字 C に含まれる部分字形を数え上げの対象から外す
- (v) *goto* (i)

以上のアルゴリズムでは、同一部分字形を含む入力対象文字が複数個選ばれる場合もある。この場合は、部分字形に外接する四角形の面積を求め、最大となるものを基本部分字形として登録する。これは、文字生成時の変倍処理において、基本部分字形の拡大がなるべく起こらないようにして手書きデータに含まれる線のブレ等が強調されない品質の良い生成結果を得るためである。

図 3.8 を用いて本アルゴリズムを説明する。この例は、「繰」と「葉」が入力対象文字に選ばれた場合を想定したものである。これら二つの漢字は、両方とも同一の部分字形「木」を含んでおり、その出現位置属性も E で同じである。そこで、両者の外接四角形面積を比較し、より面積の大きい 75-b を字形データベースに登録する。

3.4.4 部分字形の配置

部分字形の配置は、部分字形の外接四角形の左上頂点を基準として幅と高さが一致するように変換する方法を用いる。図 3.9 に示すように基本部分形を構成する点の座標を (x_b, y_b) 、基本部分字形に外接する四角形の左上頂点座標を (X_b, Y_b) 、幅を W_b 、高さを H_b とし、生成文字の部分字形に外接する四角形の左上頂点座標を (X_g, Y_g) 、幅を W_g 、高さを H_g とすれば、生成される部分字形の座標 (x_g, y_g) は、式 3.1 で表される。

$$\begin{aligned} x_g &= (x_b - X_b) \cdot \frac{W_g}{W_b} + X_g \\ y_g &= (y_b - Y_b) \cdot \frac{H_g}{H_b} + Y_g \end{aligned} \quad (3.1)$$

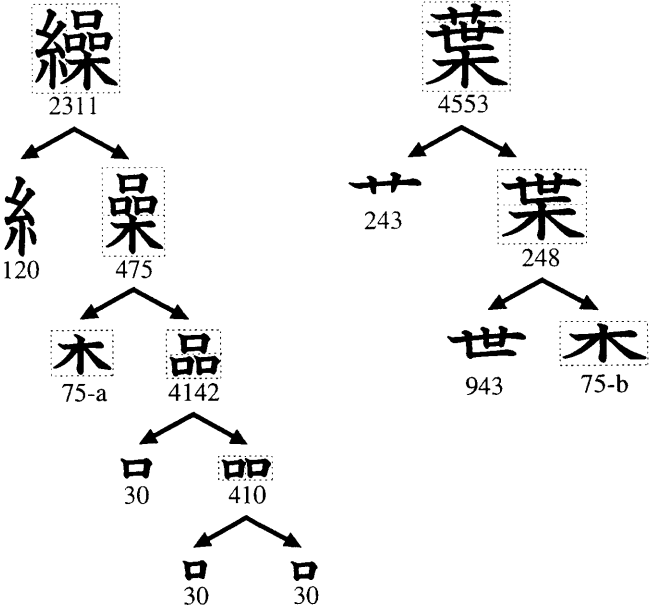


図 3.8 基本部分字形の決定
Fig. 3.8 Decision of radical part.

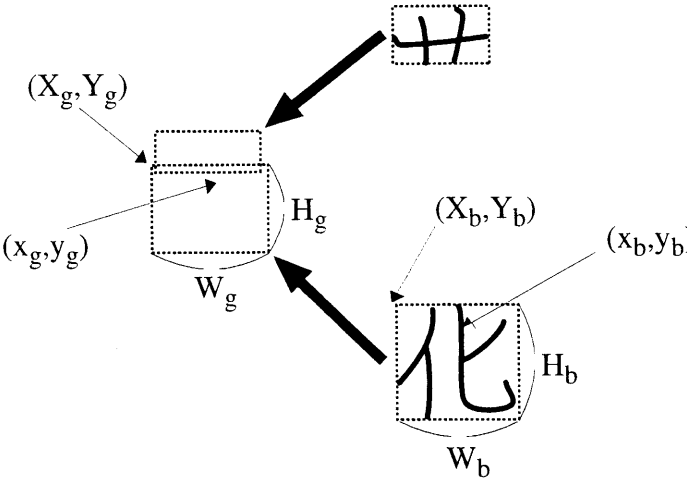


図 3.9 部分字形の配置
Fig. 3.9 Arrangement of the radical parts.

3.4.5 仮名文字の扱い

本章では、主に漢字を生成対象としているが、仮名文字は日本語文書に欠かせないものである。しかも、仮名には漢字以上に書き手の個性が現れると考えられる。仮名文字は、漢字よりもはるかに数が少ないので、全文字を登録してもユーザへの負担はそれほど大きくない。しかし、少しでも負担を軽減するために、か行、さ行、た行は濁点付き文字のみを入力し、清音の文字は濁点を取り除くことにより生成する。また、は行は半濁点付き文字のみを入力し、清音は半濁点の除去で、濁音は半濁点の位置にか行等の他の文字から抽出した濁点を配置して生成する。この結果、平仮名と片仮名で合計 50 文字の入力を省略可能となる。

3.5 生成実験

3.5.1 手書き文字データの収集

生成実験には、次の 2 組の手書き文字データを用いた。

- セット A：ペン習字の手本をトレースして入力
- セット B：研究室の女性が筆記

この 2 組の手書き文字データから生成規則と基本部分字形を抽出して手書き風フォントの生成実験を行った。

手書き文字データの入力には、ペンタブレットを用い、座標データのサンプリング速度を 20 点/秒、空間分解能を 20.48 点/mm とした。

図 3.10 に X-Window 上に作成したデータ収集システムの画面を示す。このシステムでは、構成パターンの階層表示に従って部分字形を入力させることにより、手書き文字座標データと字形データベースを対応づけている。

入力された手書き文字には大きさのばらつきを始めとする様々な変動が含まれるが、このような変動の多くが手書き文字の個性を表すものであると考え、入力データに対する補正としては、文字中心を文字枠中心に揃える位置の正規化のみを行った。

3.5.2 出現位置属性の効果

手書き入力されたオリジナルデータ並びに出現位置属性の有無による手書き風フォントの生成結果の違いを図 3.11 に示す。生成規則および基本部分字形は、共にデータセット A か

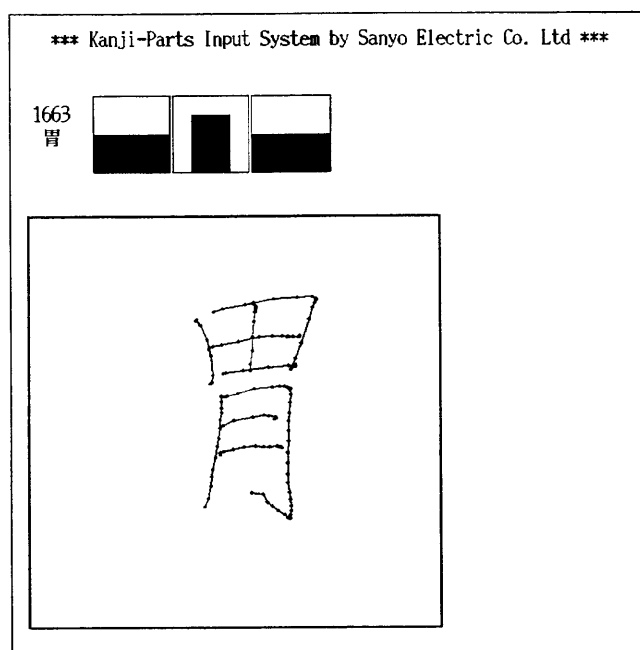


図 3.10 手書き文字データ収集システム

Fig.3.10 Input system of handwriting character data.

ら抽出したものを用いている。出現位置属性を用いない場合,「境」と「社」,「特」と「件」ではヘンとツクリに,「秋」と「季」ではヘンとカンムリに,また「処」と「夏」ではニヨウとアシに同じ部分字形が使用されてしまい,文字の生成品質を著しく低下させていることが分かる。一方,出現位置属性を用いた場合には,正しい基本部分字形が選択され,より良い生成品質の得られていることが分かる。

3.5.3 個性化

ユーザの筆記した文字から基本部分字形を抽出し,これを用いることによってユーザの個人性を反映した文字パターンの生成すなわちフォントの個性化を行う。

図 3.12, 図 3.13に示す生成結果は,それぞれデータセット A, データセット B から抽出した生成規則を用いたものである。図 3.12(a) および図 3.13(a) はデータセット A から抽出した基本部分字形を使用したものであり,図 3.12(b) および図 3.13(b) はデータセット B から抽出した基本部分字形を使用したものである。

この生成結果を観察すると,図 3.12(a) と図 3.13(a) が,よく似た文字となっており,同様に図 3.12(b) と図 3.13(b) もよく似た文字となっている。

また,図 3.14の (a) と (b) は,それぞれデータセット A と B の筆記者による筆跡そのも

言語情報システム科学
境社特件秒季処夏

(a)original handwritings

言語情報システム科学
境社特件秒季処夏

(b)results of generating without position attribute

言語情報システム科学
境社特件秒季処夏

(c)results of generating with position attribute

図 3.11 出現位置属性の効果
Fig. 3.11 Effect of position attribute.

のである。図 3.12と図 3.13の手書き風文字生成結果は、元の筆跡にもよく類似したものとなっている。

以上より、生成対象文字セット全体よりも少量の手書き文字から文字セット全体を生成するのに必要な基本部分字形を抽出して組み合わせる方法によって、ユーザの個性を反映した手書き風フォントを生成可能であることを確認した。

3.6 結言

漢字を部分字形と構成パターンで表現し、ユーザの手書き文字データから文字生成規則と基本部分字形を抽出して構築した字形データベースをもとにして手書き風フォントを自動生成する手法を提案した。

JIS X0208 規格に収録された 6,355 の漢字を分析したところ、基本部分字形 1,012 個、そ

先進的な技術と知識の応用

(a)radical parts : A

先進的な技術と知識の応用

(b)radical parts : B

図 3.12 手書き風文字の個性化（生成規則：A）

Fig.3.12 Generated results (rule : A).

先進的な技術と知識の応用

(a)radical parts : A

先進的な技術と知識の応用

(b)radical parts : B

図 3.13 手書き風文字の個性化（生成規則：B）

Fig.3.13 Generated results (rule : B).

先進的な技術と知識の応用

(a)writer : A

先進的な技術と知識の応用

(b)writer : B

図 3.14 筆跡データ

Fig.3.14 Handwriting data.

の組み合わせを示す構成パターン 105 通り、および出現位置属性が 4 種類となった。

ユーザが文字データを入力する際の負担を軽減するため、基本部分字形の抽出に用いる文字を絞り込むアルゴリズムを提案した。この結果、常用漢字 1,945 文字を生成対象とするとき、全体の 19.9%に相当する 388 文字の手書き入力から全文字を生成できることを示した。

実験により、手書き文字から抽出した基本部分字形を組み合わせで文字生成すれば、ユーザの個性を反映した文字生成が可能となる見通しを得た。

しかしながら、本手法には、以下の課題が存在している。

- 部分字形相互の位置合わせ精度を高め、生成品質を向上すること。
- 基本部分字形抽出に必要な文字数を更に削減すること。

第4章では、大域的個人性と局所的個人性に基づくアプローチによって、このような問題のないフォント生成を試みる。

第4章

大域的個人性と局所的個人性に基づく 手書き風フォント生成

4.1 緒言

本章では、ユーザが筆記した少数の文字から個人性を抽出し、これを事例ベース、すなわち多様な個人性をもつ複数の基準パターンを格納した手書き文字データベースに適用して、ユーザ固有の手書き風フォント一式を生成する。[53],[54]

本章の目標は、次の3点である。

1. ユーザの個人性を反映したフォントを生成する。
2. フォント生成の際にユーザが負担するコストをできるだけ軽減する。
3. 個人性の反映の程度を調整でき、ユーザの好みに合ったフォントを生成できるようにする。

第3点目は、ユーザの手書き文字をそのまま模倣したフォントの生成が目的ではないことを述べたものである。例えば、字の上手でないユーザは自分の筆跡をそのままデータ化することに抵抗があるだろう。すなわち、本章の目指すところは、ユーザの個人性を反映しつつ、しかもそれなりに整った美しい文字パターンを自動生成することである。

以上の目標を実現するために、できるだけ多様な複数の基準パターンを用意し、これを基底ベクトルとして少数のユーザ手書き文字データから決定した重み係数で線形結合することにより、ユーザの個人性を反映した文字パターンを生成する。

但し、線形結合だけで文字の縦横比や回転などの大域的な特徴に対応するには、非常に多くの基準パターンが必要となって実用的でない。そこで、線形結合する前に幾何学的変形を用いて、基準パターンをユーザの入力文字に概略類似させる。これによって比較的少数の基準パターンセットから多様な個人性をもつ文字パターンの生成が可能となる。

文字パターンの個人性は大きさ、縦横比、傾き、ストロークの曲率や長短などの特徴量として表せることが報告されている[55],[56]。本研究では、これらを大きさ、縦横比、傾きのように文字の概略形状を表す大域的個人性とそれら以外の特徴を表す局所的個人性に分け、上述の幾何学的変形パラメータが大域的個人性、線形結合係数が局所的個人性に相当すると考える。

4.2 システムの概要

フォント生成システムの概要を図4.1を用いて説明する。手書き文字データベースには n

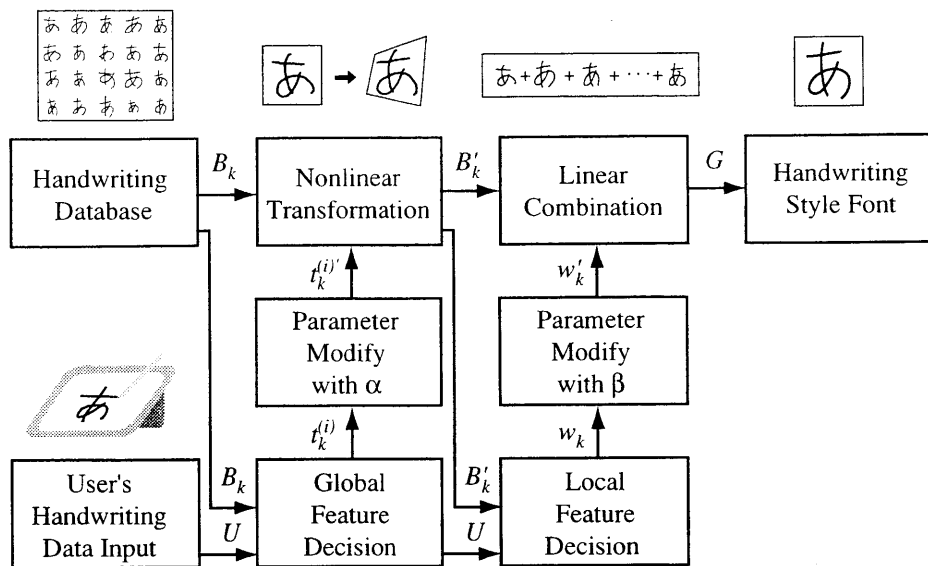


図 4.1 フォント生成システムの概要

Fig.4.1 Font generation outline.

人の筆記者から収集された基準パターン $B_k (k = 1, \dots, n)$ が格納されている。 U はユーザ入力文字であり、学習パターンとして用いる。まず、 B_k を非線形変換により幾何学的に変形し、 B'_k を得る。変形に用いる大域的個人性のパラメータ $t_k^{(i)}$ は U と B'_k との文字間距離の最小化を条件に自動決定する。次に B'_k に局所的個人性に応じた重み係数 w_k を乗じて線形結合し、手書き文字パターン G を生成する。局所的個人性のパラメータ w_k は、 G と U との文字間距離が小さくなるように自動決定する。また、ユーザはシステムが決定した個人性パラメータをそのままフォント生成に反映させるのではなく、パラメータ α と β を用いて個人性の強さを抑制することが可能である。

4.3 手書き文字パターン

4.3.1 文字データの入力

本章においても第3章と同様に文字骨格のみを文字データとして扱う。すなわち、点 $P^{(i)} (i = 1, \dots, n)$ は i 番目に標本化されたペン先位置を表し、その座標を $(x^{(i)}, y^{(i)})$ と表記する。

本研究で用いる文字データは電磁誘導方式のペンタブレット（空間分解能 0.02mm，時間分解能 20 点/秒）を使用して幅 20mm × 高さ 20mm の正方形枠内に入力したものである。

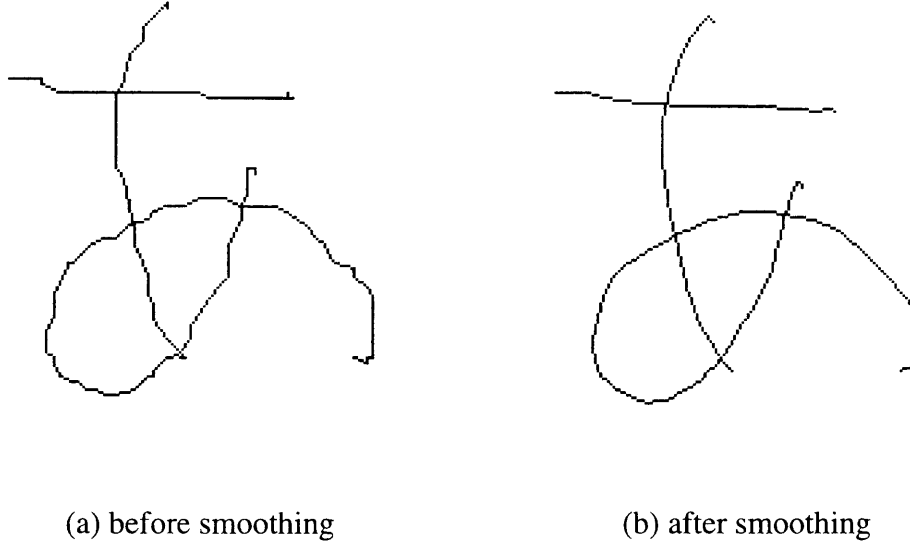


図 4.2 文字データの平滑化
Fig. 4.2 Font data smoothing.

タブレットから入力した手書き文字データには、図 4.2 (a) に示すようなペン先のぶれ等に起因するノイズが含まれる。[57] そこで、このノイズを除去するために、ペン先が同じ位置に停留する回数を重みとする移動平均処理を用いて標本点を平滑化する。

式 (4.1) は、点 $P^{(i)}$ の平滑化結果 $\tilde{P}^{(i)}$ を示すものである。図 4.2 (a) の筆記データを平滑化処理すると、図 4.2 (b) のように滑らかな筆跡が得られた。

$$\tilde{P}^{(i)} = \frac{c \cdot P^{(i-a-c)} + (a+b+1) \cdot P^{(i)} + d \cdot P^{(i+b+d)}}{a+b+c+d+1} \quad (4.1)$$

但し、

$$P^{(i-a-c)} = \dots = P^{(i-a-1)} = P^{(i-a)}$$

$$P^{(i-a)} = \dots = P^{(i-1)} = P^{(i)} = P^{(i+1)} = \dots = P^{(i+b)}$$

$$P^{(i+b+d)} = \dots = P^{(i+b+1)} = P^{(i+b)}$$

4.3.2 文字間距離

次に手書き文字の個人性を客観的に評価する指標として、式 (4.2) で文字パターン C_j と C_k の文字間距離 $D(C_j, C_k)$ を定義する。

$$D(C_j, C_k) = \min_{x_p, y_p} \left(\frac{1}{n} \sum_{i=1}^n ((x_j^{(i)} - x_k^{(i)} - x_p)^2 + (y_j^{(i)} - y_k^{(i)} - y_p)^2) \right) \quad (4.2)$$

これは、 C_j , C_k の対応する点のユークリッド距離の 2 乗平均を個人性に関係のない平行移動成分 (x_p, y_p) について最小化したものである。本研究では、距離 $D(C_j, C_k)$ の値が小さ

いほど C_j と C_k は類似しており、個人性の一致が強いとみなす。平行移動成分 (x_p, y_p) を求める処理は2つの文字パターンができるだけうまく重なるようにずらす操作に対応し、最小2乗法により容易に決定できる。

距離 $D(C_j, C_k)$ の計算には C_j, C_k の筆順およびデータ点数が一致しなければならない。そこで、ストローク対応決定法 [58],[59] を用いて C_j と C_k の筆順を一致させる。また、データ点数の一致にはストローク間で点数一定かつストローク内で点間隔一定とする正規化を用いる。

4.4 個人性を表すパラメータ

4.4.1 大域的個人性パラメータ

本研究では、幾何学的変形を用いて基準パターンの概略形状を入力パターンに類似させ、大域的個人性を生成結果に反映させる。式 (4.3) で表されるアフィン変換は、代表的な幾何学的変形処理で、拡大縮小、回転、傾斜、平行移動の各処理を同時に実行する。

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & 0 \\ r_{21} & r_{22} & 0 \\ t_x & t_y & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (4.3)$$

しかしながら、手書き文字には尻すぼみの形状のようにアフィン変換では表現できない個人性をもつものがある。そこで、本研究では文字枠方形を任意の四角形に変形する非線形変換を用いる。図 4.3 を用いてこの非線形変換を説明する。変換前の文字データ点 P から文字枠の各辺に下ろした垂線と文字枠との交点を $P_{AB}, P_{BC}, P_{CD}, P_{DA}$ とし、これらが、辺 AB , 辺 BC , 辺 CD , 辺 DA を内分する比率が変換後も保存されるとすれば、変換後のデータ点 P' の座標 (x', y') は文字枠 $ABCD$ の幅 r_w , 高さ r_h と文字枠 $A'B'C'D'$ の頂点座標 $(a, b), (c, d), (e, f), (g, h)$ を用いて式 (4.4) で示される。式から明らかなように、この変換は非線形変換である。

$$\begin{aligned} x' &= \frac{(r_w - x)(r_h - y)a + (r_w - x)yc + xye + x(r_h - y)g}{r_w r_h} \\ y' &= \frac{(r_w - x)(r_h - y)b + (r_w - x)yd + xyf + x(r_h - y)h}{r_w r_h} \end{aligned} \quad (4.4)$$

式 (4.4) の非線形変換は文字パターンの変形と同時に平行移動も行うが、本研究では文字の平行移動を個人性とは考えない。そこで、非線形変換後の文字枠 $A'B'C'D'$ の頂点座標を各々 $(a, b), (t_k^{(1)} + a, t_k^{(2)} + b), (t_k^{(3)} + a, t_k^{(4)} + b), (t_k^{(5)} + a, t_k^{(6)} + b)$ とおいて平行移動成分 a, b

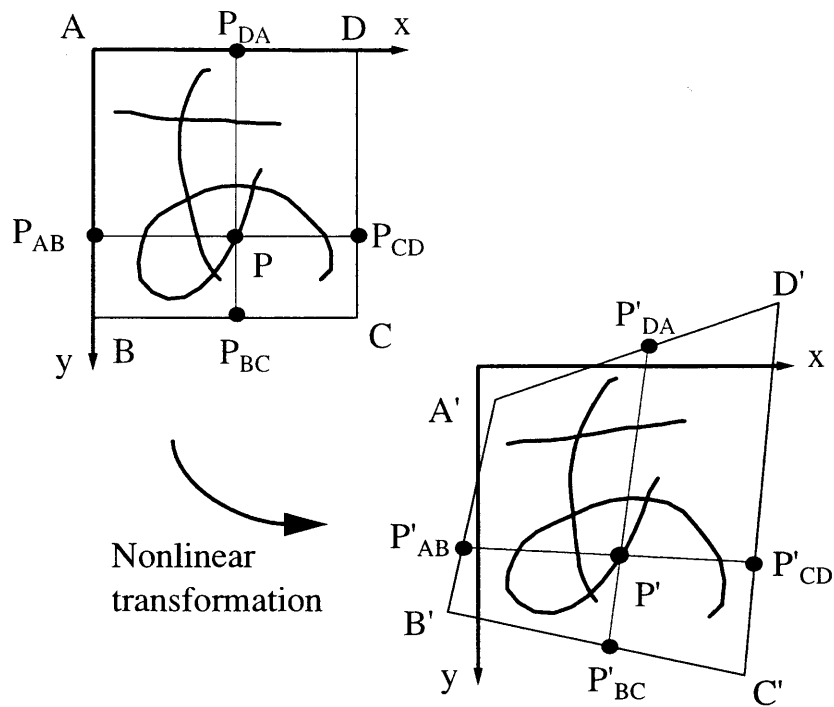


図 4.3 非線形変換

Fig.4.3 Nonlinear transformation.

を分離し、 $t_k^{(i)} (i = 1 \dots 6)$ を大域的個人性を表すパラメータとする。パラメータの決定には、変換前の文字パターンと変換後の文字パターンの文字間距離を最小とする最小2乗法を用いる。

この非線形変換の有効性を確かめるため、次の予備実験を行った。

[予備実験 1] 図 4.4(a) に示す 20 人分の手書き文字 C_j から No.20 の C_{20} を目標パターンに選び、No.20 以外の文字パターン $C_k (k = 1, \dots, 19)$ を $D(C_{20}, C'_k)$ が最小となるように非線形変換して図 4.4(b) の非線形変換結果 C'_k を得た。

各文字パターンの下に示す C_{20} との文字間距離は、この非線形変換によって小さくなっている。

また、19 名の被験者による主観テストを実施したところ、非線形変換の前よりも後の方が C_{20} に似ているとする回答は 92.0% と高率であった。以上より、非線形変換処理が有効であると認められた。

1 あ 252	2 あ 248	3 あ 154	4 あ 267	5 あ 213
6 あ 154	7 あ 425	8 あ 326	9 あ 251	10 あ 309
11 あ 245	12 あ 360	13 あ 587	14 あ 348	15 あ 531
16 あ 394	17 あ 344	18 あ 407	19 あ 564	20 あ 0

(a) 20 Variations of the Hiragana character 'あ' included in handwriting database

1 あ 66	2 あ 176	3 あ 42	4 あ 176	5 あ 151
6 あ 66	7 あ 203	8 あ 218	9 あ 110	10 あ 84
11 あ 95	12 あ 134	13 あ 280	14 あ 167	15 あ 285
16 あ 97	17 あ 139	18 あ 245	19 あ 155	

(b) Results of nonlinear transformation in fig. (a),
goal : character No.20 in fig. (a)

あ 18

(c) Results of linear combination in fig. (b),
goal : character No.20 in fig. (a)

図 4.4 非線形変換処理および線形結合処理の効果

Fig.4.4 Results of nonlinear transformation and linear combination.

4.4.2 局所的個人性パラメータ

局所的個人性を文字パターン生成結果に反映するための線形結合処理を式(4.5)で定義する．ここで、 $P_k^{(i)}$ は非線形変換後の基準パターン B'_k の第 i 番目の点、また $\hat{P}^{(i)}$ は生成結果 G の第 i 番目の点である．

$$\hat{P}^{(i)} = \frac{1}{N} \cdot \sum_{k=1}^n w_k \cdot P_k^{(i)} \quad (4.5)$$

$$\text{但し, } N = \sum_{k=1}^n w_k$$

また、式(4.5)の線形結合における重み係数 w_k を式(4.6)で与える．これは、入力パターン U と B'_k との文字間距離 $D(U, B'_k)$ が小さいほど B'_k の影響が生成結果に強く現れるようにするものである．ここで、 p は分母 = 0 による演算エラーやオーバーフローを防止する定数、 q は距離 D に対する感度を制御する定数である．実験では $p = 1$ 、 $q = 3$ とした．

$$w_k = \frac{1}{(p + D(U, B'_k))^q} \quad (4.6)$$

(p, q 定数: $p > 0, q \geq 1$)

線形結合処理の有効性を確かめるため、次の予備実験を行った．

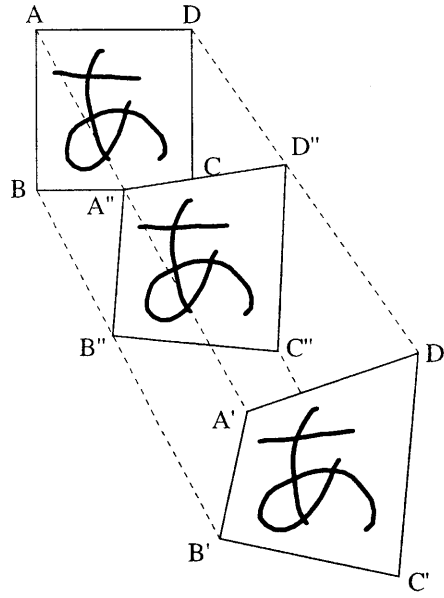
[予備実験 2] 予備実験 1 で得られた 19 個の非線形変換済パターンを線形結合して G を得た．図 4.4(c) に生成結果 G と目標パターンとの文字間距離を示す． G の目標パターンに対する距離は、図 4.4(b) のどれよりも小さいものであった．

次に、図 4.4(b) と (c) を合わせた 20 個のパターンから、最も G_{20} に似ているものを選ばせる主観テストを予備実験 1 と同じ被験者に実施した．結果は、89.5%とかなり高い割合で図 4.4(c) の線形結合結果が選ばれた．以上より、線形結合処理が有効であると認められた．

また、予備実験 1 ならびに予備実験 2 の結果は、文字間距離を用いた類似性の判定が主観テストの結果とほぼ一致し、有効であることを示している．

4.4.3 大域的個人性および局所的個人性を抑制するパラメータ

ここまでに述べた手法は、基準パターンセットごとに大域的個人性のパラメータ 6 個、局所的個人性のパラメータ 1 個、 n セットの基準パターンに対して計 $7n$ 個のパラメータを決定し、入力文字の個人性をできる限り生成結果に反映しようとするものである．しか

図 4.5 大域的個人性のパラメータ α による制御Fig.4.5 Modifying global features with α .

し，ユーザは自分の文字にそのまま類似した文字を望むとは限らない．そこで，大域的個人性および局所的個人性を抑制し，適度に平均化された文字パターンを生成するためのパラメータ α と β を導入する．

図 4.5 に示すように非線形変換前の文字枠 $ABCD$ と変換後の文字枠 $A'B'C'D'$ の対応する頂点を結び，この線分を $(1 - \alpha) : \alpha$ に内分する点を A'', B'', C'', D'' とすれば， α が 0 から 1 へ変化するにつれて四角形 $A''B''C''D''$ は $ABCD$ から $A'B'C'D'$ に変化する．このようにパラメータ α によって生成結果に及ぼす大域的個人性を制御することが可能である．

α によって制御された大域的個人性のパラメータ $t_k^{(1)'} \sim t_k^{(6)'}$ は，式 (4.7) で表せる．

$$\left. \begin{aligned} t_k^{(1)'} &= \alpha t_k^{(1)} \\ t_k^{(2)'} &= (1 - \alpha)r_h + \alpha t_k^{(2)} \\ t_k^{(3)'} &= (1 - \alpha)r_w + \alpha t_k^{(3)} \\ t_k^{(4)'} &= (1 - \alpha)r_h + \alpha t_k^{(4)} \\ t_k^{(5)'} &= (1 - \alpha)r_w + \alpha t_k^{(5)} \\ t_k^{(6)'} &= \alpha t_k^{(6)} \end{aligned} \right\} \quad (4.7)$$

同様にパラメータ β による局所的個人性の制御を式 (4.8) で与える．

$$w'_k = (1 - \beta)c + \beta w_k \quad (4.8)$$

$$\text{但し, } c = \frac{1}{n} \sum_{k=1}^n w_k$$

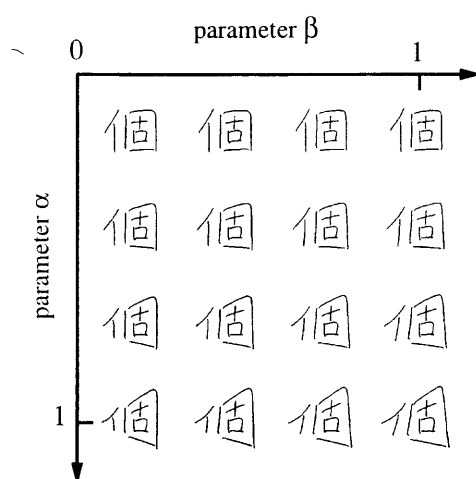


図 4.6 大域的個人性および局所的個人性制御の例
 Fig. 4.6 Example of modifying global and local features.

パラメータ β を 0 から 1 へ変化するにつれて線形結合重み w'_k は c から w_k の間を変化する。 c は w_k の平均であり、 $\beta = 0$ のときは、そのカテゴリーのすべての文字パターンを一様に足し合わせて平均化することになる。

図 4.6 は、個人性制御パラメータ α と β の値による生成結果の変化を示す例である。 $\alpha = \beta = 0$ のときは、非線形変換なし、重み一定で基準パターンを線形結合した結果である。各基準パターンのもつ個人性や歪みが打ち消し合い固有の癖がない整った文字となっている。 α を大きくするに従い、文字の左側がすぼまり右側が広がって、文字全体が台形状に変化していく。一方、 β による形状変化は α の小さいところでは目立たないが、 α の大きいところでは、文字の下側を左右に広げる変形として現れている。また、 β の影響は文字の周辺部分に強く現れ、文字中央の部分字形「古」をあまり変化させない局所的なものとなっている。

以上より、個人性制御パラメータ α と β によって、大域的個人性と局所的個人性を生成結果に反映する度合を抑制可能であることが確認できた。

4.5 生成実験

前節では、単一の学習パターンから個人性パラメータを抽出する手法を述べた。これを学習パターンと同一カテゴリーの基準パターンに適用すると生成結果にはユーザの個人性を反映できた。

本節では、複数の学習パターンから大域的個人性および局所的個人性のパラメータを決定し、これを生成パラメータとして学習パターン以外のカテゴリーを含んだフォントを生成する。そして、生成結果に反映された個人性を評価する。

4.5.1 生成パラメータの決定

手書き文字に現れる個人性は文字カテゴリーに依存し、筆記者が同一でもカテゴリーが異なれば、微妙に傾きや大きさが変化するのが一般的である。本手法では、カテゴリーに関係なく共通の生成パラメータを使用してフォントを生成する。したがって、生成パラメータは特定のカテゴリーに依存しないことが望ましい。そこで、特定の1文字ではなく、複数カテゴリーの学習パターンから生成パラメータを決定する。

まず、大域的個人性パラメータ $t_k^{(1)} \sim t_k^{(6)}$ については各学習パターンとそれに対する基準パターンの非線形変換結果との文字間距離 D を求め、その総和 D_{sum} 最小を条件に最小2乗法により決定する。 m カテゴリーのユーザ手書き文字 U が入力されたとき、距離の総和 D_{sum} は式 (4.9) で与えられる。ここで、 $C_u^{(i)} \in U$ と $C_k^{(i)} \in B_k$ は同じカテゴリー i に属する文字である。

$$D_{sum}(U, B_k) = \sum_{i=1}^m D(C_u^{(i)}, C_k^{(i)}) \quad (4.9)$$

また、局所的個人性パラメータすなわち重み係数 w_k は、式 (4.6) の D を D_{sum} で置き換えて得る。

4.5.2 実験内容

本手法によるフォント生成結果を評価するため、以下に示す条件で実験を行った。

1. 生成対象は漢字、仮名をそれぞれ4カテゴリーずつ含む文字列「個性ゆたかな文字」とする。
2. 基準パターンにはオンライン文字認識の辞書開発を目的に収集された手書き文字データ (JIS 第1水準漢字および非漢字) 20人分を用いる。データ収集に際しては、次の指示が行われた。
 - できるだけいねいに書くこと
 - つづけ字とならないこと

この基準パターンには、図 4.7に示すようにさまざまな個性を持った文字パターンが含まれている。

3. ユーザ入力文字は基準パターンとは全く独立に本実験のために 19 名の被験者から収集した。また、被験者には基準パターンと同様にていねいに書くことを指示した。図 4.8に、被験者 No.1～5 によるユーザ入力文字パターンの例を示す。
4. 入力文字カテゴリーには生成対象の「個性ゆたかな文字」のほかに漢字と平仮名による各被験者の氏名を選んだ。氏名を選んだのはユーザが最も書き慣れた文字であり、個人性抽出にふさわしいと考えたからである。

また、生成したフォントを実際に使用する場面を想定して、約 400 文字の文章を生成する実験も行った。

4.5.3 実験結果と評価

以下に本実験により生成した手書き風文字パターンを示し、客観的類似順位および主観的類似順位を用いた評価を行う。

(1) 生成結果

図 4.9は $\alpha = \beta = 1$ として、各被験者の個人性パラメータが最もストレートに結果に反映するように生成した文字パターンである。ここで、図の (a)～(d) は、入力に用いたカテゴリー「個」、「ゆ」、「個性ゆた」、各自の氏名を示すものである。(a)～(d) を比較すると被験者による生成結果の違いだけでなく、同一被験者でも入力カテゴリーによって生成結果に差が見られる。これは、同一被験者から得られる個人性パラメータが、入力文字カテゴリーによって異なることを示している。

学習パターンに漢字 1 文字を用いた (a) と平仮名 1 文字を用いた (b) を比較すると、(a) は (b) に比べて歪の発生が少なく、(b) では入力カテゴリー「ゆ」以外にかなりの歪が見られる。これは、漢字に比べ画数の少ない平仮名は字形の変動が大きく、その 1 文字に合わせた生成パラメータでは他の文字が大きく変形されてしまうことを示すものである。複数の入力文字を用いた (c) と (d) では生成パラメータが平均化され、(a)、(b) よりも歪の少ない安定した生成結果となっている。以上より、整った生成結果を得るには、入力文字パターンに漢字、平仮名を混在することが有効と考えられる。

個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字
 個性 ゆ た か な 文 字

図 4.7 基準パターンの例
 Fig.4.7 Examples of base patterns.

No.1 個性 ゆ た か な 文 字
 No.2 個性 ゆ た か な 文 字
 No.3 個性 ゆ た か な 文 字
 No.4 個性 ゆ た か な 文 字
 No.5 個性 ゆ た か な 文 字

図 4.8 手書き文字データの例
 Fig. 4.8 Examples of users' handwriting data.

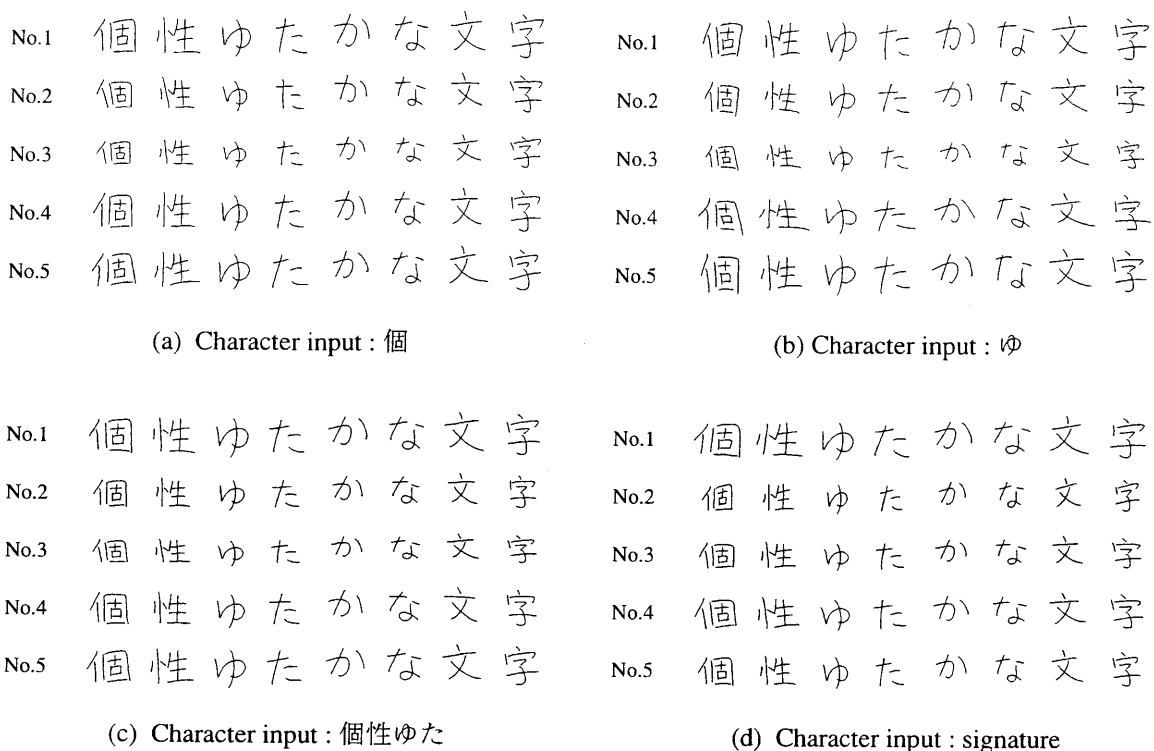
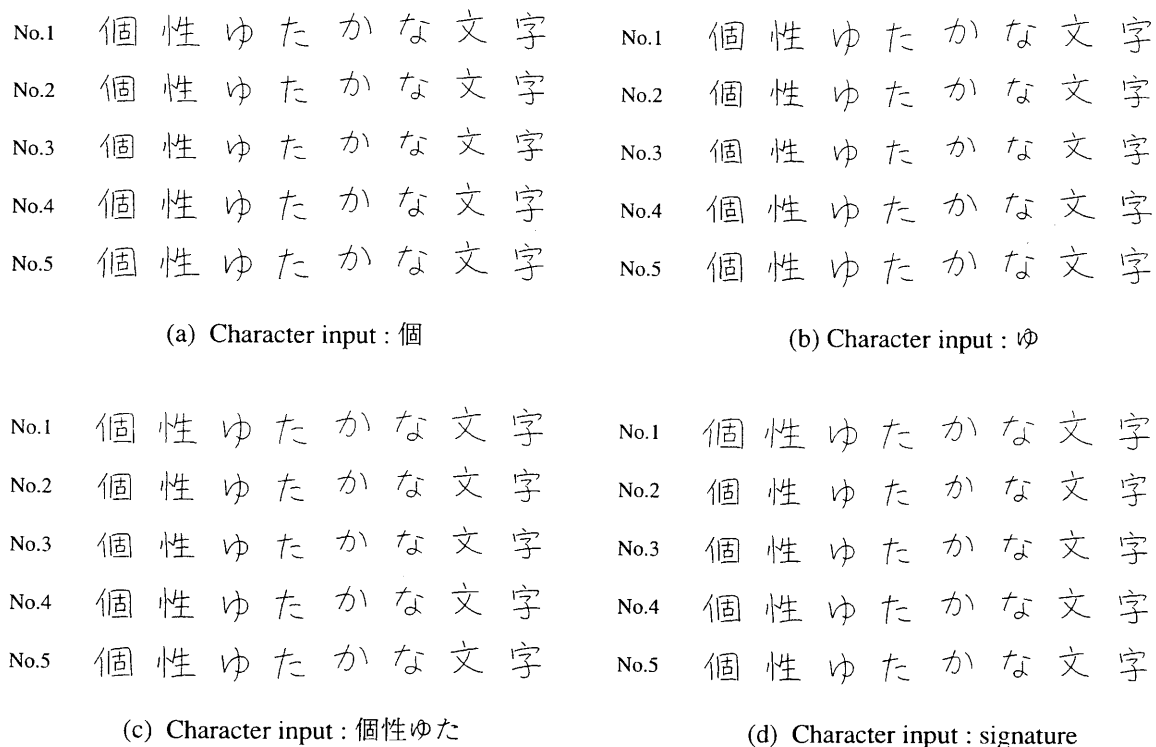
図 4.9 手書き風フォント生成結果 ($\alpha = \beta = 1$)Fig. 4.9 Results of generating handwriting style fonts ($\alpha = \beta = 1$).図 4.10 手書き風フォント生成結果 ($\alpha = \beta = 0.5$)Fig. 4.10 Results of generating handwriting style fonts ($\alpha = \beta = 0.5$).

図 4.10は $\alpha = \beta = 0.5$ とし、個人性を抑制したときの生成結果である。図 4.9に比べるとユーザによる文字パターンの違いが穏やかで、個人性の抑制された平均的な文字となっている。また、図 4.9に比べて (b) での歪発生がかなり押さえられている。このようにユーザが整った文字の生成を希望するときには、 α, β による制御が有効である。

また、被験者 No.1～No.5 の入力文字パターンに基づいて生成した文章をそれぞれ図 4.11～図 4.15に示す。生成されたフォントによって、文章から受けるイメージは、大きく異なり、フォント個性化の効果が現れている。

(2) 客観的類似順位に基づく評価

客観的類似順位とは、あるユーザの入力した手書き文字と生成された文字パターンに関する類似性の順位であり、以下に示す手順で求められる。

1. 被験者 u の筆記したカテゴリー i の手書き文字を $C_u^{(i)}$ 、それを入力パターンに用いたときのフォント生成結果を $\hat{C}_u^{(i)}$ とする。
2. 被験者 s の手書き文字パターン $C_s^{(i)}$ と被験者 t の入力に基づいて生成された文字パターン $\hat{C}_t^{(i)}$ との距離 $D(C_s^{(i)}, \hat{C}_t^{(i)})$ を $D_{s,t}^{(i)}$ とする。
3. i, s を固定して、 t について $D_{s,t}^{(i)}$ を求める。
4. $D_{s,t}^{(i)}$ の昇順に $\hat{C}_t^{(i)}$ に順位を付け、これを客観的類似順位とする。

図 4.16は、生成結果 $\hat{C}_u^{(i)}$ の自身の入力 $C_u^{(i)}$ に対する客観的類似順位が m 位以内となるものの全体に対する割合を示したものである。客観的類似順位が被験者の入力した手書き文字と無関係であれば、この割合は、直線 (e) に近くなり、類似があれば (e) から離れ、直線 (f)、すなわち 100%に近いものとなるはずである。結果は、いずれも直線 (e) から離れ、(a), (c), (b), (d) の順に (f) に近くなった。類似順位 5 位以内の割合は 81.6%～42.1%であった。

以上より、生成結果に個人性が反映されていること、生成対象に入力文字カテゴリーが含まれると類似度が高くなること、 α, β によって生成結果に反映される個人性を抑制できることが確認できた。

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザプリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

(a) $\alpha = \beta = 1$

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザプリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

(b) $\alpha = \beta = 0.5$

図 4.11 手書き風文章生成結果（被験者 No.1）

Fig. 4.11 Results of generating handwritings (testee No.1).

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザープリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

(a) $\alpha = \beta = 1$

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザープリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

(b) $\alpha = \beta = 0.5$

図 4.12 手書き風文章生成結果（被験者 No.2）

Fig. 4.12 Results of generating handwritings (testee No.2).

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザープリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

$$(a) \alpha = \beta = 1$$

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザープリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

$$(b) \alpha = \beta = 0.5$$

図 4.13 手書き風文章生成結果（被験者 No.3）

Fig. 4.13 Results of generating handwritings (testee No.3).

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザープリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

$$(a) \alpha = \beta = 1$$

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザープリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

$$(b) \alpha = \beta = 0.5$$

図 4.14 手書き風文章生成結果（被験者 No.4）

Fig. 4.14 Results of generating handwritings (testee No.4).

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザープリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

$$(a) \alpha = \beta = 1$$

プリンタの高解像度化、半導体技術の進歩によるメモリの大容量化やマイクロプロセッサの高速化、さらに電子機器向け高品位フォントの開発が行われ、日本語ワードプロセッサやパソコン用のプリンタで、邦文タイプライタに匹敵する美しいフォントが利用可能となった。この結果、邦文タイプライタはその使命を完全に終えることとなった。1980年代半ばに登場したPDL（ページ記述言語）ベースのレーザープリンタは、アウトラインフォント技術に基づく高品位フォントを扱う機構を標準で備えていた。しかし、データフォーマットが非公開だったため、利用者がフォントを追加することは難しかった。ところが、GUI（グラフィカルユーザインタフェース）ベースのパソコンOSが、アウトラインフォントを標準でサポートするようになると、そのフォーマットも公開され、サードパーティーのフォントベンダーが独自に開発したフォントも流通するようになった。

$$(b) \alpha = \beta = 0.5.$$

図 4.15 手書き風文章生成結果（被験者 No.5）

Fig. 4.15 Results of generating handwritings (testee No.5).

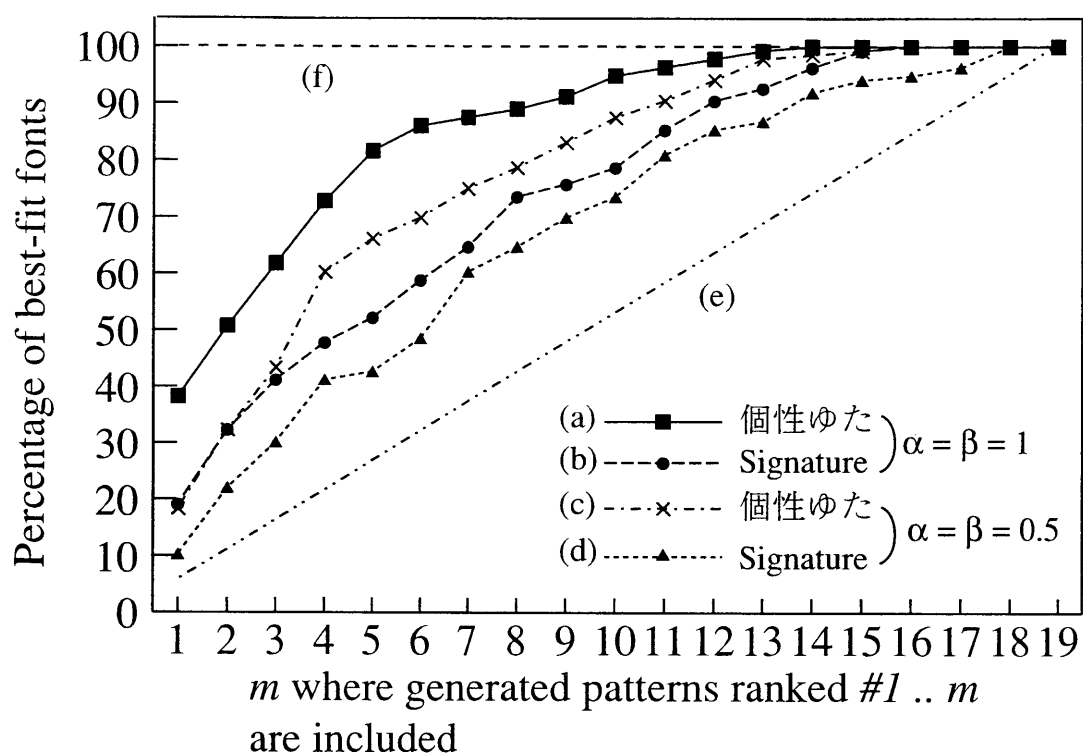


図 4.16 客観的類似順位 m 位以内に入る生成結果の割合
 Fig. 4.16 Ratio of objectively ranked fonts to best-fit fonts.

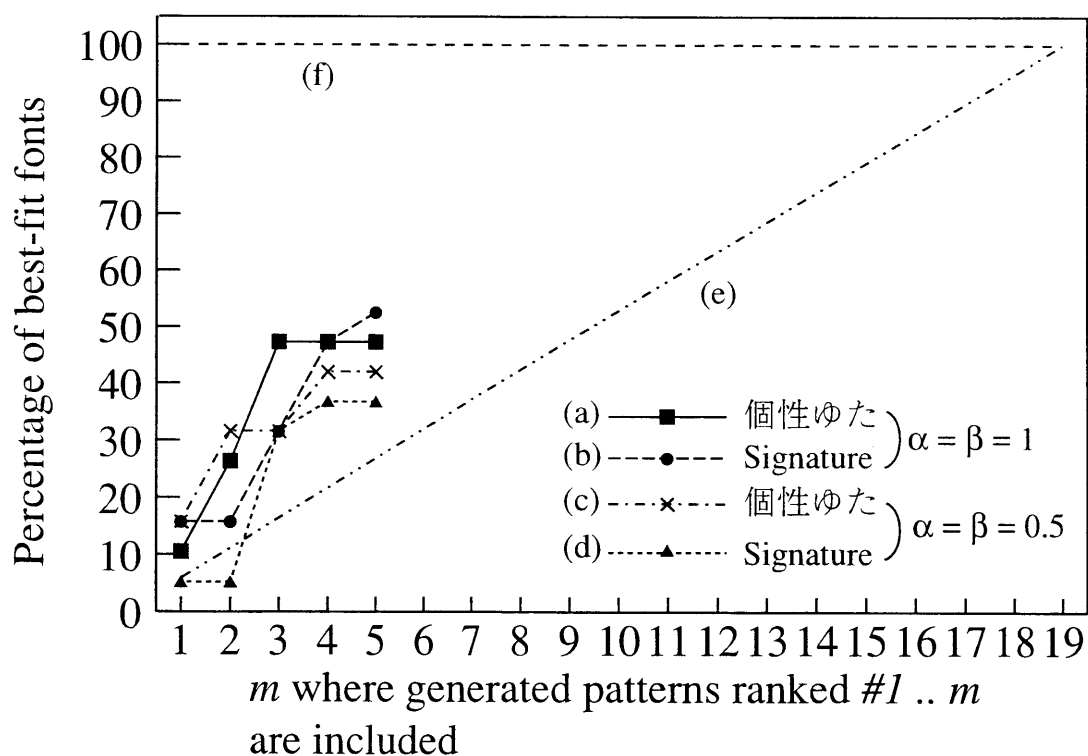


図 4.17 主観的類似順位 m 位以内に入る生成結果の割合
 Fig. 4.17 Ratio of subjectively ranked fonts to best-fit fonts.

(3) 主観的類似順位に基づく評価

主観的類似順位とは、手書き文字入力と生成結果との類似度を主観テストに基づいて決定した順位である。主観テストは以下の条件で実施した。

1. 被験者は、入力文字パターンを筆記した19名とする。
2. 被験者には、各被験者の個人性パラメータを用いて生成した19人分の文字パターン「個性ゆたかな文字」を提示する。
3. 被験者に次の指示を行い、生成結果に順位を付けてもらう。これを主観的類似順位とする。

「以下の文字パターンの中には、あなたの筆跡から個性を抽出して生成したものが含まれています。あなたの筆跡に最も近いと思うものから順に上位5位までの番号を回答欄に記入してください。」

先の客観的類似順位が文字ごとに求めたものであるのに対し、この主観テストは文字列全体の類似順位を求めたものである。また、主観によってすべてに順位をつけるのは困難であると判断し、5位までに限定した。

主観テストの結果を図4.17に示す。(a)~(d)は直線(e)と(f)の間を取り、類似順位5位以内の割合は52.6%~36.8%であった。以上より、被験者による主観テストからも生成結果に個人性が反映されていることが認められた。

4.6 結言

本章では、少数のユーザ手書き文字から個人性パラメータを抽出し、これを複数の基準パターンに適用してユーザの個人性を反映した手書き風フォント一式を生成する手法を提案した。この手法によれば、大量の文字入力や複雑なルール作成は不要であり、ユーザの負担するコストは非常に小さくてすむ。19名の被験者による実験を行い、客観的類似順位および主観的類似順位を用いて評価した結果、文字パターン生成結果にユーザの個人性を反映していることが認められた。また、個人性を抑制するパラメータを用いると、個人性の強い文字パターンから癖のない平均的な文字パターンまでユーザの好みに合わせて生成結果を変更できることを示した。

本章の手法によれば，ユーザは少なくとも 1 文字を入力するだけで自分の個性を反映した手書き風フォント生成できるため，第 3 章で提案した手法と比較してユーザの負担を大幅に軽減可能である．

本手法の生成結果は手書き文字データベース内の基準パターンに依存する．少量の基準パターンセットから幅広い個人性を反映した文字パターンを生成できることが望ましいが，最適な基準パターンセットを獲得する手法については今後の課題としたい．

第 5 章

結論

5.1 研究結果の概要

本研究では、事例ベースに基づいて、ユーザの個人性を反映した固有のフォントを自動生成する手法を提案した。また、文字間距離に基づく客観テストと被験者による主観テストを行い、ユーザの入力した文字に類似した手書き風フォントが生成できることを示した。

最近では、文書の作成にパーソナルコンピュータや日本語ワードプロセッサを用いることが、すっかり定着し、個人でも商業印刷並の出力結果を得ることが可能となっている。ところが、このような出力は、ユーザにかかわりなく横並びの没個性なものになりがちである。年賀状の宛名や文面に筆文字フォントが好んで用いられるのは、個性あふれる手書き文字への回帰現象と考えることもできる。ネットワーク社会の急速な進展によって、電子メールや WWW によるコミュニケーションが主流となっても発信者の温もりを伝える手書き文字への需要は高いであろう。

しかし、自分の筆跡そのものをデータ化しようとする、日常文書での使用に限定しても常用漢字等 2,000 文字以上の筆記が必要である。これでは、ユーザへの負担が大き過ぎる。そこで、本研究では、ごく一部の文字を筆記するだけで必要な文字すべてを含むフォント一式を自動生成できる手法を提案したのである。

第 2 章では、まず初めに本研究の背景として、活字や写植文字盤といったアナログフォントから電子的に文字形状を記憶するデジタルフォントに至るフォント技術とフォントの制作手法について述べた。ユーザ専用の個性的なフォントの生成を試みる本研究は、計算機処理によって自由自在に加工できるベクトルフォント技術が登場した結果、可能となったものである。

次に文字パターンの自動生成に関する従来の研究について述べた。ユーザの手書き文字に類似した文字パターンを生成しようとする研究は、従来から行われているが、少量の手書き文字をもとに個人性を反映したフォント一式を自動生成しようとする研究は、あまりなかった。

第 3 章では、事例ベースに基づく個性的なフォント生成手法の一つとして、漢字の構造的特徴を利用する方法、すなわちユーザの手書き文字から基本部分字形を抽出し、それを組み合わせて漢字を生成する方法を提案した。JIS X0208 規格に収録された 6,355 の漢字を基本部分字形に分解したところ、1,012 個の基本部分字形と、その組み合わせを示す 105 通りの構成パターン、および 4 種類の出現位置属性に整理することができた。また、基本部分字形の収集に用いる漢字を基本部分字形を多く含むものから順に選択するアルゴリズム

ムの開発を行った。この結果、常用漢字 1,945 文字を生成する場合、ユーザが筆記しなければならない漢字を 388 文字すなわち生成対象全体の 19.9%にまで削減し、ユーザの筆記労力を軽減することができた。生成実験の結果、ユーザごとに基本部分字形を取り替えれば、個性的な手書き風文字を生成することができることを示した。

第 4 章では、第 3 章で提案した部分字形組み合わせを用いた手書き風フォント生成法よりも、更に少量の手書き文字入力から個性的フォントを生成可能な手法について述べた。従来の手書き文字生成法が、ただ 1 組の基準パターンを変形するものであるのに対し、この手法は、複数の多様な筆跡を集め、基準パターンに用いるものである。ユーザは、少なくとも 1 文字を入力するだけで、大域的個人性を制御する非線形変換係数と局所的個人性を制御する線形結合重みを自動決定し、自分だけのフォントを生成できる。また、2 つの個人性を個別に調整するパラメータを導入して、個人性の強い文字パターンから癖のない平均的な文字パターンまで、ユーザによる個人性制御も可能とした。文字間距離に基づく客観的評価と 19 名の被験者による主観的評価により、手書き風フォント生成結果にユーザの個人性が反映されていることを確認した。

5.2 今後の課題

部分字形組み合わせによる手書き風フォント生成手法は、ユーザの筆跡そのものをフォントデータ化することに比べれば、ユーザへの負担をはるかに小さくできる。また、ユーザの筆跡から抽出した部分字形を直接生成に用いるため、ユーザの個性を強く反映した手書き風フォントの生成が可能である。しかし、実用化を目指すには、次の課題を解決することが必要である。

- 部分字形相互の微妙な位置ずれを除去し、生成品質を向上させること。
- 基本部分字形抽出に必要な文字数を更に削減し、ユーザへの負担を軽減すること。

大域的個人性と局所的個人性に基づく手書き風フォント生成法では、その生成結果が手書き文字データベース内の基準パターンに大きく依存する。少量の基準パターンセットから幅広い個人性を反映した文字パターンを生成することが望ましいが、最適な基準パターンセットを獲得する手法については今後の課題である。

更に、本研究では、文字の骨格データのみを扱ってきたが、筆やペン等さまざまな筆記具をシミュレートするには、骨格データに輪郭パターンを貼り付けたり、ストロークのにじみやかすれを表現する必要がある。また、生成したフォントをパーソナルコンピュータ

などで使用するには、TrueType や Type1 といった標準フォントフォーマットへ変換するコンバータの開発も必要である。

謝辞

本論文をまとめるに当たり，本研究の全過程を通じ，終始ご親切な御指導を賜りました岐阜大学工学部応用情報学科池田尚志教授に心から感謝の意を表します。

本研究を開始する機会を与えて頂いた三洋電機株式会社常務取締役研究開発本部長桑野幸徳氏，三洋インフォメーションビジネス株式会社亀田勇社長，同商品開発センター武田昭二所長に心から感謝致します。

日頃御指導いただく三洋電機株式会社ハイパーメディア研究所鈴木治所長並びに同通信システム研究部富田義数部長，同ヒューマンインタフェース研究室堀井洋主管研究員に深謝致します。三洋電機株式会社研究開発本部豊倉完治主管企画員には，本研究の随所で惜しみない御援助を頂きました。本当にありがとうございました。

有益な御討論，御助言を頂いた岐阜大学地域科学部地域政策講座應江黔助教授並びに岐阜大学工学部応用情報学科兵藤安昭助手に感謝します。

文字認識研究用手書き文字データを快く御提供くださいました三洋電機株式会社研究開発本部ハイパーメディア研究所ヒューマンインタフェース研究室岡本正義主任研究員に感謝します。また，同研究室の皆様には手書き文字データの収集，生成結果の評価実験に多大な御協力を頂きました。どうもありがとうございました。

最後に，本研究を遂行するにあたって御指導，御協力を頂きながらお名前を挙げられなかった多くの方々に感謝申し上げます。

参考文献

- [1] 紀田順一郎, “日本語大博物館,” ジャストシステム, Jan. 1994.
- [2] 日本情報科学(株) 編著, “フォントバイブル,” ラッセル社, Jan. 1995.
- [3] “新フォント関連用語集,” 財団法人日本規格協会文字フォント開発・普及センター, Mar. 1993.
- [4] アルベルティーン・ガウアー, “文字の歴史,” 原書房, Dec. 1987.
- [5] 渡辺茂, “漢字と図形,” NHK ブックス, Sept. 1976.
- [6] 白石和也, 工藤剛, 河地知木, “文字の歴史とデザイン,” 九州大学出版会, Jul. 1984.
- [7] 澤田善彦, “フォントデザイン　ーアナログからデジタルフォントまでー,” 平成2年度フォント技術動向報告会報告書, pp.127-158, Mar. 1991.
- [8] 林隆男, “書体を創る,” ジャストシステム, Nov. 1996.
- [9] 島袋徹, 酒井勝正, 荒川直人, “デジタルフォントのすべて,” 日経 EP, pp.102-115, 1988.
- [10] “特集: デスクトップパブリッシング,” 情報処理学会誌, Vol.31, No.11, pp.1482-1585, Nov. 1990.
- [11] 安本護, 武田昭二, “高品位フォント,” 三洋電機技術報告, Vol.22, No.3, pp.113-120, Oct. 1990.
- [12] 金井聡, 加藤朝美, 安本護, 西本恭博, “通信カラオケ用 MPEG プレーヤ,” 三洋電機技術報告, Vol.26, No.2, pp.26-35, Sept. 1994.

- [13] 中込清皓, “テレビの文字とテレビ朝日のテロップなどのシステム,” 平成6年度フォント技術動向報告会報告書, pp.89-97, Mar. 1994.
- [14] 伊藤裕幸, “テレビにおける文字表現,” 平成6年度フォント技術動向報告会報告書, pp.98-103, Mar. 1994.
- [15] 宮部善幸, “3DO プレーヤにおける階調フォント,” 平成6年度フォント技術動向報告会報告書, pp.79-85, Mar. 1994.
- [16] “昭和62年度高品質フォントの標準化調査研究報告書,” 財団法人日本規格協会情報技術標準化センター, Mar. 1988.
- [17] “平成元年度高品質フォントの標準化調査研究報告書,” 財団法人日本規格協会文字フォント開発・普及センター, Mar. 1990.
- [18] “平成2年度高品質フォントの標準化調査研究報告書,” 財団法人日本規格協会文字フォント開発・普及センター, Mar. 1991.
- [19] 財団法人日本規格協会文字フォント開発・普及センター編, “文字フォント開発・普及センターのあゆみ,” Mar. 1997.
- [20] 中里西克, 安本護, 豊倉完治, “部品データ共有によるアウトラインフォント圧縮の一手法,” 1995年度東海支部連合大会論文集, pp.308, Sept. 1995.
- [21] Peter Karow, “Digital Formats for Typefaces (邦題: 書体のデジタル・フォーマット),” URW Verlag, May. 1989.
- [22] “MACROMEDIA FONTOGRAPHER Version 4.1 ユーザーズガイド,” Altsys Corporation, July 1994.
- [23] 陳和明, 小沢慎治, “多様な明朝体文字の規則的な生成,” 信学論 (D-II), Vol.J72-D-II, No.9, pp.1423-1431, Sept. 1989.
- [24] 上原徹三, 鍵政秀子, 国西元英, 下井憲司, “骨格ベクトル方式による漢字フォントの形状特性,” 信学論 (D-II), Vol.J72-D-II, No.11, pp.1807-1815, Nov. 1989.
- [25] 上原徹三, 国西元英, 下井憲司, 鍵政秀子, “骨格ベクトル方式による文字形状の表現と生成,” 信学論 (D-II), Vol.J74-D-II, No.8, pp.1020-1031, Aug. 1991.

- [26] 上原徹三, 国西元英, 下位憲司, 鍵政秀子, 菊池純男, “ストローク種別に基づく漢字形状生成方式,” 情処学論, Vol.31, No.2, pp.209-218, Feb. 1991.
- [27] 長橋宏, 安居院猛, “漢字パターンの生成法について,” 信学技報, PRU80-15, pp.57-64, 1980.
- [28] 安居院猛, 中嶋正之, 長橋宏, “部分パターンの位置関係を利用した手書き漢字の表現法,” 信学論 (D), Vol.J60-D, No.12, pp.1109-1116, Dec. 1977.
- [29] 長橋宏, 安居院猛, “手書き漢字パターンの符号化について,” 信学論 (D), Vol.J61-D, No.11, pp.803-1116, Nov. 1978.
- [30] 田中哲朗, 石井裕一郎, 竹内幹雄, 和田英一, “プログラム肉付けによる複数漢字書体間のスケルトンデータの共有,” 情処学論, vol.36, No.1, pp.177-186 Jan. 1995.
- [31] 田中哲朗, 石井裕一郎, 岩崎英哉, 長橋賢児, 和田英一, “部品合成による漢字スケルトンフォントの作成,” 情処学論, vol.36, No.9, pp.2122-2131 Sept. 1995.
- [32] 奥村彰二, 前田正弘, “漢字画像から文字要素の自動抽出,” 情処学論, vol.32, No.1, pp.51-61, Jan. 1991.
- [33] 張憲栄, 真田英彦, 手塚慶一, “漢字楷書毛筆字体の計算機による生成,” 信学論 (D), Vol.J67-D, No.5, pp.599-606, May 1984.
- [34] 張憲栄, 真田英彦, 手塚慶一, “階層分解合成法による隷書体漢字の生成,” 信学論 (D), Vol.J68-D, No.8, pp.1489-1496, Aug. 1985.
- [35] 張憲栄, 季華妹, 真田英彦, 手塚慶一, “自由連筆の可能な毛筆書き平仮名の生成,” 信学論 (D-II), Vol.J76-DII, No.9, pp.1868-1877, Sept. 1993.
- [36] 塩野充, “非線形な幾何学的ひずみを用いた手書き風文字パターン生成の一手法,” 信学論 (D-II), Vol.J74-D-II, No.2, pp.209-219, Feb. 1991.
- [37] 木島裕二, 山口由紀子, 鎌田洋, “文字清書装置,” 特許公報, 特公平 7-9665, Feb. 1995.
- [38] 山崎敏範, 中村公計, 服部哲郎, “書字知識を利用する毛筆書体文字の生成,” 信学技報, AI92-11, pp.75-80, 1992.

- [39] 中村剛志, 黒田崇, 伊藤英則, 世木博久, “筆記速度のファジィ評価方法を導入した毛筆生成システムについて,” 日本ファジィ学会誌, vol.7, No.2, pp.371-379, 1995.
- [40] 中村剛志, 真野淳治, 世木博久, 伊藤英則, “毛筆フォントの掠れ・滲み処理システムについて,” 情処学論, vol.38, No.5, pp.1008-1015, May 1997.
- [41] 吉村ミツ, 吉村功, “筆者認識研究の現段階と今後の動向,” 信学技報, PRMU96-48, pp.81-90, June 1996.
- [42] 石井健一郎, “変形文字パターン発生法とその応用,” 信学論 (D), Vol.J66-D, No.11, pp.1270-1277, Nov. 1983.
- [43] 石井健一郎, “変形文字を用いた識別辞書の設計,” 信学論 (D-II), Vol.J72-D-II, No.5, pp.669-677, May 1989.
- [44] 倉掛正治, 石井健一郎, “変形文字パターンを用いた手書き文字認識系の評価,” 信学論 (D-II), Vol.J74-D-II, No.9, pp.1208-1216, Sept. 1991.
- [45] 大澤一爽, “文字の科学,” 法政大学出版局, pp.129-146, 1985.
- [46] 喜多伸之, “個人筆記モデルを用いた手書き文字認識の検討,” 画像の認識・理解シンポジウム MIRU'92, pp.247, July 1992.
- [47] 松尾守之, 近藤正三, “パターンの変動モデル,” 信学技報, PRU79-16, pp.39-46, 1979.
- [48] 松尾守之, 金子浩敏, 近藤正三, “変動解析法による文字パターンの特徴抽出ー再急降下法を用いた変動解析の一方法,” 昭 54 信学全大, Vol.5, pp.318, 1979.
- [49] ブーワット・アッタチュー, 近藤正三, “手書き過程のモデル化とその評価,” 信学技報, PRU86-15, June 1986.
- [50] 安本護, 池田尚志, 豊倉完治, “手書き風フォントの自動生成について,” 第 43 回情報処理学会全国大会論文集, Vol.2, pp.427-428, Sept. 1994.
- [51] 安本護, 池田尚志, 豊倉完治, “部分字形組合せを用いた手書き風フォントの自動生成,” 信学技報, PRU94-83, pp.87-95, 1994.

- [52] M. Yasumoto, T. Ikeda, J. Ying and K. Toyokura, "A Generation Method of Hand-written Style Font with Individuality from Example Base," International Conference on Chinese Computing '96, Singapore, System & Software, no.II.1, pp.361-368, June 1996.
- [53] 安本護, 池田尚志, 堀井洋, "非線形変換と線形結合を用いたユーザ固有の手書き風フォントの一生成法," 1996 年度電子情報通信学会ソサイエティ大会論文集, Vol.D, pp.337, Sept. 1996.
- [54] 安本護, 池田尚志, 堀井洋, "大域的個人性と局所的個人性に基づく手書き風フォントの生成," 信学論 (D-II), Vol.J80-D-II, No.11, pp.2930-2939, Nov. 1997.
- [55] 吉村ミツ, 木村文隆, "手書き文字の個性が現れる特性の機械的計測化とその解析," 信学論 (D), Vol.J63-D, No.9, pp.795-802, Sept. 1980.
- [56] 中村善一, 豊田順一, "書写技能に基づく筆跡に現われる個人性の抽出," 信学論 (D-II), Vol.J77-D-II, No.3, pp.510-518, Mar. 1994.
- [57] 小高和巳, 荒川弘, 増田功, "ストロークの点近似による手書き文字のオンライン認識," 信学論 (D), Vol.J63-D, No.2, pp.153-160, Feb. 1980.
- [58] 小高和巳, 若原徹, 増田功, "筆順に依存しないオンライン手書き文字認識アルゴリズム," 信学論 (D), Vol.J65-D, No.6, pp.679-686, June 1982.
- [59] 若原徹, 小高和巳, 梅田三千雄, "選択的ストローク結合による画数・筆順に依存しないオンライン文字認識," 信学論 (D), Vol.J66-D, No.5, pp.593-600, May 1983.

研究業績

学術研究論文

1. M. Yasumoto, T. Ikeda, J. Ying and K. Toyokura, "A Generation Method of Handwritten Style Font with Individuality from Example Base," International Conference on Chinese Computing '96, Singapore, System & Software, no.II.1, pp.361-368, June 1996.
2. 安本護, 池田尚志, 堀井洋, "大域的個人性と局所的個人性に基づく手書き風フォントの生成," 信学論 (D-II), Vol.J80-D-II, No.11, pp.2930-2939, Nov. 1997.

講演(研究会)

1. 安本護, 池田尚志, 豊倉完治, "部分字形組合せを用いた手書き風フォントの自動生成," 信学技報, PRU94-83, pp.87-95, 1994.

一般講演

1. 安本護, 池田尚志, 豊倉完治, "手書き風フォントの自動生成について," 第43回情報処理学会全国大会論文集, Vol.2, pp.427-428, Sept. 1994.
2. 中里西克, 安本護, 豊倉完治, "部品データ共有によるアウトラインフォント圧縮の一手法," 1995年度東海支部連合大会論文集, pp.308, Sept. 1995.
3. 安本護, 池田尚志, 堀井洋, "非線形変換と線形結合を用いたユーザ固有の手書き風フォントの一生成法," 1996年度電子情報通信学会ソサイエティ大会論文集, Vol.D, pp.337, Sept. 1996.

その他

1. 安本護, 武田昭二, "高品位フォント," 三洋電機技術報告, Vol.22, No.3, pp.113-120, Oct. 1990.