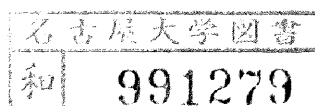


オンライン手書き文字図形認識に  
関する研究

昭和 62 年度



村 瀬 洋

## まえがき

本論文は、筆者が日本電信電話株式会社武蔵野電気通信研究所において行なってきた手書き文字・図形のオンライン認識に関する研究についてまとめたものである。

文字や図形は、人間同士の情報伝達的手段として、古くから重要な役割を果たして来た。もし、このような文字や図形を利用して人間と機械との対話が可能となれば、より円滑で効率的な人間と機械とのインタフェースが達成される。そのため、文字や図形のパターン認識の工学的な実現が望まれている。従来のオンライン手書き文字・図形認識では、その認識能力の低さ故に、種々の筆記制限が筆記者に課せられ、良好な人間と機械とのインタフェースが阻害されていた。筆記制限には、文字や図形をシステムが定めた画数や筆順で筆記すること、文字を枠内に筆記すること、シンボル毎に図形の切り出し位置を指示することなどがある。本論文の第1の主眼点は、これらの筆記制限を解除することにある。一方、人間が文字や図形を認識する際には種々の知識を利用している。高精度な機械認識を実現するには、人間の場合と同様に知識の利用は不可欠である。本論文の第2の主眼点は、知識を利用した高精度な認識系を実現することにある。

本論文は8章から構成され、まず図形部分の認識に関する基本技術の検討（2、3、4章）、次に文字部分の認識に関する基本技術の検討（5、6章）、最後に文字と図形の混在認識システムの構成（7章）から組み立てられている。

第1章では、機械による手書きの文字と図形の認識について概説するとともに、これらの研究の動向を述べ、本論文の目的を明確にする。

第2章では、基本図形（フローチャート等の図面を構成するシンボル）の筆記特性に関する分析、および画数や筆順に依存しない基本図形の認識手法について検討する。

第3章では、基本図形群とそれらを接続する直線群が複合して構成される手書き図形（例えば、フローチャート、ブロック図、論理回路図など）から基本図形や直線を自動的にセグメンテーションし、認識する手法について検討する。

第4章では、基本図形間の接続ルールを知識として利用する高精度な手書き図形認識について検討する。

第5章では、白紙紙面上あるいは罫線上に自由書式で筆記された文字列を認識する手法を検討する。

第6章では、言語処理、文字認識、文字の切り出しの上位概念から下位概念までを統合した認識系を提案し、文字認識や言語処理が文字の切り出し処理に与える効果について述べる。

第7章では、文字と図形の混在した図面から、文字や図形を自動的に分離し各々を認識するシステムについて述べる。

第8章では、本論文全編を通じての結論をまとめ、残された問題点および今後の課題について述べる。

本論文で述べるオンライン手書き文字図形の認識手法の提案により、筆記制限が少なく同時に高精度で効率の良い認識系を実現することができた。解除された筆記制限は、各基本図形をシステムが定めた筆順・画数で筆記すること、基本図形ごとに切り出し位置を指示すること、文字の

枠内筆記の制限等である。また、図形認識におけるシンボル間の接続ルールや、文字列認識における言語情報の利用など、認識における上位概念利用の有効性を示すことができた。

本論文の成果が、手書き文字・図形認識の発展に寄与し、真に情報処理の自動化や機械化に役立つこと、またパターン認識の原理的研究の一つとなることを期待する。

昭和62年4月

村瀬 洋

# 目次

第1章 序論	1
1.1 文字・図形認識の概要	1
1.2 手書き文字・図形認識の動向	4
1.3 本研究の目的	15
第2章 画数・筆順に依存しない基本図形認識法	20
2.1 緒言	20
2.2 手書きされた基本図形の性質	22
2.3 画数・筆順自由整合法	27
2.3.1 手書き情報入力	28
2.3.2 前処理	29
2.3.3 シンボルに関する辞書の記述	30
2.3.4 候補端点との対応付け	31
2.3.5 候補ストローク列の生成	33
2.3.6 候補端点の連結	40
2.3.7 DPマッチングによる相異度の計算	41
2.3.8 シンボルの認識	43
2.4 認識実験	43
2.5 考察	45
2.5.1 単純マッチング法との比較	45
2.5.2 候補ストローク列の生成数	48
2.5.3 候補端点連結の効果	51
2.6 結言	51
第3章 図形のセグメンテーションと認識法	53
3.1 緒言	53
3.2 筆記条件に関する調査	54
3.3 認識対象	58
3.4 図形中からの候補図形の抽出	59
3.4.1 手書き線図形入力	59
3.4.2 シンボルに対する候補図形の抽出	60
3.4.3 候補図形の抽出手順	61
3.4.4 シンボルの縦横比による棄却	64
3.4.5 候補端点連結処理の影響	64
3.4.6 DPマッチングによる相異度の計算	65
3.5 候補ラティス法	66
3.5.1 候補ラティスの作成	66
3.5.2 動的計画法による候補ラティスの探索	67
3.6 図形のオンライン認識実験	71
3.7 考察	71

3.7.1	認識時間	71
3.7.2	候補図形の抽出の効率化	72
3.7.3	誤認識の分析	75
3.8	結言	76
第4章	接続関係知識を導入した手書き図形の高精度認識法	78
4.1	緒言	78
4.2	接続ルールを導入した図形認識法	79
4.2.1	接続ルールを導入した認識アルゴリズム	79
4.2.2	手書き図形入力	82
4.2.3	候補図形の抽出	83
4.2.4	候補ラティスの仮の探索	85
4.2.5	接続関係の記述	86
4.2.6	接続ルールの適用	88
4.2.7	認識結果の整形出力	92
4.3	認識実験結果	98
4.4	考察	99
4.4.1	接続ルール導入の効果	99
4.4.2	候補ラティスの修正と探索の反復実行回数	106
4.4.3	認識時間	106
4.4.4	論理回路図への応用	107
4.5	結言	108
第5章	自由書式手書き文字列の認識法	109
5.1	緒言	109
5.2	認識対象	111
5.2.1	文字列データ	111
5.2.2	辞書作成データ	112
5.3	手書き文字列の性質	114
5.3.1	文字間隔の分布	114
5.3.2	分離文字の存在	116
5.4	認識アルゴリズム（候補文字ラティス法）	117
5.4.1	認識アルゴリズムの概要	117
5.4.2	手書き文字列入力	118
5.4.3	文字列の大きさの正規化	118
5.4.4	基本セグメントへの分割	119
5.4.5	候補文字の生成	120
5.4.6	候補文字の認識	122
5.4.7	候補文字ラティスの作成	127
5.4.8	候補文字ラティスの探索	128
5.4.9	分離有意文字処理	129
5.5	認識実験結果	131

5.6	考察	-----	134
5.6.1	本手法の適用範囲	-----	134
5.6.2	計算量	-----	134
5.6.3	他手法との比較	-----	136
5.6.4	分離有意文字処理の拡張	-----	137
5.7	結言	-----	139
第6章 言語情報を利用した自由書式手書き文字列の高精度認識法			-----140
6.1	緒言	-----	141
6.2	知識を導入した文字のセグメンテーションの考え方	-----	142
6.2.1	知識の役割	-----	142
6.2.2	文字認識を利用した文字のセグメンテーションの限界	-----	143
6.2.3	使用データ	-----	143
6.2.4	セグメンテーション不能文字の分析	-----	143
6.3	言語情報を利用した文字のセグメンテーションと認識	-----	148
6.3.1	候補文字ラティス	-----	149
6.3.2	言語処理	-----	152
6.3.3	反復的な候補文字ラティスの探索	-----	154
6.3.4	動的計画法による第K次最短経路の探索	-----	155
6.4	評価実験	-----	159
6.4.1	使用データ	-----	159
6.4.2	認識実験	-----	161
6.4.3	言語処理適用の反復回数	-----	166
6.5	考察	-----	167
6.5.1	従来手法（分離有意文字処理）との比較	-----	167
6.5.2	縦書きへの応用	-----	167
6.5.3	文字列の構造の階層的記述	-----	169
6.5.4	文字認識後処理法との関係	-----	170
6.5.5	文字間隔が候補文字ラティス法に与える影響	-----	171
6.5.6	候補文字認識における距離値の安定性について	-----	173
6.6	結言	-----	175
第7章 文字・図形混在図面認識システム			-----177
7.1	緒言	-----	177
7.2	システム構成	-----	178
7.3	文字と図形の分離法	-----	179
7.3.1	処理対象	-----	180
7.3.2	ストロークの大きさに着目した文字と図形の分離	-----	181
7.3.3	文字の大きさの推定	-----	181
7.3.4	ストロークの大きさによる文字と図形の分離	-----	184
7.3.5	ストロークの分類	-----	185
7.3.6	評価関数による文字と図形の分離	-----	188

7.3.7	文字と図形の分離実験	-----	190
7.4	文字と図形の回転の正規化法	-----	192
7.4.1	文字列の方向の正規化	-----	192
7.4.2	文字と図形の回転の正規化	-----	195
7.4.3	文字列認識および図形認識	-----	197
7.5	文字列認識専用処理装置	-----	200
7.5.1	専用処理装置の必要性について	-----	201
7.5.2	アルゴリズム上での処理の高速化	-----	201
7.5.3	専用処理装置のハードウェア構成	-----	203
7.5.4	専用処理装置の能力	-----	209
7.6	結言	-----	208
第8章 結論			-----211
謝辞			-----216
参考文献			-----218
付録			-----226
A.	シンボル辞書 (JIS-C6270)	-----	227
B.	文字の標準辞書	-----	232
C.	分離有意文字の一覧	-----	244
D.	言語処理における接続規則の一覧	-----	249

# 第 1 章 序論

## 1. 1 文字図形認識の概要

最近の電子計算機技術の発展は、L S I 技術やネットワーク技術に支えられ、高速化、大容量化、そして低価格化へと目覚ましい。それに伴い、計算機の応用範囲は広がり、処理対象は数値データだけでなく文字や図形の処理へと拡大されてきている。更に、利用者も計算機の専門家以外の一般大衆にまで拡大しつつある。このような背景の中で、計算機と人間との結合（マンマシンインタフェース）をいかに円滑に行なうかが情報処理における重要な課題となっている。

文字や図形は、人間同士の情報伝達的手段として、古くから利用されて来た。もし、文字や図形を数値に変換することなく、これらを直接用いることにより人間と機械との対話が可能となれば、より円滑で効率的なマンマシンインタフェースが達成される。文字や図形のパターン認識技術は、それを可能にすることができる。そのため、文字や図形のパターン認識の工学的な実現が望まれている。手書きの文字・図形認識の研究は、このような理由で近年、特に注目を浴びている。

手書きの文字・図形認識を認識対象、入力形態の観点から分類した図を図 1. 1 に示す。認識対象を分類すると、まず文字と図形に大分類できる。更に文字については、1 文字ずつ枠内に筆記された個別文字を認識する書式固定文字認識と、白紙紙面上に文字単位に区切ることなく連続的に筆記された文字列を認識する自由書式文字認識とに分けられる。本論文では文字については、後者の自由書式文字認識を扱う。一方、図形については、フローチャートや回路図面等の意味のある（人間が定義したシンボルにより構成される）図形を読み取る定形図形の認識と、漫画やイラストのような意味のはっきりしない図形を認識する不定形図形の認識に分類される。本論文では図形については、前者の定形図形認識を扱う。

文字や図形の認識技術は、その入力形態によりオフライン認識とオンライン認識の 2 つに大別される。オフライン認識では、既に筆記された文字や図形を F A X あるいはスキャナー等の光電変換により 2 次元の画像として計算機に入力し、この画像に対して認識処理を施す。そのため、過去に蓄積された図面を計算機に入力する際に有効である。オフライン型の文字認識は古くから研究がなされ [ 1 ]、最初は活字の認識を、その後手書きの片仮名や平仮名の認識を、現在では手書き漢字認識を中心に研究がなされている。オフライン型の図形認識の研究も最近になって特に活発化し、回路



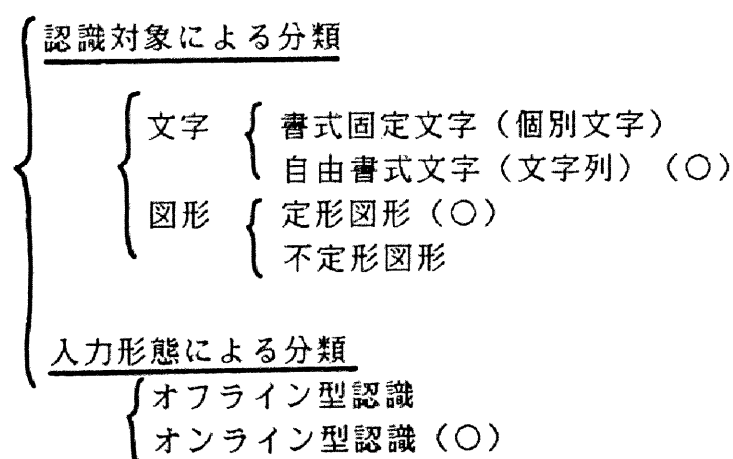
図面の認識 [2-6]、フローチャートの認識 [7]、天気図の認識 [8-9] や楽譜の認識 [10] などがなされている。一方、オンライン認識は、通常データタブレットと専用ペンを用いることにより、手書きの筆跡情報を X Y 座標の時系列として計算機に入力する。そして、入力すると同時に、この時系列に認識処理を施す。本研究では後者のオンライン型の認識について取り扱う。なお、オンライン認識の研究の歴史等については、1. 2 節で詳細に述べる。

オンライン認識はタブレットなどの特殊な装置が必要ではあるものの、オフライン認識に比較して多数の優れた点を有している。その特長を以下にまとめる。

(1) 文字や図形を入力すると同時に認識結果を得ることができる。そのため、認識結果を確認できることで筆記者に対して安心感を与えるとと共に、誤認識やリジェクトに対してもすぐに訂正することが可能である。

(2) 対話的なシステムとの整合性がよく、入力、編集、出力を一貫して行なうことができる。

(3) タブレットを利用することにより、ストローク等の筆跡情報を安定に抽出することが可能であり、認識アルゴリズムを構築する際に有利である。ちなみに、オフライン型の図形認識では細線化 (ストローク抽出の前処理として使用されることが多い) だけでも困難な処理である。



但し (○) は本論文の扱う範囲

図 1. 1 文字図形認識の分類

本論文で提案する文字・図形認識技術には以下に示すような幅広い応用が考えられる。

①ワードプロセッサ（WP）の入力に利用する。文字認識技術の導入により、手書き入力のWPが実現できる。更に、図形認識を用いることにより、図表を含む文書の清書を行なうことが可能となる。

②窓口業務の伝票入力に利用する。

③CADの入力に利用する。例えば論理回路図面の入力などが容易にできる。

④計算機のプログラムを図形入力する。例えば、フローチャートから直接計算機の言語に変換する。

本研究ではオンラインの文字・図形の認識アルゴリズムの基礎を確立し、操作性の良好な手書きによる人と機械とのコミュニケーションを可能にすることにより、人間にとって真に扱い易い将来の高度情報通信システムの実現に寄与することを狙う。

なお、図1.1では、認識対象、入力形態により、文字と図形の認識を分類したが、これらの技術は相互に密接に関係している。例えば、個別文字の認識技術は、認識用の標準パターン（辞書）を長方形や円などの図形認識における基本図形に変更することで線図形認識の一部に応用することが可能である。また、オンラインの文字列認識の技術はオフラインの文字列認識にほとんどそのまま利用することができる。そのため、本論文では図1.1に○で示した対象を仮に設定したが、本論文で提案した手法や基礎的な検討項目は、オフライン型の文字・図形認識や、更には一般の文字・図形の認識技術全般に貢献するものである。

## 1.2 手書き文字・図形認識の動向

この節では、オンライン手書き文字・図形認識の歴史について概観する。更に、この歴史を参考にして、オンライン文字・図形認識系を構築する上での問題点を検討する。表1.1に歴史を年代順に示す。オンライン文字・図形の認識の歴史は、入力装置の進歩に従って、以下に示す4期に大分類することができる。

### ①第1期：（～1963年頃）

特殊な入力装置を用い、認識手法は入力装置に依存していた。

### ②第2期：（1963年頃～1966年頃）

筆跡は主にライトペンから入力され、認識手法としては領域分割法やストロークの位置関係などを利用する手法が主流であった。

### ③第3期：（1966年頃～1980年頃）

筆跡はタブレットから入力された。ストロークの詳細な形状を利用することが可能となり、基本ストロークに分け識別木を利用する手法が主流となった。

### ④第4期：（1980年頃～現在）

筆跡はタブレットから入力されるが、入力時の筆記制限の緩和や、文字に加えて図形の認識の導入などが研究課題となっている。

以下、順追って説明する。

#### （1）第1期：（～1963年頃）

最初のオンライン文字図形認識の研究は、1957年にベル研究所のT. L. Diamondが開発したStylator [11]である。この装置は、電話会社で用いる料金請求票の自動処理を目的とした文字認識装置で、0～9までの手書きの数字をそれが書かれると同時に認識することができる。筆記者は図1.2に示す特殊な入力装置の上で数字を書く。この入力装置は、A～Gの7本のゾンデ（電極）を有している。スタイラスペンで数字を筆記すると、筆跡に対応して、各ゾンデが刺激を受ける。刺激を受けたゾンデのシーケンスによって、数字を認識する。この手法は、文字の筆記枠を適当な領域に分割し、筆跡に対応して得られる領域番号を特徴に文字を認識する手法に受け継がれている。その他、第1期の研究としては、ベル研のHarmonが開発したガイドライン法 [12] や、イリノイ大のBrownが開発した7領域法

[16] などがあるが、いずれも特殊な電極の上に文字を筆記する手法であった。またこれらの認識手法は、当然のことながら、電極の形状に依存したものであった。

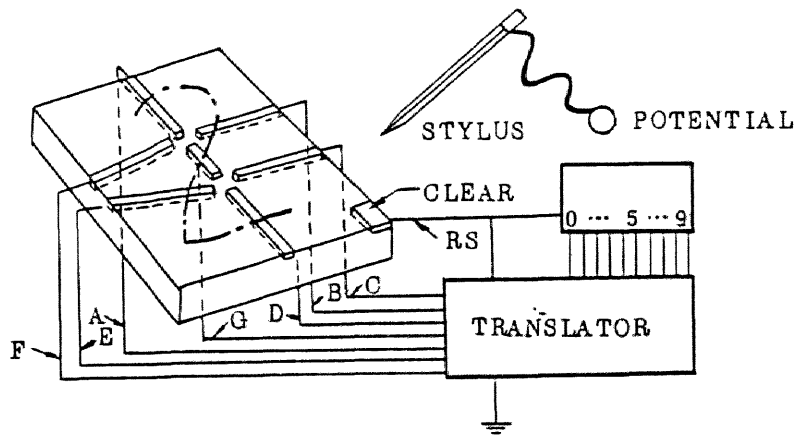


図1.2 スタイレータ

(2) 第2期：(1963年頃～1966年頃)

1960年代に入るとライトペンとCRTの組み合わせによる図形の入力手法が開発された。ライトペンを有効に利用した代表的なシステムとして、1963年にMITのSutherlandによって開発されたSKETCHPADシステム[14]がある。これは、ライトペンによって指示された箇所に直線や円を書くことができるインタラクティブな図形処理システムである(図1.3参照)。このシステムは、CAD等の、高度な編集機能等を有する図形処理の基礎となった。このライトペンの開発により2次元情報の入力が容易に行なえるようになり、その結果1960年代前半には、文字図形認識のための入力装置としてライトペンが使用された。

ライトペンによる筆跡入力により、入力装置に依存しない認識アルゴリズムの開発が可能となった。1963年にT. Marrillはストロークの幾何学的情報を利用して、仮説検定により文字を認識した[15]。例えば、「もし入力が数字の1であるならば1本の直線が存在する」などの仮説を検定することにより、文字を認識する。彼の方式では、これらの仮説を全て人間が設定する必要があり、認識対象の拡張性に劣るという問題が存在した。しかし、2つの非常に先進的な思想を含んでいる。1点目は、認識にトップダウン的なアプローチを持ち込んだことである。2点目は、複合文字の解析が可能であること、つまり文字のセグメンテーションが可能なこと

ある。この思想は、その後あまり利用されなかったが、本研究の方針を定める上で影響を与えた。

1964年にMITのTeitelmanは4領域法[17]を提案した。領域法は基本的には第1期の手法であるが、第1期では、領域の分割は電極の形状でなされていた。しかし、ライトペンの利用で領域をプログラムにより設定することができるようになり、より柔軟な領域設定が可能となった。

第2期では、ライトペンの開発により、認識アルゴリズムに柔軟性が出てきた。しかし、ライトペンは文字のような複雑な図形を入力するには操作性が悪く、またストロークの細かい形状を正確に取り込むことは困難であった。そのため、文字図形認識のためにライトペンが利用された期間は短かった。

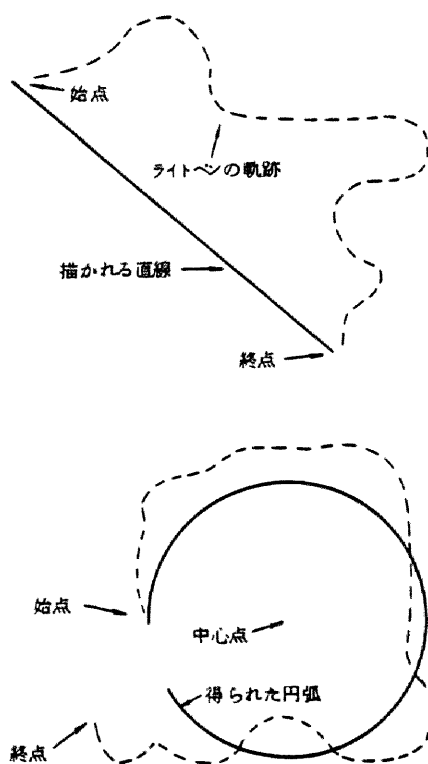


図1.3 スケッチパッドにおける直線と円弧の入力

### (3) 第3期：(1966年頃～1978年頃)

1966年にRAND社が高性能なデータタブレットを開発した。データタブレットは、その後、各社から商品化され現在に至っている。データタブレットの出現により、紙にボールペンで筆記するのと同等の感覚で、文字や図形情報が入力できるようになり、これを機会にオンライン文字図形認識の研究は本格化した。

1966年にはRAND社のG. F. Gronerによって、タブレット入力による文字認識システムが発表された[19, 20]。このシステムは、タブレットからのデータに対して行なう平滑化などの前処理と、文字の認識論理の部分(図1.4参照)から構成され、その後のオンライン手書き文字認識システムの構成法に大きな影響を与えた。また、時間情報などを利用した文字のセグメンテーション法などにも検討が加えられている。

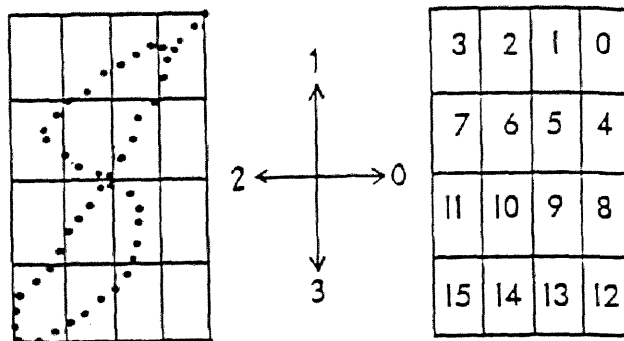
我が国では、1968年に京大の萩原らによって、ストロークの方向指数系列を特徴として、オートマトンで数字を認識する手法が提案された[25]。また、1972年に京大の柏岡らが漢字の認識を初めて行なった[26]。それ以来、国内での文字認識の対象は漢字へと変わっていった。

初期の漢字認識の研究は、多数の漢字のカテゴリーをより効率良く分類するために、ストロークを基本ストロークに分類し、その後には識別木などを利用して認識する手法が主流であった。これらの手法は構造解析的な手法と呼ばれる。代表的な研究としては、1973年に日立の寺井らが行なった手法(図1.5参照)[32, 33]や、1974年に阪大の牧野らが行なった手法[34]などがある。初期の構造解析的な手法は、最初の基本ストロークへの分類が誤ると後段の処理でそれを訂正することが困難であり、また識別木の構成が複雑になるなどの問題点を有していたが、その問題点の克服をねらい、1981年に広島大学の磯道ら[41]が、同年東京農工大の高橋ら[43]などが研究を発展させている。一方、1977年頃から、入力パターンと全ての標準パターンとの間でパターン間距離を求め、それが最小となるパターンを認識結果とする手法が拡大して来た。これらの手法はパターン整合法的な手法と呼ばれる。代表的な手法としては、NTTの荒川らは1976年にフーリエ係数を利用して距離を計算する手法[35]を、更に1978年にNTTの小高らはストロークを点近似し点間のユークリッド距離を利用してパターン間距離を計算する手法[37]を提案した。また、1982年に日電の迫江らは、方向系列のDPマッチングを利用してパターン間距離を計算する手法[46]を提案した。これらのパターン整合法的

な手法は、標準パターン辞書が大きくなり計算量も大きくなるが、処理系を構築するのが容易であり高い認識精度が得られることから、その後広く利用されるようになっていく。

第3期には多数の研究者がオンライン文字認識の研究を行ない、多数の優れた認識アルゴリズムが提案された。しかし、これらのほとんどは画数や筆順は定められた順序で入力されるものと仮定されており、更に各文字は1文字単位でセグメンテーションされているものとしている。そのため、筆記者にとっては扱い易いものではなかった。

また線図形認識では1978年にCWR大のLINらによってフローチャートを認識した例[39]があるが、これも文字認識手法をそのままフローチャートに適用したもので、フローチャートのシンボルを決まった筆順で筆記する必要があり、その上シンボルのセグメンテーション位置をその都度指示する必要があり、扱いやすいものではなかった。



1. Six 4-direction segments (left-down-right-down-left-up) encoded as 2-3-0-3-2-1.
2. Corners at positions 7 and 15.
3. Starting point at position 0.
4. Ending point at position 0.

図1.4 G. F. Gronerの方法

		CODE	SHAPE
FUNDAMENTAL STROKES	simple	A	— /
		B	↓
		C	/ /
		D	、 、
	complex	P	ㄇ ㄇ ㄇ 一
		Q	ㄷ ㄷ ㄷ
		R	ㄣ ㄣ
		S	ㄱ ㄴ ㄷ ㄹ ㄺ
		T	ㄱ
		U	ㄴ
		V	ㄷ

図 1. 5 寺井の方法

(4) 第4期：(1980年頃～現在)

第3期以前では、制限をつけて筆記された文字をいかに高い精度で認識するかが研究の主眼であった。オンラインの文字認識は人力が扱いにくいとする評判があったのも1970年代である。しかし、1980年頃になるとその研究方針が見直された。最終的により扱いやすいものを実現するためには、制限が解除された文字や図形を高い精度で認識する必要がある。

1974年に阪大の牧野らが提案した手法[34]では、構造解析的な手法を使用し、画数は固定であるが、筆順は自由であった。また、1978年にNTTの小高、増田らはパターン整合法に筆順自由を導入した(図1.6参照)[38]。しかし、この段階ではまだ画数は固定であった。1981年には、筆順自由、画数固定の筆記条件ではあるものの、NTTで手書き入力WPが試作され、オンライン文字認識の有効な応用例が示された。1982年の日電の迫江らは画数が自由な手法を提案した[46]。但し筆順は辞書に登録するため準自由である。1982年にNTTの若原らはストロークを選択的[47]に、あるいは結合規則[48]を用いて結合する手法により本質的に画数と筆順を自由にする手法を提案している。



また第4期に入ると文字認識に加えて、線図形の認識が重要な研究課題となった。1977年に T. Jarvisは図形認識を利用した図形編集システムを提案し、図形認識の応用の広さを示した [62]。また1985年日立的横山らは同様の線図形認識法 [71] を提案した。

つまり、第4期の研究では、入力時の筆記制限の緩和や、文字に加えて図形の認識の導入などが研究課題となっている。現在のオンライン文字・図形認識の研究段階は、この第4期に位置付けることができる。

本研究は、第4期に位置し、1982年にシンボルの画数筆順自由、かつセグメンテーションの不要な線図形の認識法を提案し [52-61]、更に1986年には、文字の枠内筆記の制限を解除する文字認識法を提案した [72-77]。これにより、自由書式で罫線上や、白紙上に筆記した文字列を読み取ることが可能となっている。

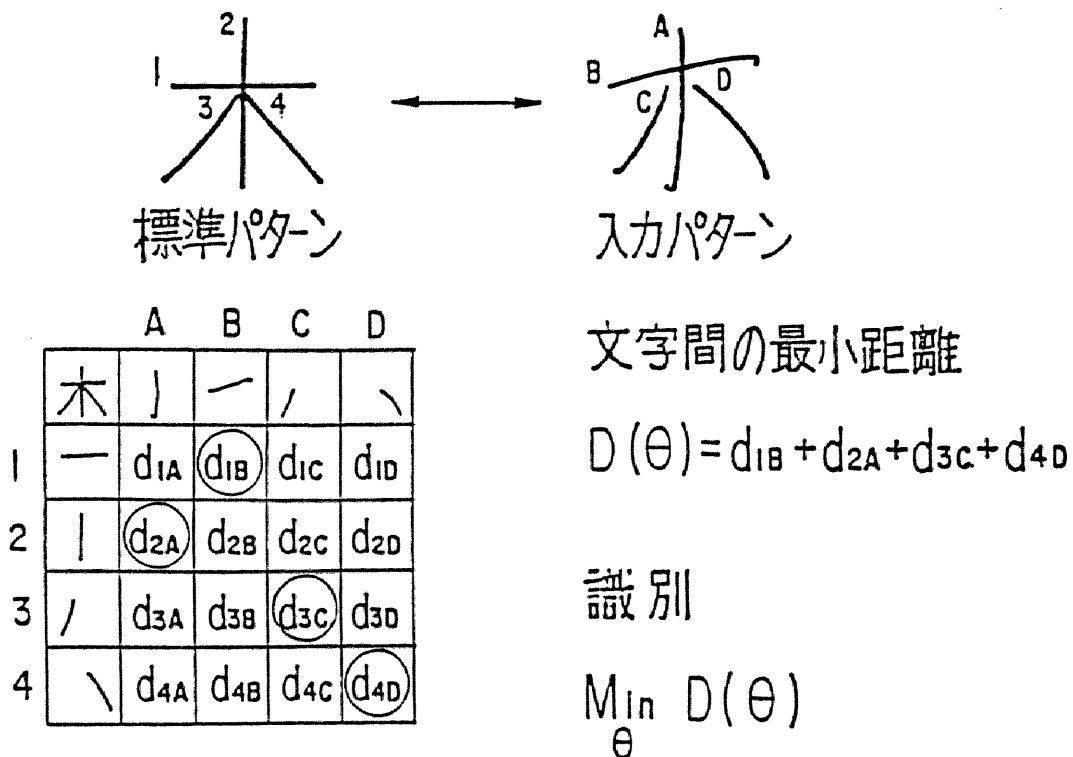


図1.6 小高、増田の方法

表 1. 1 オンライン文字認識の歴史 (A)

年代 機関  (文献)	(入力装置) 認識手法	認識対象 (文字) <図形> ○困難な対象 △普通 ×易しい	筆記制限				
			切り出し		画数筆順		
			◎自由 ○準自由 ×指示	◎自由 ○準自由 ×固定	文字	図形	
1957 <sup>(11)</sup> ベル研 Dimond	(特殊入力装置) ゾンデ法	(数字) ×	×		×		
1962 <sup>(12)</sup> ベル研 Harmon	(特殊入力装置) ガイドライン法	(英単語 10個) ×	×		×		
1962 <sup>(13)</sup> MIT Eden	基本要素への分解	(英単語) ×	×		×		
1963 <sup>(14)</sup> MIT Sutherland	(ライトペン) 編集機能を持つ	<直線、円> ×		×			×
1963 <sup>(15)</sup> BBN Marrill	(ライトペン) 仮設検定	(英数字) ×	◎		×		
1964 <sup>(16)</sup> イリノイ大 Brown	(特殊装置) 7領域法	(英数記号) ×	×		×		
1964 <sup>(17)</sup> MIT Teitelman	(ライトペン) 4領域法	(英字) ×	×		×		
1964 <sup>(18)</sup> P. M. Mermelstein	ストロークのマッチング	(英単語) △	×		×		
1966 <sup>(19,20)</sup> RAND Groner	(タブレット) 方向コード	(英数記号) <フローチャート> △	○	○	×		×
1967 <sup>(21)</sup> ピッツバーグ大 Li	(ライトペン) 方向コード	(漢字 14種) △	×		×		
1967 <sup>(22)</sup> Inforonics Nugent	(タブレット) ストロークの位相特徴	(英数字) ×	×		×		

表 1. 1 オンライン文字認識の歴史 (B)

年代 機関  (文献)	(入力装置) 認識手法	認識対象 (文字) <図形> ○困難な対象 △普通 ×易しい	筆記制限			
			切り出し		画数筆順	
			◎自由	◎自由	○準自由	○準自由
×指示	×指示	×指示	×指示			
文字	図形	文字	図形			
1969 <sup>(23)</sup> Cal. 大 Miller	(タブレット) 方向指数列のマッチング	(英数字) ×	×		×	
1969 <sup>(24)</sup> 日電 Hanaki	(タブレット) 方向指数列	(片仮名)	×		×	
1971 <sup>(26)</sup> 東大 藤崎	(タブレット) フーリエ係数	(数字) ×	×		×	
1971 <sup>(28)</sup> 京大 柏岡	(タブレット) 方向指数系列を 順序論理で解析	(漢字) ○	×		×	
1973 <sup>(29)</sup> NPGS Powers	(タブレット) 方向指数列	(数字)	×		×	
1973 <sup>(30)</sup> 日電 古川	(タブレット) 基本ストロークに分類 識別木	(漢字) ○	×		×	
1973 <sup>(33)</sup> 日立 寺井	(タブレット) 基本ストロークに分類 識別木	(漢字) ○	×		×	
1974 <sup>(35)</sup> NTT 荒川	(タブレット) フーリエ係数	(仮名、英数字) ×	×		×	
1974 <sup>(34)</sup> 阪大 牧野	(タブレット) 基本ストロークに分類 識別木	(漢字) ○	×		○	
1976 <sup>(36)</sup> 京大 藤原	(タブレット) 方向指数列DPマッチング	(漢字) ○	×		×	
1978 <sup>(37,38)</sup> NTT 小高	(タブレット) 点近似マッチング	(漢字) ○	×		○	

表 1. 1 オンライン文字認識の歴史 (C)

年代 機関  (文献)	(入力装置) 認識手法	認識対象 (文字) <図形> ○困難な対象 △普通 ×易しい	筆記制限			
			切り出し		画数筆順	
			◎自由	○準自由	◎自由	○準自由
			×指示	×固定		
			文字	図形	文字	図形
1978 (39) CWR大 Lin	(タブレット) ストロークのマッチング	<フローチャート> △		×		×
1979 (40) 富士通 加藤	(タブレット) ストロークの相対位置行列	(漢字) ○	×		×	
1981 (41) 広島大 磯道	(タブレット) 基本ストローク DPマッチング	(漢字) ○	×		×	
1981 (42) IBM Yhap	(タブレット) 基本ストローク 識別木	(漢字) ○	×		×	
1981 (43) 農工大 高橋	(タブレット) 基本ストローク 識別木	(漢字) ○	×		×	
1982 (44) IBM Tappert	(タブレット) DPマッチング	(英数字) △	◎		×	
1982 (45) 慶応大 横山	(タブレット) 基本ストローク 識別論理	(漢字) ○	×		×	
1982 (46) 日電 迫江	(タブレット) 方向指数列DPマッチング	(漢字) ○	×		○	
1982 (47-49) NTT 若原	(タブレット) ストロークの結合 DPマッチング	(漢字)	×		◎	
1982 (51) 富士通 加藤	(タブレット) 方向変化のマッチング	<直線、円、楕円> △		×		×
1982 (52-61) NTT 村瀬	(タブレット) グラフ探索	<フローチャート> ○		◎		◎

表 1. 1 オンライン文字認識の歴史 (D)

年代 機関  (文献)	(入力装置) 認識手法	認識対象 (文字) <図形> ○ 困難な対象 △ 普通 × 易しい	筆記制限			
			切り出し		画数筆順	
			文字	図形	文字	図形
1983 (63) 松下 富本	(タブレット) 基本ストローク 識別論理	(漢字) ○	×		×	
1984 (64) ノイヘル大 Ye	(タブレット) ストロークマッチング	(漢字) ○	×		○	
1984 (65) 沖 ゆるき	(タブレット) 基本ストローク 識別木	(漢字) ○	×		×	
1984 (66) シャープ 広瀬	(タブレット) 点近似マッチング	(漢字) ○	×		○	
1984 (67) 名工大 佐藤	(タブレット) DP マッチング	(漢字) ○	×		○	
1984 (69) 沖電気 矢野	(タブレット) 線解析法	<フローチャート> ○		×		×
1984 (68) NTT 児島	(タブレット)	<フローチャート> ○		◎		◎
1984 (70) 早稲田大 影沢	(タブレット) 基本ストロークに分類	(漢字) <フローチャート> ○	×	×	×	×
1985 (71) 日立 横山	(タブレット) 線分セグメント法	(漢字) <フローチャート> ○	×	◎	○	◎
1986 (72-83) NTT 村瀬	(タブレット) グラフ探索	(漢字) ○	◎		○	

### 1.3 本研究の目的

文字や図形は、人間同士の情報伝達になくてはならないものである。この文字や図形を機械に認識させることは、2つの大きな意義がある。

第1点は、人間と機械とのコミュニケーションをより円滑に行なうことが可能となるという工学的な意義である。文字や図形を機械に入力する方法は、数値やコードを直接入力するかつての方法から、仮名漢字変換やポインティングデバイス等を利用することにより直感的に情報を機械に入力する方法に変化してきた。つまり、機械に人間を適応させる時代から、人間に機械を適応させる時代へと変化してきた。その一環として、手で書いた文字や図形をそのまま機械で読み取ることが出来れば更に人間の負担が低減されるであろうという期待が生じてきた。この工学的な観点から従来の認識技術を見ると、そこには種々の問題が含まれている。従来のオンライン手書き文字・図形認識では、その認識能力の低さ故に、種々の筆記制限が筆記者に課せられ、良好な人間と機械とのインタフェースが阻害されていた。具体的な筆記制限としては、文字や図形をシステムが定めた画数や筆順で筆記すること、文字を枠内に筆記すること、シンボル毎に図形の切り出し位置を指示（セグメンテーション）することなどがある。これらの筆記制限は、筆記者に対して精神的かつ肉体的な疲労をもたらすことになる。また、各種の筆記制限は、文字・図形の認識技術の応用範囲を制限することにもなる。本研究の第1の目的は、筆記制限の少ない文字・図形認識系を構築することにある。

第2点は、人間が如何に文字や図形を読んでいるかを解明し、これを機械に代行させようとする人工知能的な立場であり、ここには学問的な意義がある。人間の認識機構を解明する研究は、実際には古くから心理学の分野や、認知科学の分野〔112〕でなされている。しかし、これらの分野では未だに、機械上にインストールすれば文字や図形が精度良く認識できるというような結果は得られていない。そのため、この立場での研究の進め方は、工学者の直感に頼らざるを得ない。ただ言えることは、人間は種々の知識（情報）を巧みに利用して文字や図形を認識していることである。文字を読み取る場合を例に取れば、単独の文字では読み取れない文字も、文章として見れば読むことができる。また、図形として見ればどこが文字の切れ目か分からない文字列も、文字の概念を導入することにより文字をセグメンテーションすることができる。これらの知識の利用は、文字の認識の場合だけでなく、人間がフローチャートのような図形を認識する際にも同様になされている。これらの知識を機械で扱えるよう

になれば、精度の高い文字や図形の機械認識が実現可能となる。本研究の第2の目的は、これらの知識を利用した、高精度な文字・図形認識系を構築することにある。

本論文は8章から構成され、まず図形部分の認識に関する検討（2、3、4章）、次に文字部分の認識に関する検討（5、6章）、最後に文字・図形の混在図面の認識システム（7章）から組み立てられている。本論文の構成を以下に示す。

第2章の「画数・筆順に依存しない基本図形の認識法」では、基本図形（フローチャート等の図面を構成する個々のシンボル）の筆記特性に関する分析、およびその認識手法について検討する。特に、入力ストローク列に対応した標準ストローク列を自動的に生成し、入力図形と標準図形との間で整合をとる画数・筆順自由整合法を提案する。この手法により、1種類の標準シンボルを登録するだけで、任意の画数と筆順で筆記された基本図形を認識することが可能となる。また、ストロークの方向と位置を用いたDPマッチングをストローク間の相異度尺度とすれば、手書き変形に強い認識が実現できることを示す。

第3章の「図形のセグメンテーションと認識法」では、基本図形（シンボル）群とそれらを接続する直線群が複合して構成される手書き図形（例えば、フローチャート、ブロック図、論理回路図など）を対象に、そこからシンボルや直線を自動的にセグメンテーションし、認識する手法について検討する。ここでは、画数・筆順自由整合法により入力図形中から各シンボルに対する候補図形を抽出する手法を検討する。またシンボルのセグメンテーションと認識とを融合した候補ラティス法の提案を行なう。この手法を利用すれば、図形全体として最適な解釈を認識結果とすることができ、セグメンテーションの高精度化が達成される。これによりシンボルの切り出し位置を指示しなければならないとする筆記時の制限が解除されることを示す。

第4章の「接続関係知識を導入した図形の高精度認識法」では、シンボル間の接続ルールを知識として利用する高精度な手書き図形認識について検討する。第3章の手法で得られた認識結果について、シンボル間の接続関係を記述し、接続関係が予め登録した接続ルールに矛盾する場合にはそれを修正する手法を提案する。この手法の導入により、包含関係を持つシンボル（フローチャート等に含まれるシンボルには、あ

るシンボルが他のシンボルの一部となるような包含関係をもつシンボルが多数ある。)のセグメンテーションや、形状が類似したシンボルの認識に対して、良好な結果が得られることを示す。

第5章の「自由書式手書き文字列の認識法」では、白紙紙面上あるいは罫線上に自由書式で筆記された文字列を認識する手法を検討する。日本語には偏とつくり等、左右に分離する文字が多数存在するため、単に図形的なまとまりだけで文字をセグメンテーションするわけにはいかない。そこで、文字のセグメンテーションと個々の文字の認識を融合した候補文字ラティス法を提案し、その手法を評価する。この手法により、分離文字に対しても高いセグメンテーション精度が得られる。これにより、文字を枠内に筆記する必要がなくなることを示す。

第6章の「言語情報を利用した自由書式文字列の高精度認識法」では、言語処理、文字認識、文字のセグメンテーションの上位概念から下位概念までを統合した認識系を提案し、文字認識や言語処理が文字のセグメンテーション処理に与える効果について述べる。文字認識の情報だけを文字のセグメンテーションに利用した候補文字ラティス法(5章の手法)では、文字が偏とつくり等に分離しそれらが別の文字になる場合には正しく認識されない場合が多い(例えば、「説明」が「説日月」と誤認識される)。それに対処するため、言語情報を文字のセグメンテーションに効果的にフィードバックする手法を提案する。この手法により、文字のセグメンテーションの高精度化が実現できること、更に、本手法は文字認識の後处理的な効果もあり文字認識率の向上にも寄与することを述べる。

第7章の「文字・図形混在認識システム」では、ストロークの長さや形状に関する性質を利用した文字・図形の分離手法、筆記時の筆点の方向特性を利用した文字列や図形の回転の正規化手法について検討する。また、手書き文字列の認識を効率的に実現するための専用処理装置について述べる。

第8章の「結論」では、本論文全編を通じての結論をまとめ、残された問題点や今後の研究の課題について述べる。



本論文で述べるオンライン手書き文字・図形の認識手法の提案により、まず、筆記制限が少ない認識系を実現することができる。解除する筆記制限は、各基本図形をシステムが定めた筆順・画数で筆記すること（2章）、基本図形ごとに切り出し位置を指示すること（3章）、文字の枠内筆記の制限（5章）等である。更に、文字や図形の認識において上位概念（知識）を利用した高精度で効率の良い認識系を実現することができる。導入した上位概念は、図形認識におけるシンボル間の接続ルール（4章）や、文字列認識における言語情報（6章）などである。

本論文では、人間が各種の知識を利用して文字や図形を認識しているように、上位概念と下位の情報とを融合して図形や文字を認識することにより、全体として高い認識精度あるいはセグメンテーション精度を達成しようとすることを基本方針としている。この基本方針は全編を通じて貫かれている。具体的な実現方法として、上位概念と下位の情報とを融合するために、ラティス表現を提案する。ラティスは、認識の一般的な問題を1次元の問題に変換し、それを2端子グラフで表現する手法であり、文字や図形の階層的な記述が可能となる。本論文では、この基本方針が、図形認識でもそして文字認識でも適用できることを示す。この基本方針を基に、本論文全体の構成を示したブロック図を図1.7に示す。

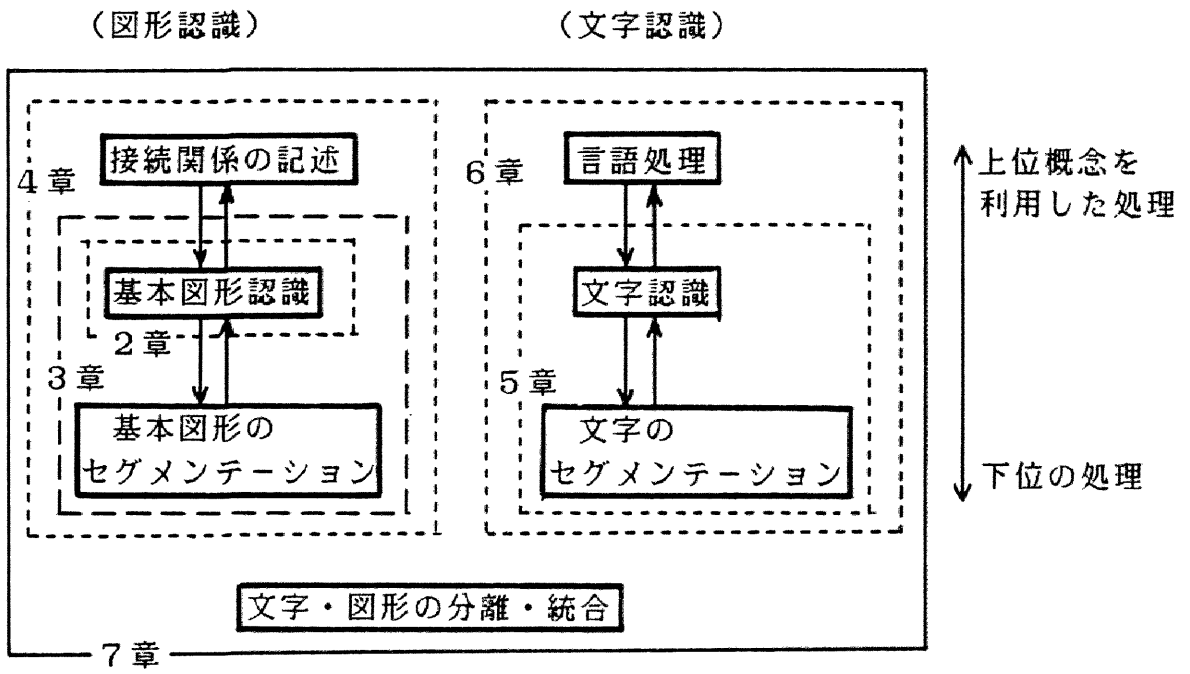


図 1. 7 本論文全体の構成

## 第2章 画数・筆順に依存しない 基本図形認識法

### 2.1 緒言

手書き線図形の認識は、手書き図面の自動清書や手書き図形による計算機との対話等、幅広い応用範囲を持つ。このため、従来より手書きされた図面をFAXから入力して認識するオフライン型の認識法[2-10]や、タブレット上に手書きしつつある図面を逐次認識するオンライン型の認識法[39, 51]について検討が進められてきた。本論文では1章で述べたようにオンライン型の認識を扱う。オンライン型の認識を採用する理由は、誤認識の発見や修正がその場で対話的に行え、また入力と編集の作業を融合しやすく、マンマシンインタフェースの観点からも有望と考えるからである。

オンライン図形認識では、タブレットから筆記順序に従って図形データを収集するため、筆順や画数に関する情報を容易に知ることが出来る。従来のオンライン図形認識アルゴリズムは、この筆順や画数に関する情報を積極的に利用していた。そのため、認識アルゴリズムは簡単になるものの、一方では、筆記者に筆順や画数に関する制限が課せらるることになった。しかし、「正方形」を手書きする場合を例に取ってみてもわかるように、図形を筆記する際には一般に決まった筆順や画数は存在しない。また、それを筆記者に覚えさせるには筆記者の負担が大きくなる。そのため従来のオンライン図形認識法は必ずしも筆記者にとって扱いやすいものではなかった。この章では、筆記制限の緩和を目指した画数や筆順に依存しない手書き図形の認識法[52, 57]について述べる。その認識法を、以下「画数・筆順自由整合法」と呼ぶことにする。

本論文で扱う図形は、文書中によく出現するブロック図やフローチャートあるいは技術資料中に出現する回路図面などとする。これらはいずれも基本図形（シンボル）とそれを接続する直線から構成されている。この章ではまず図形認識の基本にあたる個々の基本図形の認識について取り上げる。

ここで提案する画数・筆順自由整合法は以下に示す3つの特長を持つ。

(1) この手法は、入力ストローク列に対応した標準ストローク列を自動的に生成し、次に入力図形と標準図形との間で整合を取る手法であり、1種類の標準シンボルを登録するだけで、任意の画数と筆順で筆記されたシンボルを認識することが可能である。

標準ストロークの自動生成は、標準シンボルを有向グラフで表現し、このグラフの経路を探索することにより実現している。

(2) 標準ストロークを生成するために、まず入力ストロークの端点と標準シンボルの候補端点(有向グラフのノードに対応した点)との対応付けを行なう。しかし、複雑な図形を雑に筆記した場合には、この対応付けを誤る危険がある。そこで、これらに対しては候補端点の対応付けに冗長性を持たせる処理を導入する。この処理を候補端点連結処理と名付ける。

(3) 手書き変形に強い認識法を確立する為に、ストローク間の相異度(2つのストローク形状の違いを表現する値)の計算に、ストロークの方向と位置を用いたDPマッチングを採用する。

この章で提案する「画数・筆順自由整合法」を従来手法と比較する。比較の概念図を、図2.1に示す。従来手法では、各シンボルに対して1つ(または、複数)の標準ストロークを持ち、それと入力ストロークとの単純マッチングにより、シンボルを認識していた。そのため、入力時の画数や筆順は、標準ストロークと同じでなければならない。それに対し本手法では、入力の画数と筆順の変形に対応した標準ストロークを自動的に生成し、それとの間でマッチングを行なう。これにより、任意の画数と筆順に対処できる。このアプローチは、変形の大きい平仮名文字を認識するために、著者が以前提案した「変動吸収特性核を用いた認識法」[78-83]と同様の考え方に基づく。

以下、本章では、2.2節で基本図形を自由に筆記者に筆記させた際の、筆記特性の分析について述べ、2.3節で画数・筆順自由整合法のアルゴリズム、2.4節で画数・筆順自由整合法による認識実験、2.5節で画数・筆順自由整合法の考察を述べる。

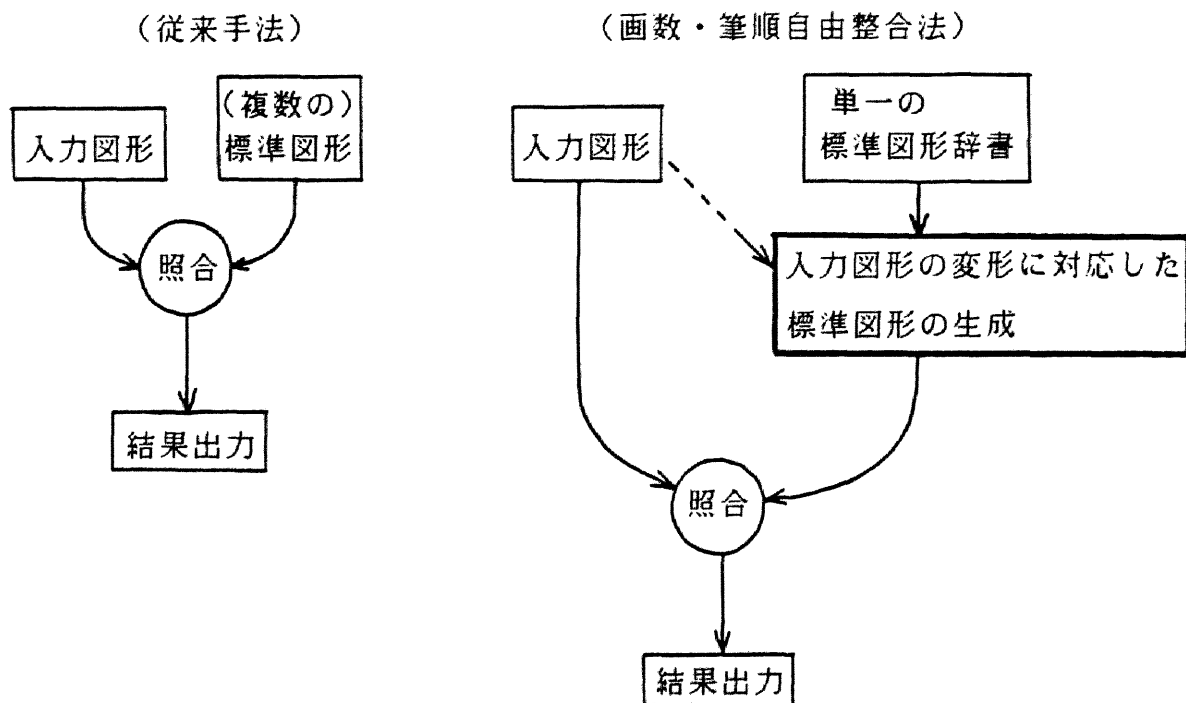


図2. 1 本手法と従来手法との比較

## 2. 2 手書きされた基本図形の性質 (アンケート調査)

シンボルを認識する為のアルゴリズムを開発する前に、実際にシンボルはどのような画数と筆順により筆記されているかの調査を行なった。対象とするシンボルを図2. 2に示す。これらのシンボルはフローチャートやブロック図のためにJIS-C6270で定められたものである。調査は、11名の研究所内の研究者を対象に図2. 3に示す用紙を配付しアンケートを行ない、その内容を分析することにより行なった。アンケートの内容は、

①フローチャートやブロック図などに含まれる基本図形はどのように筆記される可能性があるか、

②筆記用紙は、白紙あるいは方眼紙(10mm, 8mm, 6mm方眼)のどれが筆記し易いか、

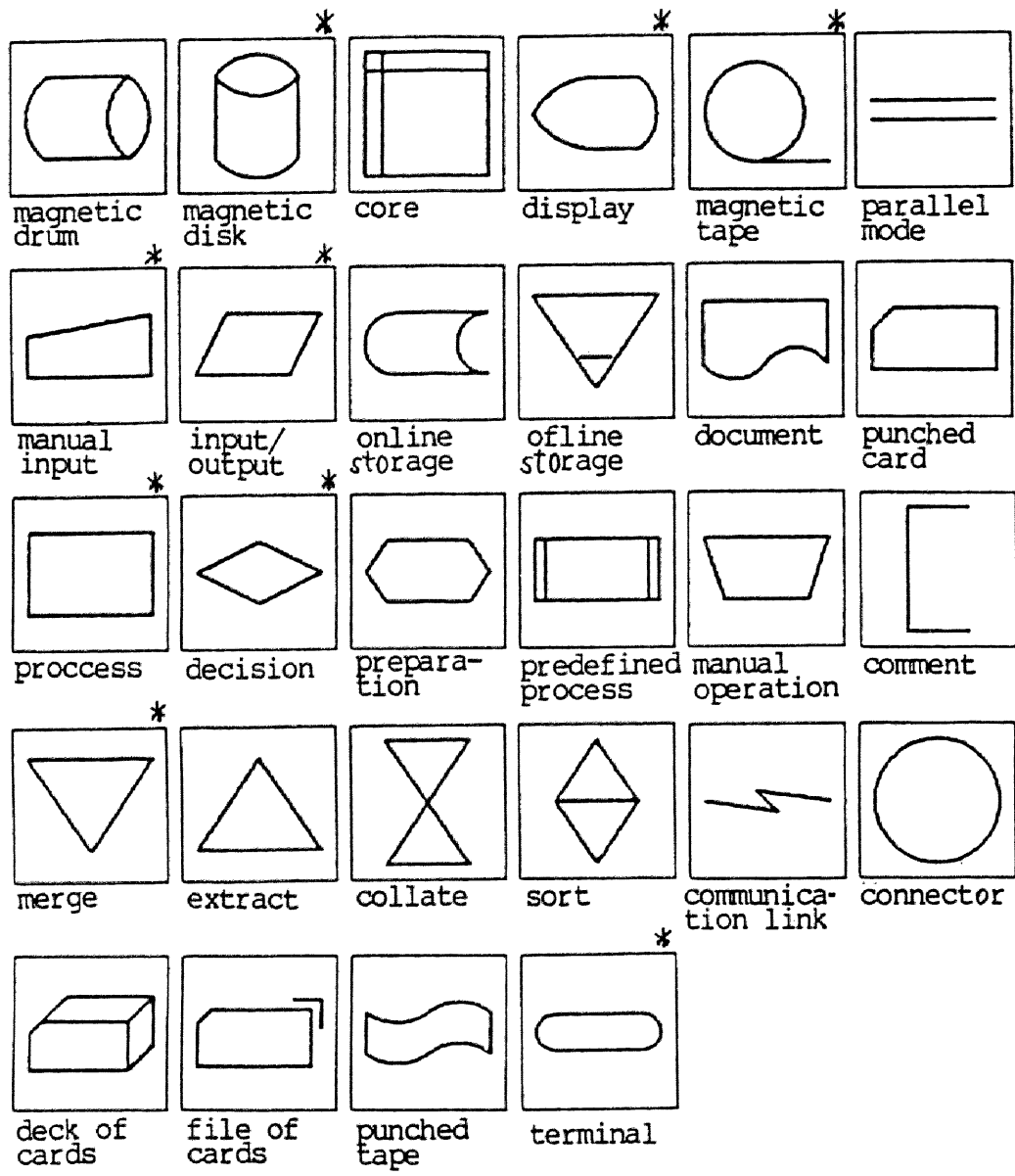
の2点である。

まず、①の結果について分析した。②の結果については、第3章の3. 2で分析す

る。

フローチャートなどに出現する基本図形は、漢字などと異なり、決められた書き順は存在しない。そのため制限を与えずに筆記させたシンボルのデータを調査すると、筆順と画数にはほとんど規則性が見られない。例えば、磁気ディスクにおける画数と筆順の例を図2.4(a)に示す。図からわかるように、磁気ディスクの場合には今回の調査だけでも10種類の画数と筆順の例が観測された。各シンボルに対する画数の分布を表2.1に示す。

しかし、画数や筆順には規則性はあまり見られないものの、ストロークの端点となる可能性のある箇所は少数の箇所に限られるという結果が得られた。しかも、シンボルを直線と円弧の基本ストロークで近似したとすれば、その箇所は基本ストロークの交点に位置していることがわかった。これら基本ストロークの交点を候補端点と呼ぶことにする。磁気ディスクの候補端点を図2.4(b)に示す。



\*は予備実験に使用したシンボル (2.4 参照)

図2.2 シンボルの例 (JIS. C6270)

# アンケートのおお願い

基八 村瀬 洋

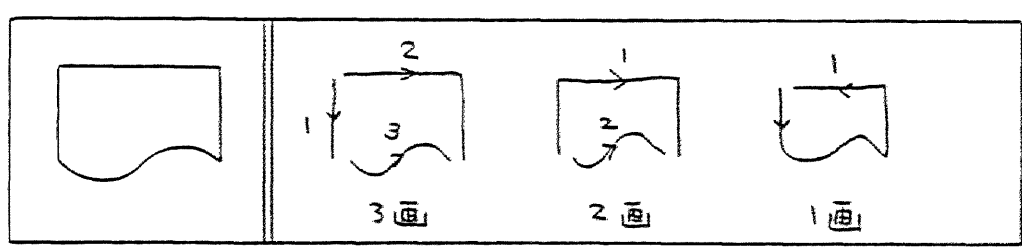
## 1. 主旨

タブレットから入力した手書きのフローチャートやブロック図等の図面を認識・理解するシステムを考えています。以下のアンケートに御協力ください。

## 2. シンボルの書き方に関する調査

筆記しやすい筆順と画数を、記入例に示すようにお答えください。複数個を記入しても結構です。

### 記入例



ここから始めてください


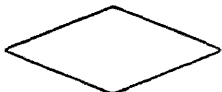
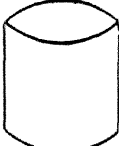
	
	
	

図 2. 3 アンケート用紙



アンケート調査に見られた画数・筆順の変動（11名分）  
磁気ディスクの場合

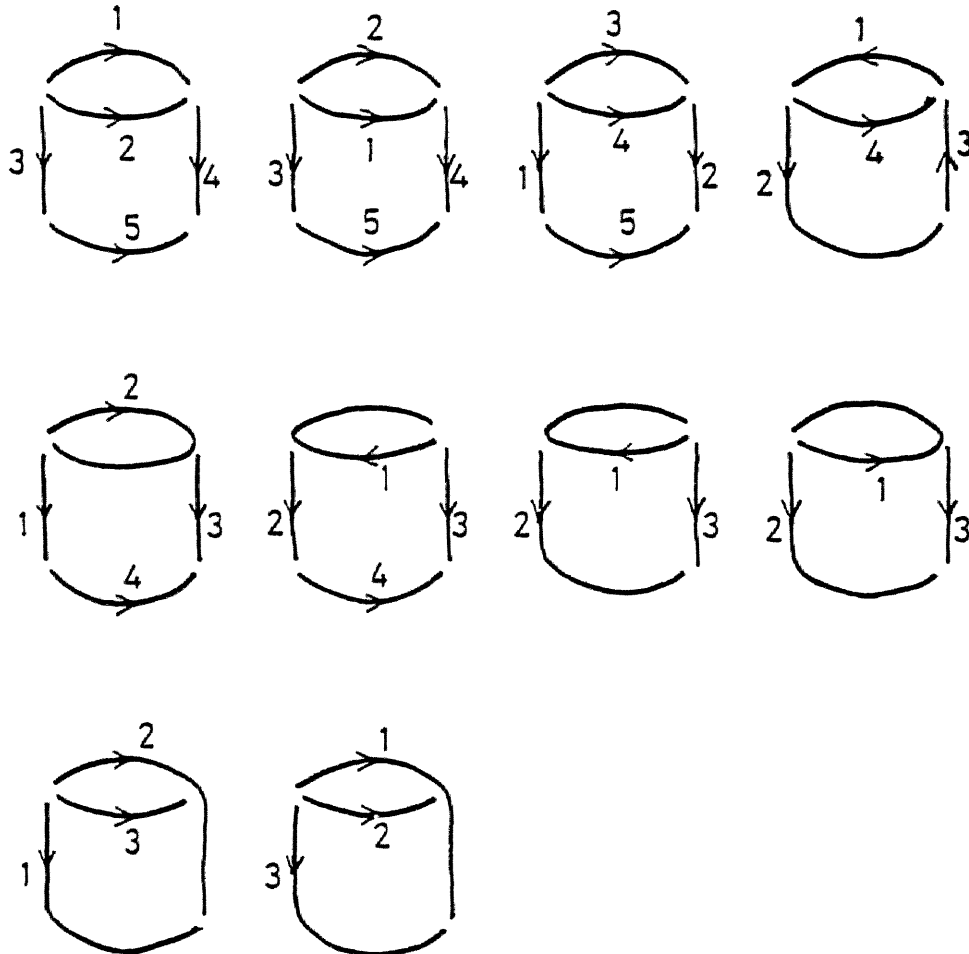


図2.4(a) 磁気ディスクに見られる画数と筆順の例

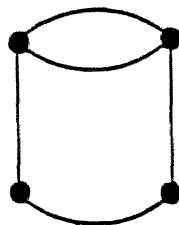


図2.4(b) 磁気ディスクの候補端点の例

表 2. 1 画数の分布

画数 シンボル名	1画	2	3	4	5	6
PROCESS	3	2	1 1	2	0	0
DOCUMENT	1	3	9	3	0	0
DECISION	4	9	4	3	0	0
TERMINAL	7	2	1	5	0	0
INPUT/OUTPUT	2	4	6	3	0	0
DISPLAY	1	5	7	1	1	0
MAGNETIC DISK	0	1	3	6	7	0
MAGNETIC TAPE	8	8	1	0	0	0
CONNECTOR	1 3	0	0	0	0	0

### 2. 3 画数・筆順自由整合法

2. 2節の結果より、フローチャートやブロック図などに含まれる基本図形（シンボル）を手書きした場合には、種々の画数や筆順での筆記が予想される。この節では、これらの多種多様な筆記順序で手書きされたシンボルを認識することを可能にする「画数・筆順自由整合法」について述べる。

本手法では任意の画数と筆順での筆記を許容しているため、入力されたシンボルがどのようなストローク順序で筆記されているのか最初の段階では予想がつかない。ここではトップダウン手法およびボトムアップ手法を併用しながら、この問題に対処する。まず、入力図形がある1種類のシンボルであるとトップダウン的に仮定する。以下、このシンボルを「仮定したシンボル」と呼ぶ。次に、入力図形が「仮定したシンボル」であるための条件を順次調べ、条件が満足される場合に、その入力図形を「仮定したシンボル」に対する候補図形とする。更に、もし入力図形が「仮定したシンボル」であるならば、どのような画数と筆順が有り得るかを、ボトムアップに得られる入力ストロークの端点の位置などの情報を利用して、全て生成する。そして、最後にストローク間のDPマッチングにより得られる相異度情報をもとに、この仮定を

検証する。そして、仮定が許容されたシンボルの中から相異度の最も小さいシンボルを認識結果とする。

図2.5に具体的な認識アルゴリズムの流れを示す。

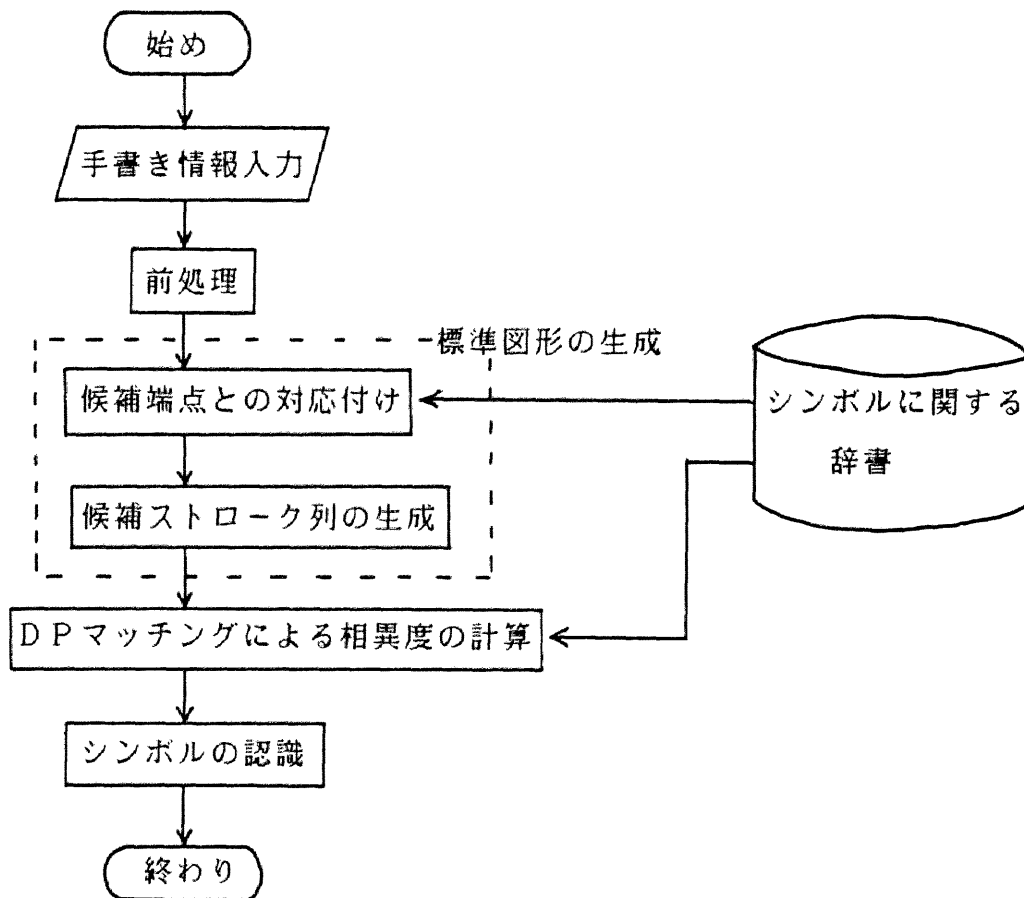


図2.5 画数・筆順自由整合法の流れ

### 2.3.1 手書き情報入力

手書き情報入力にはデータタブレットを使用する。タブレットはペンのアップダウン情報とXY座標の値を一定時間間隔でサンプリングする機能を有する。使用したタブレットは電磁結合型である。その原理は、ペンから発生する交流磁界をタブレット面内にX方向Y方向にスリット状に張った電線のループで検出し座標値を求めるものである。ペンのアップダウン情報はペンホルダーに内蔵されたマイクロスイッチでペ

ン先の上下動を検出することにより得る。ここでは筆記し易さを考慮し、ペン先の上下動の少ないノンストローク型のペンを特別に使用した。サンプリングレートは毎秒100点、空間分解能は1mmあたり10本である。

ペンのダウンからアップまでの座標値系列をストロークと呼ぶことにする。

### 2.3.2 前処理

タブレットから採集されたデータに対して、まず処理の効率化あるいは認識精度の向上の目的で、雑音点の除去と停留点の除去〔20〕を行なう。

#### (1) 雑音点の除去

電磁結合型のタブレットは、外部誘導などの影響により、筆跡から大きく離れた座標点を、雑音点として発生することがある。このような雑音点は、一般にストローク上の他の座標点から孤立した点となる。そこで、ストローク内の隣り合う2個の筆点間の距離が、あるしきい値を越える場合に、後の筆点を雑音とみなし、この筆点の座標値を前の筆点の座標値に変換する雑音除去処理を行なう。ここで、入力座標値系列を式(2.1)で示す。また、各筆点は式(2.2)によって、XY空間座標系で表現される。

$$(p_1, p_2, \dots, p_i, \dots, p_m) \quad (2.1)$$

但し、 $p_i$  : 筆点

$m$  : 筆点総数

$$p_i = (p_{ix}, p_{iy}) \quad (1 \leq i \leq m) \quad (2.2)$$

一方、処理後の座標系列を以下のように、

$$(q_1, q_2, \dots, q_i, \dots, q_m) \quad (2.3)$$

$$q_i = (q_{ix}, q_{iy}) \quad (1 \leq i \leq m) \quad (2.4)$$

と表わすと、雑音除去処理の手順は以下のようなになる。

$$q_{ix} = p_{i-1x} \quad \text{if} \quad |p_i - p_{i-1}| > D \quad (2.5)$$

$$q_{iy} = p_{i-1y} \quad \text{//} \quad (2.6)$$

$$q_{ix} = p_{ix} \quad \text{if} \quad |p_i - p_{i-1}| \leq D \quad (2.7)$$

$$q_{iy} = p_{iy} \quad // \quad (2.8)$$

但し、 $2 \leq i \leq m$ のときに(2.5)~(2.8)は適用

$$q_{1x} = p_{1x}, \quad q_{1y} = p_{1y}$$

ここで、Dは雑音除去のしきい値で、ここでは2.5cmに設定した。

## (2) 停留点の除去

タブレットで採集した筆点の座標データは、一定時間間隔でサンプリングされるため、ペンの停留した箇所やペンの移動速度が遅い箇所では、近接した筆点が多数含まれる。近接した筆点はストロークの形状を表現するのに冗長な情報となり、計算機の記憶領域を無意味に消費するだけなので、ここでは引き続き筆点間の距離が一定値を越えるように、筆点を間引く。

処理後の座標値系列を

$$(r_1, r_2, \dots, r_i, \dots, r_m) \quad (2.9)$$

$$r_i = (r_{ix}, r_{iy}) \quad (1 \leq i \leq m) \quad (2.10)$$

とすると、以下に示す関係を満たしている。

$$(r_{ix} - r_{i-1x})^2 + (r_{iy} - r_{i-1y})^2 \geq \lambda^2 \quad (2.11)$$

ここで、 $\lambda$ は間引きのパラメータであり、ここでは1mmとした。

雑音点の除去と停留点の除去を施された座標値系列は、2.3.4以降の処理で使用される。

### 2.3.3 シンボルに関する辞書の記述

認識対象シンボルの種類の拡張性を考慮すれば、シンボルに関して計算機に付与する辞書は、できるだけ簡単な記述であることが望ましい。本手法では各シンボルを直線と円弧の組み合わせで記述する。その際、アンケート調査の結果を考慮して、各シンボルを候補端点を端点とする基本ストローク群で表現する。但し、各基本ストロークは直線と円弧とする。直線は始点と終点の座標値で、円弧は始点と終点と中間点の

座標値で表現される。シンボル「磁気ディスク」に関する辞書の記述を図2.6に示す。その他のシンボルの辞書の記述を付録Aに示す。

実際にこの辞書が利用される際には大きさの正規化および縦横比の正規化がなされるため、基本ストロークの表現で使用される座標値は相対的な値でよい。

Representation format:

```
NAME="symbol name" ("list of labels")

"label":=("stroke type (arc or line)"
          ,"starting point coordinate"
          ,"ending point coordinate"
          [, "middle point coordinate"])
```

example

```
NAME="magnetic disk" (A,B,C,D,E)

A:=(arc, (0,5), (4,5), (2,6))
B:=(arc, (0,5), (4,5), (2,4))
C:=(arc, (0,1), (4,1), (2,0))
D:=(line, (0,1), (0,5))
E:=(line, (4,1), (4,5))
```

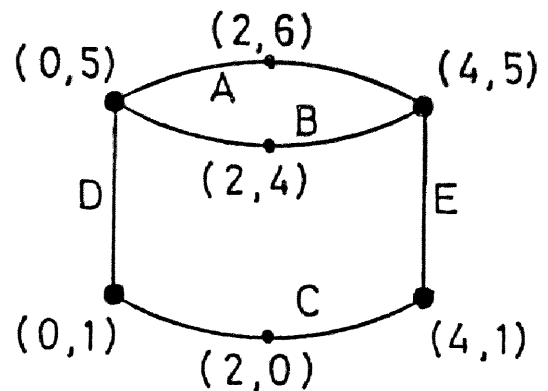


図2.6 辞書の記述例（磁気ディスクの場合）

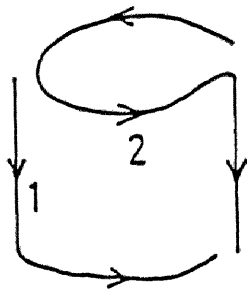
#### 2.3.4 候補端点との対応付け

入力図形を構成するストロークを入力ストロークと呼ぶ。もし、入力図形が「仮定したシンボル」であるならば、入力ストロークの端点は「仮定したシンボル」上の候補端点のどれかに対応する。そこで、まず辞書の記述を用いて「仮定したシンボル」を着目した入力図形と同じ大きさに正規化する。具体的には、縦横独立に拡大縮小し、「仮定したシンボル」に外接する矩形を入力図形のそれと同じにする。その後に入力ストロークに含まれる端点をそれぞれ最近傍の候補端点に対応付ける。図2.7(a)に示す入力ストロークの場合には、図2.7(b)の磁気ディスクが仮定された際に、

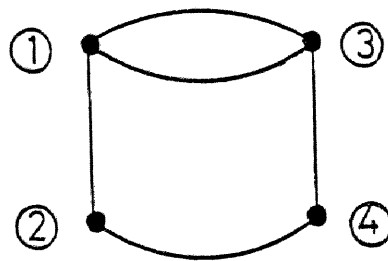
表 2. 2 に示す対応結果が得られる。

表 2. 2 候補端点への対応例

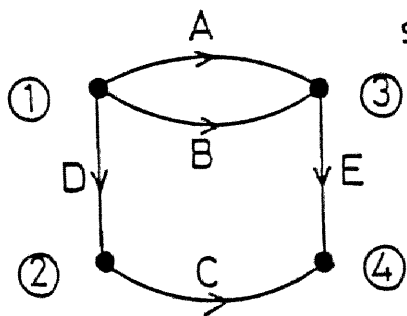
ストローク番号	始点	終点
1	1	4
2	3	4



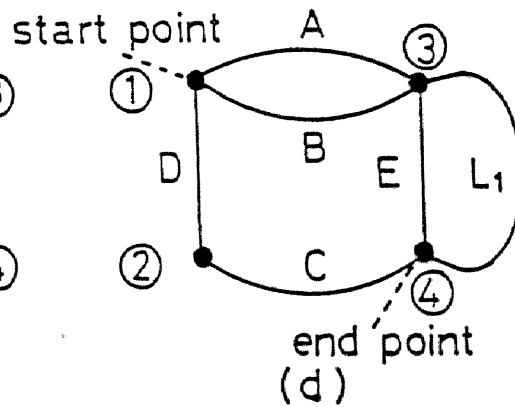
(a)



(b)



(c)



(d)

(a)Input strokes, (b)supposed symbol,  
(c)directed graph ,(d)graph in search.

図 2. 7 標準ストローク列の探索

手書き変形のため入力ストロークの端点の位置は変動するが、その変動は一定範囲内にある。もし、入力ストロークの端点が「仮定したシンボル」のどの候補端点からも一定範囲内にない場合には、着目した部分図形は「仮定したシンボル」ではないとし、他のシンボルを仮定する処理に移行する。

### 2.3.5 候補標準ストローク列の生成

前節の処理によって得られた候補端点と入力ストロークの端点との対応付け結果から、入力図形が、「仮定したシンボル」として妥当か否かを位相構造的な観点から調べる手法について述べる。もし妥当であるならば、「仮定したシンボル」上での予想されるペンの動きを、候補となる標準ストローク列（候補標準ストローク列）として全て列挙する。

まず、「仮定したシンボル」は辞書に基づき、候補端点をノードとし、基本ストローク（直線や円弧）をブランチとする有向グラフで表現する。有向グラフを採用した理由は、ブランチ上でのペンの動く方向を陽に記述するためである。ノード名を  $v$ 、ブランチ名を  $a$  とすれば、有向グラフは計算機上で、接続行列  $D = (d_{va})$ 、

$$d_{va} = \begin{cases} 1 & (\text{ノード } v \text{ がブランチ } a \text{ の始点のとき}) \\ -1 & (\text{ノード } v \text{ がブランチ } a \text{ の終点のとき}) \\ 0 & (\text{その他}) \end{cases} \quad (2.12)$$

により表現できる。

例えば、磁気ディスクの有向グラフ表現は図2.7(C)となり、接続行列は、

$$\begin{matrix} & A & B & C & D & E \\ \begin{matrix} (1) \\ (2) \\ (3) \\ (4) \end{matrix} & \left( \begin{array}{ccccc} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ -1 & -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & -1 \end{array} \right) & & & & \end{matrix} \quad (2.13)$$

となる。

ここで、入力ストロークは  $N$  本から構成されるとする。入力ストロークが「仮定し



たシンボル」として妥当であるためには、上記の有向グラフ内のすべてのブランチを1回ずつ通過し、かつ入力ストロークと対応した候補端点を始点あるいは終点とするN本のストロークからなる経路が存在することが条件となる。すなわち処理手順としては、前節で求めた候補端点と入力ストロークの端点との対応条件に制限されたN筆書きの経路を求めるグラフ問題となる。更に問題を簡単化するため、引き続きストロークの間のペンアップを1つのブランチとして追加することによりオイラーパス(Eulerian path)を求める問題に変形する。但し、オイラーパス[84]とは、全てのパスをちょうど1回ずつ通るパスのことであり、一筆書き可能(unicaursal)なパスとも称される。以下に処理手順を示す。

(1) 第 $n$  ( $1 \leq n \leq N-1$ ) ストロークの終点に対応したノードから、第 $n+1$  ストロークの始点に対応したノードに向かう新しいブランチ $L_n$ をもとの有向グラフに追加する(以下、追加ブランチと呼ぶ)。一例を図2.7(d)に示す。接続行列は、

$$\begin{array}{l}
 \begin{array}{cccccc}
 & A & B & C & D & E & L1 \\
 (1) & \left( \begin{array}{cccccc}
 1 & 1 & 0 & 1 & 0 & 0 \\
 0 & 0 & 1 & -1 & 0 & 0 \\
 -1 & -1 & 0 & 0 & 1 & -1 \\
 0 & 0 & -1 & 0 & -1 & 1
 \end{array} \right) \\
 (2) \\
 (3) \\
 (4)
 \end{array}
 \end{array} \quad (2.14)$$

となる。

(2) 第1ストロークの始点に対応したノードをパスの始点とし、第Nストロークの終点に対応したノードをパスの終点とするオイラーパスを求める。解の存在のための条件は、パスの始点と終点を除くすべてのノードの次数が偶数となることである。ノード $v$ の次数は接続行列の第 $v$ 行ベクトルより  $\sum_a |d_{va}|$  として計算できるから、ノードの偶奇条件を満足しない場合は、着目した部分図形「仮定したシンボル」ではありえないとし、他の部分図形を着目する処理に移行する。

(3) 次数に関する条件が満足される場合には、以下に示す条件を満足するようなオイラーパスをすべて求め、候補標準ストローク列とする。下記の条件は入力ストローク列の順序を考慮するために生じるものである。

①バスが追加ブランチ $L_n$  ( $n = 1, 2, \dots, N-1$ )を通過する順序は $L_1, L_2, \dots, L_{N-1}$ である。

②バスが追加ブランチを引き続いて通過することを禁止する。

③追加ブランチの逆矢印方向へのバスの通過は禁止するが、その他のブランチでは両方向へ通過可能である。

オイラーバスの探索には、ディプスファースト法を用いた。なおディプスファースト法はスタックを利用することにより実現が容易である。バスの探索が成功した場合には、着目した部分図形を「仮定したシンボル」の候補図形として抽出し、探索が失敗した場合には、着目した部分図形は「仮定したシンボル」ではありえないとし、他のシンボルを仮定する処理に移行する。

図2.7の入力例に対して候補標準ストローク列が生成される過程を、ステップごとに表2.3に示す。この表を簡単に説明する。探索系列とは、そのステップで探索しているバスを、ブランチ名の記号系列で表現したものであり、+はブランチを矢印方向に、-はブランチを逆矢印方向に通過することを意味している。各ステップでのスタック動作に着目すれば、プッシュ動作をするのはそのノードでバスの分岐の可能性が存在する場合で、ポップ動作をするのは探索がそれ以上深いレベルに進むことができない場合である。ポップ動作は探索におけるバックトラッキングを意味している。この場合バックトラッキングされるのは具体的には、探索が成功した場合と、途中で探索することが不能となった場合である。このうち、探索が成功した場合の探索系列の内容は、候補標準ストローク列としてテーブルに格納される。

表 2. 3 探索の一例 (Magnetic disk の例)

ステップ	探索系列	スタック動作
1	+D	PUSH
2	+D +C	
3	+D +C L1	PUSH
4	+D +C L1 +E	POP 探索失敗
5	+D +C L1 -B	PUSH
6	+D +C L1 -B +A	
7	+D +C L1 -B +A +E	POP 探索成功
8	+D +C L1 -A	
9	+D +C L1 -A +B	
10	+D +C L1 -A +B +E	POP 探索成功
11	+D +C -E	
12	+D +C -E -B	PUSH
13	+D +C -E -B +A	POP 探索失敗
14	+D +C -E -A	
15	+D +C -E -A +B	POP 探索失敗
16	+B	
17	+B +E	PUSH
18	+B +E L1	
19	+B +E L1 -A	
20	+B +E L1 -A +D	
21	+B +E L1 -A +D +C	POP 探索成功
22	+B +E -C	
23	+B +E -C -D	
	以下、省略	

図2.7(d)の例に対し、上記の手法でパスを探索すれば、以下に示す6組の候補標準ストローク列が生成される。

- ① +D, +C, L1, -B, +A, +E
- ② +D, +C, L1, -A, +B, +E
- ③ +B, +E, L1, -A, +D, +C
- ④ +B, -A, +D, +C, L1, +E
- ⑤ +A, +E, L1, -B, +D, +C
- ⑥ +A, -B, +D, +C, L1, +E

但し、“+D”はブランチDの矢印方向への通過を意味し、“-B”はブランチBの逆矢印方向への通過を意味する。上記の候補標準ストローク列の2組目に入力ストローク(図2.7(a)参照)との正しい対応が現われていることがわかる。

候補標準ストローク列は、上記のように「仮定したシンボル」上でのブランチの系列として求まる。各ブランチは、辞書によれば直線又は円弧で記述されている。そこで、各ブランチに対し直線や円弧の座標値系列を生成し、各ストローク毎に一定の点数で近似する。

直線と円弧の具体的な座標値系列の生成手法を、以下(1)、(2)で述べる。この座標値系列を用いて、候補標準ストローク列と入力ストロークとの相異度を計算する。相異度の計算方法は、2.3.7で述べる。

#### (1) 直線の座標値系列の生成

直線は、辞書によれば始点と終点の座標値で記述されている。以下に、この記述から、直線の座標値系列を求める手法を述べる。ここでは、始点と終点の座標値を内挿することにより求める。始点を $(X_1, Y_1)$ とし終点を $(X_2, Y_2)$ とすれば、N点から構成される座標値系列 $\{(X_n, Y_n), n=1, \dots, N\}$ の $X_n, Y_n$ は、

$$\begin{cases} X_n = \{ (n-1) \cdot X_2 + (N-n) \cdot X_1 \} / (N-1) & (2.16) \end{cases}$$

$$\begin{cases} Y_n = \{ (n-1) \cdot Y_2 + (N-n) \cdot Y_1 \} / (N-1) & (2.17) \end{cases}$$

により生成できる。

## (2) 円弧の座標値系列の生成

円弧は、辞書によれば、始点と終点と中間点の座標値で記述されている。以下に、この記述から円弧の座標値系列を生成する手順を述べる。始点を  $(X_1, Y_1)$ 、終点を  $(X_2, Y_2)$ 、中間点を  $(X_3, Y_3)$  とする。まず、これら3点の座標値から円弧の半径  $R$  と中心座標  $(a, b)$  を求める。円の方程式は、

$$(x - a)^2 + (y - b)^2 = R^2 \quad (2.18)$$

で表現することができるので、3点の座標値を各々代入すれば、

$$(X_1 - a)^2 + (Y_1 - b)^2 = R^2 \quad (2.19)$$

$$(X_2 - a)^2 + (Y_2 - b)^2 = R^2 \quad (2.20)$$

$$(X_3 - a)^2 + (Y_3 - b)^2 = R^2 \quad (2.21)$$

となる連立方程式ができる。これより  $a$ 、 $b$ 、 $R$  を求めると以下のようなになる。

$$W_1 = (X_1 - X_3)^2 + (Y_1 - Y_3)^2 \quad (2.22)$$

$$W_2 = (X_2 - X_3)^2 + (Y_2 - Y_3)^2 \quad (2.23)$$

$$A = \frac{(Y_2 - Y_3) \cdot W_1 - (Y_1 - Y_3) \cdot W_2}{2 \cdot \{ (X_1 - X_3) \cdot (Y_2 - Y_3) - (X_2 - X_3) \cdot (Y_1 - Y_3) \}} \quad (2.24)$$

$$B = \frac{-(X_2 - X_3) \cdot W_1 + (X_1 - X_3) \cdot W_2}{2 \cdot \{ (X_1 - X_3) \cdot (Y_2 - Y_3) - (X_2 - X_3) \cdot (Y_1 - Y_3) \}}$$

(2.25)

とすれば、

$$R = \sqrt{(A^2 + B^2)}$$

(2.26)

$$a = A + X_3$$

(2.27)

$$b = B + Y_3$$

(2.28)

となる。

生成する座標値系列を  $\{(X_n, Y_n), n=1, \dots, N\}$  とすれば、 $X_n$ 、 $Y_n$  は、

$$X_n = R \cdot \cos(C \cdot (n-1) + a)$$

(2.29)

$$Y_n = R \cdot \sin(C \cdot (n-1) + b)$$

(2.30)

で求めることができる。ここで、 $C$  は以下の手順で求める。 $K_1$ 、 $K_2$ 、 $K_3$  を、

$$K_1 = \tan^{-1}((Y_1 - b) / (X_1 - a))$$

(2.31)

$$K_2 = \tan^{-1}((Y_2 - b) / (X_2 - a))$$

(2.32)

$$K_3 = \tan^{-1}((Y_3 - b) / (X_3 - a))$$

(2.33)

で定義すると、 $C$  は、

$$C = \begin{cases} (K_2 - K_1) / (N - 1) \\ \quad \{K_1 < K_3 < K_2 \text{ OR } K_2 < K_3 < K_1 \text{ のとき}\} \\ (K_2 - K_1) / (N - 1) - 2\pi \\ \quad \{K_1 < K_2 < K_3 \text{ OR } K_3 < K_1 < K_2 \text{ のとき}\} \\ (K_2 - K_1) / (N - 1) + 2\pi \\ \quad \{K_2 < K_1 < K_3 \text{ OR } K_3 < K_2 < K_1 \text{ のとき}\} \end{cases} \quad (2.34)$$

である。

直線や円弧が連結したストロークについては、上記の手法により求めた直線あるいは円弧の座標値系列を連結することにより、生成する。

### 2.3.6 候補端点の連結

入力ストロークの端点を候補端点に対応付ける際に、シンボルの形状が単純な場合には、2.3.5の処理だけで正しく候補標準ストローク列が生成される。しかし、形状が複雑な場合には、入力ストロークの端点が正しい候補端点に近接した誤った候補端点に対応付けられ、入力ストロークに対応した候補標準ストローク列が生成されない場合が生じる。そこで、シンボル形状が複雑なために、シンボル中に近接した候補端点群が存在するシンボルについては、「仮定したシンボル」を表現するグラフ中の近接した候補端点に対応するノードを連結する。この処理を候補端点連結処理と呼ぶ。図2.8(a)～(c)で示すグラフは、候補端点連結処理がなされた結果、図2.8(a')～(c')に示すグラフに変形される。

このグラフに対し2.3.5の処理を施す。これにより、候補標準ストローク列の生成数は多くなるものの、複雑な形状のシンボルに対しても候補標準ストローク列が安定して生成されることになる。

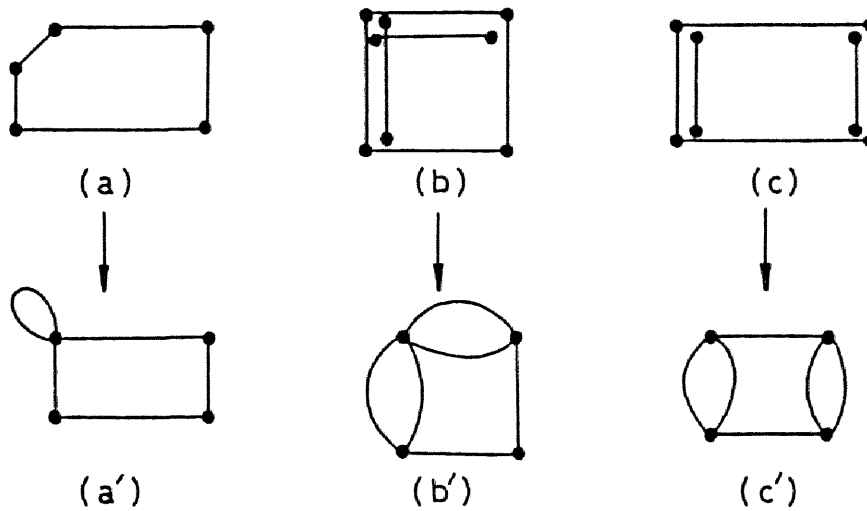


図 2. 8 候補端点の連結処理

### 2. 3. 7 DPマッチングによる相異度の計算

候補図形の抽出段階では、入力ストロークの端点位置とその接続関係だけを利用しているため、入力ストロークの細かい形状については考慮していない。そこで、次段階として、入力ストロークと2. 3. 5で生成した候補標準ストローク列との間で形状のマッチングを行い、形状の相異度を計算する。

候補標準ストローク列と入力ストロークとの形状の相異度を表現するのに適した尺度として、次に述べる3種類の相異尺度を比較検討した。相異度は入力ストロークと候補標準ストローク列との間で対応するストローク毎の相異度の和として定義する。ここでは、1ストローク対の相異度を計算する手法について説明する。入力ストロークと候補標準ストローク列との間で、対応したストロークのある1組が各々M点から構成され、両者の座標値系列が、 $\{(X_m, Y_m) \ m=1, \dots, M\}$  ,  $\{(X'_m, Y'_m) \ m=1, \dots, M\}$  であるとする。但し、Mはストロークの近似点数である。

#### (1) 相異度1: $d_1$

入力ストロークと候補標準ストロークの各々を構成する座標点を単純に1対1に対応付ける。対応付けられた点間のユークリッド距離の和を相異度とする。つまり、相



異度  $d_1$  は、

$$d_1 = \frac{1}{M} \sum_{m=1}^M \{ (X_m - X'_m)^2 + (Y_m - Y'_m)^2 \}^{\frac{1}{2}} \quad (2.35)$$

である。

### (2) 相異度 2 : $d_2$

相異度 1 においてストロークを構成する座標点の対応付けに DP マッチングを採用する。つまり、相異度  $d_2$  は、

$$d_2 = \frac{1}{M} \min_u \left\{ \sum_{m=1}^M \{ (X_m - X'_{u(m)})^2 + (Y_m - Y'_{u(m)})^2 \}^{\frac{1}{2}} \right\} \quad (2.36)$$

である。ここで  $u(m)$  は座標点間の対応関係を表わし、 $u(m)$  には以下の制約条件を加える。座標点間の対応関係の一例を図 2. 9 に示す。

$$\begin{cases} u(1) = 1 & (2.37) \\ u(M) = M & (2.38) \\ u(i) = j \text{ のとき } u(i+1) = \{ j \text{ or } j+1 \text{ or } j+2 \} & (2.39) \end{cases}$$

### (3) 相異度 3 : $d_3$

点間のユークリッド距離に加え、各点での接線方向の差を評価した DP マッチングを採用する。つまり、相異度  $d_3$  は、

$$d_3 = \frac{1}{M} \min_u \left\{ \sum_{m=1}^M \{ (X_m - X'_{u(m)})^2 + (Y_m - Y'_{u(m)})^2 + \alpha \cdot h(m, u(m)) \}^{\frac{1}{2}} \right\} \quad (2.40)$$

である。ここで  $h(i, j)$  は、一方のストロークの  $(X_i, Y_j)$  での接線方向と、他方のストロークの  $(X'_i, Y'_j)$  での接線方向の差を表わし、

$$h(i, j) = \left| \tan^{-1} \left( \frac{Y_{i+1} - Y_i}{X_{i+1} - X_i} \right) - \tan^{-1} \left( \frac{Y'_{i+1} - Y'_i}{X'_{i+1} - X'_i} \right) \right| \quad (2.41)$$

である。ここで $\alpha$ は $h(i, j)$ の係数であり、定数である。この定数は実験により最適な値に設定する。

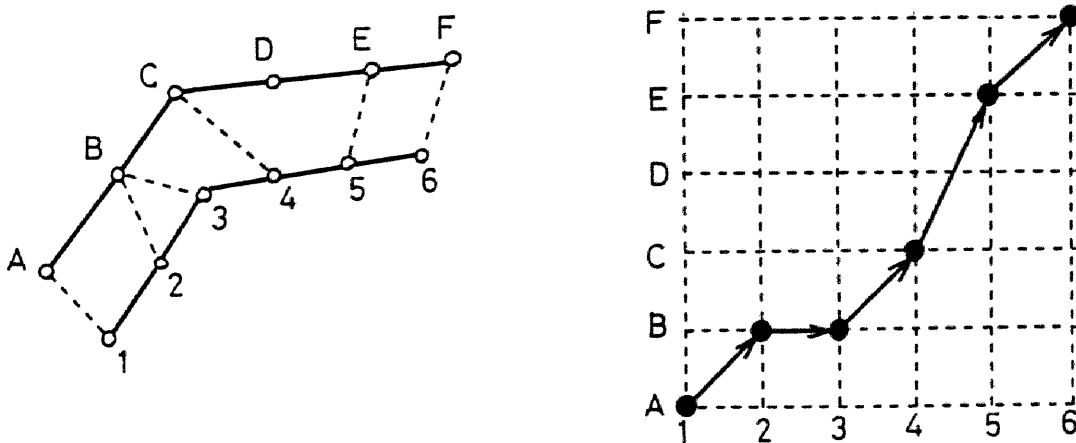


図2.9 DPマッチングによるストローク間の点の対応付け

（例えば、図中の1～6が入力ストロークで、A～Fが候補標準ストロークの場合には、DPマッチングの結果が右図のように、対応結果が左図のようになる。）

### 2.3.8 シンボルの認識

シンボルの認識は以下の手順で終了する。まず、入力図形に対して、辞書に含まれる全てのシンボルの存在を仮定する。次に、候補標準ストローク列が生成されたシンボルに対し、入力図形と各シンボルとの間のシンボル間相異度を求める。但し、シンボル間の相異度は、シンボルを構成する全てのストロークについてのストローク間相異度の和で定義する。最後に、このシンボル間相異度が最小となるシンボルを認識結果として出力する。

### 2.4 認識実験

2.3.7で示した3種類の相異度を用いてシンボル図形の認識実験を行った。使用した図形は、フローチャートによく出現する「terminal」「decision」等の9種類のシンボルであり、JISC6270で定められたシンボルの中から選択した。これ

ら9種類のシンボルを図2.2中に\*で示す。

実験に使用したデータは、これら9種類のシンボルを20名の筆記者に各4回ずつ制約なしに筆記させたものであり、合計720個のシンボルである。

相異度別の認識率を表2.4に示す。実験の結果、点間のユークリッド距離とその点での接線方向の差の評価値の和をもとにDPマッチングを行う相異度3の手法で $\alpha$ を最適な値とした際に高い認識率が得られた。 $\alpha$ に対する認識率を表2.5に示す。以上の実験に基づき、認識アルゴリズムでは相異度3を採用することにした。

表2.4 シンボルの認識率

相異度1	95.0%
相異度2	96.9
相異度3	81.0 ( $\alpha$ = 十分大きい値)
相異度3	97.3 ( $\alpha$ を最適な値)

表2.5  $\alpha$ に対する認識率

$\alpha$	認識率
0.0	96.9%
0.001	97.1
0.002	97.1
0.003	97.3
0.004	97.2
0.005	97.1
0.01	96.7
0.015	96.3
0.02	95.7
0.025	94.6
$\infty$	81.0

## 2.5 考察

### 2.5.1 単純マッチング法との比較

本論文で提案した画数・筆順自由整合法と、従来から知られている単純マッチング法との比較を行なう。ここで、単純マッチング法とは、各シンボルについて最も一般的に筆記される画数と筆順を標準ストロークとし、標準ストロークと入力ストロークとの間で相異度を計算し、相異度の最小なシンボルを認識結果とする手法である。標準ストロークは複数登録される場合もあるが、ここでは単純化するため標準ストロークは1組だけ登録する場合を考え、画数・筆順自由整合法と比較する。


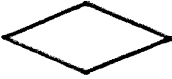



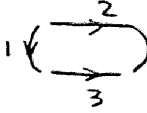
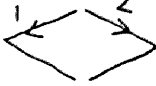
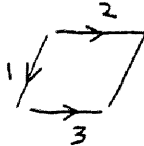
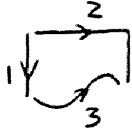
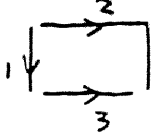
比較実験に使用した図形は、2.4節の認識実験で使用した9種類のシンボルである。

図2.10に各シンボル毎の最頻出のストローク順序を示す。単純マッチング法では、このストローク順序を標準ストロークとする。括弧内の数字は、今回のデータがどの程度このストローク順序で筆記されているかの割合を示す。単純マッチング法を用いた際の、各シンボル毎の認識率を図2.10に、またコンフュージョンマトリクスを図2.11(a)に示す。認識率が低い理由は、筆記されたストローク順序の多数が標準ストロークに登録されているストローク順序と異なっているためである。

画数・筆順自由整合法を用いた際のコンフュージョンマトリクスを図2.11(b)に示す。単純マッチング法の場合に比較して、大幅に認識率の向上が見られる。両手法の平均認識率のおよび認識時間の比較を表2.6に示す。但し、認識時間は1MIPSのミニコンピュータ(DG社のMV8000)によるものである。つまり、本手法は認識時間を2倍にしただけで、任意の画数と筆順で筆記されている基本図形を高精度に認識することが可能となったと考えられる。

表2.6 単純マッチング法との比較

	単純マッチング法	画数・筆順自由整合法
認識率	46.4%	97.3%
認識時間	0.6秒	1.2秒

					
最頻出 ストローク列 (頻度%)	 (26%)	 (26)	 (48)	 (64)	 (66)
単純 マッチング法 による認識率	23%	26	45	63	66


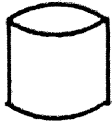


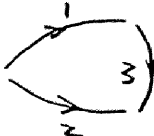
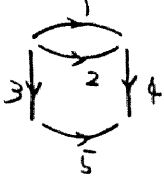

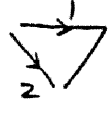
					全体
最頻出 ストローク列 (頻度%)	 (23)	 (19)	 (46)	 (43)	
単純 マッチング法 による認識率	30	42	47	72	46.3%

図2.10 単純マッチング法による認識

結果 入力	1	2	3	4	5	6	7	8	9	reject
1 :	23.%	1.	0.	0.	2.	1.	0.	0.	7.	63.
2 :	0.	26.	2.	1.	0.	2.	0.	0.	22.	45.
3 :	3.	18.	45.	0.	0.	2.	0.	0.	2.	27.
4 :	0.	3.	0.	63.	0.	5.	0.	0.	1.	26.
5 :	0.	1.	0.	0.	66.	2.	0.	0.	7.	22.
6 :	6.	5.	3.	0.	0.	30.	0.	0.	12.	42.
7 :	0.	0.	0.	0.	0.	0.	42.	0.	0.	57.
8 :	0.	0.	0.	0.	0.	0.	0.	47.	1.	51.
9 :	0.	0.	1.	2.	0.	5.	0.	0.	72.	18.

recognition rate= 46.39 %

1	'terminal'
2	'decision'
3	'in/out'
4	'document'
5	'process'
6	'display'
7	'mag. disk'
8	'mag. tape'
9	'merge'

図2. 11 (a) コンフュージョンマトリクス (単純マッチング法)

結果 入力	1	2	3	4	5	6	7	8	9	reject
1 :	92.%	0.	0.	0.	7.	0.	0.	0.	0.	0.
2 :	0.	100.	0.	0.	0.	0.	0.	0.	0.	0.
3 :	2.	0.	97.	0.	0.	0.	0.	0.	0.	0.
4 :	0.	0.	0.	98.	0.	0.	0.	1.	0.	0.
5 :	1.	0.	0.	0.	98.	0.	0.	0.	0.	0.
6 :	12.	0.	0.	0.	0.	87.	0.	0.	0.	0.
7 :	0.	0.	0.	0.	0.	0.	100.	0.	0.	0.
8 :	0.	0.	0.	0.	0.	0.	0.	100.	0.	0.
9 :	0.	0.	0.	0.	0.	0.	0.	0.	100.	0.

recognition rate= 97.22 %

図2. 11 (b) コンフュージョンマトリクス (画数筆順自由整合法)

## 2.5.2 候補標準ストローク列の生成数

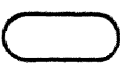
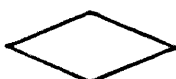


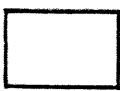

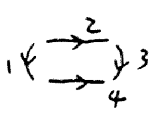

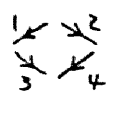

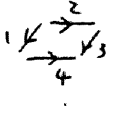

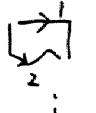

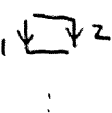
画数・筆順自由整合法により多数の標準ストロークを自動的に生成することが可能となった。実際に、どの程度まで生成することが可能なかを調査する。方法は、シンボルをグラフと見た際の全てのオイラーパスを数え上げることにより求める。生成することの可能な標準ストロークの種類を図2.12に示す。この結果より多数の画数と筆順の変形に対応した標準ストロークの生成が可能であることを確認した。

しかし、実際に生成される標準ストロークは入力ストロークの端点の位置とその接続関係から制限を受け、それほど多くない。つまり、生成数が効率的に減少していると考えることが出来る。例えば、2.4節の例では入力の磁気ディスクに対して6個の候補標準ストローク列が生成された。各シンボルに対して、平均的にどの程度の候補標準ストローク列が実際に生成されるかを調べ、その結果を表2.7に示す。シンボルの種類により、生成数の若干の違いはあるものの、全てのシンボルに対する平均生成数は1.6個であった。相異度の計算量は候補標準ストロークの生成数に比例することから、本手法により、画数と筆順が固定の場合に比較して、1.6倍のストローク間のマッチング回数（相異度の計算回数）で、任意の画数と筆順が許容されたことになる。種々の入力ストロークに対応して、各「仮定したシンボル」に対して、実際に生成された標準ストロークの一例を図2.13にしめす。

また計算時間の内訳について調査してみると、その26%が標準ストロークの生成に、残りの74%が相異度の計算に費やされていることがわかった。

表2.7 候補標準ストローク列の生成数

シンボル名	生成数	シンボル名	生成数
Terminal	1.2	Display	1.3
Decision	1.7	Mag. disk	2.8
Input/Output	1.3	Mag. tape	2.0
Document	1.1	Merge	1.9
Process	1.2		
平均	1.6		

					
標準 ストローク列 の生成数 と生成例	6 3 2 個   ⋮	6 3 2   ⋮	6 3 2   ⋮	6 3 3 0   ⋮	6 3 2   ⋮
画数・筆順 自由整合法 による認識率	92%	100	97	98	98


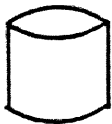










					全体
標準 ストローク列 の生成数 と生成例	6 3 3 0   ⋮	7 9 9 6   ⋮	8 4   ⋮	7 8   ⋮	2 3 3 4 6
画数・筆順 自由整合法 による認識率	87	100	100	100	97.2%

図 2. 1 2 画数・筆順自由整合法による認識




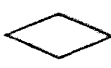







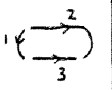
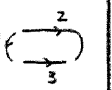
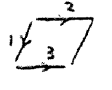
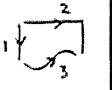
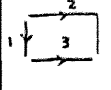
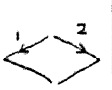
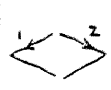
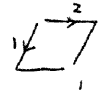
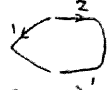
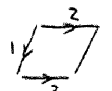
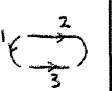
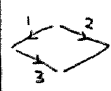
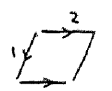
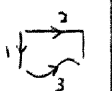
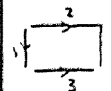
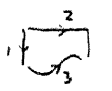
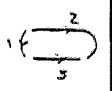
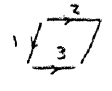
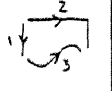
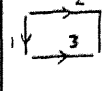

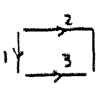
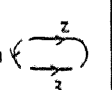
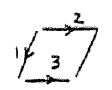
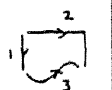
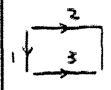
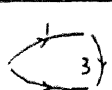
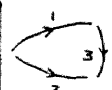
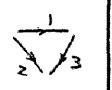
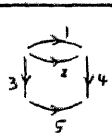
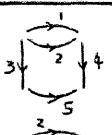

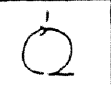
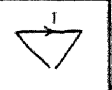
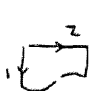
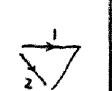
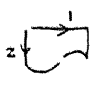

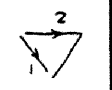
標準 入力									
									
									
									
									
									
									
									
									
									
									

図2. 13 入力例に対して実際に生成された候補標準ストローク列

### 2.5.3 候補端点連結の効果

2.3.6節で述べたように、候補端点連結処理を施さない場合には、候補端点の対応付けを誤り、候補標準ストローク列が正しく生成されない場合がある。ここでは、候補端点連結処理の効果を実験的に調査した。

実験の対象とする基本図形は、図2.2に示す28種類のシンボルとする。前にも触れたが、これらのシンボルは、フローチャートやブロック図に使用する目的でJISC6270で定められているものである。それら28種類のシンボルを含むフローチャートやブロック図などの図面を5名の筆記者に筆記させた。その図面から各シンボルを手作業で切り出してきて、それらのシンボルを使用して、候補標準ストローク列の生成実験を行なった。各シンボルの出現数は図面の性質上多少異なるが、平均で各シンボル29.3個が出現している。全体で820個のシンボルを実験に使用した。表2.8に実験結果を示す。候補端点連結処理の効果を確認できた。

表2.8 候補端点連結処理の効果

	候補端点連結処理を行わない場合	候補端点連結処理を行なった場合
正解候補ストローク列の生成率	95.4%	99.6%

## 2.6 結言

この章では、筆記制限の緩和を目指した画数や筆順に依存しない手書き図形の認識法について述べた。その認識法を、「画数・筆順自由整合法」と呼ぶ。

本論文で扱う図形は、文書中によく出現するブロック図やフローチャートあるいは技術資料中に出現する回路図面などとする。これらはいずれも基本図形（シンボル）とそれを接続する直線から構成されている。この章ではまず図形認識の基本にあたる個々のシンボルの認識について取り上げた。

ここで提案した画数・筆順自由整合法は以下に示す3つの特長を持つ。

(1) この手法は、入力ストローク列に対応した標準ストローク列を自動的に生成し、次に入力図形と標準図形との間で整合を取る手法であり、1種類の標準シンボルを登

録するだけで、任意の画数と筆順で筆記された基本図形を認識することが可能である。標準ストロークの自動生成は、標準シンボルを有向グラフで表現し、このグラフの経路を探索することにより実現している。この手法により、従来からの単純マッチング法に比較して認識時間を2倍にするだけで、任意の画数と筆順で筆記された720個（9種類で各80個のシンボル）のシンボルに対して97.3%の認識率を得ることが出来た。

(2) 複雑な図形を雑に筆記された場合、入力ストロークの端点と標準シンボルの候補端点（有向グラフのノードに対応した点）との対応付けが誤る危険がある。そこで、これらに対しては候補端点の対応付けに冗長性を持たせた候補端点連結処理を導入した。候補端点連結処理の導入により、正しく標準ストロークが生成されるようになり、820個（28種類のシンボル）のシンボルに対して生成された標準ストロークの中に正解のストロークが入る率は95.4%から99.6%に増大した。

(3) ストローク間の相異度の計算に、ストロークの方向と位置を用いたDPマッチングを採用した。その結果、手書き変形に強いシンボル認識手法を実現することができた。ストロークの近似点間のユークリッド距離の和を相異度にする手法に比較し、720個（9種類のシンボル）の基本図形に対して、認識率は95.0%から97.3%に改善された。

本章で提案した「画数・筆順自由整合法」により、1種類の標準シンボルを登録するだけで、任意の画数と筆順で筆記された基本図形（フローチャート等のシンボル）を高精度に認識することが可能となった。但し、本手法は単独の基本図形の認識法であり、複数の基本図形から構成される図面の認識にはこのままでは適用できない。第3章では、複数の基本図形から構成される図面中から基本図形を自動的にセグメンテーションし、同時に認識する手法について述べる。

## 第3章 図形のセグメンテーションと認識法

### 3.1 緒言

2章では個々の基本図形（シンボル）の認識手法について記述した。しかし、実際の図形では、複数のシンボルが複雑に連結することにより1つの図形が構成されている。この章では、複数のシンボルと直線から構成される手書き線図形を対象に、自由書式、任意の画数と筆順での筆記を許し、且つシンボル間のセグメンテーション指示を不要とするオンライン線図形認識法について述べる。

従来のオンライン認識アルゴリズムは、ストロークの単純なマッチングを線図形認識に応用したものであり、

(1) シンボル（基本図形）間のセグメンテーションを人間が指示する、

(2) 各シンボルを決められた画数と筆順で筆記する、等

の筆記時の制約が多く、素人には扱いにくいという問題点を抱えていた。これらの制約が取り除かれれば、オンライン認識型の入力は非常に扱い易いものとなる。本論文では、セグメンテーションの指示なく任意の画数と筆順で筆記された自由書式の線図形を認識するアルゴリズムについて検討した [52, 56, 57]。

認識アルゴリズムは次に示す3段階から構成した。

(1) 候補図形の抽出：入力図形の内部からシンボルらしい部分図形を候補図形としてすべて抽出する。その際、第2章で述べた画数・筆順自由整合法を基本とする。つまり、シンボル形状の構造を有向グラフで表現し、そのグラフのパスを探索することにより、任意の画数と筆順に対処して各シンボルに対する候補図形を抽出する。

(2) 候補ラティスの作成：抽出されたすべての候補図形に対しDPマッチングにより形状の相異度を計算し、候補図形のシンボル名称と相異度を候補ラティスに登録する。但し、候補ラティスは音声認識で使用される音韻ラティス [85] と同様な考えに基づくものである。

(3) 候補ラティスの探索：候補ラティスから相異度の和を最小とするシンボル系列を探索し、これを認識結果とする。つまり、基本図形のセグメンテーションと認識を同時に行う。

第3章で提案するセグメンテーション指示を不要とするオンライン図形認識手法を

「候補ラティス法」と呼ぶことにする。候補ラティス法の処理の流れの概念図を図3.1に示す。本章での主眼点は図中太枠で囲んだ箇所にある。シンボルのセグメンテーションと認識を融合することにより、シンボルを精度良くセグメンテーションする手法を実現している。

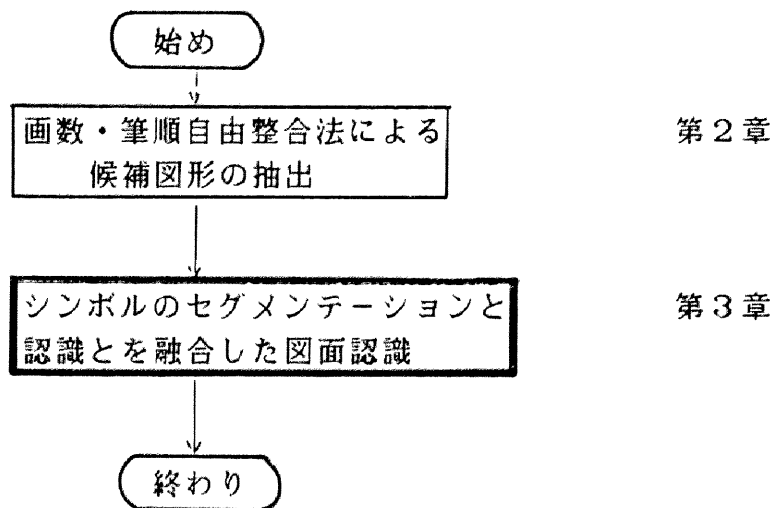


図3.1 候補ラティス法の概要

### 3.2 筆記条件に関する調査

フローチャートのような複数のシンボルと直線および文字が混在する図形を筆記する際の、筆記条件に関する筆記者の感想をアンケートにより調査した。アンケートは、11名の被験者に、文字と図形の混在したフローチャートを以下の4種類の紙に筆記させ、筆記のし易さをアンケート用紙に筆記させることにより実施した。使用したアンケート用紙を図3.2に示す。

- (1) 白紙
- (2) 10mm方眼紙
- (3) 8mm方眼紙
- (4) 6mm方眼紙

アンケートの結果をもとに、各用紙の筆記し易さをグラフで示す（図3.3）。筆記者の感想をまとめると以下のようなになる。

（1）図形部は白紙上より方眼紙上に筆記する方がやや書き易い。しかし、「図形を方眼紙の線にそって必ず書かなければいけない。」という制限がつくと、非常に書きづらい。またシンボルの大きさに制限が付くと書きづらい。

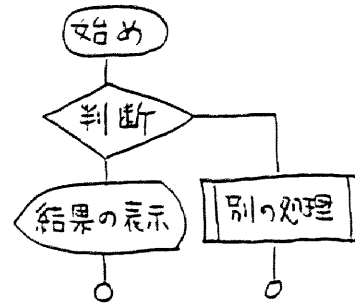
（2）文字部は、文字と図形を混在させた場合、方眼紙の方眼内に書くことは不可能である。

以上の結果より、図形認識アルゴリズムとしては、理想的には白紙紙面上に自由書式で書かれた図形を認識できる必要がある。この場合の自由書式とは、各文字やシンボルを任意の場所に任意の大きさに筆記することが可能であることを意味している。この技術が確立されれば、たとえ方眼紙を使用しても、「その方眼は単に筆記者の目安にすぎない」とする筆記条件を設定することが可能となる。

本章では自由書式に筆記された図形で、特にシンボル間でセグメンテーション位置を指定しないような図形を認識する手法を検討する。

3. 筆記用紙に関する調査

右図に示すようなフローチャートを別紙(A,B,C,D)に筆記し、筆記のしやすさを下の記入欄に記入してください。別紙(A,B,C)の場合、各文字は方眼内に記入してください。



但し

- 別紙 A : 10mm方眼紙
- 別紙 B : 8mm方眼紙
- 別紙 C : 6mm方眼紙
- 別紙 D : 白紙用紙

記入欄

用紙	A	B	C	D
図形部分の書きやすさ				
文字部分の書きやすさ				
全体としての書きやすさ				

◎, ○, △, × でお答えください。

4. 何か意見がございましたら御記入ください。

お忙しいところ、ご協力どうもありがとうございました。

図3.2 アンケート用紙

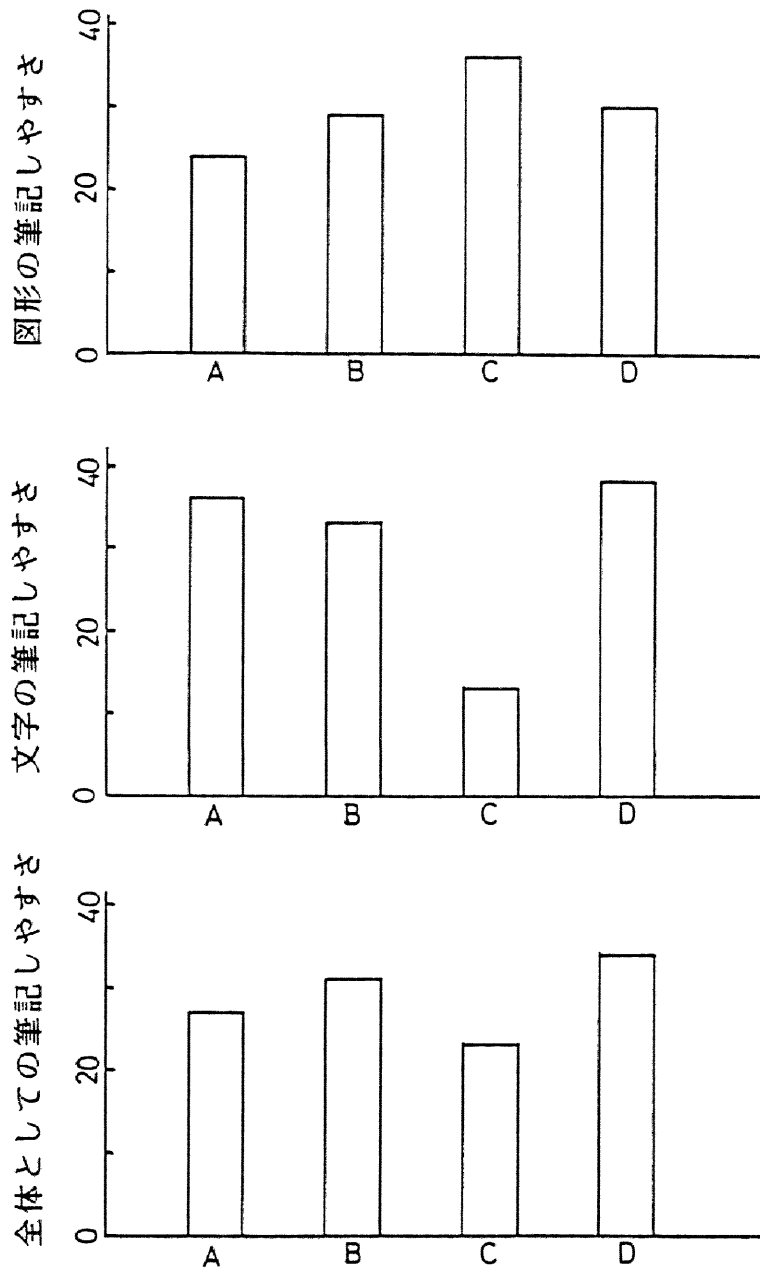


図3.3 筆記し易い用紙（アンケート結果より）



### 3.3 認識対象

本節で述べる認識手法は、シンボルと直線から構成される任意の図形を対象としている。ここでは、説明および実験の都合上、一例として、JISC6270で定められたシンボル（図2.2参照）と直線から構成されるフローチャート及びブロック図を取り上げる。

予備実験あるいは認識実験に使用するため、以下に示す2種類のデータを収集した。

#### (1) 予備実験データ：

5～9個程度のシンボルと直線より構成された5種類のフローチャートデータを収集した。その図形に出現するシンボルの種類は図2.2中\*で示した9種類である。この9種類のシンボルは相互に誤認識しにくく（シンボル間に包含関係がない）、且つ通常のフローチャートを筆記するのにほとんど不自由を感じないものである。それらの図形を6名の筆記者に各4回ずつ筆記させ、合計120枚のデータを収集した。

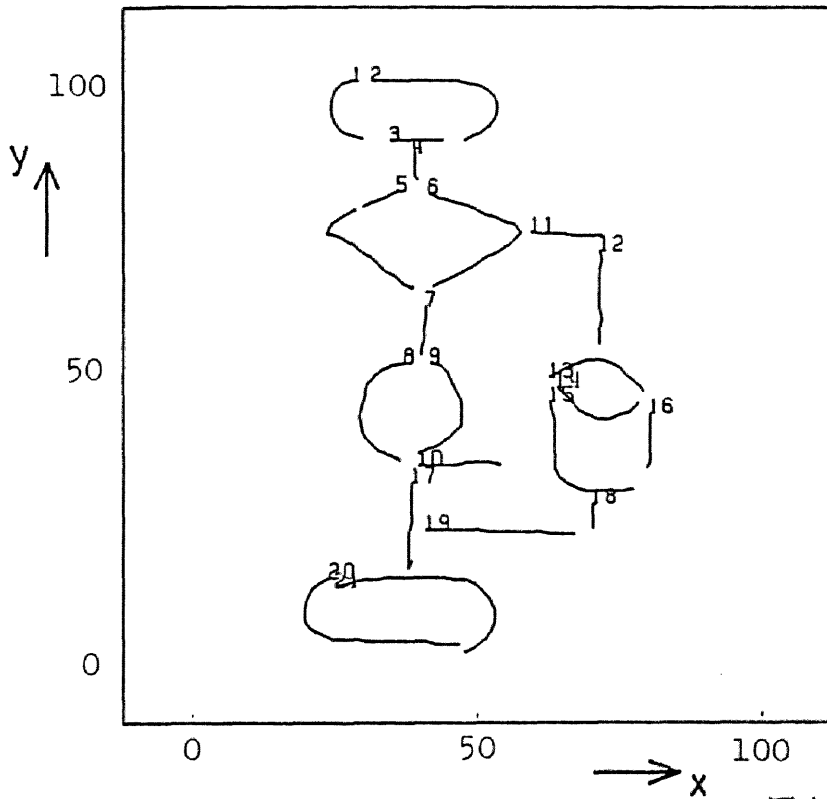
#### (2) 認識実験データ：

6～12個程度のシンボルとそれらを接続する直線からなる5種類の図形（フローチャート3種類、ブロック図2種類）を設定した。それら5種類の図形には図2.2で示した28種類すべてのシンボルの種類が含まれている。それらの図形を5名の筆記者に各4回ずつ筆記させ、合計100枚の図形データ（総シンボル数780）を収集した。

筆記にあたっての指示は以下の3点である。

- (1) 各シンボルの画数と筆順は自由である。
- (2) 各シンボルの位置と大きさは自由である（自由書式である）。
- (3) 1つのシンボルは引き続くストローク列で筆記するものの、シンボル間のセグメンテーションの指示は不要である。

認識実験データの一例を図3.4に示す。



図中の数字はストローク番号

図3.4 入力図形の例

### 3.4 図形中からの候補図形の抽出

候補ラティス法の流れを図3.5に示す。図形中から各シンボルに対する候補図形を抽出する手法を3.4節で、抽出した候補図形から最適な図形系列を選択する候補ラティスの探索を3.5節で述べる。

#### 3.4.1 手書き線図形入力

手書き線図形は、2章で述べたと同様に、データタブレットから入力する。タブレットはペンのアップダウン情報とXY座標値を一定時間間隔で検出する。タブレットから採集されたデータは前処理として、雑音点の除去と、停留点の除去が行なわれる。ペンのダウンからアップまでの座標値系列をストロークと呼ぶ。各ストロークに対して入力順にストローク番号を付与し、以下の処理に使用する。

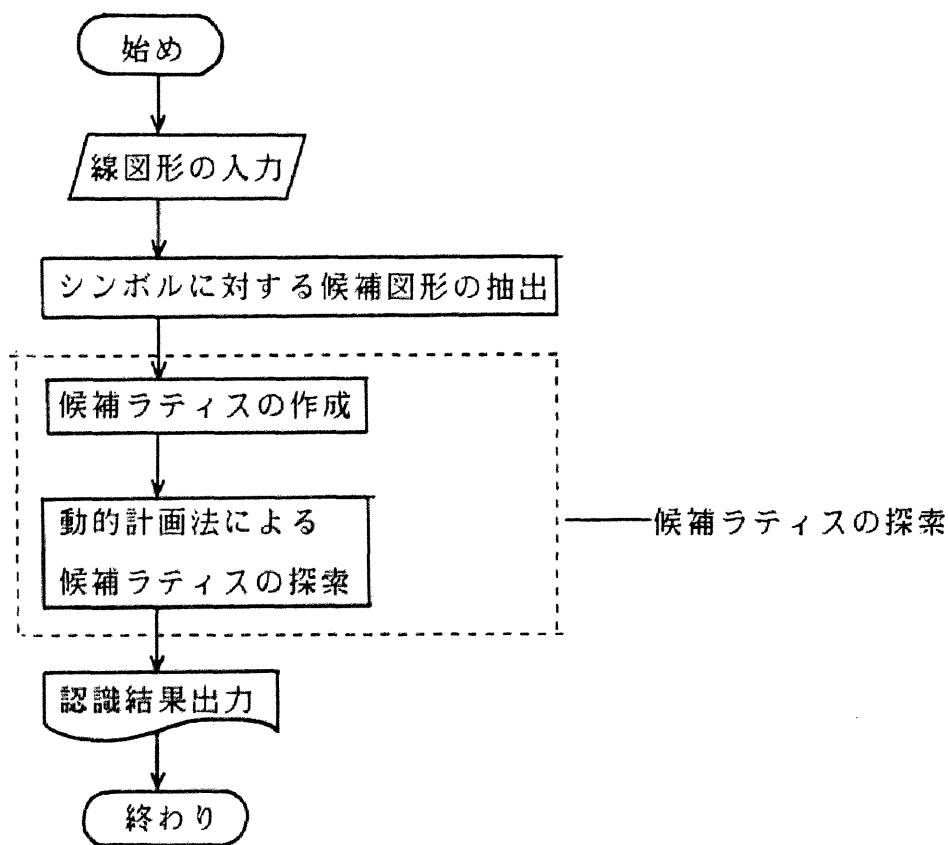


図3.5 候補ラティス法の流れ

### 3.4.2 シンボルに対する候補図形の抽出

入力図形は複数のシンボルと直線から構成されている。候補図形とは、ある種類のシンボル又は直線に対して、それである可能性を持つ入力図形中の部分図形のことである。候補図形抽出のためトップダウン手法を適用する。つまり、図2.2に示した各シンボルの存在を順次仮定し（以下「仮定したシンボル」と呼ぶ）、「仮定したシンボル」となり得る入力図形中の部分図形を、候補図形として抽出する。

そのため、まず入力図形中の部分図形（1組の引き続くストローク列）に着目し、この部分図形が「仮定したシンボル」であるための条件をストロークの端点の位置と接続関係を用いて調べる。条件を満足すればこの部分図形を候補図形とする。その際、部分図形の筆記順序に対応した候補標準ストローク列を自動生成し、任意の画数と筆

順に対処する。候補標準ストローク列の生成は、2章で述べた画数・筆順自由整合法における候補標準ストローク列の生成手法を利用する。

この処理を順次、他の部分図形に対し適用することにより候補図形の抽出は終了する。以下に各処理を説明する。

### 3.4.3 候補図形の抽出手順

図3.4で示した手書き図形から、シンボル「process」に対する候補図形を抽出する場合を例に取り、処理の基本手順を以下に具体的に説明する。なお、この処理の中で(1)(3)(4)は、2章で述べた画数・筆順自由整合法における候補標準ストロークの生成の処理を基本にして、候補図形の抽出に使用している。内容で不十分な点は2章を参照されたい。

(1) 「process」を「仮定したシンボル」とする。「仮定したシンボル」を候補端点をノードとし、候補端点間で接続した箇所をブランチとするグラフで表現する。但し、候補端点とは、そのシンボルを筆記する際のストロークの端点が存在する可能性のある点のすべてであり、2章で記述した候補端点と同一のものである。

(2) 入力図形のある部分図形(例えば、ストローク番号1, 2から構成される部分図形)に着目する

(3) 着目した部分図形の各ストロークの端点を候補端点に対応付ける。この処理は、「仮定したシンボル」の大きさを正規化し(縦横独立に拡大縮小を行なう)着目した部分図形を重合わせた後に、ストロークの各端点をそれぞれ最近傍の候補端点に対応付ける。

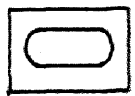
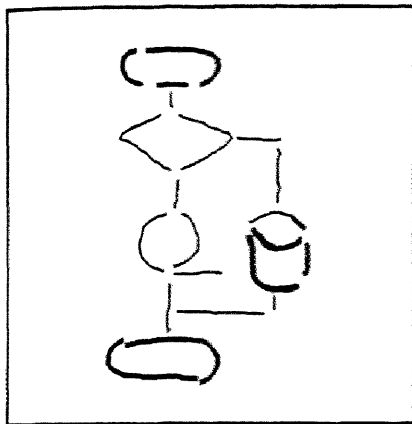
(4) (3)で対応付けられた結果を満足するパス(ペンの動きに対応)を全て列挙する。このパスを候補標準ストロークと呼び、3.4.6で利用する。この処理は、有向グラフ(「仮定したシンボル」)上でオイラーパスを探索する処理により実現できる。これら(3)(4)の処理で矛盾が生じない場合には、着目した部分図形を候補図形とする。

(5) 同様の処理を各部分図形に対しても行い、すべての候補図形を抽出する。例えば入力ストローク図形が23ストロークから構成され、「Process」の候補図形を抽出する場合を考える。「Process」は最大4画、最低1画で書かれる可能性があるので、4画の場合にはストローク番号が(1,2,3,4), (2,3,4,5), ..., (20,21,22,23)の

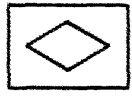
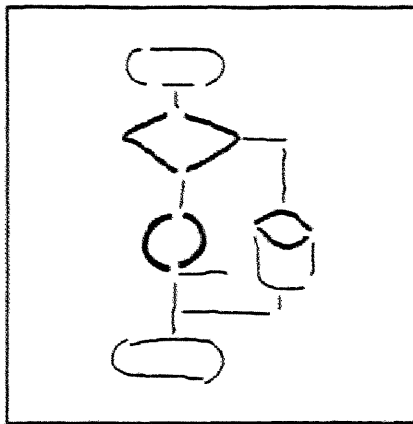
20組のストロークの組（部分図形に対応している）に対し、3画の場合には21組、  
．．、1画の場合には23組のストロークの組に対して、つまり全体で86組のスト  
ロークの組に対して上記の処理を行なう。

以上が、候補図形抽出の基本的な手順である。実際には、候補図形の抽出をより効  
率的に且つ精度を向上させるため、この基本手順に加えて、以下に説明するシンボル  
の縦横比による候補図形の棄却（3.4.4）、候補端点の連結処理（3.4.5）  
を導入している。

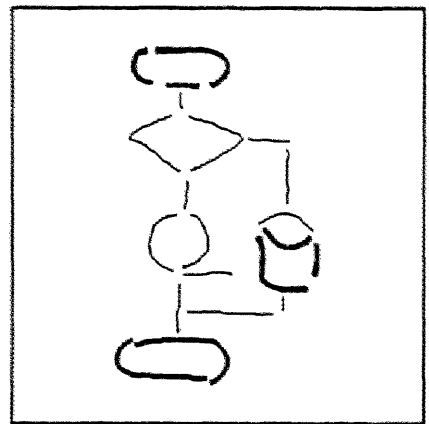
この節で述べた候補図形抽出法により、図3.4のフローチャート図形から各シン  
ボルに対する候補図形を抽出した例を図3.6に示す。この手法ではシンボルの位相  
構造的な情報を利用して候補図形を抽出している。「Process」や「Input/Output」  
などは1重のループから構成され、位相構造的には両者は類似している。またこれら  
のシンボルはフローチャートに比較的多数出現する。そのため、実際に入力図形に  
それらのシンボルが出現していなくても、それらのシンボルに対する候補図形は多数  
抽出されることになる。一方、「Magnetic disk」などの構造の複雑なシンボルは、  
位相構造的に他に類似するシンボルがなく、これらに対する候補図形の抽出数は少数  
となる。



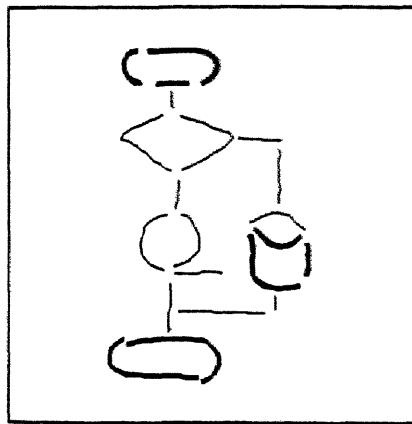
Terminal



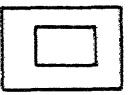
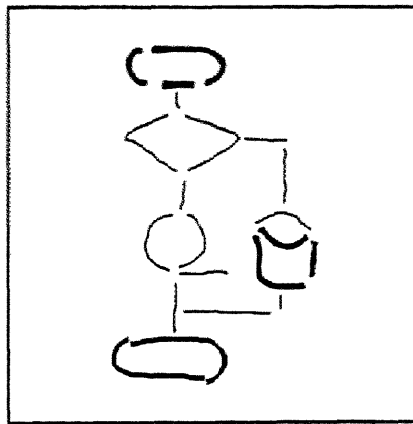
Decision



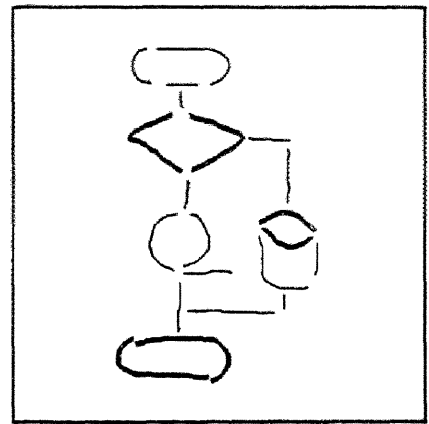
Input/Output



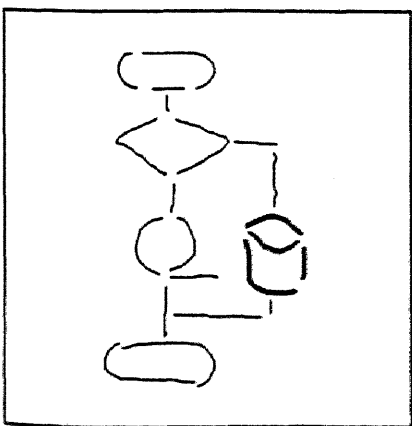
Document



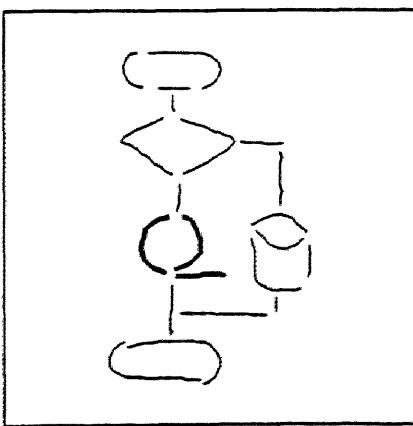
Process



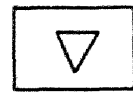
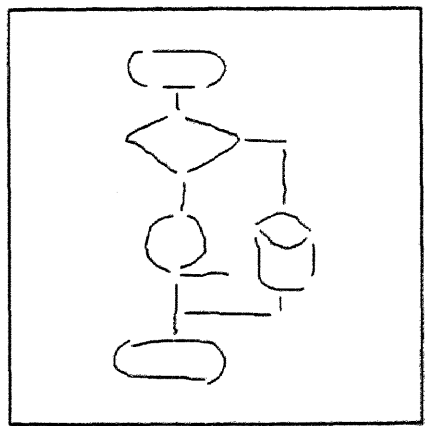
Display



Mag. disk



Mag. tape



Merge

図3.6 候補図形の抽出

### 3.4.4 シンボルの縦横比による棄却

入力図形中の各シンボルの形状は、縦横比に対して自由である。しかし、例えば、「Magnetic tape」の記号を、縦横比5以上の縦長で筆記することは考えられない。着目した部分図形が明かに、あるシンボルの候補図形でないにもかかわらず、3.4.3で示した処理により候補図形を抽出することは無意味である。そこで、着目した部分図形の縦横比が極端に大または小となる場合には3.4.3の処理を行わず棄却する。

部分図形を構成するストロークの組の座標値系列を

$$\{ (X_{nm}, Y_{nm}) \mid n=1, N; m=1, M \}$$

とする。但し、Nは各ストロークの近似点数、Mはストローク数である。縦横比Rを、

$$R = \frac{\left( \text{MAX}_{n,m} \{ Y_{nm} \} - \text{MIN}_{n,m} \{ Y_{nm} \} \right)}{\left( \text{MAX}_{n,m} \{ X_{nm} \} - \text{MIN}_{n,m} \{ X_{nm} \} \right)} \quad (3.1)$$

で定義し、縦横比Rが、

$$R > R_{MAX} \quad \text{又は} \quad R < R_{MIN} \quad (3.2)$$

の条件を満たす時に棄却する。 $R_{MIN}$ 、 $R_{MAX}$ はシンボル毎に経験的に定め、シンボルの辞書に格納する。

### 3.4.5 候補端点連結処理の影響

候補端点間が近接しているようなシンボルでは、微小な手書き変形で入力ストロークと候補端点との対応付けが誤り、正しくシンボルの認識が出来ないことがある。これについては2章で述べた。この問題と同様のことは候補図形の抽出処理でも発生する。ここでは、2章で述べたと同様の候補端点連結処理を導入し、その効果を検討する。

2. で述べた予備実験データに対して候補図形抽出実験を行った。候補端点連結処理を行った場合とそうでない場合についての候補図形抽出率を表3.1に示す。但し、候補図形抽出率は、入力図形に含まれるシンボル数に対する候補図形に含まれる正解

シンボル数の割合で定義した。候補端点連結処理により、候補図形抽出率は大幅に改善された。この結果より候補端点連結処理は、候補図形抽出の際にも効果があることを確認した。

表 3. 1 候補図形抽出率（予備実験データ）

候補端点連結処理を行わない場合	候補端点連結処理を行なった場合
95.4%	99.6%

### 3. 4. 6 DPマッチングによる相異度の計算

候補図形の抽出段階では、入力ストロークの端点位置とその接続関係のみを利用し、入力ストローク列の細かい形状については考慮していない。そこで、ストローク間のDPマッチングによる形状の相異度を導入する。ここで使用する相異度は、2章のシンボル認識の際に最も高い認識率を与えた以下の相異度を導入する（2. 4参照）。、

3. 4. 3で生成された候補標準ストローク列の1ストロークの座標値系列を  $\{P_1, P_2, \dots, P_M\}$  ( $M$ は点数) とし、入力ストローク列のこれに対応するストロークの座標値系列を  $\{Q_1, Q_2, \dots, Q_N\}$  ( $N$ は点数) とすれば、ストローク間の距離  $d$  は、DPマッチングにより、

$$d = \frac{1}{M} \min_u \left\{ \sum_{i=1}^M e(P_i, Q_{u(i)}) \right\} \quad (3.3)$$

で計算される。但し、 $e(P_i, Q_j)$  は点  $P_i$  と点  $Q_j$  の間のユークリッド距離と、その点での接線方向の方向差の加重和とする。 $u(i)$  は座標点間の対応関係を表わし、

$$\begin{cases} u(1) = 1 & (3.4) \end{cases}$$

$$\begin{cases} u(M) = N & (3.5) \end{cases}$$

$$\begin{cases} u(i) = j \text{ のとき } u(i+1) = \{j \text{ or } j+1 \text{ or } j+2\} & (3.6) \end{cases}$$



の関係を制約条件にする。

候補図形の相異度は、図形を構成する各ストロークとそれに対応する候補標準ストロークとの距離の総和で定義する。

### 3.5 候補ラティスの探索

#### 3.5.1 候補ラティスの作成

候補ラティスは、入力図形中の各ストロークがどの候補図形の一部として抽出されたかを表形式に表現したものである。まず1つのシンボルに対し、入力図形中から候補図形を抽出する。候補図形はストロークの番号列で得られる。そこで、候補ラティス内の対応するストローク番号の位置に、候補図形の名称（シンボル名）とその相異度を登録する。他のシンボルに対してもこの処理を繰り返す。又、すべてのストロークは直線の候補図形でもあるため、これも合わせて登録する。一例として図3.4の入力図形に対する候補ラティスの一部を図3.7に示す。

stroke number	8	9	10	11	12	13	14	15	16	17
	—	—	—	—	—	—	—	—	—	—
	0.31	0.29	0.01	0.01	0.01	0.17	0.20	0.32	0.02	0.02
candidate symbol	→ ○					◇				
distance	0.15					0.14				
			◇					○		
			0.24					0.28		
								○		
								0.45		
								○		
								0.45		
								◇		
								0.51		

図3.7 候補ラティス

### 3.5.2 動的計画法による候補ラティスの探索

入力図形中に含まれるシンボルのセグメンテーションおよび、シンボル名の識別を行うこと（入力図形の認識）は、候補ラティスの中からすべてのストロークを被う最適な図形系列（シンボルおよび直線からなる系列）を選択することにより実現できる。一般には、候補ラティスの中から選択可能な図形系列の組み合わせは多数存在する。そこで、第一段階として評価関数  $S$

$$\text{評価関数 } S = \frac{(\text{相異度})}{(\text{ストローク数})} \quad (3.7)$$

を導入し、その評価関数の値を最小とする図形系列を探索し、この図形系列を認識結果とする。ラティスの探索は動的計画法（DP）によって容易に実現することが可能である。

一例として、図3.7の候補ラティスを探索すると太枠の候補図形が認識結果となる。表3.2は図3.4に示した図形を認識した結果を表で示した例である。この表には、シンボルについてはその大きさと位置に関する情報が、また直線についてはその始点位置と終点位置に関する情報が含まれている。図3.8はこの認識結果に基づき、図形の認識結果を表示した例である。各シンボルの大きさと位置および直線の始点位置と終点位置は、入力図形中のそれに合わせて表示してある。

表 3. 2 認識結果の例

ストローク番号	シンボル名	位置	大きさ
1,2,3	Terminal	(39,94)	(29,10)
4	line		
5,6	Decision	(40,71)	(34,17)
7	line		
8,9,10	Mag. Tape	(42,41)	(24,18)
11	line		
12	line		
13,14,15,16	Mag. Disk	(71,39)	(17,23)
17	line		
18	line		
19	line		
20,21	Terminal	(36,6)	(33,13)

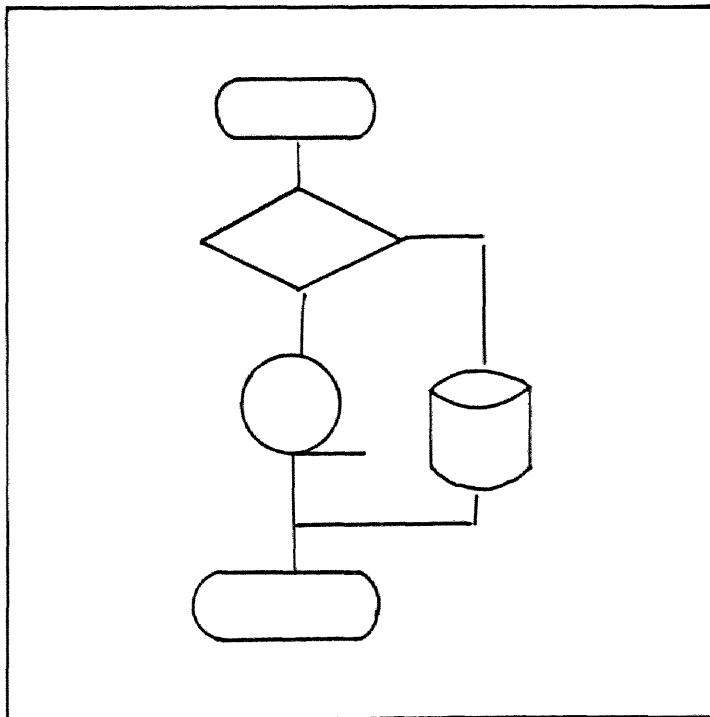


図 3. 8 認識結果

動的計画法によって、候補ラティスを探索する手法を以下に示す。評価関数を最小にするように候補ラティスを探索することは、候補ラティスをグラフと考えて、そのグラフの最短パスを求めることにより実現される。最短パスの探索は、ネットワーク理論における基本的な問題の一つである。特に、ラティス型のグラフ（2端子グラフ）の最短経路は、動的計画法で効率良く解くことができる【86】。以下に処理を説明する（図3.9を参照）。

記号の定義：

- (1) 候補ラティスのノードの数をM、ノード名を1、2、...、Mとし、始点から終点に向かって、この順に並んでいるとする。
- (2)  $T(i, j)$  をノードiとノードjを接続するブランチの重みとする。もしノードiとノードjを接続するブランチがなければ、 $T(i, j)$  は無限大の値とする。
- (3)  $U(i)$  をノードiからノードMに至る最短パスの重みとする。 $U(1)$  は候補ラティスの最短経路の重みとなる。
- (4)  $L(i)$  をノードiからノードMに至る最短パスがノードiの次に通過するノード番号とする。
- (5) ノードiからノードMに至る最短パスは、ノードiからノード $L(i)$ に至るブランチを通過する。 $P(i)$  をそのブランチに対応した候補図形の名称とする。

処理は、動的計画法により、各段階での最適解を再帰的に決定することにより実現できる。以下に、具体的な処理ステップを述べる。

・ステップ 1

初期設定として次の代入を実行する。

$$i = M - 1 \tag{3.8}$$

$$U(M) = 0 \tag{3.9}$$

・ステップ 2

$U(i)$  を以下の式で求める。

$$U(i) = \underset{i < j \leq M}{\text{MIN}} (T(i, j) + U(j)) \tag{3.10}$$

つまり、 $U(i)$  は、 $j$  が  $(i+1)$  から  $M$  までの  $U(j)$  の値を再帰的に用いることにより計算される。

$L(i)$  は  $U(i)$  を決定する（最適解を抽出する）際に用いられた  $U(j)$  の  $j$  の値に設定される。

$P(i)$  は、 $i$  と  $j$  を結合するブランチに対応した文字名称に設定される。

・ステップ 3

$i$  の値を 1 ずつ減少させステップ 2 を繰り返す。これにより、 $U(i)$ 、 $L(i)$ 、 $P(i)$  ( $1 \leq i \leq M-1$ ) が求められる。

・ステップ 4

$U(1)$  が最短パスに対応したブランチの重み和となる。最短パスは、 $L(i)$  のポインタをノード 1 ( $i=1$ ) からノード  $M$  までたぐることにより得られ、認識結果はそれに対応した  $PN(i)$  を抽出することにより得られる。

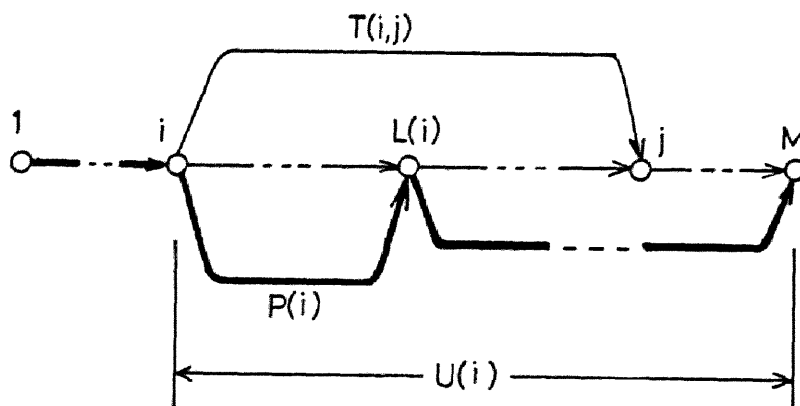


図 3. 9 ラティスの最短経路の探索

### 3.6 図形のオンライン認識実験

4. で述べた認識アルゴリズムを用いて、手書き図形の認識実験を行った。認識率を表3.3に示す。但し、認識率は、

$$\text{認識率} = \frac{(\text{正しく認識されたシンボル数})}{(\text{入力図形中に含まれるシンボル数})} \times 100 (\%) \quad (3.11)$$

で定義した。

表3.3 図形の認識率

認識対象	認識率
予備実験データ	97.9%
認識実験データ	89.8%

ここで、予備実験データが認識実験データに比較して認識率が高い理由は、予備実験データではシンボル間の包含関係が少なく、且つシンボル間で形状的に混同しないようにシンボルセットを選択したため、シンボルのセグメンテーション誤りやシンボルの認識誤りが少なかった為である。これらの誤認識の分析は3.7.3節で述べる。

結論としては、フルセットのシンボルから構成される100枚のフローチャートとブロック図(3.3節で述べた認識実験データ)に対して、89.8%の認識率が得られた。

### 3.7 考察

#### 3.7.1 認識時間

ミニコン(1MIPS)上でFORTRAN77を用いシミュレーション実験を行った際のストローク数と認識時間の関係を図3.10に示す。これより、候補ラティス法による認識時間は1ストローク当り5秒程度で、ストロークの増加にほぼ比例していて、

組み合わせ的な増大はないことが分かる。更に、処理時間の約80%はDPマッチングに費やされているので、この箇所をハードウェア化することにより認識時間の短縮が容易である。

図形を筆記する際、筆記時間はほぼストローク数に比例する。オンライン認識の場合、筆記と同時にストロークを認識装置に取り込み認識処理の多くの部分（処理時間の大半をしめる候補図形の抽出と相異度の計算）を実現することが出来る。そのため、1ストローク当たりの平均認識時間を、1ストローク当たりの平均筆記時間以下で実現できれば、多くのストローク数から構成される大きな図形の認識も、実時間で認識することが可能である。

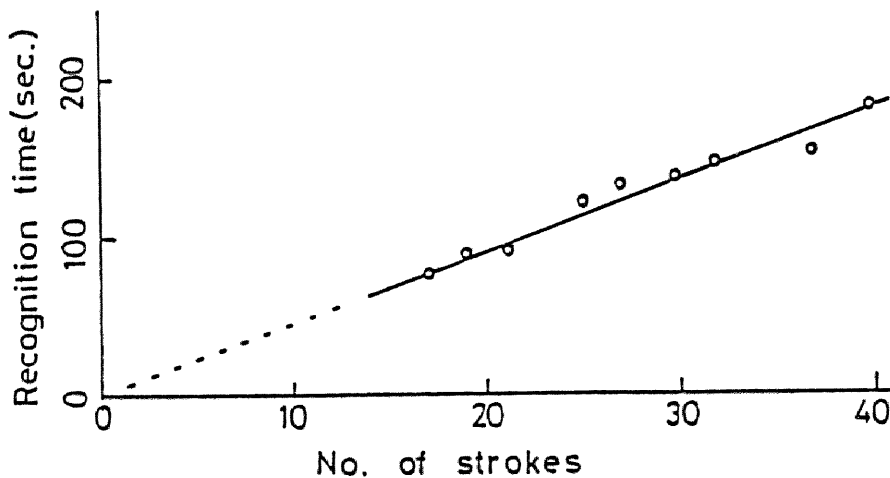


図3. 10 ストローク数に対する認識時間

### 3. 7. 2 候補図形の抽出の効率化

3. 7. 1 節で述べたように、認識時間の大半は、候補図形と標準図形との間で相異度を計算するDPマッチングに費やされている。そこで、不要な候補図形を棄却し、処理量を低減する手法について考察する。但し、候補図形を過剰に棄却することは正しい候補図形をも棄却することになり、認識率の低下を招く。ここでは、最適な棄却パラメータを選定するため、以下の2点を検討する。

### (1) 候補端点对応付けでの棄却

入力図形の部分図形を構成するストロークの端点を、「仮定したシンボル」の候補端点对応付ける際に、この両者の端点間の距離が、あるしきい値にない場合には、この候補図形を棄却する手法を検討する。

120枚のフローチャートデータ（データ1）を使用し、端点間の距離のしきい値をパラメータとして、認識率と平均候補図形数を求めた。その結果を図3.11に示す。但し、平均候補図形数は、入力図形中のシンボル1個に対して抽出された候補図形数の平均値である。また、端点間の距離は着目した部分図形の大きさ（図形を囲む長方形の長辺の長さ）を1.0とした場合の相対的な値である。

図3.11より、認識率を下げないしきい値の範囲から、0.35を選択することが可能である。その結果、平均候補図形数は8.3個から6個に減少する。この処理により棄却される候補図形については処理を途中で打ち切ることができ、棄却された候補図形については、相異度の計算、および候補ラティスへの登録処理は不要となる。

### (2) 相異度計算での棄却

相異度の大きな候補図形は、候補ラティスに登録する必要はない。そこで、あるしきい値以上の相異度を持つ候補図形を棄却する。相異度のしきい値をパラメータとし、認識率と平均候補図形数を求めた。その結果を図3.12に示す。図3.12からわかるように、認識率を下げないしきい値の範囲から0.1を選択することが可能である。その結果、平均候補図形数は6個から2.8個に減少する。この処理により、相異度がしきい値を越えた時点で相異度計算を打ち切り、対象とする候補図形を棄却することができる。そのため棄却される候補図形については、相異度の計算時間の短縮、および候補ラティスへの登録処理は不要になる。

以上説明した(1)(2)の処理を候補ラティス法に組み込み、認識のシミュレーション実験を行なったところ、認識率を下げることなく、候補ラティス法の全処理時間を約50%低減することが可能となることを確認した。



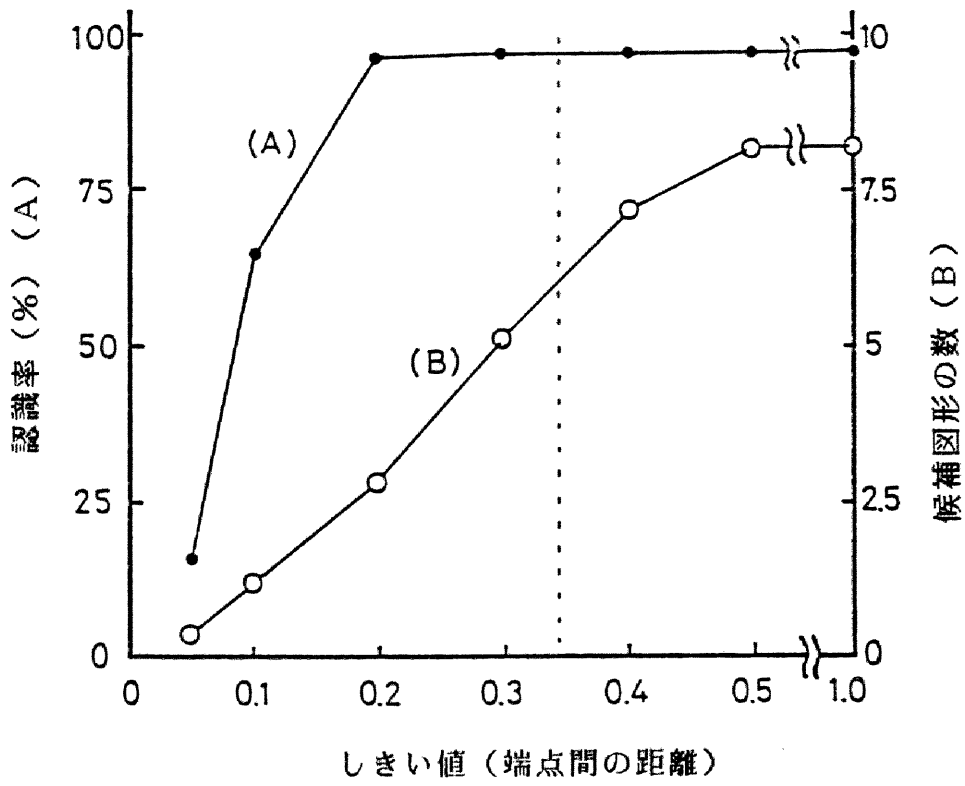


図3. 11 候補端点对应付けでの棄却

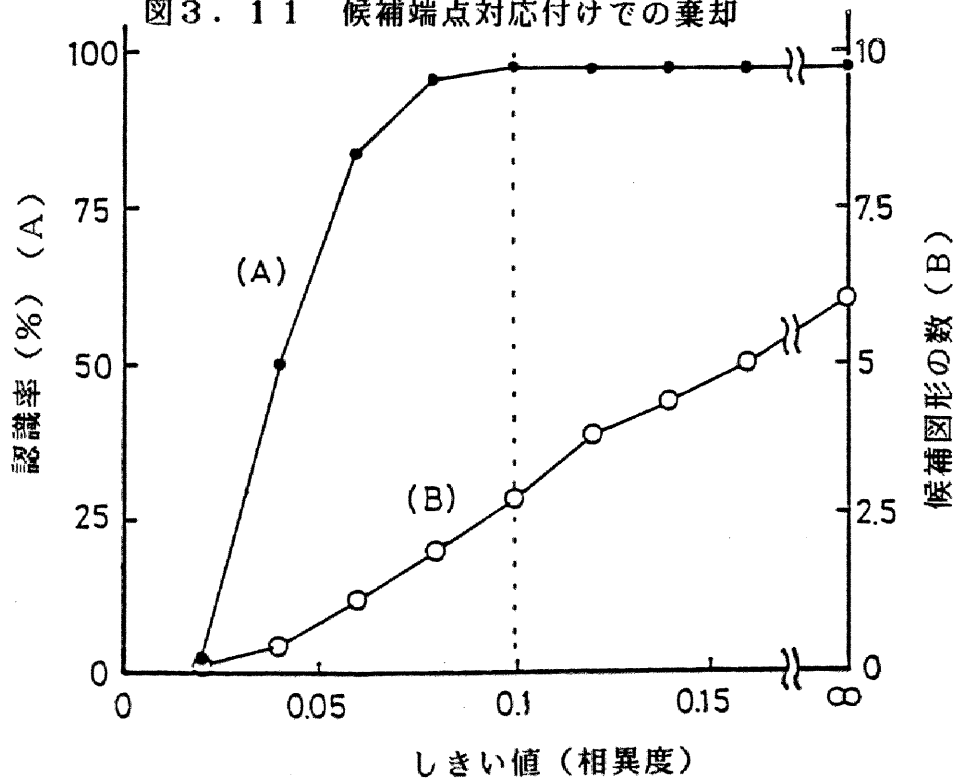


図3. 12 相異度計算での棄却

### 3.7.3 誤認識の分析

誤読例を分析すると、セグメンテーションの失敗とシンボル名の誤読に大別される。以下に、分析内容を述べる。

(1) セグメンテーションの失敗は2.6%存在した。その内訳を表3.4に示す。一例を図3.13に示す。主な原因は、一方のシンボルの部分図形が他のシンボルであるというシンボル相互間の包含関係にある。今回使用した認識対象に対するシンボル間の包含関係を表3.5に示す。包含関係にあるシンボルの認識では、両者の評価関数の値は類似するため、誤読する危険が高い。

表3.4 セグメンテーションの失敗例

「core」 -----> 「process」 + 2本の直線	0.9%
「sort」 -----> 「decision」 + 直線	0.3%
「Predefined process」 ---> 「Process」 + 2本の直線	0.2%
その他	1.2%

表3.5 シンボル間の包含関係

被包含シンボル	包含シンボル
「Predefined process」	「Process」
「Collate」	「Merge」 「Extract」
「Sort」	「Decision」 「Merge」 「Extract」
「Offline strage」	「Merge」
「Deck of cards」	「Punched card」
「File of cards」	「Punched card」
「Magnetic tape」	「Connector」
「Core」 「Process」	

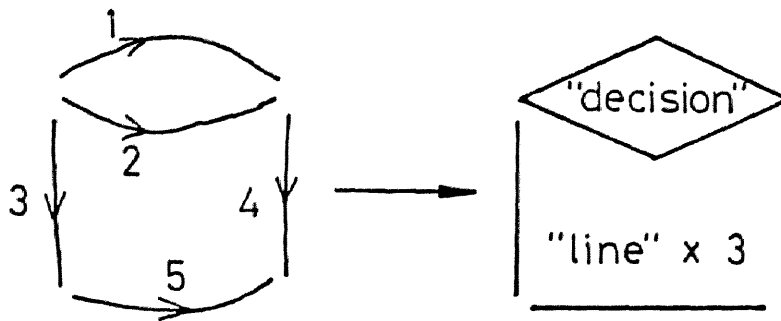


図 3. 13 誤認識例

(2) シンボル名の誤認識は 7.9% 存在した。代表的な失敗には、「terminal」を「preparation」に誤認識した例 2.3%、「connector」を「preparation」に誤読した例 2.3% 等があげられる。原因はシンボル間の形状の類似にある。誤読例は、人間でも手書きシンボルを単独に提示されたら誤読するような例である。

上記の 2 種類の誤りが発生する理由は、シンボル個々の形状しか利用していないことによる。人間が図形からシンボルを正しくセグメンテーションし、更に各シンボルを正しく認識できる理由は、人間はシンボル個々の形状のみならず、図形の全体情報を利用している為である。図形の全体情報としては、例えば、シンボル間の接続関係や、図形の持つ意味などが考えられる。

### 3. 8 結言

手書き線図形の認識は、手書き図形の自動清書や手書き図形による計算機との対話等、幅広い応用範囲を持つ。この章では、複数のシンボルと直線から構成される手書き線図形を対象に、筆記制限の少ないオンライン線図形認識法として、「候補文字ラティス法」を提案した。

候補ラティス法の特長は、

(1) 認識アルゴリズムが図形中から自動的にシンボルを切り出すことが可能である。つまり、シンボル（基本図形）間のセグメンテーションを人間が指示する必要がなくなる。

(2) 各シンボルを自由な画数と筆順で筆記することが可能である。

この両者の特長により、オンライン認識型の入力は非常に扱い易いものとなる。

この章での結果を以下にまとめる。

(1) 入力図形の内部からシンボルらしい部分図形を候補図形としてすべてを安定に抽出する手法を提案した。各シンボルの形状の構造を有向グラフで表現し、有向グラフのパスを探索することにより、任意の画数と筆順で筆記されたシンボルに対して候補図形を抽出することが可能になった。構造の複雑なシンボルに対しても安定な候補図形抽出を実現するため、候補端点連結処理を導入した。100枚の図形データに対して、候補図形抽出率は99.6%を達成した。

(2) セグメンテーションとシンボルの認識を同時に実現する候補ラティス法を考案した。これは、シンボルの名称と相異度を登録した候補ラティスから、相異度の和を最小とするシンボル系列を探索し、これを認識結果とする手法である。この手法により入力図形を、図形全体の形状に着目した最適なシンボル及び直線の系列に認識することが可能になった。100枚の図形データに対して89.8%の認識率を得ることができた。

(3) 候補図形の抽出処理を効率化し、全体の計算時間を短縮する手法を提案した。具体的には、ストロークの端点を候補端点に対応付ける際に、端点間の距離で棄却する処理の導入と、相異度の計算でその値により棄却する処理の導入である。その結果、全体の計算時間を50%短縮することが可能となった。

以上、候補ラティス法により自由書式で筆記された線図形を認識することが可能となった。しかし、この手法には、包含関係にあるシンボル間で本質的にセグメンテーションの誤りが発生すること、類似形状のシンボル間で認識の混同が生じやすいことなどの問題点がある。第4章では、包含関係にあるシンボル間でのセグメンテーションの失敗や、類似形状のシンボル間での誤読に対処するため、上位概念としてシンボル間の接続関係を利用した「接続ルールを導入した候補ラティス法」を提案し、認識精度の向上を図る。

## 第4章 接続関係知識を導入した 手書き図形の高精度認識

### 4.1 緒言

3章では、図形を手書きする際の筆記制限の緩和について特に検討をした。そして、手書きフローチャートを対象に、任意の画数と筆順での筆記を許し、且つシンボル間のセグメンテーション指示を不要とする図形認識法として候補文字ラティス法を提案し、有効性を確認した。しかし、この手法には、

(1) 形状の類似した他のシンボルに誤認識しやすい。

(2) 包含関係にあるシンボル間でセグメンテーションの失敗を招きやすい。

等の問題点が残されていた。

上記の誤りが発生する理由は、図形を認識する際に、シンボル個々の形状しか利用していないことによる。人間がかなり複雑な図形であっても、図形中からシンボルを正しくセグメンテーションし、更に各シンボルを正しく認識できる理由は、人間はシンボル個々の形状のみならず、図形の全体情報を利用している為である。図形の全体情報としては、例えば、シンボル間の接続関係や、図形の持つ意味などが考えられる。

この章では、上位概念としてシンボル間の接続ルールを導入した手書き図形の認識手法について述べる [54, 55, 59, 60, 61]。これにより、シンボル間の接続関係や、各シンボルの有する機能をも認識に有効に利用することが可能になり、高精度な図形認識が達成される。

認識アルゴリズムは、次に示す3段階から構成した(図4.1参照)。

(1) 候補図形の抽出：入力図形の内部からシンボルらしい部分図形を候補図形としてすべて抽出する。その際、シンボル形状の構造を有向グラフで表現し、そのグラフのパスを探索することにより、任意の画数と筆順に対処する。

(2) 候補ラティス：抽出されたすべての候補図形に対しDPマッチングにより形状の相異度を計算し、候補図形のシンボル名称と相異度を候補ラティスに登録する。候補ラティスから相異度の和を最小とするシンボル系列を探索し、これを”仮の認識結果”とする。つまり、シンボルのセグメンテーションと認識を同時に行う。

(3) 接続ルール：”仮の認識結果”における各シンボルの接続関係を記述する。この接続関係が上位概念として導入した接続ルールに矛盾する場合には候補ラティス

を修正し再度シンボル系列の探索を行う。この処理を反復することにより、接続ルールに矛盾しないシンボル系列を最終認識結果として得る。

本章での主眼点は図4.1中の太枠で囲んだ箇所にある。シンボル間の接続関係を記述し、接続関係に矛盾が生じる場合にはそれを修正する処理を導入した。これにより、シンボルの切り出し精度と、認識精度を向上させることが可能となる。

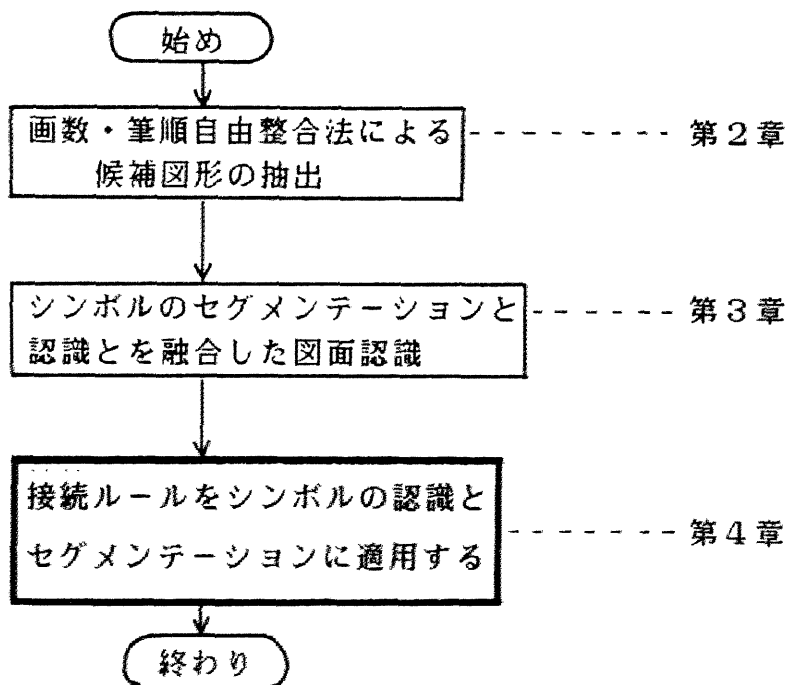


図4.1 本章の主眼点

以下、4.2節で接続ルールを導入した図形認識アルゴリズムについて、4.3節で認識実験の結果を、4.4節で接続ルール導入の効果や、この手法を論理回路図などの他の図形に適用した場合の拡張性などについての検討を述べる。

## 4.2 接続ルールを導入した図形認識法

### 4.2.1 接続ルールを導入した認識アルゴリズム

この節では、接続ルールを導入した図形の高精度認識手法について述べる。この手

法は3つの特長を持つ。

- ①各シンボルの画数と筆順は任意である。
- ②各シンボルのセグメンテーションの指示を気にすることなく筆記できる。
- ③接続ルールを導入した高精度な認識が可能である。

つまり、3章で述べた①②の特長に加えて、新規に③の特長を追加した。処理のブロック図を図4.2に示す。処理は、入力図形から候補図形を抽出して候補ラティスを作成する段階と、候補ラティスから接続ルールを考慮しながら全体として矛盾のないシンボルの組み合わせを選択する段階から構成した。また、このブロック図からもわかるように、処理は「標準図形・接続ルールに関する知識のブロック」と「制御のブロック」とに分離されている。そのため、知識の部分のみを交換することによって、この手法を種々の図形認識に容易に適用することが可能である。

認識アルゴリズムを手続き的に示した流れ図が図4.3である。以下、流れ図に従って説明する。

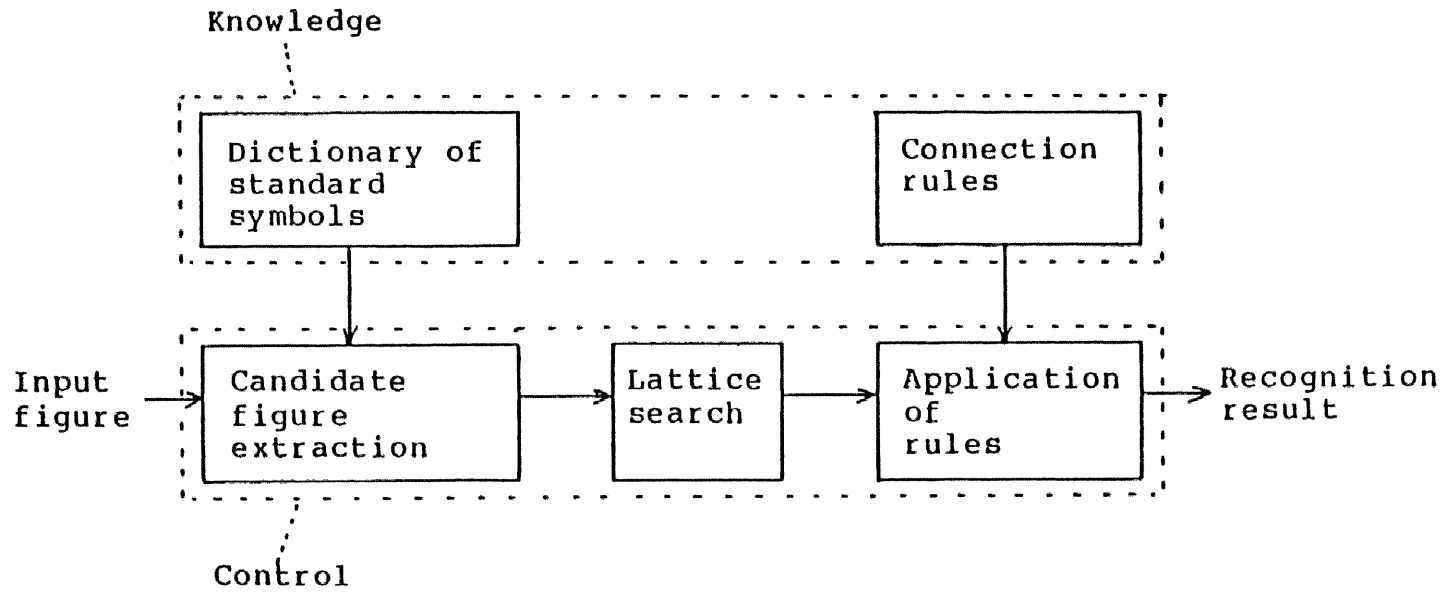


図4.2 システムブロック図



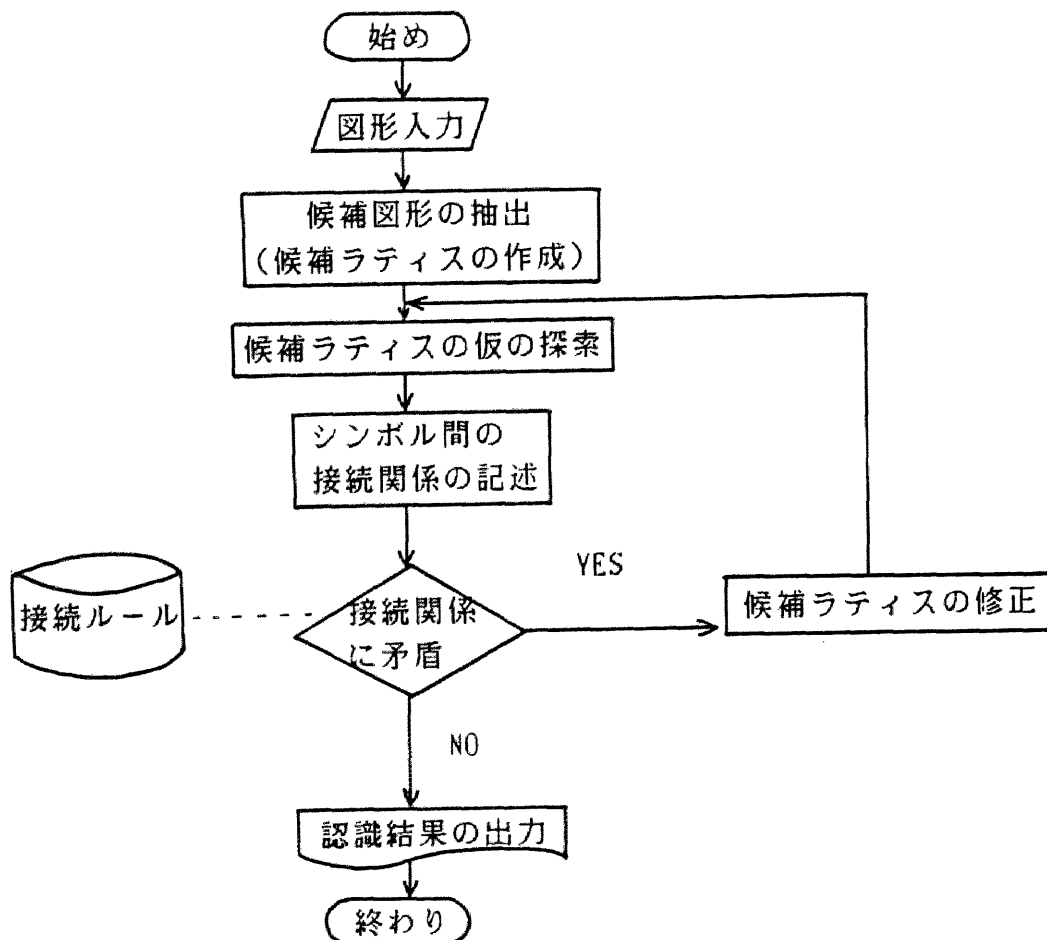
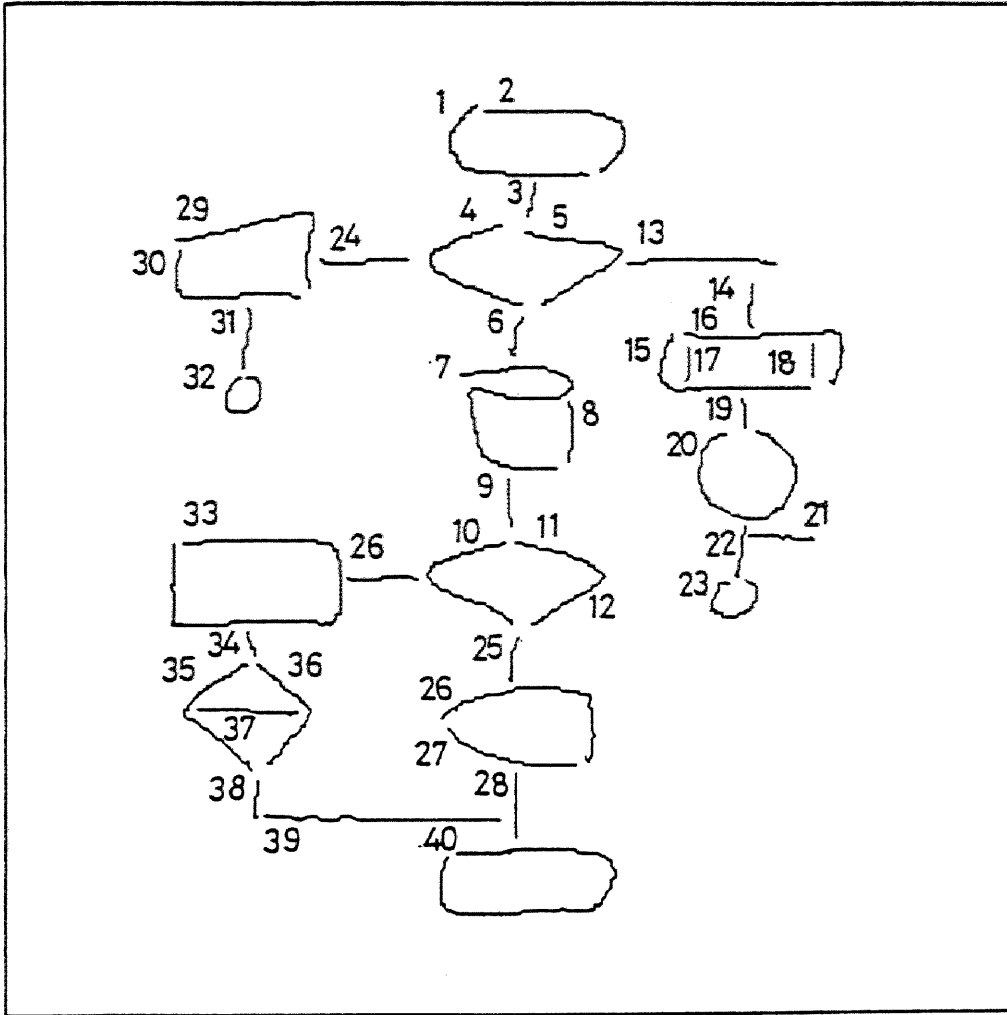


図4.3 接続ルールを導入した図形認識アルゴリズム

#### 4.2.2 手書き図形入力

手書き線図形は3章と同様にデータタブレットから入力する。タブレットから入力されたデータは、前処理として、雑音点の除去、停留点の除去がなされ、以下の処理に使用される。各ストロークには入力順にストローク番号を付与する。入力図形の例を図4.4に示す。



図中の数字はストローク番号

図4.4 入力図形の例

#### 4.2.3 候補図形の抽出

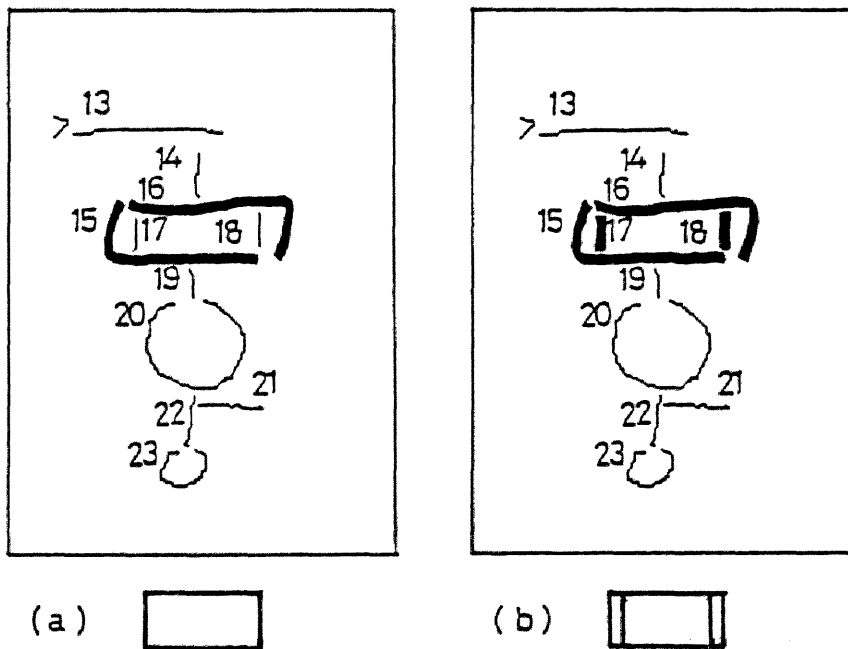
入力図形はシンボルと直線から構成される図形である。この入力図形中から、あるシンボルに対して、そのシンボルである可能性を持つ入力図形中の部分図形を抽出し、それを候補図形とする。候補図形の抽出は、ストロークの端点の位置とその端点間の接続関係を調べ、入力図形中で、端点間の接続関係に関する条件を満足する部分図形を探索することにより行なわれる。

「Process」に対する候補図形を図4.5(a)に示す。また、「Predefine proc

cess」に対する候補図形を図4.5(b)に示す。

抽出された候補図形に対しては、標準シンボルと候補図形との間で、ストロークの方向とユークリッド距離に着目したDPマッチングにより、相異度を計算する。相異度は、候補図形がどの程度標準のシンボルに近いかを示す量である。

次に、候補図形と、それに対する相異度を用いて、候補ラティスを作成する。候補ラティスは、入力図形中の各ストロークがどの候補図形の一部として抽出されたかを表形式に表現したものである。ある一つのシンボルに対する候補図形はストロークの番号列で得られる。そこで、候補ラティス内の対応するストローク番号の位置に、候補図形の名称(シンボル名)とその相異度を登録する。他のシンボルに対してもこの処理を繰り返す。又、すべてのストロークは直線の候補図形でもあるため、これも合わせて登録する。一例として図4.4の入力図形に対する候補ラティスの一部を図4.6に示す。



図中の数字はストローク番号

図4.5 候補図形の抽出

(a : Process, b : predefined process)

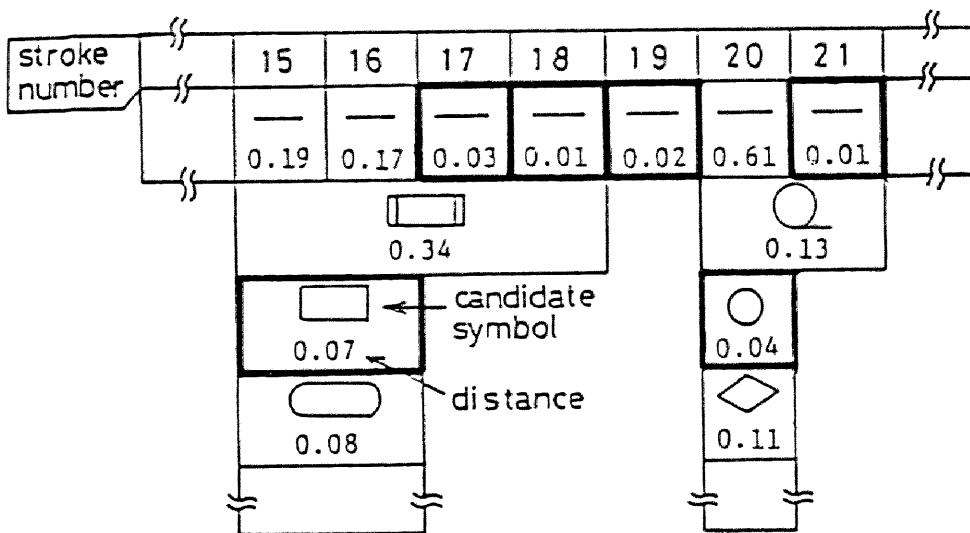


図4.6 候補ラティス

#### 4.2.4 候補ラティスの仮の探索

入力図形中に含まれるシンボルのセグメンテーションおよび、シンボル名の識別を行うこと（入力図形の認識）は、候補ラティスの中からすべてのストロークを被う最適な図形系列（シンボルおよび直線からなる系列）を選択することにより実現できる。一般には、候補ラティスの中から選択可能な図形系列の組み合わせは多数存在する。そこで、第一段階として評価関数  $S$

$$\text{評価関数 } S = \frac{(\text{相異度})}{(\text{ストローク数})} \quad (4.1)$$

を導入し、その評価関数の値を最小とする図形系列を動的計画法（DP）により探索する。探索の結果得られた図形系列を”仮の認識結果”とする。この仮の認識結果は、3章で述べた候補ラティス法による認識結果と同一である。

一例として、図4.6の候補ラティスを探索すると太棒の候補図形が”仮の認識結果”となる。図4.7は、”仮の認識結果”を入力図形（図4.4）の対応するストロークの位置と大きさに合わせて表示した例である。この例では、破線内に示した箇所で

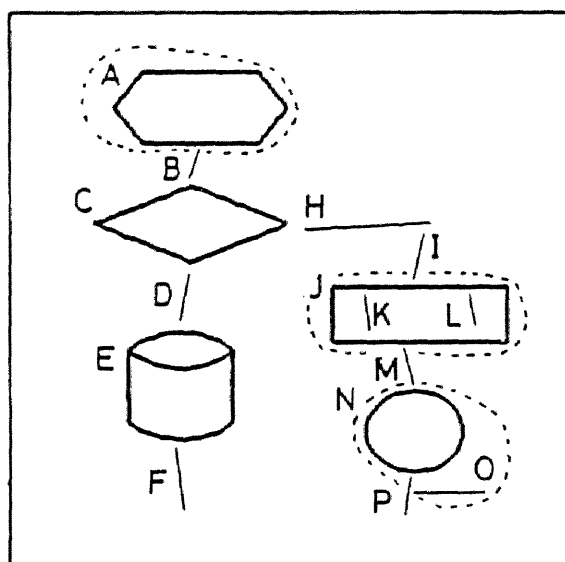
誤認識が生じている。

上記の手法のみで、認識実験データ（3.3(2)参照）を認識すると、3章で述べたように、認識率は89.8%であった。但し、認識率は、

$$\text{認識率} = \frac{(\text{正しく認識されたシンボル数})}{(\text{入力図形中に含まれるシンボル数})} \times 100.0 (\%) \quad (4.2)$$

で定義した。

次段階としてシンボル間の接続関係やシンボルの持つ意味などの大局情報を上位概念として接続ルールの形で利用し、“仮の認識結果”を修正する手法を導入する。



点線内は誤認識

A~Pはid

図4.7 仮の認識結果

#### 4.2.5 接続関係の記述

仮の認識結果は、複数のシンボルとそのシンボル間を接続する直線により構成されている。そこで、シンボル相互の接続関係を以下のように記述する。

(1) 各シンボルや直線は、相互のシンボルを区別するための識別記号 (id) と、接続する対象を示す接続ポインタ (Connection pointer) を持つ。

(2) 各シンボルや直線は、原パターンを参照するためのストローク番号 (Stroke number) を持つ。これにより、各シンボルの原パターン上での位置、大きさ、縦横比などの情報を参照することが可能になる。

上記の記述で、(1) は接続ルールの適用の際に必要である。また、(2) は認識結果を整形して出力する際に必要となる。表4. 1に、図4. 7に示した”仮の認識結果”の接続関係を記述した例を示す。表4. 1中の id は、図4. 7中の記号A～Pに対応している。表4. 1によれば、例えば記号Cのシンボル「decision」は、入力図形中 (図4. 4) のストローク番号の4, 5から構成され、記号B, D, Hの直線に接続されていることがわかる。

表4. 1 シンボル間の接続関係の記述

stroke number	id	symbol name	connection pointer
1,2	A *	preparation	B
3	B	line	A,C
4,5	C	decision	B,D,H
6	D	line	C,E
7,8	E	magnetic disk	D,F
9	F	line	E,G
10,11,12	G	decision	F,S,T
13	H	line	C,I
14	I	line	H,J
15,16	J *	process	J,K,L,M
17	K	line	J,J
18	L	line	J,J
19	M	line	J,N
20	N *	connector	M,O,P
21	O	line	N
22	P	line	N,Q

#### 4. 2. 6 接続ルールの適用

フローチャートなどのような複数のシンボルから構成される図形には、意味の上からは存在し得ないシンボル間の接続関係がある。例えば、「terminal」がフローチャートの分岐点に存在したり、直線が二つのシンボルを接続する役割を果していない等は、フローチャートとして明らかに矛盾している。そこで、このようなシンボル間の関係を「矛盾した接続関係」として登録しておく。”仮の認識結果”の接続関係が矛盾する場合には候補ラティスを修正する。ここではこの手順を接続ルールと呼ぶ。接続ルールを適用したのち、再度認識しなおす。接続関係に矛盾がなくなるまで、この処理を反復実行して矛盾のない認識結果を最終的に決定する。

接続ルールは、性質の異なる各種ルールの記述部  $B_i$  ( $i$  はルールの種類を示す) と、そのルールが作用するシンボル名称部  $S_{ij}$  ( $j$  はシンボルの種類を示す) という構成で表現する。

今回使用したフローチャートの例に対して登録した接続ルールを表 4. 2 に示す。この例では  $B_i$  と  $S_{ij}$  の組み合わせにより 52 種類の接続ルールが存在することになる。但し、 $B_4$  に対するルールはシンボルの接続に関するものでないが、ここでは説明の都合上これらを総称して接続ルールと呼んでいる。接続関係の記述が、上記接続ルール 52 個のうち 1 個でも満足しなければ、そのシンボルに対応する候補図形を候補ラティスから削除し、候補ラティスの探索処理を再度実行する。接続関係に矛盾がなくなるまで上記処理を反復実行し、矛盾が解消した段階での候補ラティスの探索結果を最終的な認識結果とする。

これら 52 個のルール ( $R_1 \sim R_{52}$ ) は IF, THEN ルールの形式で表現することが可能である。その例を以下に示す。

$R_1$  : IF (直線の両端点が、同一の「Process」に接続している)  
THEN (その「Process」をラティスから取り除く) (4.3)

$R_2$  : IF (「Terminal」に接続する直線が 2 本以上である)  
THEN (その「Terminal」をラティスから取り除く) (4.4)

.....

$R_i$  : IF ( . . . )  
THEN ( . . . ) (4.5)

.....

R 5 2 : I F (「Terminal」の大きさが、他のシンボルの平均的な大きさに  
比較して非常に小さい)  
T H E N (その「Terminal」をラティスから取り除く) (4.6)

表4. 1に示した記述例に、この接続ルールを適用すると、4箇所のシンボルで接続関係に矛盾が生じる。そのシンボルを表4. 1中に\*印で示す。そこで、矛盾したシンボルに対応した候補図形を図4. 6に示した候補ラティスから削除(候補ラティスの修正)し、再び候補ラティスを探索する。但し、探索は4. 2. 4節で述べた仮の認識結果を求める手法を再度適用する。

この例の場合には、候補ラティスの修正と探索の処理を2回反復実行した後に認識結果が得られる。1回反復した後の仮の認識結果を図4. 8(a)に示す。この段階で接続関係に矛盾する箇所を破線で示す。図4. 8(b)は、2回反復した後の最終認識結果である。反復に伴う認識結果の変化を表4. 3に示す。

認識した結果を入力図形中のシンボルの位置と大きさに対応させて整形して出力した例を図4. 9に示す。図形の整形手法については、次節で説明する。

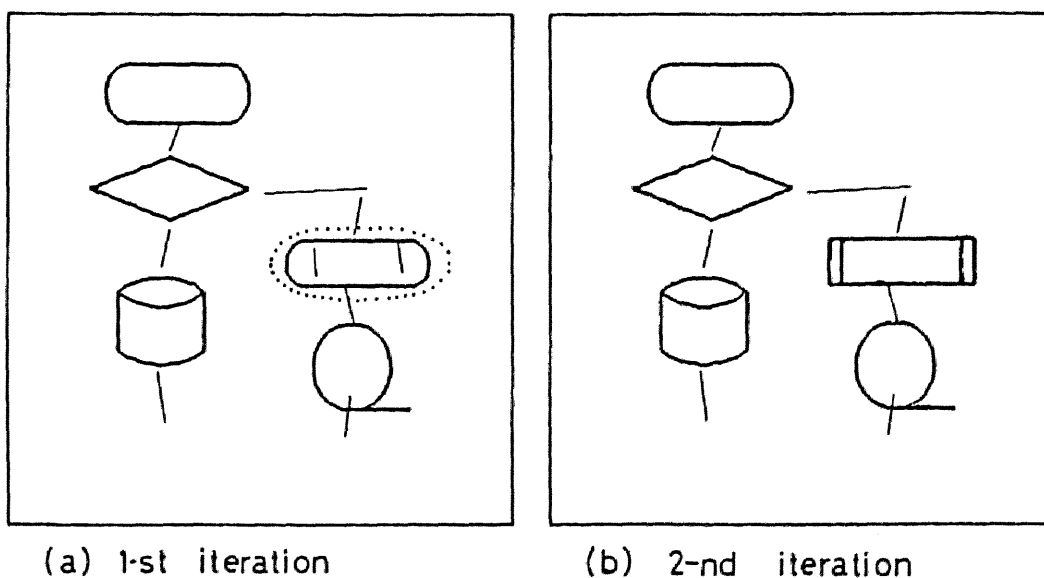


図4. 8 仮の認識結果(繰り返し回数1回、2回)  
(図中で点線内は誤認識を表わす)



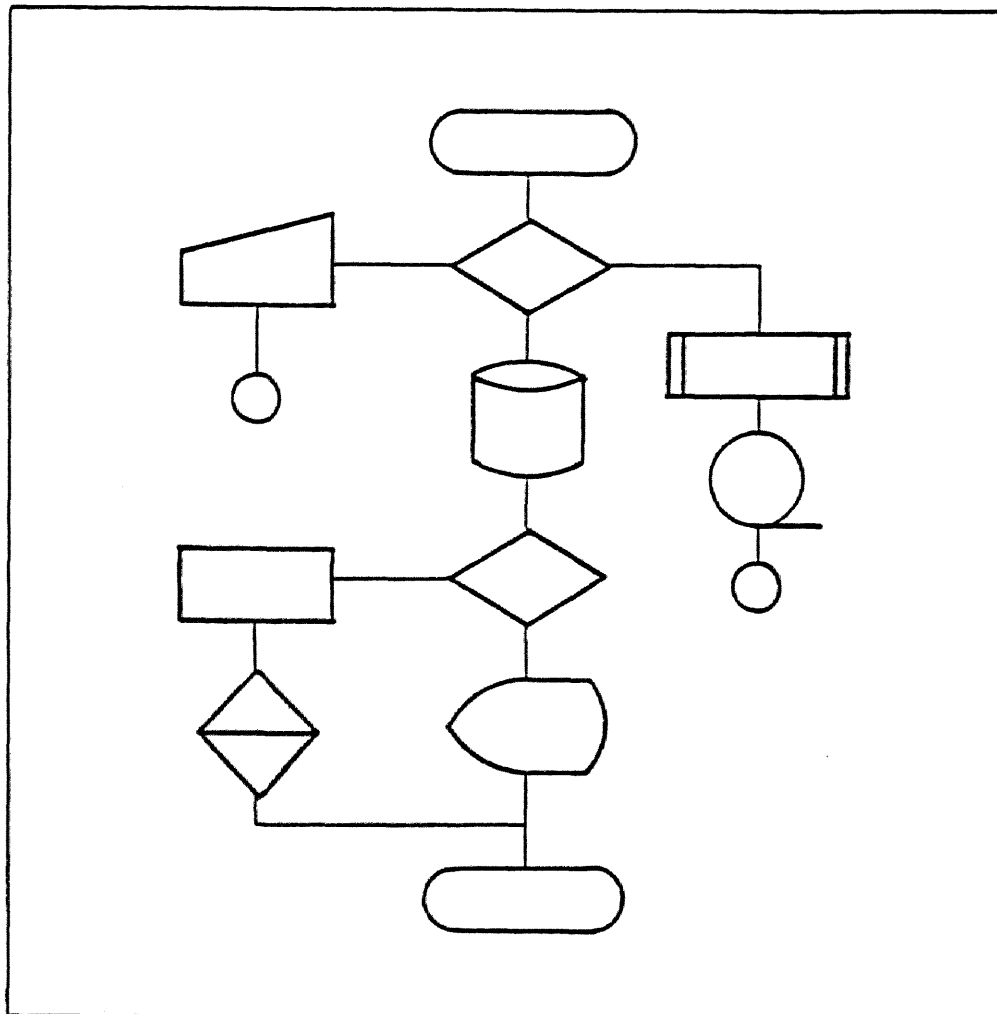


図 4. 9 認識結果の整形出力

表 4. 2 登録した接続ルール

ルールの記述部		作用するシンボル名称部	
$B_1$	直線はシンボル内部には存在しない.	$S_{1,j}$	すべてのシンボル $j=1\sim 28$
$B_2$	シンボルに接続する直線数は1である.	$S_{2,j}$	「Terminal」 「Connector」 $j=1\sim 2$
$B_3$	シンボルに接続する直線数は2以上である.	$S_{3,j}$	「Preparation」 「Decision」 等 $j=1\sim 6$
$B_4$	極端に小さいシンボルは存在しない.	$S_{4,j}$	「Terminal」 「Display」 等 $j=1\sim 16$

表4.3 仮の認識結果

stroke number	1st tentative recognition result	2nd tentative recognition result	3rd tentative recognition result (the final)
.			
.			
10	} decision	} decision	} desision
11			
12			
13	line	line	line
14	line	line	line
15	} process (*)	} terminal (*)	} predefined process
16			
17			
18	line	line	} line
19	line	line	
20	connector (*)	} magnetic tape	} magnetic tape
21	line		
22	line	line	line
.			
.			
.			

(\*): illegal symbol

#### 4. 2. 7 認識結果の整形出力

認識結果の各シンボルと直線を入力図形中の対応するストロークの位置と大きさに合わせてそのまま表示すると、不自然な表示となる。そこで、認識結果の表示において、シンボル間の接続関係の記述に基づき、且つ原パターンにおけるシンボルの配置を考慮した以下の整形処理を導入する。

##### (1) 整形処理に適した図形データの構造

整形処理においては、以下の条件を満足し、シンボルの位置などを柔軟に変更することの可能なデータ構造が必要となる。

①あるシンボルに着目したとき、そのシンボルに接続するすべてのシンボルを簡単にアクセスできること。

②シンボルだけを、あるいは直線だけを簡単にアクセスできること。

③シンボルについては、その中心位置座標、大きさを、直線については、その両端点の位置座標が簡単にアクセスできること。

上記の条件を満足するデータ構造として、CADなどの図形処理でよく使用される Linked list structure [87] を採用する。つまり、シンボルや直線はそれぞれブロックで記述し、相互にポインターでアクセスすることの可能な構造を採用する。図4.10に各ブロックが持つ情報の内容を示す。図4.11にこのデータ構造によるフローチャートの記述例を示す。

専用図形 block

リングポインター
シンボル名
位置 X座標
位置 Y座標
大きさ 横方向
大きさ 縦方向
接続ポインター
◇
◇
⋮

直線 block

リングポインター
直線タイプ
始点 X座標
◇ Y座標
◇ 接続環境
◇ 接続ポインター
終点 X座標
◇ Y座標
◇ 接続環境
◇ 接続ポインター

----- (水平、垂直、その他)

{ 接続無し  
 シンボルに接続  
 直線の端点に接続  
 直線の midpoint に接続

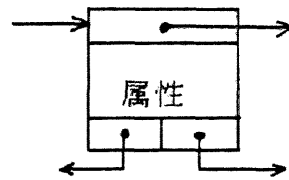
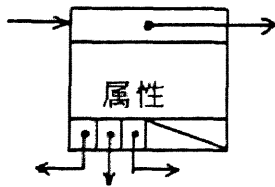
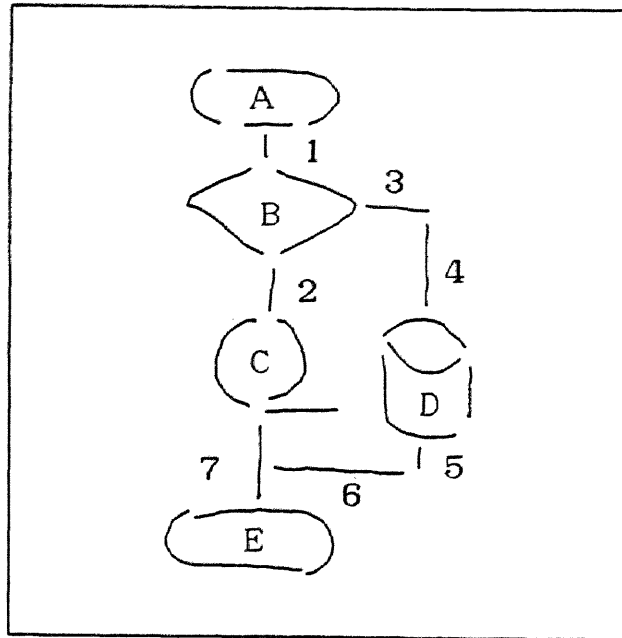


図4.10 データ構造



入力図形内のA～E，1～7は、記述内のそれに対応  
 記述内のターミネータは各ブロックのリングポインターに接続

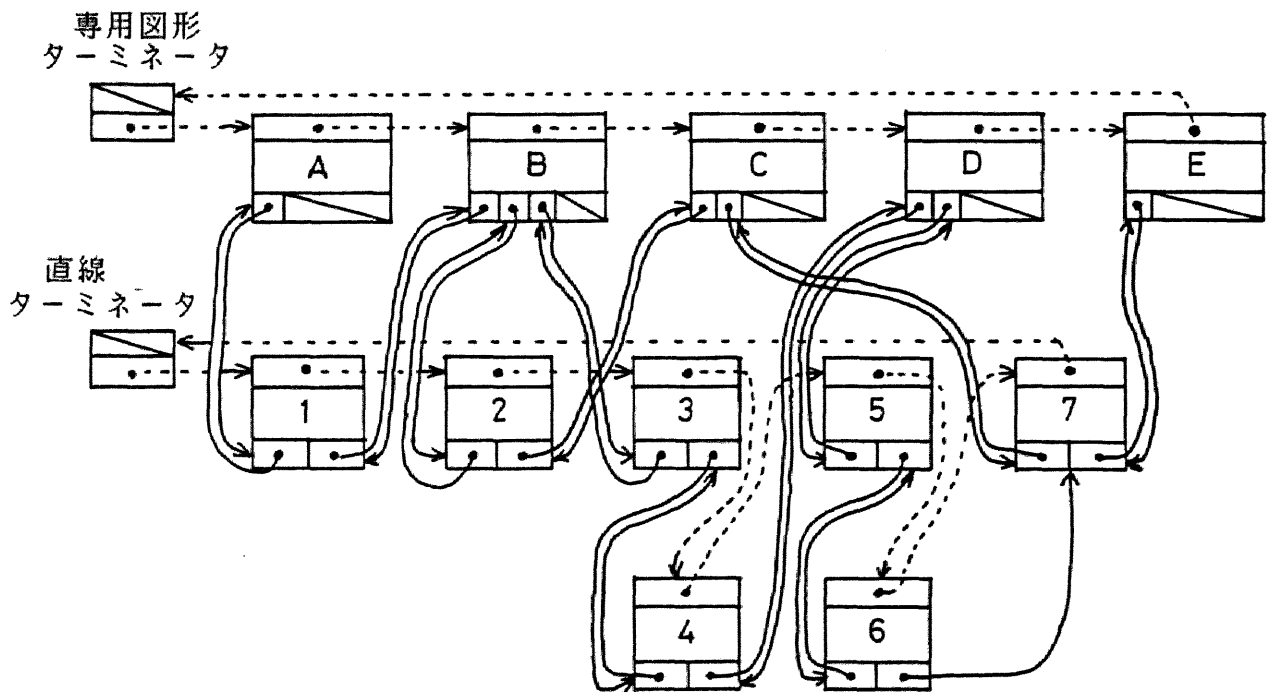


図4. 1 1 図形の記述例

## (2) 整形の手順

図形の整形は以下の手順に従って行なう。各手順の説明を図4.12に示す。

- ①直線と認識された部分図形が直線らしくない場合には、これを2つの直線要素に分割する。(本認識手法では、図形中の部分図形でシンボルとして解釈できない箇所は直線として認識されるため、直線と認識された部分図形が直線らしくないことは良く発生する。)
- ②あるシンボルのX方向の位置が他のシンボルのそれと類似している場合は、シンボルのX方向の位置を一致させる。
- ③直線の端点がシンボルと近接している場合には、それをシンボルに連結する。
- ④水平(垂直)に近い直線は、水平(垂直)に正規化する。
- ⑤直線の端点同士が近接している場合には、それらを連結する。
- ⑥直線の端点が他の直線に近接している場合には、それを直線に連結する。

この処理に従って、認識結果を整形して出力した例を図4.13に示す。この図形整形法によって、大半の図形を清書することが可能である。しかし、このアルゴリズムではシンボル間の相互の位置関係を詳細に調べていないため、図4.13(c)の「Display」と「Magnetic tape」のようにシンボル間が重なる場合がまれに発生する。


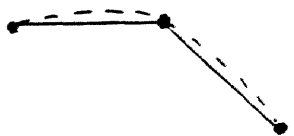
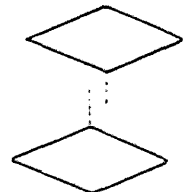
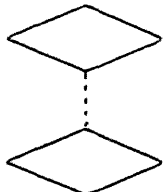
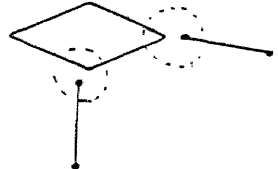
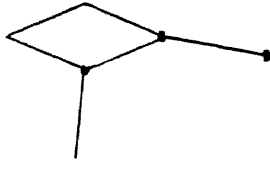
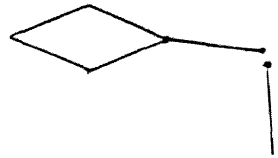
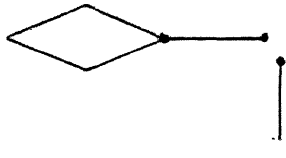
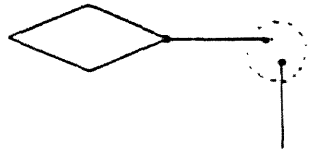
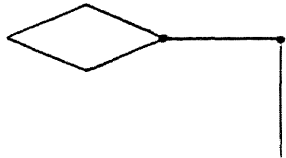
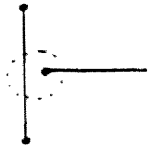
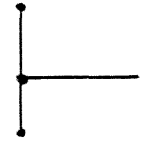
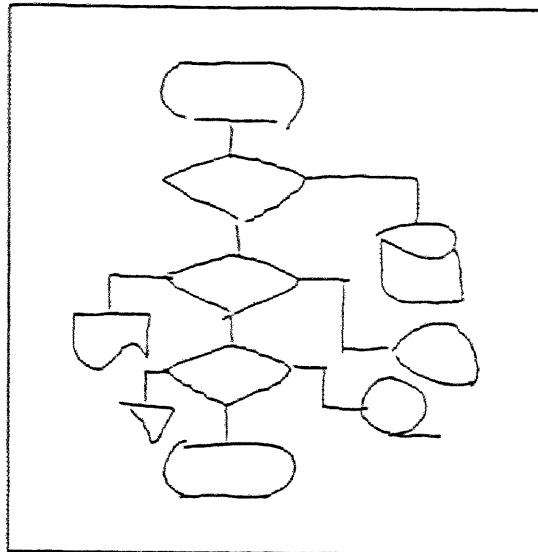
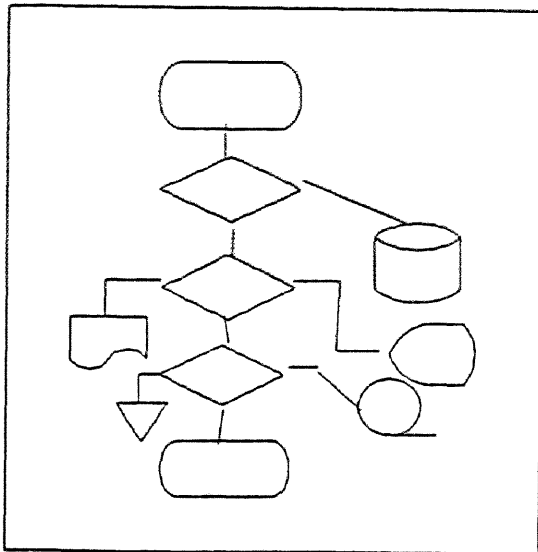
	整形前	整形後
1		
2		
3		
4		
5		
6		

图 4. 1 2 整形处理

(a) 入力図形



(b) 認識結果をそのまま出力



(c) 認識結果の整形出力

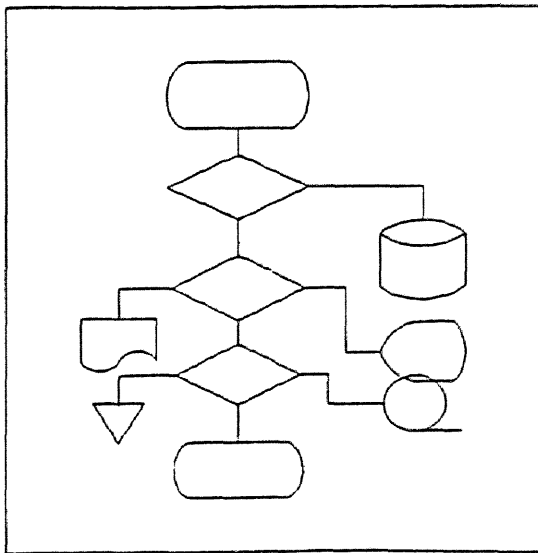


図4.13 整形処理の一例



#### 4.3 認識実験結果

4. で述べた認識アルゴリズムを用いて、手書き線図形の認識実験を行った。100枚のフローチャートとブロック図（2. 参照）に対する認識率を表4. 4に示す。接続ルールの導入により、6. 3%の認識率の改善が見られ、96. 1%の認識率が達成された。

表4. 4 認識率

接続ルールの無い場合	接続ルールを導入した場合
89. 8%	96. 1%

## 4.4 考察

### 4.4.1 接続ルール導入の効果

接続ルール無しで認識した際の誤認識を分析すると、原因別に以下の①～④に分類可能である。

①：シンボル間の包含関係によるセグメンテーションの誤り。

②：フローチャートの開始点や終了点に存在するシンボルをそこには存在し得ないシンボルに誤認識。

③：フローチャートの分岐点や中間点に存在するシンボルをそこには存在し得ないシンボルに誤認識。

④：上記以外の誤認識。

の4種類である。表4.2で述べた接続ルールB1～B4は、それぞれ上記①～④の誤認識の原因を排除するためのものである。接続ルール導入の効果を表5に示す。また、各接続ルールを適用した際のコンフュージョンマトリックスの変化を図4.14(a)～(d)に示す。各接続ルールの適用により対応する誤認識が減少し、各接続ルールの有効性が確認された。図中の○は次の段階で改善される誤認識を示す。ルールの追加による認識率の向上を図4.15に示す。また、接続ルールB1～B4を導入して正しく認識できた図形の例を図4.16(a)に示す。一方、接続ルールを導入しても正しく認識出来なかった図形の例を図4.16(b)に示す。

接続ルールB1～B4を導入しても修正不能であった3.9%の誤読には、セグメンテーションの誤り0.4%（誤認識①）、「connector」を「display」に誤読したものの0.5%（誤認識④）「conncter」を「terminal」に誤読したものの0.4%（誤認識④），等があった。これらに対しては、更に他の接続ルールを導入する必要がある。

Recognition results Input symbols	PROCESS	DECISION	PREPARATION	PREDEFINED PROCESS	MANUAL OPERATION	MERGE	COLLATE	SORT	MANUAL INPUT	INPUT/OUTPUT	ONLINE storage	OFFLINE storage	DOCUMENT	PUNCHED CARD	DECK OF CARDS	PUNCHED TAPE	MAGNETIC TAPE	MAGNETIC DRUM	MAGNETIC DISK	CORE	DISPLAY	COMMUNICATION LINK	CONNECTOR	TERMINAL	segmentation error	
	PROCESS	76																								1
DECISION	60																									
PREPARATION	20																									
PREDEFINED PROCESS	19																									1
MANUAL OPERATION	20																									
MERGE	19																									1
COLLATE	20																									1
SORT	18																									②
MANUAL INPUT	18																									1
INPUT/OUTPUT	18																									1
ONLINE storage	3																									3
OFFLINE storage																										
DOCUMENT																										
PUNCHED CARD	1																									1
DECK OF CARDS																										1
PUNCHED TAPE																										
MAGNETIC TAPE																										
MAGNETIC DRUM																										2
MAGNETIC DISK																										
CORE																										③
DISPLAY	4																									
COMMUNICATION LINK																										
CONNECTOR	18																									
TERMINAL	18																									

図 4. 14(a) コンフュージョンマトリクス (ルール無し)

図中の○は次のルール適用段階で改善される誤認識を示す

Recognition results Input symbols	PROCESS	DECISION	PREPARATION	PREDEFINED PROCESS	MANUAL OPERATION	MERGE	COLLATE	SORT	MANUAL INPUT	INPUT/OUTPUT	ONLINE storage	OFFLINE storage	DOCUMENT	PUNCHED CARD	DECK OF CARDS	PUNCHED TAPE	MAGNETIC TAPE	MAGNETIC DRUM	MAGNETIC DISK	CORE	DISPLAY	COMMUNICATION LINK	CONNECTOR	TERMINAL	segmentation error	
	PROCESS	76																								1
DECISION	60																									
PREPARATION	20																									
PREDEFINED PROCESS	20																									
MANUAL OPERATION	20																									
MERGE	19																									1
COLLATE	20																									
SORT	20																									
MANUAL INPUT	18																									1
INPUT/OUTPUT	18																									1
ONLINE storage	3																									3
OFFLINE storage																										
DOCUMENT																										
PUNCHED CARD	1																									1
DECK OF CARDS																										1
PUNCHED TAPE																										
MAGNETIC TAPE																										
MAGNETIC DRUM																										
MAGNETIC DISK																										
CORE																										
DISPLAY	④																									
COMMUNICATION LINK																										
CONNECTOR	⑧																									
TERMINAL	⑧																									

図 4. 14(b) コンフュージョンマトリクス (ルール1適用)

Recognition results Input symbols	PROCESS	DECISION	PREPARATION	PREDEFINED PROCESS	MANUAL OPERATION	MERGE	COLLATE	SORT	MANUAL INPUT	INPUT/OUTPUT	ONLINE STORAGE	OFFLINE STORAGE	DOCUMENT	PUNCHED CARD	DECK OF CARDS	PUNCHED TAPE	MAGNETIC TAPE	MAGNETIC DRUM	MAGNETIC DISK	CORE	DISPLAY	COMMUNICATION LINK	CONNECTOR	TERMINAL	segmentation error	
PROCESS	76																									
DECISION		60																								①
PREPARATION			20																							
PREDEFINED PROCESS				20																						
MANUAL OPERATION					20																					
MERGE						20																				
COLLATE							20																			
SORT								20																		
MANUAL INPUT									18																	1
INPUT/OUTPUT										18					1											①
ONLINE STORAGE											14															①
OFFLINE STORAGE												20														
DOCUMENT													20													
PUNCHED CARD														19												1
DECK OF CARDS															19											1
PUNCHED TAPE																1										
MAGNETIC TAPE																	19									
MAGNETIC DRUM																		20								
MAGNETIC DISK																			60							
CORE																				60						
DISPLAY																					20					
COMMUNICATION LINK																						58				1
CONNECTOR																							60			
TERMINAL																							41	7		
segmentation error																										54

図4. 14(c) コンフュージョンマトリクス (ルール 1, 2 適用)

Recognition results Input symbols	PROCESS	DECISION	PREPARATION	PREDEFINED PROCESS	MANUAL OPERATION	MERGE	COLLATE	SORT	MANUAL INPUT	INPUT/OUTPUT	ONLINE STORAGE	OFFLINE STORAGE	DOCUMENT	PUNCHED CARD	DECK OF CARDS	PUNCHED TAPE	MAGNETIC TAPE	MAGNETIC DRUM	MAGNETIC DISK	CORE	DISPLAY	COMMUNICATION LINK	CONNECTOR	TERMINAL	segmentation error	
PROCESS	77																									
DECISION		60																								
PREPARATION			20																							
PREDEFINED PROCESS				20																						
MANUAL OPERATION					20																					
MERGE						20																				
COLLATE							20																			
SORT								20																		
MANUAL INPUT									18																	1
INPUT/OUTPUT										19					1											
ONLINE STORAGE											15				2											
OFFLINE STORAGE												20														
DOCUMENT													20													
PUNCHED CARD														19												1
DECK OF CARDS															19											1
PUNCHED TAPE																1										
MAGNETIC TAPE																	19									
MAGNETIC DRUM																		20								
MAGNETIC DISK																			60							
CORE																				60						
DISPLAY																					20					
COMMUNICATION LINK																						58				1
CONNECTOR																							60			
TERMINAL																							41	7		
segmentation error																										54

図4. 14(d) コンフュージョンマトリクス (ルール 1~3 適用)

Recognition results / Input symbols	PROCESS	DECISION	PREPARATION	PREDEFINED PROCESS	MANUAL OPERATION	MERGE	COLLATE	SORT	MANUAL INPUT	INPUT/OUTPUT	ONLINE STORAGE	OFFLINE STORAGE	DOCUMENT	PUNCHED CARD	DECK OF CARDS	PUNCHED TAPE	MAGNETIC TAPE	MAGNETIC DRUM	MAGNETIC DISK	CORE	DISPLAY	COMMUNICATION LINK	CONNECTOR	TERMINAL	segmentation error	
PROCESS	77				2				1																	
DECISION		60																								
PREPARATION			20																							
PREDEFINED PROCESS				20																						
MANUAL OPERATION					20																					
MERGE						20																				
COLLATE							20																			
SORT								20																		
MANUAL INPUT									18																	
INPUT/OUTPUT										19					1											1
ONLINE STORAGE											15					2										
OFFLINE STORAGE												20														
DOCUMENT													20													
PUNCHED CARD														19												1
DECK OF CARDS															19											1
PUNCHED TAPE																1										
MAGNETIC TAPE																	19									
MAGNETIC DRUM																		20								
MAGNETIC DISK																			60							
CORE																				60						
DISPLAY																					20					
COMMUNICATION LINK																						58				1
CONNECTOR																							60			
TERMINAL																								4	52	3
segmentation error																								2	54	

図4. 14(e) コンフュージョンマトリクス (ルール1~4適用)

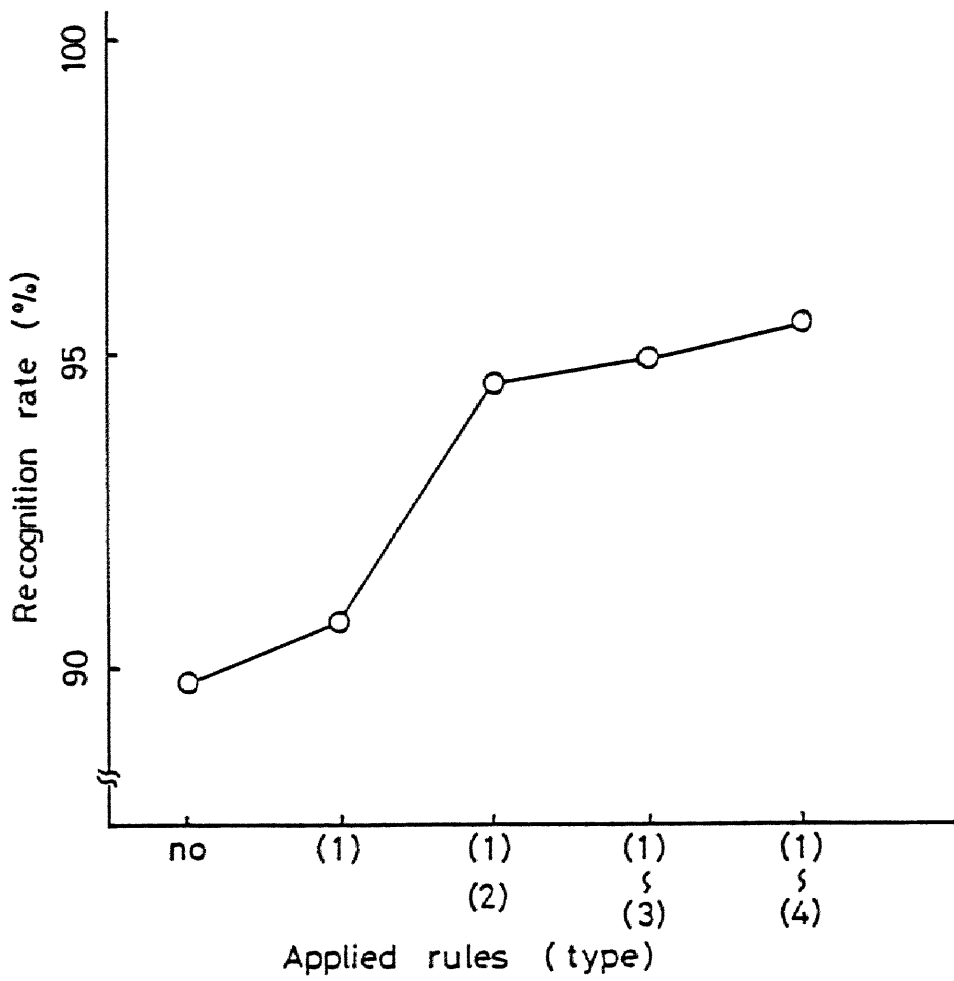


図4. 15 適用ルールに対する認識率

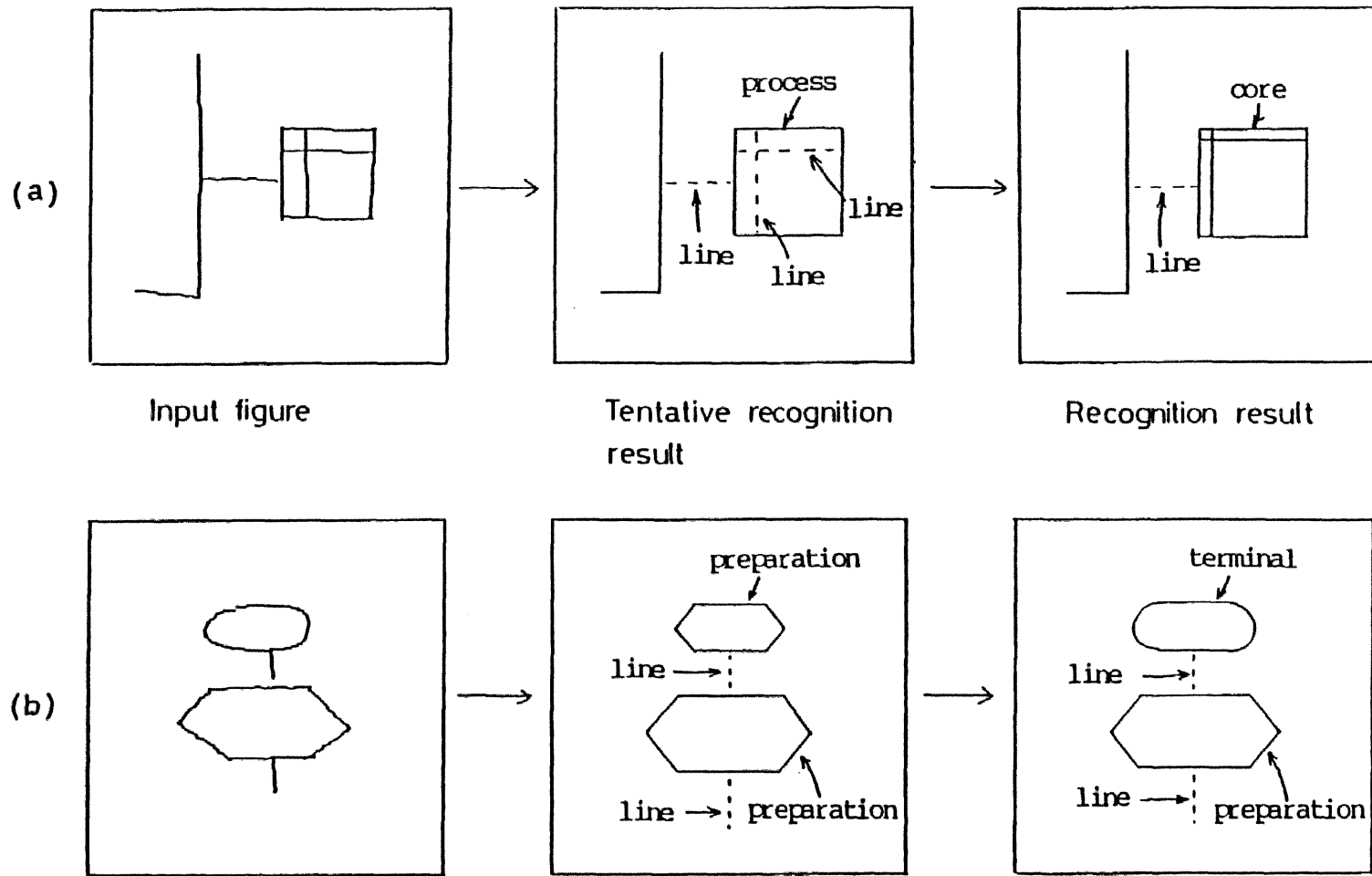


図4.16 (a) 接続ルールの導入により正しく認識できた例

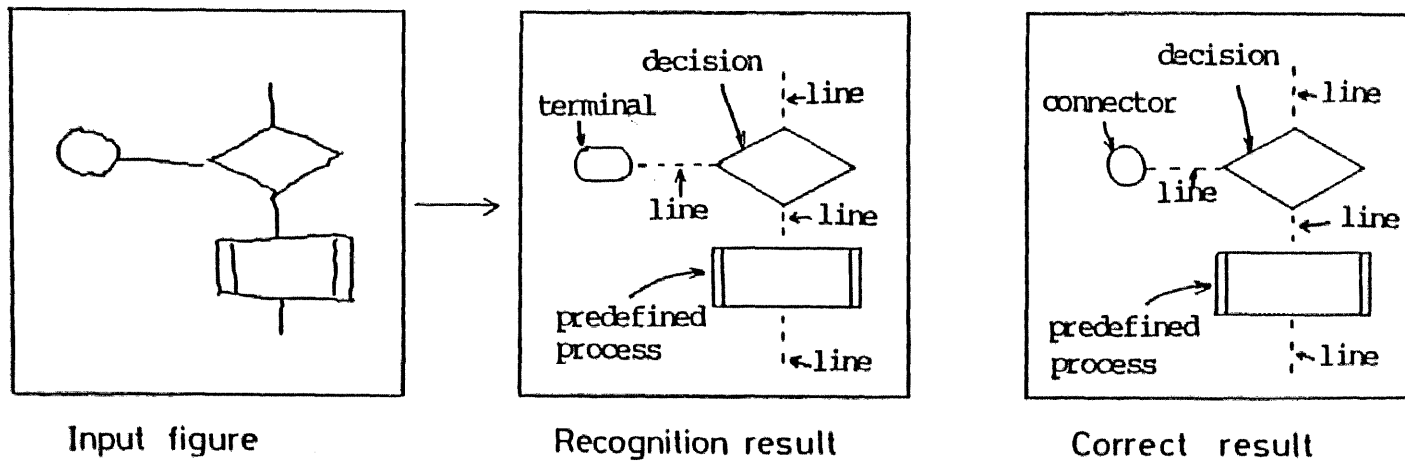


図4.16(b) 接続ルールにかかわらず正しく認識できなかった例



#### 4.4.2 候補ラティスの修正と探索の反復実行回数

この章で述べた図形認識法では、認識した結果が接続ルールで与えた接続関係に矛盾しなくなるまで候補ラティスの修正と探索を反復実行する。認識実験での反復実行回数の頻度分布を調査した。結果を図4.17に示す。また、候補ラティスの修正と探索の反復実行を途中で打ち切った時の認識率を図4.17に重ねて示す。反復実行回数が0とは、接続ルールを適用しないことを意味している。この例では、40%のデータが1回以上の反復実行を受け、それにより認識率は6%程度向上することがわかる。反復実行回数が4回で認識率は飽和している。ここでは処理の発散の危険性を考慮し、調査結果に基づき、候補ラティスの修正と探索の反復実行は4回で打ち切ることにする。

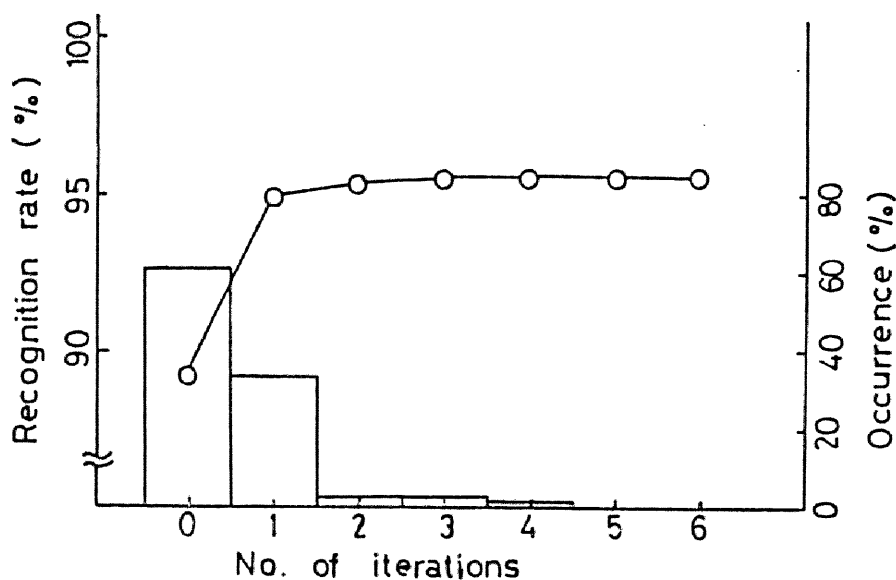


図4.17 接続ルール適用の繰り返し回数と認識率

#### 4.4.3 認識時間

本処理は大別して、候補ラティスの作成処理と、候補ラティスの探索の処理から構成される。ミニコン(1MIPS、FORTRAN77)によるシミュレーション実験により、図形の認識処理の大半は候補ラティスの作成に費やされていることを確認した。その理由は、候補ラティスの探索の処理(候補ラティスから仮の認識結果を探索する処理、

および、接続ルールによる候補ラティスの修正と探索の反復実行)は簡単な記号処理により実現可能なためである。つまり、この章で述べた手法による図形の認識時間は3章で述べた候補文字ラティス法のそれと、ほとんど同等であり、1ストローク当たり5秒程度である。

#### 4.4.4 論理回路図への応用

この章で述べた図形認識手法は、シンボルと直線から構成される図形ならば、シンボルの辞書と接続ルールのテーブルを用意するだけで、認識可能である。ここでは認識対象の拡張性を確認する目的で、論理回路図にこの図形認識法を適用してみた。辞書にはシンボル「and」「or」「nand」「exor」「not」と、それらを90度、180度、270度回転したシンボルと、「connect」「block」との合計26種類を登録した。接続ルールには、「not」は1入力1出力である。」等、7種類のルールを登録した。3章(3.3)で述べたと同様の筆記条件で40枚(総シンボル数は205個)のデータを収集し、認識実験を行った。接続ルール無しで90.2%、接続ルールを導入して93.2%の認識率が得られた。認識したデータの一例を図4.18に示す。

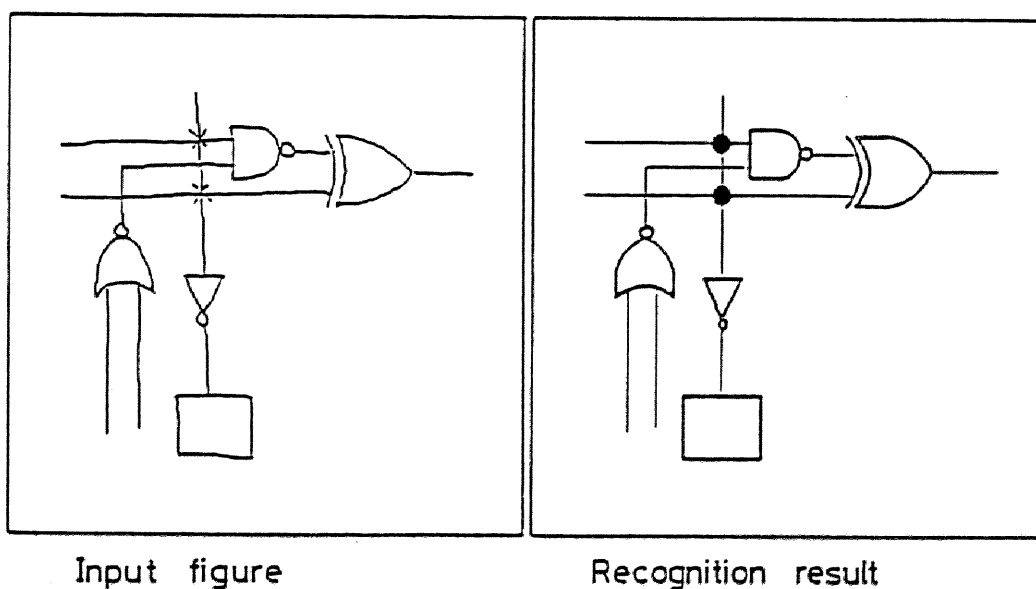


図4.18 論理回路図への応用

#### 4.5 結言

人間はかなり複雑な図形であっても、図形中からシンボルを正しくセグメンテーションし、更に各シンボルを正しく認識できる。その理由は、人間はシンボル個々の形状のみならず、図形の全体情報を利用している為である。図形の全体情報としては、例えば、シンボル間の接続関係や、図形の持つ意味などが考えられる。

この章では、このような人間の認識能力に迫る能力を機械に付与するために、図形の全体情報を反映した上位概念を利用することを検討した。具体的には、3章で述べたような局所的な形状だけを利用した認識法ではシンボルのセグメンテーションや認識が困難な図形に対しても、精度良く認識するために、上位概念として接続ルールを導入した候補ラティス法を提案した。これにより、シンボル間の接続関係や、各シンボルの有する機能をも認識に有効に利用することが可能になり、高精度な図形認識が達成された。自由書式で筆記されたフローチャート、ブロック図、論理回路図などの図形を例に、この手法の能力を検討した。以下に、その結果をまとめる。

(1) 包含関係にあるシンボル間でのセグメンテーションの失敗や、類似形状のシンボル間での誤認識に対処するため、上位概念として接続ルールを導入した。入力図形を仮に認識し、シンボル間の接続関係を記述し、この接続関係に矛盾が存在する場合には、候補ラティスを修正し候補ラティスの探索処理を反復実行する手法を提案した。その結果、フローチャートやブロック図に対する認識率は89.8%から96.1%に向上した。これにより、意味を持つ線図形の認識において、接続ルールの有用性が確認できた。

(2) 論理回路図に対し93.2%の認識率が得られた。この章で述べた手法は、複数のシンボルと直線から構成される線図形の範囲ならば、シンボルの辞書と接続ルールのテーブルを用意するだけで、容易に他の図形に拡張して適用できることを確認した。

以上、本手法により自由書式で筆記された図形を高精度にオンライン認識することが可能となった。残された課題としては、接続ルールをより最適化すること、およびイラストのようにシンボルと直線から構成されていない図形を含む図形の認識法を開発すること、認識された図形を的確に記述することなどがあげられる。図形の記述法としては例えばPDL [88]などが考えられる。

## 第5章 自由書式手書き文字列の 認識法

### 5.1. 緒言

2章～4章では、図面内の図形部分の認識について述べてきた。以下の5章と6章では、図面内の文字部分の認識について述べる。この章では特に自由書式で筆記された文字列を認識するのに基本となる、文字のセグメンテーションと、文字の認識とを融合した手法について検討を行なう。

近年の文書処理の発展に伴い、手書き漢字認識による円滑なマンマシンインターフェイスの実現が望まれている。これら実現のための基本技術として、低品質文字認識[46, 48]や、自由書式文字認識等、筆記時の制限の少ない認識技術の確立が必要となっている。この章で述べるオンライン自由書式手書き文字列の認識は、即時性や対話性が良好であるというオンラインの特長に加え、各文字の枠内筆記の制限を解除するという特長を持つ。枠内筆記の制限が解除されれば、大量文書を入力する際の精神的・肉体的な負担は低減される。また、この認識技術は図面中の文字のような任意の場所に大小混在した文字や、罫線上に筆記された文字の認識等広い応用範囲を持つ。

従来、自由書式文字列を認識する手法は、文字と文字を分離するセグメンテーション段階と、個々の文字を認識する文字認識段階から構成されていた。セグメンテーション手法には、筆記者がセグメンテーションを意識して入力する手法として、文字と文字の間でピック等の指示を与える手法や、1文字を書き終えてから次の文字を書き初めるまでの時間間隔を利用する手法[19]があるが、いずれも筆記者の負担が大きくなる。一方、英文字列を対象として自動的にセグメンテーションを行う手法[89-92]、ストロークの横軸への投影を利用する手法(以下、単純投影法と呼ぶ。)等がある。しかし、漢字のように左右・上下に分離する可能性の高い対象では、ストロークの位置関係の利用だけで文字を確実にセグメンテーションすることは困難となっている。この他にOCR型文字列を対象とした報告[93-104]があるが、これらの手法をそのままオンライン型の文字列に適用することは不適當である。他方、文字認識には多数の手法が提案されているが、文字が正しくセグメンテーションされなければ、これら文字認識技術は適用出来ない。

この章では、タブレット上に手書きされた自由書式の文字列を認識することが可能

な「候補文字ラティス法」を提案する [72, 74]。ここでは、自由書式文字列の一例として、セグメンテーション指示なく横方向に手書きされた仮名漢字文字列を扱う。「候補文字ラティス法」は、3章で述べた線図形認識法「候補ラティス法」を文字認識用に改良した手法である。「候補文字ラティス法」は各文字のセグメンテーションと各文字の認識を同時に実現する手法である。個々のストローク形状等の下位の特徴から、文字の包含関係等の上位の特徴までを有機的に利用することが可能なため、高精度に文字列を認識することが可能である。本手法の特長を以下に述べる。

(1) 手書き文字列の文字間隔の分散は大きく、文字が重なる場合から、文字間が1文字分以上離れる場合まで存在する。本方式では本質的に文字間隔(文字ピッチ)の情報を使用しないため、文字間隔の分散が問題にならない。

(2) 漢字や平仮名には、“科”, “川”, “い”等の分離文字が多数存在する。本手法では各文字の認識情報が自動的に文字のセグメンテーションに利用されるため、分離した部分図形が他の文字にならなければ、分離文字は正しく認識される。

(3) 漢字には、“理”, “炉”等のように左右に分離し、分離した各部分図形が他の文字になる場合がある。この場合には、文字の縦横比を利用した分離有意文字処理を導入している。

候補文字ラティス法を、従来手法と比較して概観する(図5.1)。従来手法では、文字列からまず文字をセグメンテーションし、次に文字認識を行なう。一方、本手法では、文字のセグメンテーションと認識を融合している。そのため、文字認識から得られる情報が自動的に文字のセグメンテーションにフィードバックされ、高精度な文字列認識が達成される。この章で主張する点は、文字のセグメンテーションと認識を融合する手法の提案である。その箇所を図5.1中に太枠で示す。

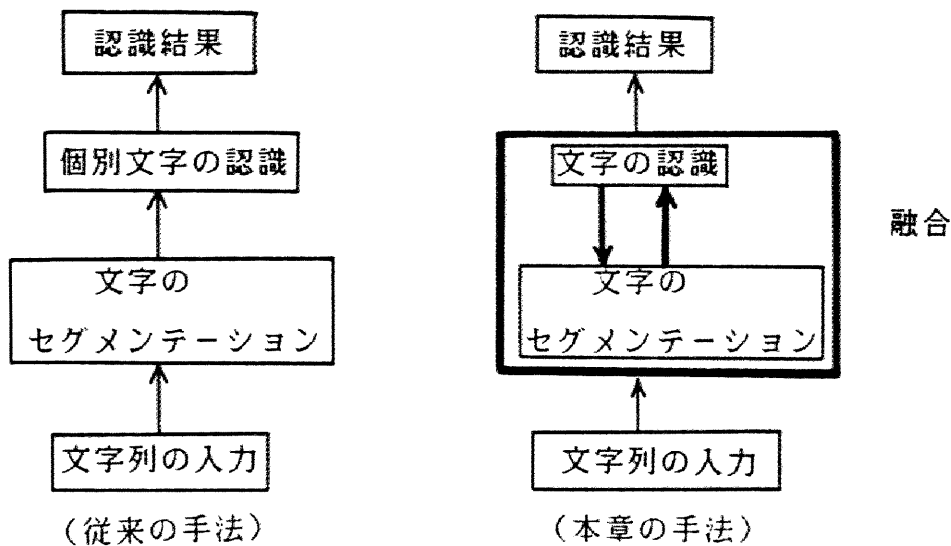


図5.1 本手法と従来手法との比較

以下、5.2で認識対象を、5.3で手書き文字列の性質を、5.4で認識アルゴリズムを、5.5で認識実験結果を述べ、5.6で考察を加える。

## 5.2 認識対象

日本語文書には縦書きと横書きが存在するが、最近では横書きの文書の割合が極めて高く、技術図面中に出現する文字列にも横書きが多い。また横書き文字列の認識技術が確立されればこの技術を縦書き文字列に拡張することは容易であることから、認識対象として横書き文字列を扱う。その際、筆記者の負担を軽減するため、不定ピッチ、文字の大小混在、等の自由書式を許す。文字列は常用漢字1945字種と平仮名46字種の文字から構成されるものとする。

### 5.2.1 文字列データ

本章で述べる認識アルゴリズムの効果を確認するため、認識実験用の文字列データを以下の条件で収集した。

- (1) 横罫線（1 cm間隔）の用紙に、文字列を自由なピッチで筆記させた。
- (2) 各文字の筆順は自由であるが、画数は守らせた。
- (3) 文字列として以下の5文例を用意し、それを筆記させた。

①地理的に離れた場所での

- ② 詳細な検討を続ける
- ③ 高騒音下での通話対策
- ④ 安定でしかも量産に向く
- ⑤ 利用するための手引

- (4) 文字ピッチを密・普通・粗の指示で、同一文例を3回づつ筆記させた。
- (5) 7名の筆記者に筆記させた。

データ収集数は（7名、各5文例、各3回）の105文例であり、文字総数は1050文字である。文字列収集用紙の一例を図5.2に、収集した文字列の一例を図5.3に示す。

#### 5.2.2 辞書作成データ

文字認識に用いる標準パターンの作成には、常用漢字と平仮名のデータを利用した。文字の種類は常用漢字1945字種、平仮名46字種、画数変動文字157字種（例えば，“子”は正しくは3画であるが2画で筆記されやすい。このような文字）の合計2148文字である。30名分の文字（64440文字）の内20名分（42960文字）の文字を用いて標準パターンを作成（5.4.6.(2)参照）した。また、残りの10名分の文字（21480文字）を予備実験の文字認識実験（5.4.6(6)）に使用した。

今回辞書に登録した標準文字の一覧を付録Bに示す。付録Bは、標準文字を画数順に並べたものであり、各文字には、JIS漢字コードと、画数、全体の通し番号、画数内での通し番号が記されている。また、その作成方法は5.4.6(2)で述べている。図中不自然な文字も見られるが、これは単純に平均パターンを作成したためである。

注意 ①文字列を枠内に、丁寧に、お書きください。

②各文字の画数は守ってください。

③文字間隔については、普:普通に、密:密に、粗:粗に、筆記ください。

\*記入例 (集積回路の動作特性を)

<普> 集積回路の動作特性を

<密> 集積回路の動作特性を

<粗> 集積回路の動作特性を

・記入1 (地理的に離れた場所での)

<普> 地理的に離れた場所での

<密> 地理的に離れた場所での

<粗> 地理的に離れた場所での

・記入2 (詳細な検討を続ける)

<普> 詳細な検討を続ける

<密> 詳細な検討を続ける

<粗> 詳細な検討を続ける

図5.2 文字列収集用紙の一例



# 地理的に離れた場所以での

図5.3 文字列の一例

## 5.3 手書き文字列の性質

手書き文字列の性質を予備実験として調査する。

### 5.3.1 文字間隔の分布

手書き文字列の文字間隔の分散は印刷文字の場合と異なり極めて大きい。図5.4 (a)に、5.2.2で述べた文字列データについて、X軸（横方向）に着目した文字間隔の分布を示す。文字間隔に指示を与えた場合の指示別の文字間隔の分布を図5.4 (b)に示す。但し、文字間隔の値は、文字列の縦幅を1.0とするように大きさを正規化（詳細は5.4.3にて述べる。）した後の値である。負の文字間隔は文字が重なり合うことを意味する。図5.4より、手書きでは文字が10%程度重なる場合から、文字間が1文字分以上離れる場合まで存在することが確認された。この事実は、手書き文字列のセグメンテーションに単純投影法の適用が困難であることを示す。

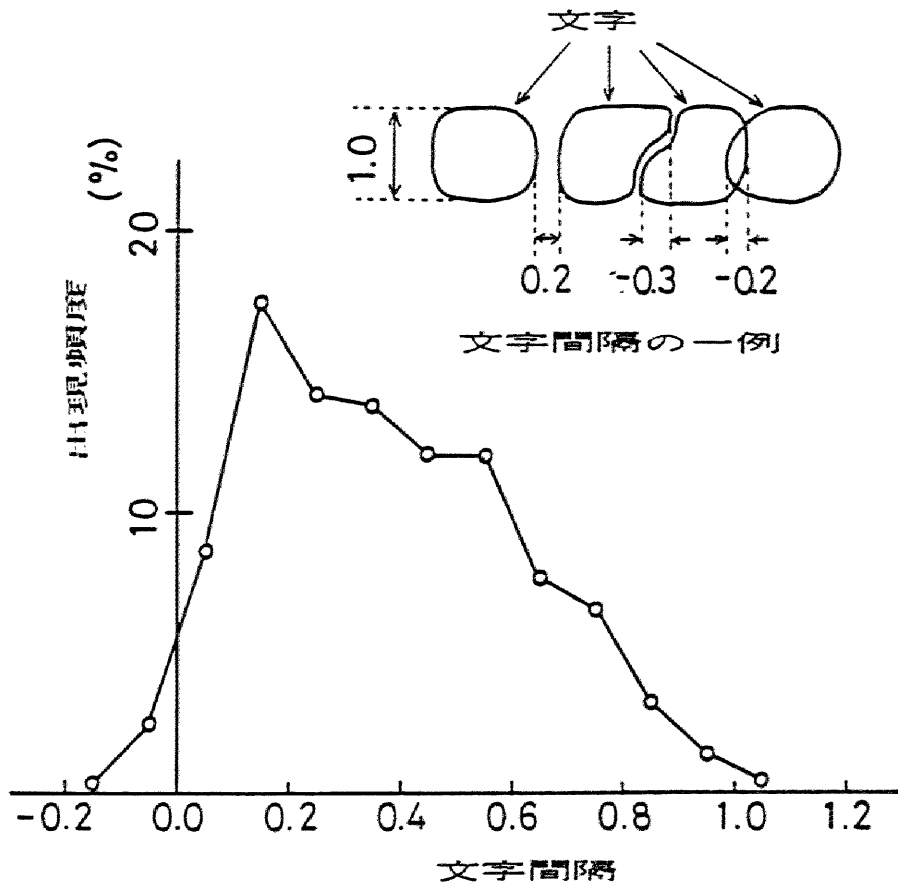


図5.4 (a) 文字間隔の分布 (全体)

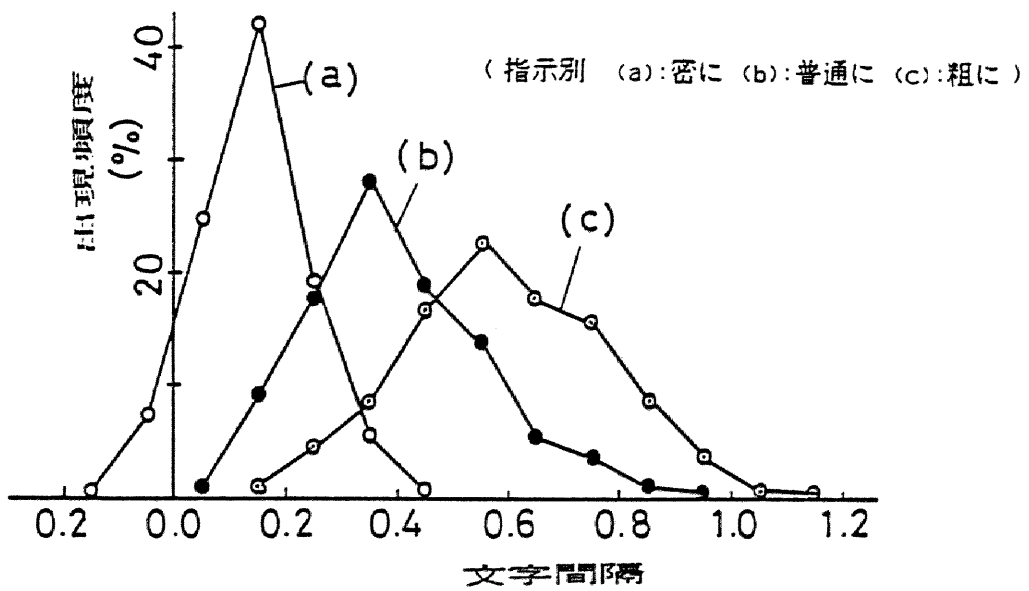


図5.4 (b) 文字間隔の分布 (指示別)

### 5.3.2 分離文字の存在

漢字や平仮名には、“科”，“川”，“い”等のように、左右に分離する文字が多数存在する。それ以外にも、“口”の第1ストロークが第2、第3ストロークと分離するような、手書き変形によりストロークの一部が左右に分離する場合なども存在する。そこで、文字の分離度を調査した。文字がMストロークから構成される場合に、第1ストロークから第i ( $1 \leq i < M$ ) ストロークまでの最大値を $X_1$ とし、第i+1ストロークから第MストロークまでのX座標の最小値を $X_2$ とした際、 $(X_2 - X_1) > T$  (Tは定数)となるiが1個でも存在すれば、この文字は間隔Tで分離すると呼ぶことにする。図5.5に、間隔Tに対する分離文字の割合を示す。但し、対象は画数変動文字を含む常用漢字と平仮名2148字種を各5サンプルずつ、合計10740文字である。各文字は前処理として大きさ1.0 (縦幅と横幅の最大値を1.0とする。)に正規化する。図5.5よりX軸への投影が分離するような文字 ( $T > 0$ )は43%存在することが確認される。この事実は、文字のセグメンテーションに単純投影法を適用したならば、単独文字が複数の図形に分離される可能性が高いことを示す。

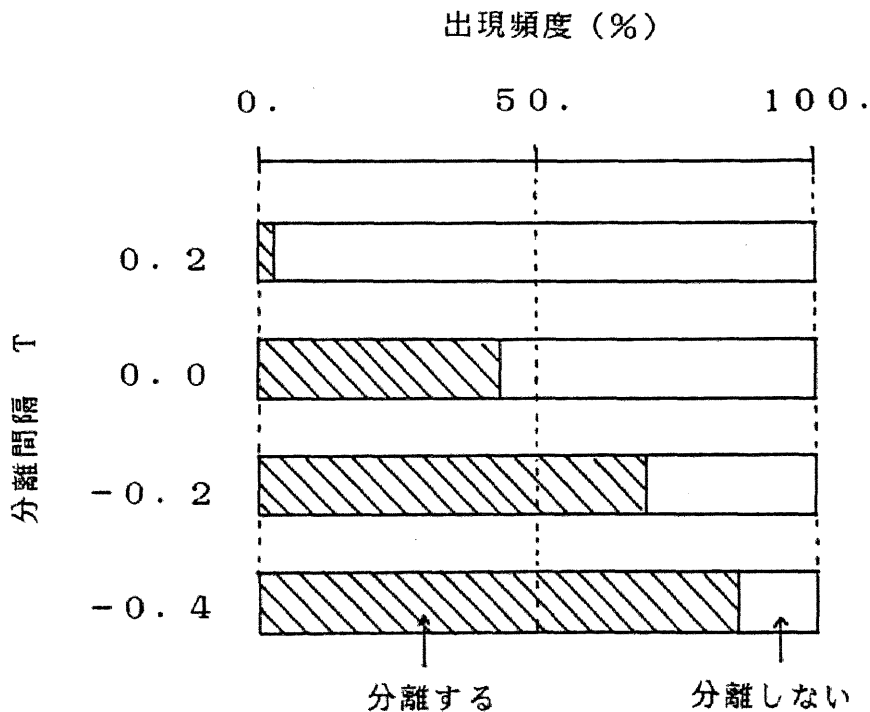


図5.5 分離間隔Tに対する分離文字の割合

## 5.4 認識アルゴリズム（候補文字ラティス法）

### 5.4.1 認識アルゴリズムの概要

本章で提案する候補文字ラティス法は文字のセグメンテーションと認識を同時に実現することにより、セグメンテーション精度を高める手法である。図5.6に、処理の流れを示す。処理は、文字列から基本セグメントへの分解、基本セグメントから候補文字の生成、候補文字の認識と文字認識結果の候補文字ラティスへの登録、候補文字ラティスの探索、分離有意文字処理の各段階から構成される。以下、各処理を詳細に述べる。

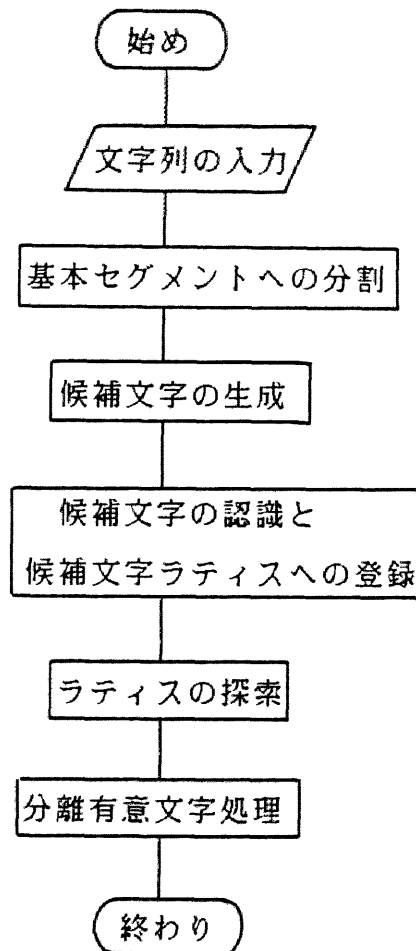


図5.6 候補文字ラティス法の流れ

#### 5.4.2 手書き文字列入力

手書き情報の入力にはデータタブレットを使用する。タブレットはペンのアップダウン情報とXY座標の値を一定時間間隔でサンプリングする機能を有する。ペンのダウンからアップまでの座標値系列をストロークと呼ぶことにする。入力ストローク数がM, 第j (1 ≤ j ≤ M) ストロークの座標点数がN<sub>j</sub> の場合の座標値系列を { (X<sub>ij</sub>, Y<sub>ij</sub>) i=1, ..., N<sub>j</sub>; j=1, ..., M } とする。

#### 5.4.3 文字列の大きさの正規化

前処理として、文字列全体の大きさの正規化とストローク毎の座標点数の規格化を行う。大きさは、各座標点のY座標の2次モーメントの値が一定になるように正規化する。正規化後の座標値系列を { (X'<sub>ij</sub>, Y'<sub>ij</sub>) i=1, ..., N<sub>j</sub>; j=1, ..., M } とすれば、X'<sub>ij</sub>, Y'<sub>ij</sub> は以下の式により与えられる。

$$\begin{cases} X'_{ij} = A \cdot X_{ij} & (5.1) \\ Y'_{ij} = A \cdot (Y_{ij} - \bar{Y}) & (5.2) \end{cases}$$

但し、

$$A = 0.25 \cdot \frac{1}{\sqrt{Y_m}} \quad (5.3)$$

$$Y_m = \frac{1}{P} \sum_{j=1}^M \sum_{i=1}^{N_j} (Y_{ij} - \bar{Y})^2 \quad (5.4)$$

$$\bar{Y} = \frac{1}{P} \sum_{j=1}^M \sum_{i=1}^{N_j} Y_{ij} \quad (5.5)$$

$$P = \sum_{j=1}^M N_j \quad (5.6)$$

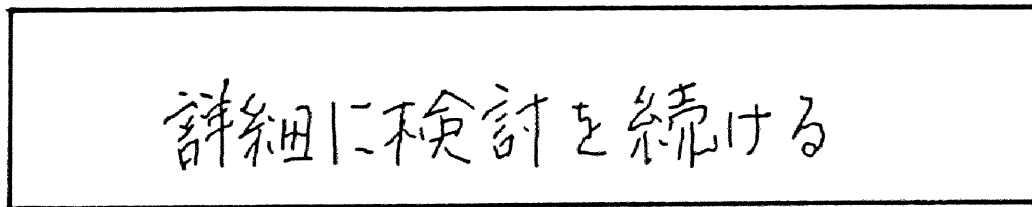
である。ここで、Pは総座標点数、 $\bar{Y}$ はY座標の平均値、Y<sub>m</sub>はY座標に関する2次モーメントを表わす。この変換により、変換後のY座標の2次モーメントは0.25に正規化される。これは文字列の平均的な縦幅(高さ)を1.0とすることに相当する。図5.7に入力文字列と、正規化後の文字列を示す。

一方、入力された座標点は一定時間間隔でサンプリングされたものであるため、座

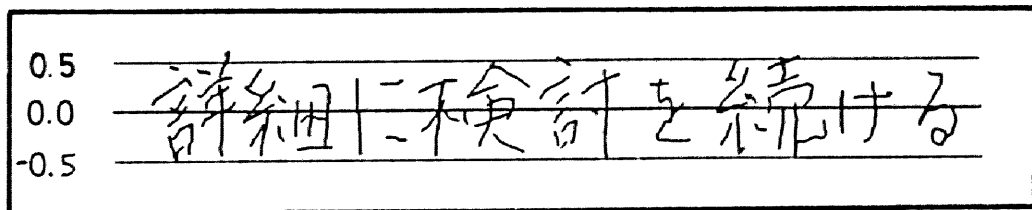
標点間の間隔は筆記速度により変動する。そこで座標点間の間隔が一定値入となるように座標値系列を変換する。変換の手法は2.3.2(2)の停留点の除去法と同等の手法を採用する。 $\lambda$ は文字を表現するのに十分な値に設定した。変換後の座標値系列を

$$\{ (X''_{ij}, Y''_{ij}) \mid i=1, \dots, N'_j; j=1, \dots, M \} \quad (5.7)$$

とする。但し、 $N'_j$ は第  $j$  ストロークの座標点数である。



( a ) 入力文字列



( b ) 正規化後の文字列

図5.7 文字列の大きさの正規化

#### 5.4.4 基本セグメントへの分割

文字列を構成する入力ストローク列を複数の基本セグメントに分割する。方針は、基本セグメントの中に文字間の分離箇所が含まれない範囲で、なるべく大きなストロークの組が基本セグメントになるように、分割することにある。基本セグメントの決定パラメータを  $Th$  とする。第1ストロークから第  $k$  ( $1 \leq k < M$ ) ストロークまでの  $X$  座標の最大値を  $X_1$  とし、第  $k+1$  ストロークから第  $M$  ストロークまでの  $X$  座標の最小値を  $X_2$  とし、

$$(X2 - X1) > Th$$

(5.8)

を満足する  $k$  をすべて求める。各  $k$  に対し第  $k$  ストロークと第  $k + 1$  ストロークとの間で入力ストローク列を分割し、分割された各ストロークの組を基本セグメントとする。図 5. 8 に、 $Th$  に対する基本セグメントの例を示す。 $Th$  を小さくすれば基本セグメントの単位が小さくなるが、 $Th$  を大きくしすぎれば基本セグメント内に文字間の分離箇所が含まれる。図 5. 5 より、通常横方向に手書きされる文字列では文字間の重さなりは 15% 以下と考えられる。この結果より、 $Th$  として  $-0.15$  を設定する。分割後の基本セグメントを  $e_1, \dots, e_t$  とする。但し、 $t$  は基本セグメント総数である。

#### 5. 4. 5 候補文字の生成

文字列中の各文字は基本セグメント ( $e_1, \dots, e_t$ ) を組み合わせることにより構成されるが、この段階では組み合わせ方は不明である。そこで、基本セグメントを組み合わせ、組合わせた図形が文字らしくなるすべての場合に対し、この図形を候補文字とする。基本セグメントを組合わせた図形が候補文字となるための条件は、

- ① 図形は引き続き基本セグメントから構成される。
- ② 図形の横幅は 2. 0 以下 (経験的に十分な値) である。
- ③ 図形を囲む長方形の長辺は 0. 2 以上 (経験的に十分な値) である。
- ④ 図形のストローク数は 23 以下 (常用漢字の最大画数は 23) である。

と設定した。図 5. 8 で示した基本セグメント ( $Th = -0.15$ ) から生成した候補文字を図 5. 9 に示す。候補文字数は実際の文字列に比較して一般に多数生成される。図 5. 9 に示した例では、11 文字から構成される文字列に対し、24 の基本セグメント ( $e_1, \dots, e_{24}$ ) が抽出され、基本セグメントの組み合わせにより 98 個の候補文字が生成される。この場合、実際の文字数に比較し約 10 倍の候補文字が生成されることになる。

Input character string

地理的に離れた場所での

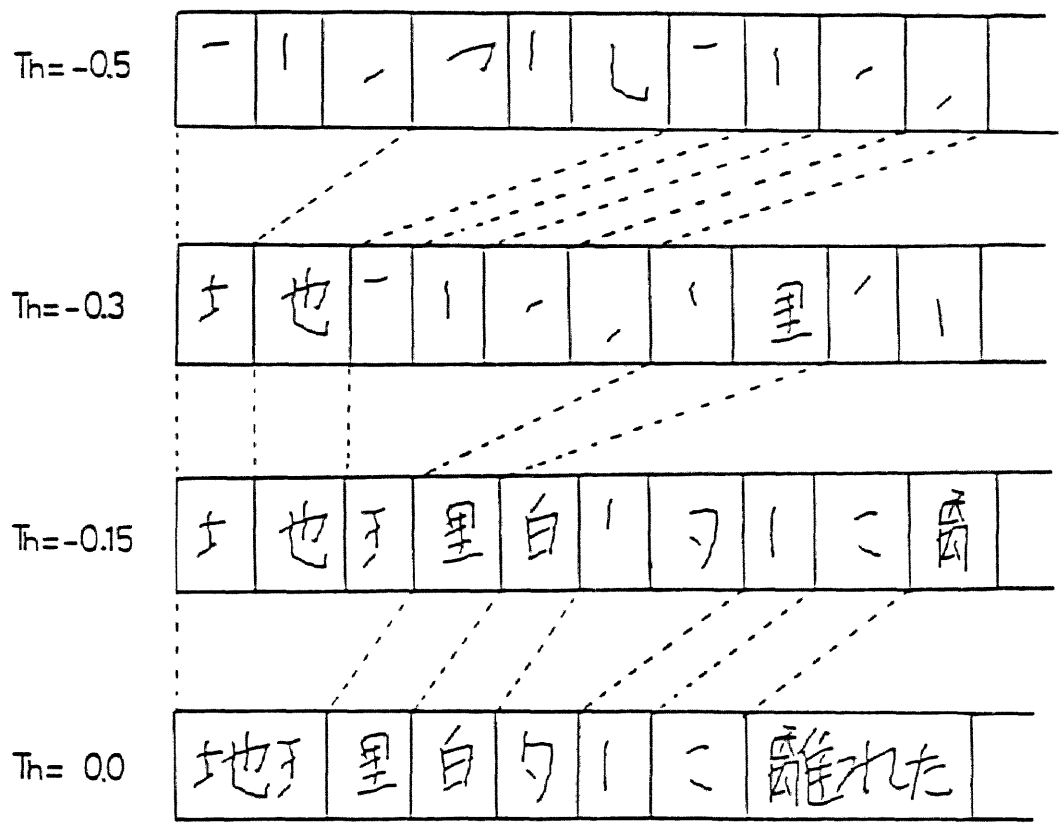


図5.8 重なる許容度Thに対する基本セグメントの例



Basic segment	e1	e2	e3	e4	e5	e6	e7
	土	也	牙	'	里	白	'
	土	也	牙	'	里	白	'
	地		牙	里		白	
		也		里	白		
	地			里		白	
		也		里		白	
			理		的		

図5.9 候補文字の生成

#### 5.4.6 候補文字の認識

前節で生成された各候補文字を認識することにより、各候補文字はどの文字らしいか（候補文字認識結果）、どの程度その文字らしいか（相異度）を求める。以下、候補文字の認識手順を述べる。

##### (1) 前処理

前処理として、候補文字の座標値系列の重心と2次モーメントを利用して、候補文字の位置と大きさの正規化を行う。候補文字の座標値系列を  $\{(U_{ij}, V_{ij}) \mid i=1, \dots, K_j; j=1, \dots, L\}$ （但し、 $L$ はストローク数、 $K_j$ は第 $j$ ストロークの座標点数）とし、正規化後の座標値系列を  $\{(U'_{ij}, V'_{ij}) \mid i=1, \dots, K_j; j=1, \dots, L\}$ とした場合、 $U'_{ij}, V'_{ij}$ を

$$\begin{cases} U'_{ij} = B \cdot (U_{ij} - \bar{U}) & (5.9) \\ V'_{ij} = B \cdot (V_{ij} - \bar{V}) & (5.10) \end{cases}$$

で定義する。但し、

$$\bar{U} = \frac{1}{Q} \sum_{j=1}^r \sum_{k=1}^{K_j} U_{ij} \quad (5.11)$$

$$\bar{V} = \frac{1}{Q} \sum_{j=1}^r \sum_{k=1}^{K_j} V_{ij} \quad (5.12)$$

$$Q = \sum_{j=1}^r K_j \quad (5.13)$$

$$B = 0.5 \cdot \frac{1}{\sqrt{m}} \quad (5.14)$$

$$m = \frac{1}{Q} \sum_{j=1}^r \sum_{k=1}^{K_j} \{ (U_{ij} - \bar{U})^2 + (V_{ij} - \bar{V})^2 \} \quad (5.15)$$

である。ここで、 $Q$ は総座標点数、 $\bar{U}$ 、 $\bar{V}$ は各 $U_{ij}$ 、 $V_{ij}$ の平均値、 $m$ は $(\bar{U}$ 、 $\bar{V})$ の回りの2次モーメント、 $B$ は正規化係数である。

## (2) 候補文字認識用の辞書の作成

画数変動文字を含んだ常用漢字と平仮名2148カテゴリーについて辞書を作成する。辞書の作成手順を以下に示す。最初に、辞書作成用文字の各ストロークの座標系列を一定の点数（ここでは15点）で近似し、前処理として上記(1)の処理を施す。次に各カテゴリーに対し20個の文字を使用し、対応する座標点毎に座標値の平均値を求める。最後に、座標点間の距離が一定値になるように変換する(4.3の後半の処理を参照)。この座標値系列を候補文字認識用の辞書とする。

## (3) ストローク間の相異度の定義（位置と方向のDPマッチング）

ストローク間の相異度は(4)で述べる文字間の相異度の計算に使用される。2本のストローク間の相異度をストローク各点の位置（座標値）とその点での接線方向の値のDPマッチングにより定義する。DPマッチングは本手法を画数変動を含む低品質文字認識に拡張する際に有効になる。一方のストロークの座標値系列を $\{ (U_i,$

$V_i \ (i=1, \dots, I)$  とし、他方を  $\{(U'_j, V'_j) \ j=1, \dots, J\}$  とする場合、両者のストローク間相異度  $d$  を、

$$d = \frac{1}{J} \min_u \left\{ \sum_{j=1}^J (g(i, j, u(j)) + \alpha \cdot h(i, j, u(j))) \right\} \quad (5.16)$$

で定義する。但し、 $I, J$  はストロークの近似点数であり、 $I$  と  $J$  とは一般に異なる。 $g(i, j)$  は  $(U_i, V_i)$  と  $(U'_j, V'_j)$  の間のユークリッド距離を表わす関数で、

$$g(i, j) = \sqrt{(U_i - U'_j)^2 + (V_i - V'_j)^2} \quad (5.17)$$

となる。 $h(i, j)$  は、点  $(U_i, V_i)$  から点  $(U_{i+1}, V_{i+1})$  に向かうベクトルと点  $(U'_j, V'_j)$  から点  $(U'_{j+1}, V'_{j+1})$  に向かうベクトルのなす角度で定義する。 $u(j)$  は座標点間の対応関係を表わす関数であるが、不適当な対応を防止するため制限条件として、

$$\begin{cases} u(1) = 1 & (5.18) \\ u(J) = I & (5.19) \end{cases}$$

$$u(j+1) = \{u(j) \text{ or } u(j) + 1 \text{ or } u(j) + 2\} \quad (\text{但し、} 1 < j < J) \quad (5.20)$$

を課した。 $\alpha$  は接線方向の特徴の寄与度をあらわす定数であり、(6) で述べる予備実験で設定される。

#### (4) 文字間相異度の計算

候補文字は複数のストロークから構成される。まず、辞書の中から候補文字のストローク数 ( $L$  とする) と同じストローク数の標準文字を選択し、選択された標準文字のすべてと候補文字との間で文字間の相異度を求める。文字間の相異度は対応するストローク間の相異度 ((3) 参照) の和で定義する。但し、候補文字の筆順は自由であるため、ストロークの対応は最初から決定できない。そこで、ストローク間相異度

行列 [38] を用いて、対応するストロークを決定する。標準文字の第  $i$  ( $1 \leq i \leq L$ ) ストロークと、候補文字の第  $j$  ( $1 \leq j \leq L$ ) ストロークとのストローク間相異度を  $d_{ij}$  とすれば、ストローク間相異度行列は、

$$\begin{pmatrix} d_{11} & d_{12} & \cdot & \cdot & d_{1L} \\ d_{21} & d_{22} & \cdot & \cdot & d_{2L} \\ \cdot & \cdot & d_{ij} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ d_{L1} & d_{L2} & \cdot & \cdot & d_{LL} \end{pmatrix} \quad (5.21)$$

となる。この行列の各行から最小の値を選択し（対応するストロークの決定）、それらの和を文字間の相異度とする。つまり、文字間の相異度  $D$  は、

$$D = \sum_{j=1}^L \{ \min_{i=1, \dots, L} (d_{ij}) \} \quad (5.22)$$

で定義される。

#### (5) 候補文字の認識

候補文字と、候補文字に等しいストローク数を持つすべての標準文字（辞書）との間で文字間相異度を求め、最小の相異度を持つ標準文字のコードを認識結果とする。候補文字の認識結果とその相異度は 4.7 で述べる候補文字ラティスに登録される。候補文字の認識例を図 5.10 に示す。図中の  $\circ \times \triangle$  はストロークあたりの相異度を基準に著者が設定したものであり、これは次段階の処理では使用されない。

候補文字	構成セグメント	認識結果	相異度	$\frac{\text{相異度}}{\text{ストローク}}$	文字らしさ
	e1	士	0.72	0.24	△
	e1, e2	地	1.22	0.20	○
	e1, e2, e3	班	2.40	0.24	△
	e2	世	1.01	0.34	×
	e2, e3	作	2.73	0.39	×
	e2, e3, e4	性	3.40	0.42	×
	e3	王	1.61	0.40	×
	e3, e4, e5	理	1.85	0.17	○

図5.10 候補文字の認識

## (6) 文字認識実験 (予備実験)

(1) ~ (5) で述べた認識手法を評価するため、未学習文字 21950 文字 (各カテゴリ 10 文字) を用いて認識実験を行った。図 5. 11 に  $\alpha$  ( (3) 参照) をパラメータとした認識率を示す。 $\alpha = 0.3$  とすることにより、1 ~ 3 画に対して 98.8%、4 ~ 23 画に対して 99.9%、平均で 99.8% の認識率を得ることを確認した。特に、接線方向特徴の利用は、平仮名等の画数の少ない文字に有効であることも確認した。

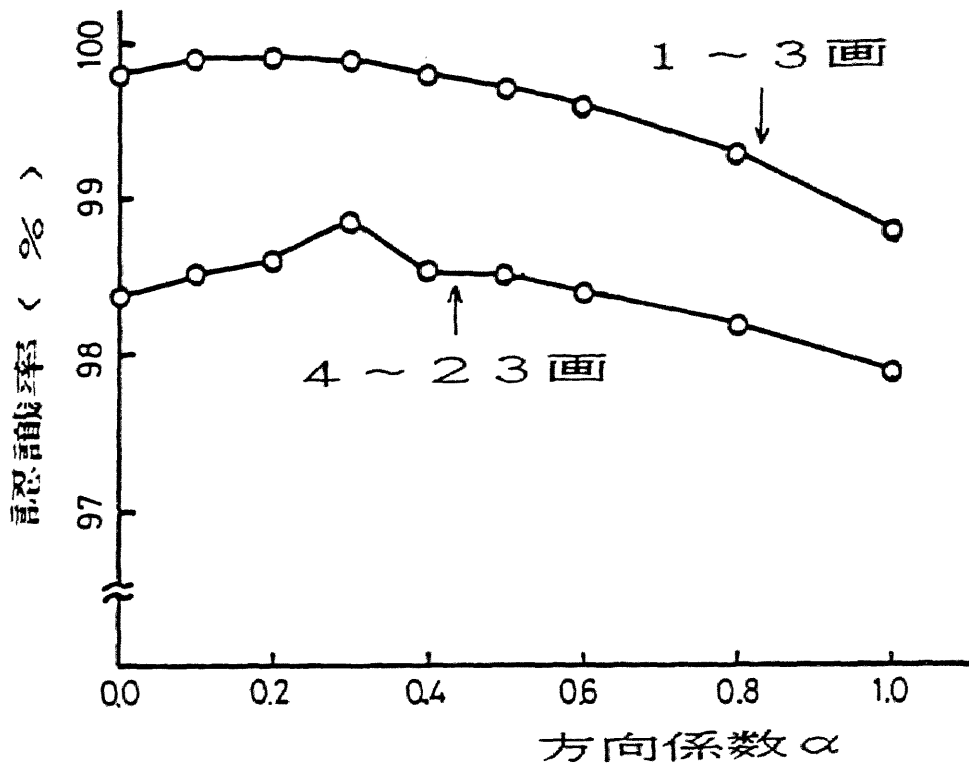


図 5. 11 方向係数  $\alpha$  に対する文字認識率

### 5. 4. 7 候補文字ラティスの作成

候補文字ラティスは、入力文字列中の各基本セグメントがどの候補文字の一部であるかの情報、各候補文字の認識結果、及び認識時の相異度を表形式に表現したものである。図 5. 9 に示した表の各欄に候補文字の認識結果と相異度を代入することにより、候補文字ラティスが生成される。候補文字ラティスの例を図 5. 12 に示す。

Basic segment	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10
	士	也	王	く	耳	白	く	又	し	こ
	0.72	1.01	1.61	0.39	1.98	1.16	0.47	0.94	0.46	0.56
	地 1.22		げ 1.89		略 3.14		勺 0.74		に 0.87	
candidate character	作 2.73		里 1.37		自 1.57		孔 1.23			
	班 2.40		軸 2.66		双 1.78					
distance	性 3.40			軸 3.27			加 2.07			
	理 1.85			的 1.76						
	理 3.70			雌 4.30						

図5. 12 候補文字ラティスの例

#### 5. 4. 8 候補文字ラティスの探索

入力文字列中に含まれる各文字をセグメンテーションし、各文字を認識することは、候補文字ラティスの中からすべての基本セグメントを被う最適な文字系列を選択することにより実現できる。候補文字ラティスは2端子の重み付きグラフ（候補文字の認識結果をブランチに、相異度をその重みとするグラフ）により表現できる。図5. 12に示す候補文字ラティスに対するグラフ表現の例を図5. 13に示す。すべての基本セグメントを被う文字系列の選択は、このグラフの一方の端子から他方の端子に向かうパスの選択に対応する。このパスは一般に多数存在する。そこで、評価関数  $S$  を導入し、 $S$  を最小にするパスを探索する。パスの探索には動的計画法を採用した。 $S$

は、

$$S = \sum_{\text{ブランチ} \in \text{パス}} (\text{ブランチの重み}) \quad (5.23)$$

で定義する。つまり、ブランチの重みの和を最小にするパスを選択する。このパスに対応する文字系列を仮の認識結果とする。図5.12中に、太枠で仮の認識結果を示す。

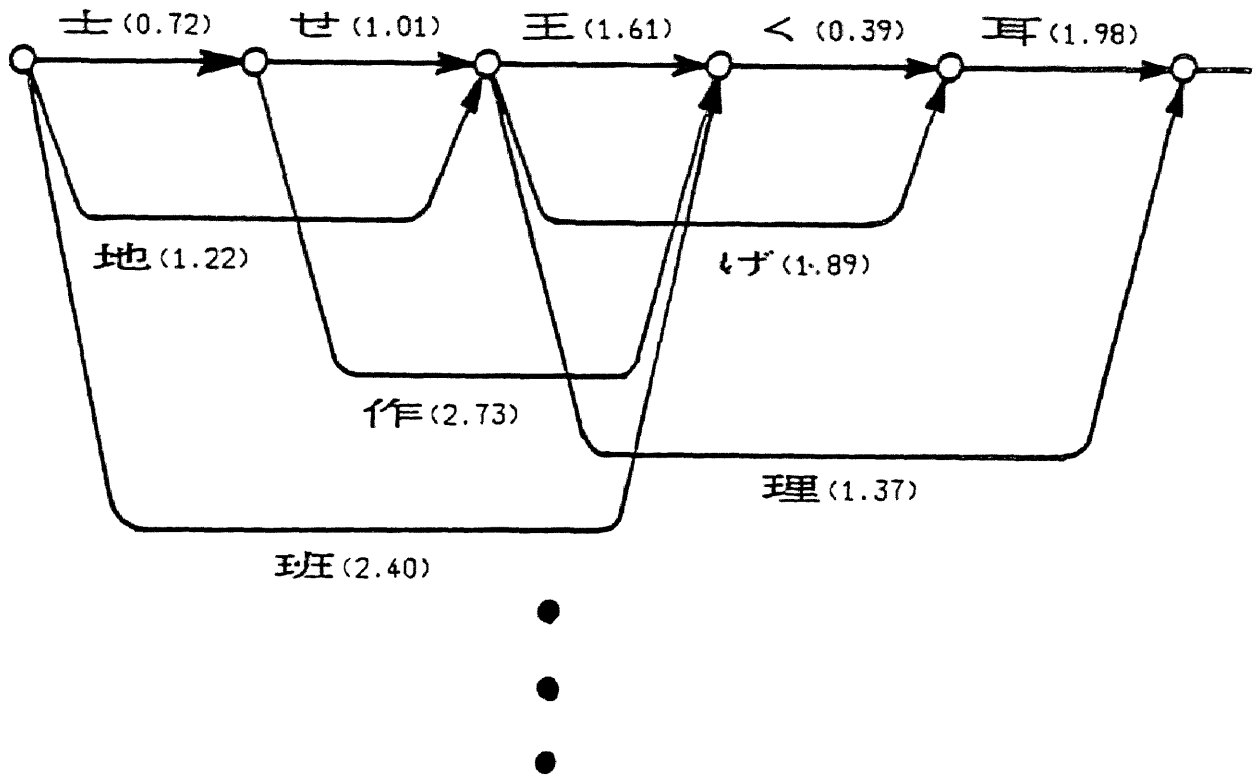


図5.13 候補文字ラティスのグラフ表現

#### 5.4.9 分離有意文字処理

5.4.8までの処理により、たとえ文字が偏と旁、或は手書き変形のため左右分離した場合でも、左右の部分図形の少なくとも一方が文字として意味を持たなければ



文字列は正しく認識できる。しかし、文字が左右に分離し、且つ分離した各部分図形がどちらも文字として意味を有する場合（この文字を分離有意文字と呼ぶ）には、5.4.8までの処理では必ずしも正しく認識されない。例えば、「説明」という文字列が、「言」、「兌」、「日」、「月」の4個の基本セグメントに分割された場合には、図5.14のような2端子グラフが生成される。この場合には、「兌」という文字が存在しないために「説」は正しく認識されるが、分離有意文字である「明」は、「明」と認識されるか「日」+「月」と認識されるかは文字間の相異度の情報だけでは正しく決定できない。分離有意文字は常用漢字1945字種中266字種存在する。分離有意文字の一例を図5.15に、その全てを付録Cに示す。

ある分離有意文字Fを $F = f_l + f_r$ と表現する（例えば、 $F = \text{「明」}$ 、 $f_l = \text{「日」}$ 、 $f_r = \text{「月」}$ ）。候補文字ラティス中にFとそれに対応して $f_l$ 、 $f_r$ なる候補文字が存在し、且つ仮の認識結果（5.4.8参照）としてそのFまたは $f_l$ 、 $f_r$ なる文字列が選択された場合には、正しく認識されていない可能性がある。そこで、この場合には文字Fの縦横比と、文字 $f_l$ と $f_r$ の縦横比の平均値を比較し、その値が1.0に近い方の文字（文字列）に修正する。この処理を分離有意文字処理と呼ぶ。

分離有意文字処理を経て選択された候補文字の認識結果の系列を、入力文字列の最終認識結果とする。

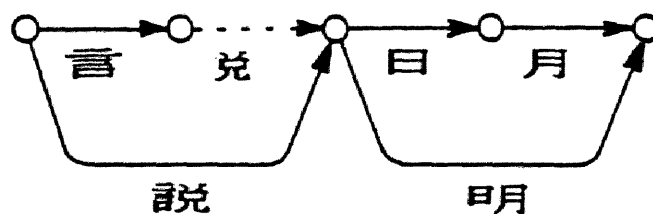


図5.14 文字列「説明」の場合のグラフ表現

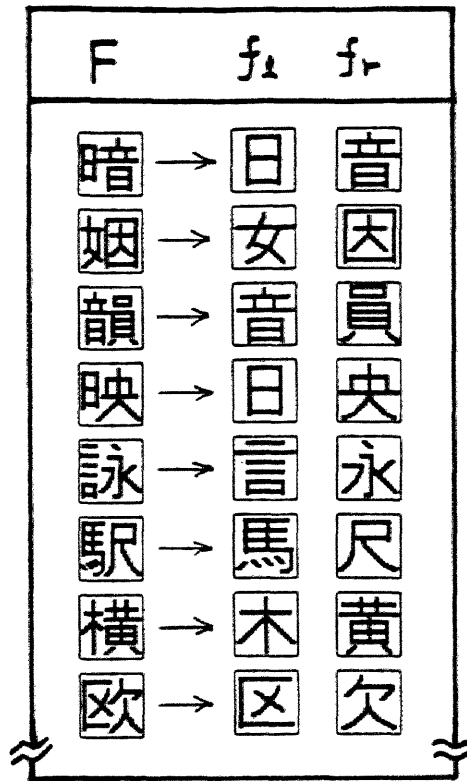


図5. 15 分離有意文字の一例

### 5. 5 認識実験結果

5. 4で述べた認識手法の効果を確認するため、105個の文字列データ（2. 1参照）を認識対象の一例として認識実験を行った。

#### （1）分離有意文字処理を行わない場合

候補文字ラティス中から単純に相異度の和を最小とする文字系列を選択する手法（5. 4. 1～5. 4. 8参照）により文字列データを認識した結果、認識率は95. 6%であった。但し、認識率を、

$$\text{認識率} = \frac{(\text{正しく認識された文字数})}{(\text{総文字数})} \cdot 100\% \quad (5.24)$$

で定義する。表5.1に誤読5.1%の内訳を示す。

表5.1 認識率と誤認識内訳

		分離有意文字処理	
		無し	有り
認識率		95.6%	96.3%
セグメンテーション誤り	基本セグメント分割誤り	0.6	0.6
	ラティス探索誤り	2.8	2.1
文字認識誤り		1.0	1.0

(2) 分離有意文字処理を行った場合

(1) に示した誤読中、セグメンテーション誤りの一部は分離有意文字処理を導入することにより改善される。その結果、認識率は96.3%に向上した。表5.1に誤読3.7%の内訳を、以下に誤読例を示す。図5.16に誤読した入力パターンの一例を示す。

(1) 基本セグメント分割誤り

「通話」→「嚇」+「舌」(1個所)

「に検」→「論」(1個所)

他 (全4文字分)

(2) ラティス探索誤り

「所」→「戸」+「斤」(3個所)

「的」→「白」+「勺」(2個所)

「続」→「糸」+「売」(2個所)

「話」→「言」+「舌」(1個所)

「話」→「言」+「吉」(1個所)

「も」→「こ」+「し」（4個所）

「に」→「し」+「こ」（2個所）

「か」→「力」+「し」（1個所）

他 （全29文字分）

(3) 文字認識誤り

「細」→「組」（1文字）

「で」→「ぞ」（1文字）

他 （全10文字分）

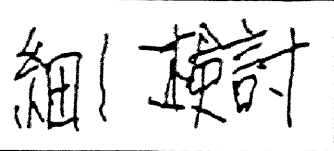
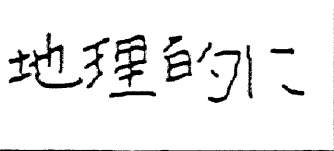
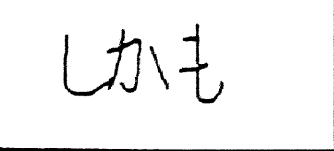
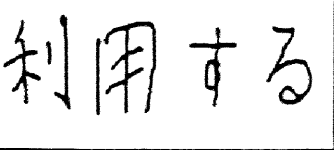
input samples		recognition results
	細に検討	糸 ? 討
	地理的に	地理白夕に
	しかも	し力しも
	利用する	利申する

図5.16 誤読文字列例

## 5.6 考察

### 5.6.1 本手法の適用範囲

本手法は文字のセグメンテーションに文字ピッチや文字の大きさ等の情報を使用していないため、以下のような文字列を認識することが可能である。

①文字の大きさに比較して $T_h$  (4.4参照)までの文字間の重なりを許容する(図5.17(1))。

②完全な不定ピッチを許容する(図5.17(2))。

③同一文字列内に文字の大小混在を許容する(図5.17(3))。

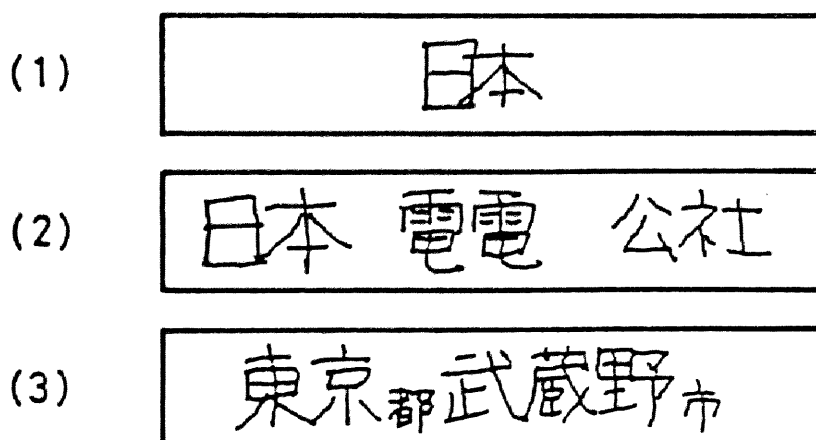


図5.17 可読文字列例

### 5.6.2 計算量

本手法では、基本セグメントの組み合わせから生成されるすべての候補文字を認識し、セグメンテーションを高精度化しているため、候補文字認識回数が多くなり、計算量は増大する。例えば、図5.2に示した文字列の例では、1文字に対し必要となる平均的な候補文字認識回数は約9回、そのための認識時間は1MIPSのミニコン(言語としてFORTRAN77を使用)でシミュレーションしたところ約18秒であった。必要な認識時間は文字の重なり許容度 $T_h$ に依存する。その結果を認識率とともに図5.18と表5.2に示す。今回は $T_h$ を $-0.15$ に設定(5.4.4参照)したが、この値は認識率を下げない範囲で最少の計算時間となっている。文字間の重なり以上に大

きい低品質な文字列を認識する為には、 $T_h$  を小さくする必要があり、この場合認識時間は増加する。

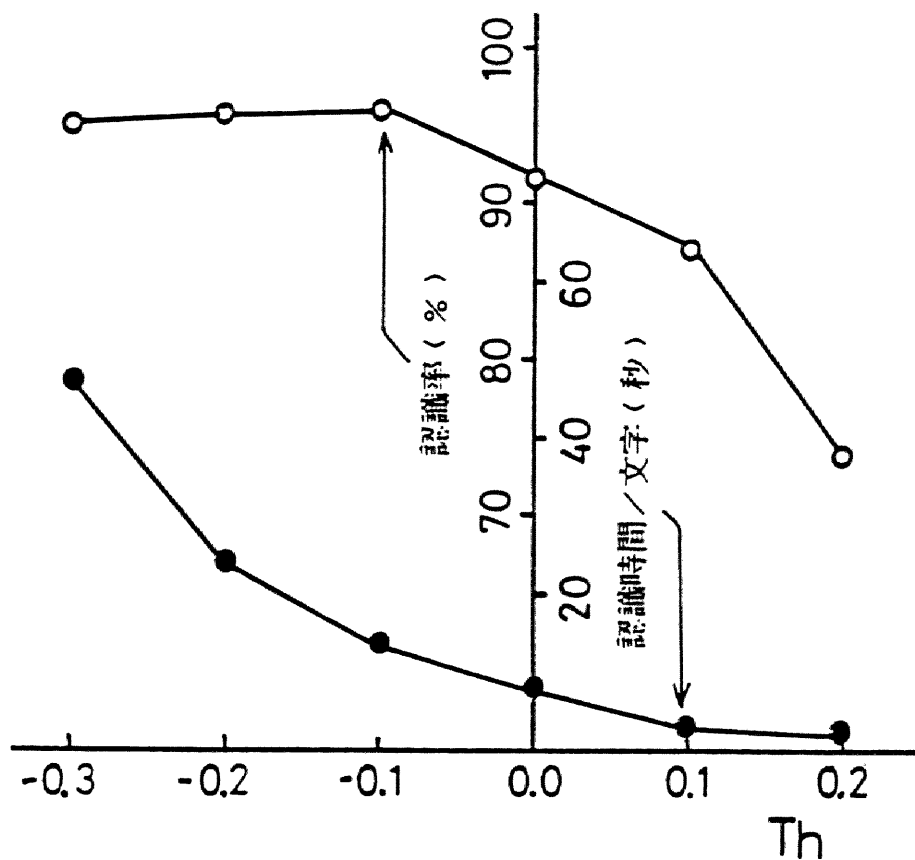


図5. 18 1文字あたりの認識時間と認識率 対  $T_h$

表5.2 重なるの許容度 (Th) に対する認識率と認識回数

重なるの許容度 (Th)	認識回数 (時間) / 文字	認識率
-0.3	23.4 (47秒)	94.8%
-0.25	17.9 (36)	95.4
-0.2	12.2 (24)	95.5
-0.15	8.8 (18)	95.6
-0.1	7.2 (14)	95.8
-0.05	5.6 (11)	94.6
0.0	4.0 (8)	93.3
0.05	2.7 (5)	91.1
0.1	1.8 (3)	87.3
0.15	1.1 (2)	81.3
0.2	0.7 (1)	74.1
0.25	0.7 (1)	68.1

### 5.6.3 他手法との比較

単純投影法により文字列から文字をセグメンテーションし、各文字を文字認識する手法と、本手法とを比較した。単純投影法は文字列を構成する各ストロークを横軸に投影し、その影の間隔がT (いき値) 以上になる箇所で文字をセグメンテーションする手法である。文字列データ105文例に対し、Tをパラメータとしてセグメンテーションの実験を行った。セグメンテーションの成功率を図5.19に示す。但し、図5.19には $T > 0$ の場合のみを示す。Tを小さな値に設定すれば1文字が複数の図形に分離されるため誤読は増加し、Tを大きな値に設定すれば文字間が連結するため誤読は増加する。Tを最適に設定することにより68.5%のセグメンテーション成功率が得られた。これを文字認識したところ、文字列の認識率は68.0%であった。表5.3に、本手法との認識率の比較を示す。

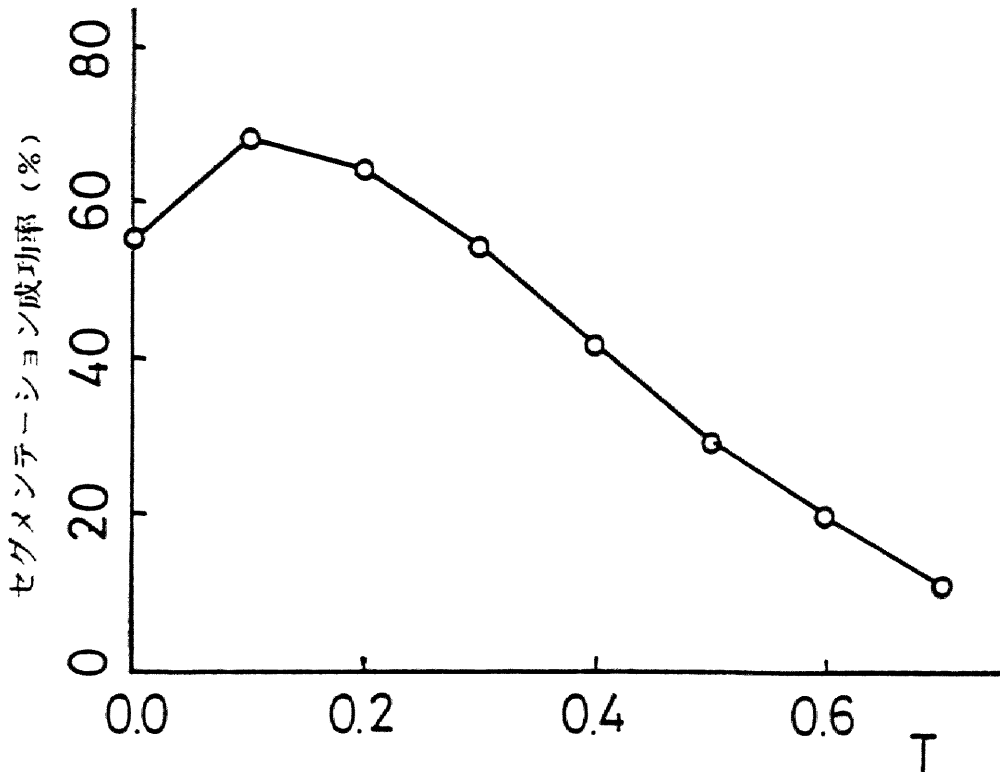


図5. 19 単純投影法におけるTに対するセグメンテーション成功率

表5. 3 文字列の認識率

単純投影法	候補文字ラティス法
68.0%	96.3%

#### 5. 6. 4 分離有意文字処理の拡張

本手法では分離有意文字対策として文字の縦横比を利用したが、これ以外に単語知識に基づく文字間の接続関係を利用することもできる。例えば、“動”は左右分離しそれが接続して“無重力”等の単語を作るが、“欲”は左右分離すると単語を作らないため“谷欠”と認識するより“欲”と認識すべきである。学研国語大辞典の見出し語約8万単語により調査した結果、分離有意文字266字種中15字種については、



左右に分離した文字が接続して単語をつくる（注1）。つまり、残り251字種に対しては左右分離した文字の接続を含む単語はない。この場合には左右に分離しないと判定すれば良い。しかし、固有名詞や2単語が連続する場合をも考慮すると、上記251字種に対しても左右の文字が接続する可能性があり、上記の処理だけでは不十分である。これらについての本質的な解決法は、6章で述べるような言語処理の導入である。

（注1）学研国語大辞典によれば、分離有意文字266字種中で、左右に分離した文字が接続して単語をつくる、あるいは単語の一部となるものが15字種存在する。以下に示す。

- |      |    |                  |        |
|------|----|------------------|--------|
| (1)  | 好： | 女子（じょし）          |        |
|      |    | 小女子（こうなご）        | <合計2個> |
| (2)  | 姓： | 女生（じょせい）         |        |
|      |    | 女生徒（じょせいと）       | <合計2個> |
| (3)  | 誠： | 言成り（いいなり）        | <合計1個> |
| (4)  | 双： | 又又（またまた）         | <合計1個> |
| (5)  | 相： | 木目（もくめ、きめ）       |        |
|      |    | 木目込細工（きめこみざいく）   |        |
|      |    | 木目込人形（きめこみにんぎょう） |        |
|      |    | 木目細か（きめこまか）      |        |
|      |    | 木目込み（きめこみ）       | <合計5個> |
| (6)  | 駐： | 馬主（ばしゅ）          | <合計1個> |
| (7)  | 脹： | 月長石（げっちょうせき）     | <合計1個> |
| (8)  | 動： | 重力（じゅうりょく）       |        |
|      |    | 無重力（むじゅうりょく）     | <合計2個> |
| (9)  | 畑： | 火田（かでん）          | <合計1個> |
| (10) | 鉢： | 金本位（きんほんい）       | <合計1個> |
| (11) | 坊： | 土方（どかた）          | <合計1個> |
| (12) | 妨： | 女方（おんながた）        | <合計1個> |
| (13) | 岬： | 穿山甲（せんざんこう）      | <合計1個> |
| (14) | 明： | 閑日月（かんじつげつ）      |        |

三日月（みかづき）	
短日月（たんじつげつ）	
二日月（ふつかづき）	
日月（じつげつ）	<合計 5 個>
(15) 林： 木木（きぎ）	<合計 1 個>

## 5.7 結言

本章では、真に扱い易い入力形態を追求するため、セグメンテーション指示なく横方向に手書きされた文字列の認識手法として「候補文字ラティス法」を考案し、その性能を実験的に確認した。以下に、まとめる。

(1) 文字列からの文字のセグメンテーションと各文字の認識を有機的に結合したことにより、文字のピッチ等の情報を全く使用することのない高精度な文字列認識法を実現した。その結果、完全な不定ピッチ文字列、文字間の重なりを許容する文字列、大小文字混在の文字列の認識が可能となった。文字列データ 105 文例に対し 96.3% の認識率を得た。

(2) 文字の縦横比を利用した分離有意文字処理を導入したことにより、“理”，“炉”等の2文字が左右に配置されることにより構成される分離有意文字のセグメンテーション精度は向上し、導入しない際のセグメンテーション誤り 3.4% を 2.7% に低減することが可能になった。

この章で提案した「候補文字ラティス法」により、1文字1文字を枠内に筆記することなく、自由書式で筆記されたオンライン手書き文字列を認識することが可能となった。しかし、この手法だけでは分離有意文字に対する認識精度を向上させることは困難である。そこで、6章では、文字列の認識処理を階層化し、文字列からの文字の切り出しと、個々の文字の認識と、言語処理を統合する。これにより、言語知識等の上位概念を導入し、より高精度の文字列認識手法について検討を加える。

## 第6章 言語情報を利用した自由書式 手書き文字列の高精度認識法

### 6.1 緒言

白紙紙面上あるいは罫線上に自由書式で筆記できる入力手法を確立するため、5章では、文字のセグメンテーションと文字認識を融合した文字列認識手法を述べた。しかし、日本語文章には「明」（「日」＋「月」）のように、偏とつくりに分離し、偏とつくりがそれ自身で文字となる場合が多数含まれる。このような場合には、文字認識の結果を文字のセグメンテーションに利用しても、文字のセグメンテーションが一意に決定できない。5章ではこれに対して、文字を囲む矩形の縦横比を利用した分離有意文字処理の検討を加えたが、これは本質的な解決ではない。一方、文字認識の結果を文字のセグメンテーションにフィードバックする手法等〔83, 84, 89〕も提案されているが、これらも同様の問題点を有している。文字認識の情報を利用するだけで文字列から文字を正確にセグメンテーションし、個々の文字を認識することは困難である。人間がこれらの文字を含む文字列からでも簡単に文字をセグメンテーションし、認識しているのは、更に上位の知識である言語情報をも利用しているためと考えられる。

言語情報を文字認識に利用する手法としては、文字認識で得られた候補文字の中から言語的に正しい文字列を選出する手法〔105-107〕、複数の文字を単位に照合する手法〔108〕が提案されている。しかし、それらはいずれも文字の認識精度を向上させるための文字認識後処理法であり、文字の切り出し精度を向上させるために言語情報を利用する手法ではなかった。

この章では、文字のセグメンテーション、文字認識、言語処理を統合した手書き文字列の認識系を構築し、その認識系を利用して言語情報を文字のセグメンテーションに有効にフィードバックする手法を提案する〔75-77〕。同様の考え方は連続音声認識〔85〕でもなされている。次に、本手法の効果を実験的に確認する。ここではオンライン手書き文字列データに対して検討するが、本手法は必ずしもオンライン文字列データを対象とする必要はない。また実験は横書き文字列を対象に行なったが、本手法を縦書き文字列に拡張することは容易である。本手法により、自由書式で筆記された文字列から文字を高精度に切り出すと同時に認識することが可能となる。

この手法の特長を以下に示す。

(1) ラティス(2端子有向グラフ)の記述を2段階(候補文字ラティスと単語ラティス)に利用することにより、上位概念から下位概念までを有機的に結合することができる。

(2) ラティス上のブランチの距離和がK番目に最短となる経路を、動的計画法により探索する方法を提案した。これにより、言語情報を文字のセグメンテーションと文字認識の両者に有効にフィードバックすることが可能となる。これにより、高精度に自由書式で筆記された文字列を認識することができる。

6章での主眼点は、言語処理を導入した文字列からの文字のセグメンテーションと認識である。この章の位置付けを図6.1に示す。

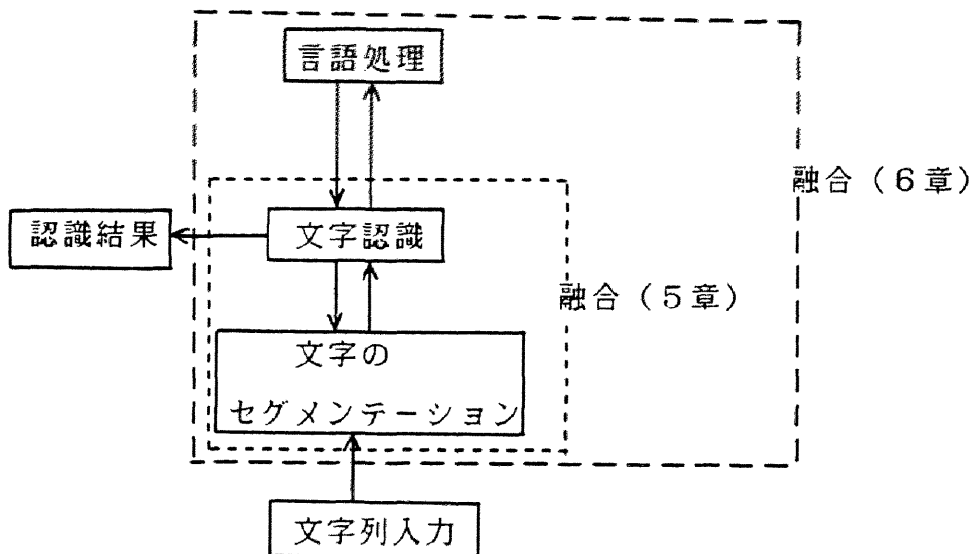


図6.1 この章の位置付け

以下、6.2で言語情報などの上位の知識を導入した文字のセグメンテーションの考え方を、6.3で言語情報を利用した文字のセグメンテーションと認識手法を、6.4で評価実験の結果を、6.5で考察を述べる。

## 6.2 知識を導入した文字のセグメンテーションの考え方

### 6.2.1 知識の役割

自由書式（不定ピッチ）で筆記された文字列から個々の文字を精度良く切り出すことは、自由書式文字列を認識する際に特有の重要でかつ困難な課題である。しかし、人間は、かなり自由に筆記された文字列からでも文字をセグメンテーションすることができる。それは、人間がさまざまな知識を有機的に利用しているためである。人間が利用する知識は、文字毎の図形的なまとまりや言語的な文脈情報など、下位の知識から上位の知識まで多様である。ここでは、図6.2に示すようにセグメンテーションに有効な知識を3段階に分類し、その役割を説明する。

	枠無し筆記文字列	セグメンテーションに必要な知識								
①	日本 <small>の</small>	図形のまとまり								
②	<table style="border-collapse: collapse; margin: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">高</td> <td style="border: 1px solid black; padding: 2px;">雀</td> <td style="border: 1px solid black; padding: 2px;">散</td> <td style="border: 1px solid black; padding: 2px;">し</td> </tr> <tr> <td style="text-align: center;">1文字</td> <td style="text-align: center;">1文字</td> <td></td> <td></td> </tr> </table>	高	雀	散	し	1文字	1文字			文字形状（文字認識）
高	雀	散	し							
1文字	1文字									
③	<table style="border-collapse: collapse; margin: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">詳</td> <td style="border: 1px solid black; padding: 2px;">系</td> <td style="border: 1px solid black; padding: 2px;">田</td> <td style="border: 1px solid black; padding: 2px;">な</td> </tr> <tr> <td colspan="4" style="text-align: center;">形容動詞</td> </tr> </table>	詳	系	田	な	形容動詞				言語情報（文節処理）
詳	系	田	な							
形容動詞										

図6.2 枠無し筆記文字列からの文字のセグメンテーション

①この例では、各文字が左右に分離しない文字から構成されている。そのため、図形のまとまりを利用することにより文字のセグメンテーションは可能である。

②この例では、漢字が偏とつくりで分離している。そのため、図形的なまとまりを考慮するだけでは文字のセグメンテーションは困難である。しかし、偏だけあるいはつくりだけに着目すればそれは文字として意味を持たない。そこで、文字として意味を持つ箇所を区切ること、つまり文字認識の導入により文字のセグメンテーションは

可能である。

③この例では、漢字が偏とつくりに分離し且つその分離図形は他の文字となる。このような例では、文字認識を導入しただけでは、「詳」「細」「に」以外に、「言」「羊」「糸」「田」「に」と認識される危険がある。しかし、単語として見れば後者のような読み方は存在しない。つまり、言語処理の導入により文字のセグメンテーションが可能である。

このように、精度の良い文字のセグメンテーションを実現するためには、下位から上位までの各段階の知識を有効に活用する必要がある。

### 6. 2. 2 文字認識を利用した文字のセグメンテーションの限界

左右に分離した文字や、文字間が接触する文字をもセグメンテーションすることができるように、従来、文字認識をセグメンテーションに利用する手法が提案されている。この手法は6.2.1 ②で述べたように、文字認識に関する知識を導入した手法であるが、6.2.1 ③で述べたような限界を持つ。ここでは1例として、各文字のセグメンテーションと認識を同時に実現する候補文字ラティス法（5章参照）を取り上げ、文字認識を利用したセグメンテーション法の限界を実験的に明らかにする。特に、1文字が複数の文字に分離して認識されるような切り出し不能文字（6.2.1 ③参照）の字種について分析し、更に上位の知識（言語や単語）を導入した場合の改善効果を明らかにする。

分析は、個々の文字を文字列として認識し、それがどのようにセグメンテーション誤りをするかを調査することにより行なった。候補文字ラティス法は本章で提案する手法に包含されている。該当箇所は6. 3. 1であり、手法の簡単な説明はそこで行なう。

### 6. 2. 3 使用データ

対象字種は、常用漢字（1945字種）、平仮名（46字種）と画数変動文字（157字種；2画で筆記された「子」など）の合計2148字種である。これらの字種を40名の筆者に筆記させ、20名分（計42960文字）を候補文字ラティス法の標準辞書作成用に、残り20名分（未知文字）を分析用に使用した。

### 6. 2. 4 セグメンテーション不能文字の分析

1文字が複数の文字に分離することによりセグメンテーションを誤る可能性のある字種を切り出し不能文字とする。分析の結果、セグメンテーション不能文字は2148字種中、207字種（9.7%）存在することを確認した。

セグメンテーション不能文字を、発生する原因により、以下の2種類に分類する。

（1）第1種セグメンテーション不能文字：偏とつくりに分離し、それぞれが文字となることが原因で文字のセグメンテーション誤りが発生する文字。第1種セグメンテーション不能文字は122字種（5.7%）存在した。その例を図6.3（a）に示す。

（2）第2種セグメンテーション不能文字：上記以外の理由で文字のセグメンテーション誤りの発生する文字。例えば、手書き変形により文字が2つの図形に分離し、それぞれを文字として誤認識することで、セグメンテーション誤りが発生する文字などである。但し、1文字が多少重なった複数の図形から構成された場合でも、その図形が複数の文字と解釈されれば、その文字も第2種として扱う（例えば、「も」→「こ」+「し」）。その理由は、本章で述べる文字のセグメンテーション法では、文字間が重なるような品質の悪い文字列をも処理対象とするためである。第2種セグメンテーション不能文字は85字種（4.0%）存在する。その例を図6.3（b）に示す。

全字種中9.7%存在するセグメンテーション不能文字は、文字認識の情報だけを利用している従来手法では、本質的に正しくセグメンテーションされず、これが文字認識を利用した切り出し法の限界となる。

つまり、これらについては言語情報などのより高度な知識を利用することで、改善効果が期待できる。

更に参考までに、文字認識の誤り易い例を図6.3（c）に示すが、これらについても言語処理を導入することにより、改善が期待される。

 136 405A ( 4, 37) 切 ↓ 七+刀	 202 3879 ( 5, 23) 功 ↓ 工+力	 303 3558 ( 6, 22) 吸 ↓ 口+及	 340 4721 ( 6, 59) 如 ↓ 女+口
 628 3068 ( 8, 79) 取 ↓ 耳+又	 847 416A ( 9, 108) 相 ↓ 木+目	 1069 473C (10, 141) 納 ↓ 糸+内	 1273 4630 (11, 134) 動 ↓ 重+力
 1400 386C (12, 68) 詞 ↓ 言+司	 1820 3074 (15, 38) 諸 ↓ 言+者	 1983 4E70 (17, 31) 齡 ↓ 齒+令	 2009 485D (18, 26) 翻 ↓ 番+羽

図6.3 (a) 第1種セグメンテーション不能文字の例




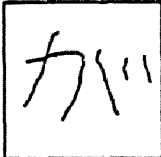




 81 242B ( 3, 35) か ↓ 力+し	 89 2462 ( 3, 43) も ↓ こ+し	 121 3926 ( 4, 22) 孔 ↓ 子+し	 204 392A ( 5, 25) 巧 ↓ 工+ち
 276 242C ( 5, 97) か ↓ 力+じ	 508 4830 ( 7, 109) 判 ↓ 半+り	 535 4E64 ( 7, 136) 冷 ↓ こ+令	 679 4748 ( 8, 130) 波 ↓ 三+皮
 808 3C6D ( 9, 69) 狩 ↓ 千+守	 1236 405F (11, 97) 設 ↓ 言+支	 1260 4422 (11, 121) 帳 ↓ け+長	 1333 302E (12, 1) 握 ↓ 工+屋

図6.3 (b) 第2種セグメンテーション不能文字の例

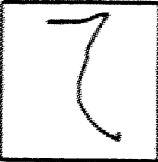


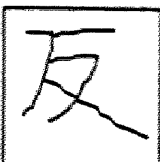




 6 2446 ( 1, 6) て ↓ 乙	 68 403C ( 3, 22) 夕 ↓ 勺	 112 376E ( 4, 13) 日 ↓ 月	 146 483F ( 4, 47) 反 ↓ 友
 224 3F3D ( 5, 45) 申 ↓ 甲	 562 322I ( 8, 13) 押 ↓ 抽	 1993 3A3F (18, 10) 鎖 ↓ 鎮	 71 4267 ( 3, 25) 大 ↓ 丈

図6.3(c) 文字認識誤りの例

### 6.3 言語情報を利用した文字のセグメンテーションと認識

本章では、言語情報を効果的に文字切り出しに活用する方法について述べる。そのため、文字のセグメンテーション、文字認識、言語処理の三階層から構成される認識系を採用する。文字のセグメンテーション層と、文字認識層との結合には候補文字ラティスを、文字認識層と言語処理層との結合には単語ラティスを利用する。ラティスとはセグメンテーションと認識のあいまい性を2端子グラフで表現したものである。処理のブロック図を図6.4に示す。

処理の手順は、①まず入力文字列を、候補文字ラティスで記述する。②そのラティスから適切な文字のセグメンテーション位置と、文字認識結果に対応した文字系列を探索し単語ラティスを作成する。③それを言語的に検定する。④言語的に許容されるまで②と③の処理を反復的に繰り返し、最適な文字系列を得る。

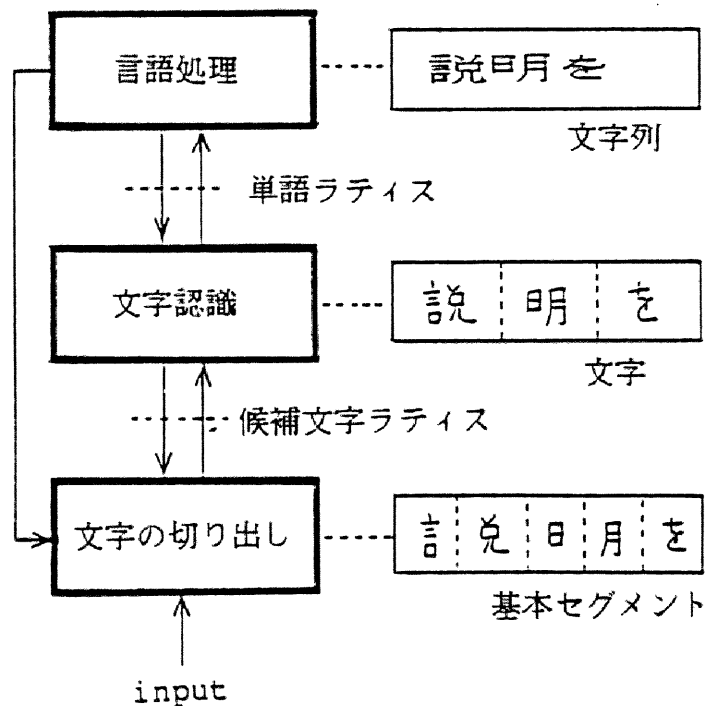


図6.4 処理のブロック図

### 6.3.1 候補文字ラティス

候補文字ラティスの作成手順を、以下に簡単に説明する。

#### (1) 基本セグメントへの分割

文字列を構成している入力ストローク列を、基本セグメントに分割する。基本セグメントは偏やつくりあるいは更に微小な孤立図形などに対応する部分図形である。これは、各ストロークをX軸に射影し、その重なりがあるいき値以下の箇所文字列を分割することにより得られる。実験ではいき値として、文字列の高さに対し15%の重なりまで許容する値に設定した。この値は、罫線上に自由ピッチで筆記される文字列では、文字間の重なりは最悪でも15%である(5.3.1参照)という分析結果に基づく。

#### (2) 候補文字の生成

基本セグメントを組み合わせて図形を生成し、その外郭矩形の大きさが一定値以下(文字列の高さに対し図形の横幅が200%以下とする。この値は実験的に定めた(5.4.5参照)となる図形を候補文字とする。生成された候補文字の例を図6.5に示す。

#### (3) 候補文字の認識

各候補文字を認識し、認識結果の上位N位とその相異度を求める。文字認識は、ストロークのDP(動的計画法)マッチングにより求めた相異度を尺度とする画数固定、筆順自由のオンライン認識手法を採用した。Nの値としては3を設定した。これは、上位3位までに正しい認識結果がほとんど含まれるという予備実験の結果より決定した。参考として、2.2.1で示した文字データに対する累積分類率(上位n位までに正解が入る割合)を表6.1に示す。

Input character string	説明を行う								
Basic segment	e1	e2	e3	e4	e5	e6	e7	e8	e9
	言	説	日	/	月	を	レ	,	行
	言	説	日	/	月	を	レ	,	行
	説		日	月を		レ			
	説		月	を	レ	行			
	説明			月を		レ	行		
	説明			月を		レ			
	説明			月を		レ			
	説明			月を		レ			

図 6. 5 候補文字の生成

表 6. 1 累積分類率

N位	1	2	3	4
分類率	99.38%	99.89	99.96	99.97

#### (4) 候補文字ラティスによる記述

候補文字ラティスは、入力文字列中の各基本セグメントがどの候補文字の一部であるかの情報と、各候補文字の上位N位の認識結果と、それらを認識した時の標準文字との相異度を表現したものである。これは、2端子のグラフで記述することができ、各ブランチに認識結果とその相異度が付属している。図6.5に対する候補文字ラティスのグラフ表現を図6.6に示す。

候補文字ラティスから評価関数を最小とする経路を探索し、その経路に対応する文字系列を第1次認識候補とする。但し、評価関数Fは、

$$F = \sum_{\text{候補文字}} \left( \begin{array}{c} \text{候補文字の} \\ \text{相異度} \end{array} \right) \times \left( \begin{array}{c} \text{候補文字を構成する} \\ \text{基本セグメント数} \end{array} \right) \quad (6.1)$$

と定義する。つまり、相異度とセグメント数を掛けた値をブランチの重みとした重み付きグラフの最短経路を探索することにより実現できる。最短経路の探索には動的計画法を採用した。第1次認識候補は文献3の候補文字ラティス法による認識結果と同等なものとなる。図6.6の例では、第1次認識候補は「説」「日」「月」「を」「行」「う」となる。

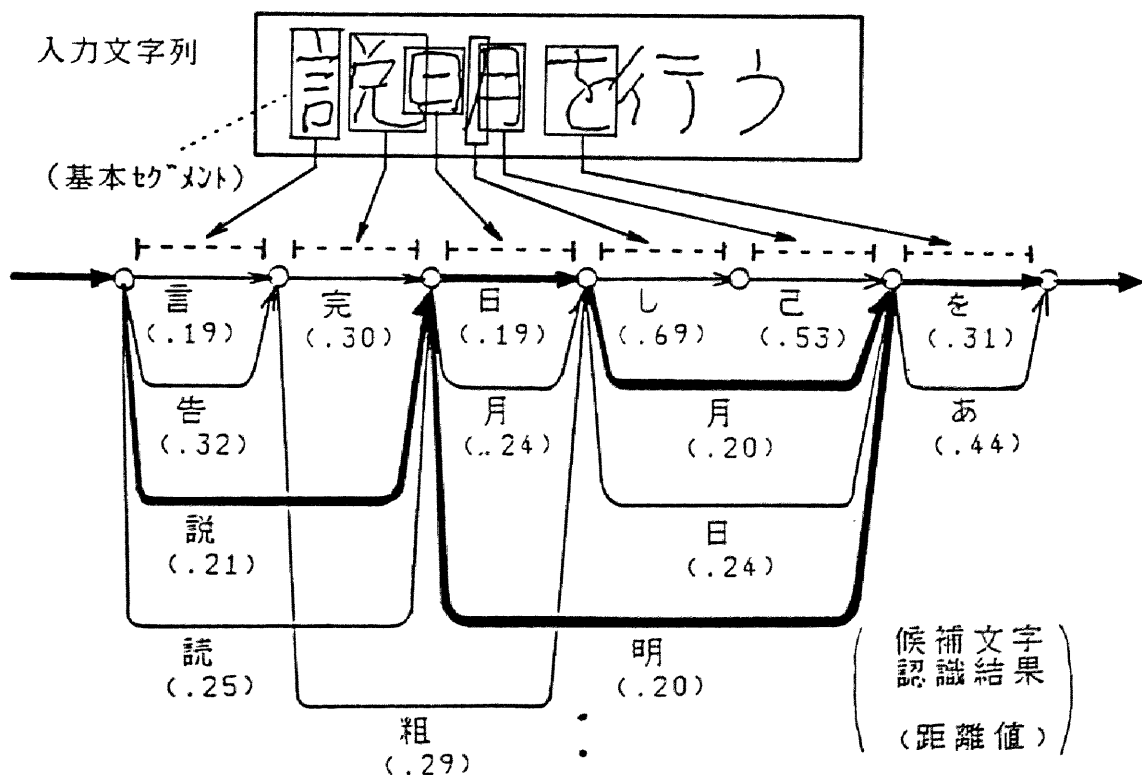


図6.6 候補文字ラティス

### 6.3.2 言語処理

第1次認識候補が言語的に矛盾するか否かを本節で述べる言語処理で検定する。この検定は比較的ゆるく行なう。もし、言語的に矛盾するならば候補文字ラティスから他の文字列を選択する処理に移行する。矛盾しなければ、その結果を最終認識結果とする。言語処理には、以下に示すような単純な方法を採用した。

#### (1) 単語照合

ここでは、字種が平仮名から非平仮名に変化する点で範囲を制限し、これを照合範囲とし、処理の効率化を図る。照合範囲内の文字列と単語辞書との間で照合を行ない、照合の取れた単語とその品詞名は全て単語ラティスに登録する。単語辞書は、学研国語大辞典[110]を基に作成したもので、約8万語からなる。「説日月を」、「行う」、「説明を」に対する単語ラティスの例を図6.7(a)~(c)に示す。

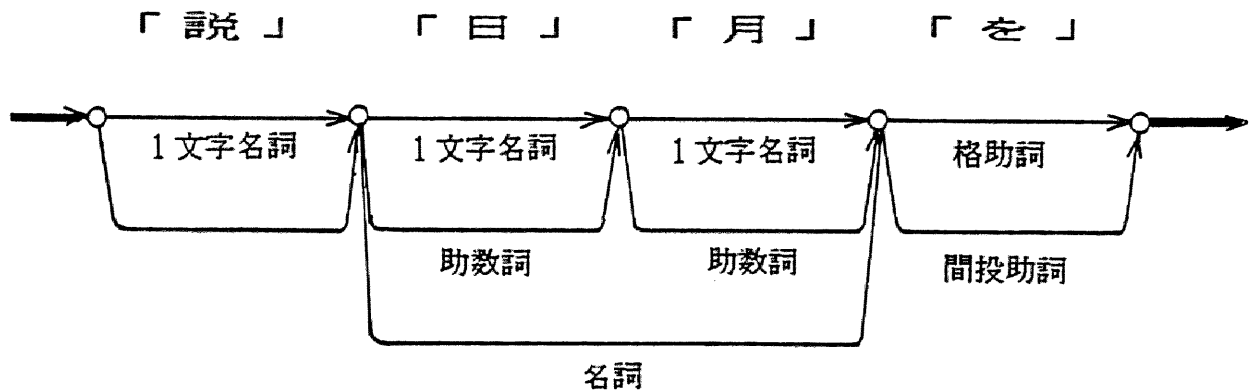


図6.7(a) 単語ラティス「説日月を」

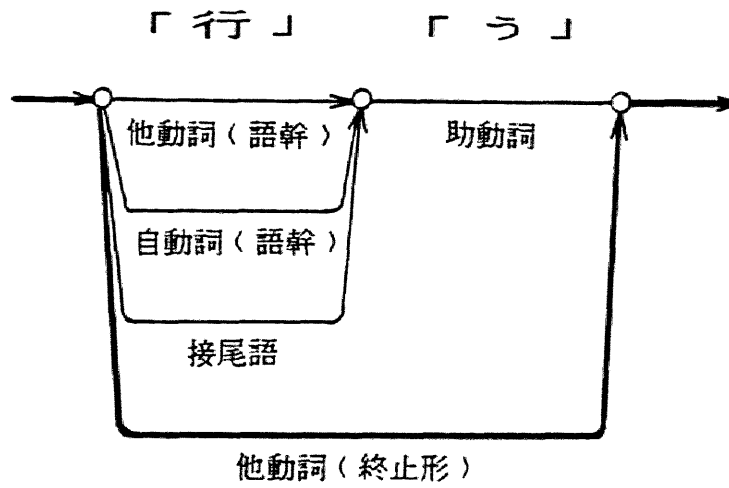


図6.7(b) 単語ラティス「行なう」

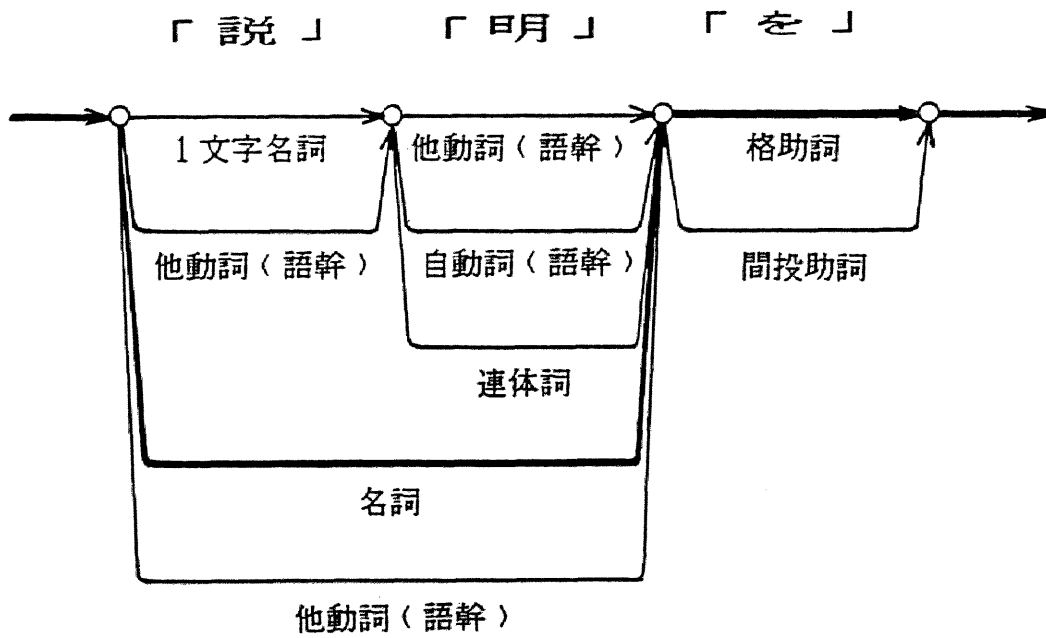


図6.7(c) 単語ラティス「説明を」



## (2) 単語ラティスの探索による言語的な検定

単語ラティスは候補文字ラティスと同様に2端子グラフで表現される。ここでは、単語間の接続検定を行ないながらグラフを探索する。接続規則として約130項目(文献105を基に作成)が登録されている。接続規則の1例を以下に示す。

・名詞 + 接尾語                    → 名詞                    (6.2)

・名詞 + 格助詞 {と}               → 文節                    (6.3)

等

(付録Dに接続規則の詳細を示す。)

この接続規則の例では、名詞は接尾語に接続し、接続した文字列は名詞として扱うことを示す。また、名詞は格助詞の「と」と接続し、これは文節として扱う。

探索の結果、対象文字列が文節等になる場合には、この文字列は言語的に許容されるとき、他の照合範囲に処理を進める。ここで文節等とは、文節、名詞、文節が連結した連文節のいずれかとする。また、文字列が文節等にならない場合にはこの範囲に対応する候補文字ラティスを再探索する。図6.7(c)の例(「説明を」)は言語的に許容されるが、図6.6の例に対する最短経路では、「説日月を」(図6.7(a)参照)の箇所が許容されない。

### 6.3.3 反復的な候補文字ラティスの探索

言語検定処理により、文字列中で言語的に矛盾している箇所を検出することができる。そこで、この箇所に対して言語的に正しい他の解釈がないかを調べるため、候補文字ラティスの再探索を行なう。言語処理における照合範囲に対応する候補文字ラティスは、文字列全体に対応する候補文字ラティスの部分グラフであり、やはり2端子グラフとなる。

再探索では、文字認識の候補とセグメンテーション箇所の候補の両者を考慮して、文字列全体としての次候補を選出する。ここでは、候補文字ラティス上で、ブランチの重みの和が2番目に最小となる経路に対応する文字列を、文字列の次候補とする。ラティスにおいて、ブランチの重みの和がK番目に最小となる経路を第K次の最短経路と呼ぶことにする。第K次最短経路の探索手法の詳細を6.3.4に述べる。

この文字列に対して再び6.3.2で示した言語処理を適用し、言語的に矛盾がなくなるまで処理を繰り返す。無限ループ状態になることを抑えるため、反復回数が1

0より大となる文字列では、認識を棄却する。

#### 6.3.4 動的計画法による第K次最短経路の探索

第K次最短経路は、K番目に最適な政策を取ったときの最短経路のことをいう。第K次最短経路の探索は、ネットワーク理論における最短経路の探索問題と類似の問題であり、動的計画法で効率良く解くことができる[111]。この探索は本章で提案する手法の要となるので、探索法の詳細を以下に述べる。

記号の説明：

①候補文字ラティスのノードの数をM、ノード名を1、2、...、Mとし、始点から終点に向かってこの順に並んでいるとする(図6.8参照)。

②候補文字を認識した際の第1位から第N位までの認識結果を候補として利用する場合を考える。この場合には、ノードiとノードjを接続するブランチはN本となる。N本中の第n番目のブランチの重み(6.3.1(4)の評価関数Fを参照)を $T(i, j, n)$ とする。もし、ノードiとノードjを接続するブランチがなければ、 $T(i, j, n)$ は無量大の値とする。

③ノードiからノードMに至る第K次最短経路の重みを $U(K; i)$ とする。特別な場合として、 $U(1; 1)$ は候補文字ラティスの最短経路の重みとなる。

④ノードiからノードMに至る第K次最短経路が、ノードiの次にノードtを通る場合を考える。但しノードiとノードtを結ぶブランチは②で述べたようにN本存在するが、この第K次最短経路はこれらN本のブランチの中で第n番目を通過するとする。更に、この経路はノードtからノードMに至る第K'次最短経路を通過するとする。その際に、このiとKに対するノード名tを $L(K; i)$ で表わし、nの値を $P(K; i)$ で表わし、K'の値を $L'(K; i)$ で表わす。

⑤ $\text{MIN}_{1 \leq j \leq M}(K; X(j))$ を、 $X(1), X(2), \dots, X(M)$ の中

でK番目に小さい $X(j)$ の値を抽出する関数とする。

処理は、動的計画法により、各段階での最適解を再帰的に決定することにより実現できる。以下に、具体的な処理ステップを述べる。

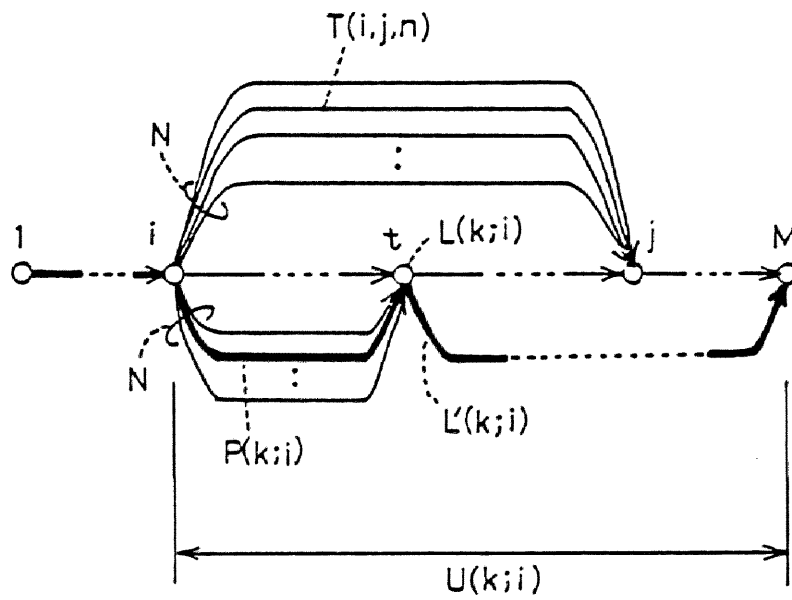


図 6. 8 第 K 次最短経路の探索

・ステップ 1

初期設定として次の代入を実行する。

$$i = M - 1 \quad (6.4)$$

$$\left\{ \begin{array}{l} U(1;M) = 0 \end{array} \right. \quad (6.5)$$

$$\left\{ \begin{array}{l} U(2;M) = \text{無限大} \end{array} \right. \quad (6.6)$$

$$\left\{ \begin{array}{l} \dots \\ U(K;M) = \text{無限大} \end{array} \right. \quad (6.7)$$

・ステップ 2

関数  $U(k;i)$   $\{k=1, \dots, K\}$  を以下の式で求める。

$$U(k;i) = \text{MIN}(k; T(i, j, n) + U(k'; j)) \quad (6.8)$$

$$\left( \begin{array}{l} i < j \leq M \\ 1 \leq n \leq N \\ 1 \leq k' \leq k \end{array} \right)$$

ここで、右辺の  $\text{MIN}(k; \dots)$  を決定する際に使用された  $j, n, k'$  の値を  $j', n', k''$  とし以下の代入を実行する。

$$\left\{ \begin{array}{l} L(k;i) = j' \end{array} \right. \quad (6.9)$$

$$\left\{ \begin{array}{l} L'(k;i) = k'' \end{array} \right. \quad (6.10)$$

$$\left\{ \begin{array}{l} P(k;i) = n' \end{array} \right. \quad (6.11)$$

つまり、 $U(k;i)$ 、 $L(k;i)$ 、 $L'(k;i)$ 、 $P(k;i)$  は、 $j$  を  $(i+1)$  から  $M$  まで変えた際の  $U(k'; j)$  の値を再帰的に利用することにより求められる。

・ステップ 3

$i$  の値を 1 ずつ減少させステップ 2 を繰り返す。 $i$  が 1 となった時点で処理を停止させる。これにより、 $U(k;i)$ 、 $L(k;i)$ 、 $L'(k;i)$ 、 $P(k;i)$  ( $1 \leq k \leq K$ ;  $1 \leq i \leq M-1$ ) が求められる。

・ステップ 4

$L(k; i)$ 、 $L'(k; i)$ 、 $P(k; i)$  のポインターをノード 1 ( $i = 1$ ,  $k = K$ ) からノード  $M$  までたぐる。これにより得られる経路が第  $K$  次最短経路となる。第  $K$  次認識候補は第  $K$  次最短経路に対応した候補文字認識結果を抽出することにより得られる。

具体的な例を以下に示す。下記の表は、 $i$ 、 $k$  に対する ( $L(k; i)$ ,  $L'(k; i)$ , 対応した文字名) の値を表現したものである。例えば、 $i = 1$ ,  $k = 1$  の箇所に記された (3, 1, 説) は、 $L(1; 1)$  が 3、 $L'(1; 1)$  が 1、そのブランチに対応した文字名が「説」であることを意味する。

$k \backslash i$	1	2	3	4	5
1	(3, 1, 説)	(3, 2, 説)	(2, 1, 言)	(3, 1, 読)	(3, 3, 説)
2	(3, 1, 完)	(3, 2, 完)	(3, 1, 克)	(3, 2, 克)	(3, 1, 弟)
3	(4, 1, 日)	(6, 1, 明)	(4, 2, 日)	(4, 1, 月)	(4, 3, 日)
4	(6, 1, 月)	(6, 1, 日)	(6, 1, 丹)	(6, 2, 月)	(6, 3, 月)
5	(7, 1, 壮)	(7, 1, 吐)	(6, 1, 口)	(7, 1, 江)	(6, 1, 士)
6	(7, 1, を)	(7, 1, あ)	(7, 1, せ)	(7, 2, を)	(7, 2, せ)

例えば、第 2 次最短経路を求めるには、まず上記の表の (1, 2) {1 行 2 列目、つまり、 $i = 1$ ,  $k = 2$ } に着目する。最初の文字が「説」で次に参照される箇所が (3, 2) であることが読み取れる。表の (3, 2) に着目すれば次の文字は「明」である。以下、表を順にたどることにより第 2 次最短経路に対応する認識結果が得られる。図 6. 6 のラティスに対する第 1 次から第 5 次までの最短経路に対応する文字列を以下に示す。この例では、第 2 次最短経路に対応する文字列で言語的に許容される。この単語ラティスを図 6. 7 (c) に示す。

第1次最短経路に対応する文字列：「説」「日」「月」「を」

第2次最短経路に対応する文字列：「説」「明」「を」

第3次最短経路に対応する文字列：「言」「完」「日」「月」「を」

第4次最短経路に対応する文字列：「読」「日」「月」「を」

第5次最短経路に対応する文字列：「説」「日」「日」「を」

この例では、第2次認識候補に言語的に許容される文字列が出現しているため、この候補を最終認識結果として出力する。

## 6.4 評価実験

### 6.4.1 使用データ

実験には次に示す2種類の文字列データを使用した。また候補文字の認識用の標準辞書には2.2.1で述べた辞書作成用データを使用した。

#### (1) データ1

手書きで文字ピッチが変動した文字列に対する本手法の効果を調査するため、以下の条件で文字列を収集した。

①横罫線（1cm間隔）の用紙に、文字列を自由なピッチで筆記させた。

②各文字の筆順は自由であるが画数は守らせた。

③文字列として5文例（但し、言語処理部で未登録な単語は含まない。）を用意し、それを各3回ずつ筆記させた。

データ収集数は7名、5文例、各3回の105文例であり、総文字数は1050文字である。データ1の例を図6.9(a)に示す。

詳細に検討を続ける

高騒音下での通話対策

安定でしかも量産に向く

利用するための手引

図6.9(a) データ1の例

物品の提示など使用者の利便を

装置構成が簡易である点で

起因する誤動作の防止に必要な

低速走査法は走査度が

図6.9(b) データ2の例

## (2) データ2

本手法を多数の文例に対して検討するため、500文例（未登録単語を含まない10文字程度から構成される文字列）を技術文書の中から適当に選択し、その文例から手書き文字列を合成して実験に用いた。その方法は、あらかじめ収集した常用漢字と平仮名の20名分2148字種を、文例に従って横方向に配置した（20名\*異なる25文例）。但し、この文字データは候補文字認識の辞書作成に使用した文字データとは異なるもの、つまり未知文字を使用した。その際、文字間隔は、手書きの平均的な値を調査し、その値（文字列の高さに比較して15%の値）に設定した。このデータは合計500文例、5709文字から構成される。データ2の例を図6.9(b)に示す。

### 6.4.2 認識実験

本手法により上記のデータ1、データ2を認識した。本手法では、個々の文字の切り出しと文字の認識が同時に終了する。文字列からの文字のセグメンテーション率（文字切り出し率）、セグメンテーションされた文字の認識率（文字認識率）、文字列としての文字の認識率（文字列認識率）を求めた。結果を表6.2に示す。但し、文字切り出し率、文字認識率はそれぞれ、

$$\text{文字切り出し率} = \frac{\text{正しく切り出された文字数}}{\text{入力文字数}} \times 100(\%) \quad (6.12)$$

$$\text{文字認識率} = \frac{\text{正しく認識された文字数}}{\text{正しく切り出された文字数}} \times 100(\%) \quad (6.13)$$

$$\text{文字列認識率} = (\text{文字切り出し率}) \times (\text{文字認識率}) \quad (6.14)$$

により与えられる。



表6.2 実験結果

項目	データ名	言語処理	
		無し	有り
文字 切り出し率	データ1	96.7%	98.8
	データ2	97.1	99.3
	全体	97.0	99.2
文字 認識率	データ1	99.2	99.9
	データ2	99.1	99.8
	全体	99.1	99.8
文字列 認識率	データ1	95.9	98.7
	データ2	96.2	99.1
	全体	96.1	99.0

この結果から分かるように、言語情報を導入しない候補文字ラティス法に比較し、本手法により全体として、切り出し率は97.0% から99.2% に向上し、言語処理が文字のセグメンテーションに有効であることを確認した。また、本手法は文字認識の後処理的な効果もあり、文字認識率は99.1% から99.8% に向上した。その結果文字列全体の認識率としては96.1% から99.0% に向上した。実際の手書き変動を含んだデータ1と多種類の文例を集めたデータ2ではほぼ同等の結果が得られている。

言語処理の導入により正しく文字のセグメンテーションができた例を図6.10(a)に、正しく文字認識できた例を図6.10(b)に、正しく文字のセグメンテーションができなかった例を図6.10(c)に示す。セグメンテーション誤りの原因は、今回の実験では言語検定を比較的緩く設定したため、図6.10(c)のような文字列も言語的に許容されたことによる。

本手法では、言語処理部を単純化するため比較的緩い言語仕様とした。そのため、上記の例のように「動作」を「重力作」(名詞+接尾語)と解釈されたりする。このような例は比較的少ないため、ここでは特に対策を設けなかった。また、この実験で使用した文字のデータには未登録語は含まれていないが、実際のデータには未登録語が含まれている可能性がある。未登録語に対しては第1次認識候補で正解が出たとしても、候補文字ラティスの再探索により誤った認識結果に変化させてしまう危険性が

ある。未登録語の対策についても今回の実験では設けていない。このような言語処理部の強化は残された課題である。

検討を糸売ける

- 正解： 「検」「討」「を」「続」「け」「る」  
 第1次認識候補： 「検」「討」「を」「糸」「売」「け」「る」  
 第2次認識候補： 「検」「討」「を」「糸」「売」「け」「そ」  
 第3次認識候補： 「検」「討」「を」「続」「け」「る」 ---> 言語的に許容

設計仕様に

- 正解： 「設」「計」「仕」「様」「に」  
 第1次認識候補： 「言」「支」「言」「十」「仕」「様」「に」  
 第2次認識候補： 「設」「言」「十」「仕」「様」「に」  
 第3次認識候補： 「言」「支」「計」「仕」「様」「に」  
 第4次認識候補： 「言」「支」「言」「す」「仕」「様」「に」  
 第5次認識候補： 「設」「計」「仕」「様」「に」 -----> 言語的に許容

場合にはほぼ等しい

- 正解： 「場」「合」「に」「ほ」「ぼ」「等」「し」「い」  
 第1次認識候補： 「場」「合」「し」「こ」「ほ」「ぼ」「等」「し」「い」  
 第2次認識候補： 「場」「合」「く」「こ」「ほ」「ぼ」「等」「し」「い」  
 第3次認識候補： 「場」「合」「し」「二」「ほ」「ぼ」「等」「し」「い」  
 第4次認識候補： 「場」「合」「く」「二」「ほ」「ぼ」「等」「し」「い」  
 第5次認識候補： 「場」「合」「に」「ほ」「ぼ」「等」「し」「い」  
 -----> 言語的に許容

図6. 10 (a) 言語処理により正しくセグメンテーションされた例

送話者の

正解： 「送」「話」「者」「の」

第1次認識候補： 「迷」「話」「者」「の」

第2次認識候補： 「迷」 「言」 「舌」「者」「の」

第3次認識候補： 「送」「話」「者」「の」 -----> 言語的に許容

対象に応じて

正解： 「対」「象」「に」「応」「じ」「て」

第1次認識候補： 「対」「象」「に」「応」「じ」「乙」

第2次認識候補： 「対」「象」「に」「応」「じ」「て」 --> 言語的に許容

図6.10(b) 言語処理により正しく文字認識できた例

しかも量産に向く

- 正解： 「し」「か」「も」「量」「産」「に」「向」「く」
- 第1次認識候補： 「し」「力」「ほ」「量」「産」「し」「拘」「く」
- 第2次認識候補： 「し」「刀」「ほ」「量」「産」「し」「拘」「ん」
- 第3次認識候補： 「し」「忙」「量」「産」「し」「拘」「し」
- 第4次認識候補： 「し」「か」「も」「量」「産」「し」「こ」「向」「く」
- 第5次認識候補： 「し」「か」「も」「量」「産」「し」「二」「向」「く」
- 第6次認識候補： 「し」「か」「も」「量」「産」「し」「こ」「向」「ん」
- 第7次認識候補： 「し」「か」「も」「量」「産」「し」「二」「向」「ん」
- 第8次認識候補： 「し」「か」「も」「量」「産」「し」「拍」「く」
- 第9次認識候補： 「し」「か」「も」「量」「産」「し」「こ」「向」「し」
- 第10次認識候補： 「し」「か」「も」「量」「産」「し」「招」「く」

-----> 言語的に許容

重力作時の

- 正解： 「動」「作」「時」「の」
- 第1次認識候補： 「重」「力」「作」「時」「の」-----> 言語的に許容

図6. 10 (c) 言語処理により正しくセグメンテーションされなかった例

### 6. 4. 3 言語処理適用の反復回数

本手法は、言語処理を反復的に適用することにより、候補文字ラティスの多数の経路の中から最適な文字のセグメンテーション位置と文字認識結果に対応した経路を選出する手法である。データ1とデータ2の全605文例に対して、実際に適用された言語処理の反復回数を調査した。結果を図6. 11に示す。これより、1~5回の反復で大半の文字列は正しく修正されることがわかる。また誤認識の内容を分析すると、11回以上の反復を行なった場合には、棄却されるべき文字列が誤認識される場合が多くなる。そのため、オンライン文字列データの場合には6~10回程度で反復回数を制限することが適当である。

計算時間は図6. 6の人力文字列の場合で約100秒であった。その内訳は、候補文字ラティスの生成に約90秒、言語処理の1回の反復に約5秒かかり、この例では2回の反復がなされた。但し、実験は1MIPSのミニコン(FORTRAN 使用)で行なった。

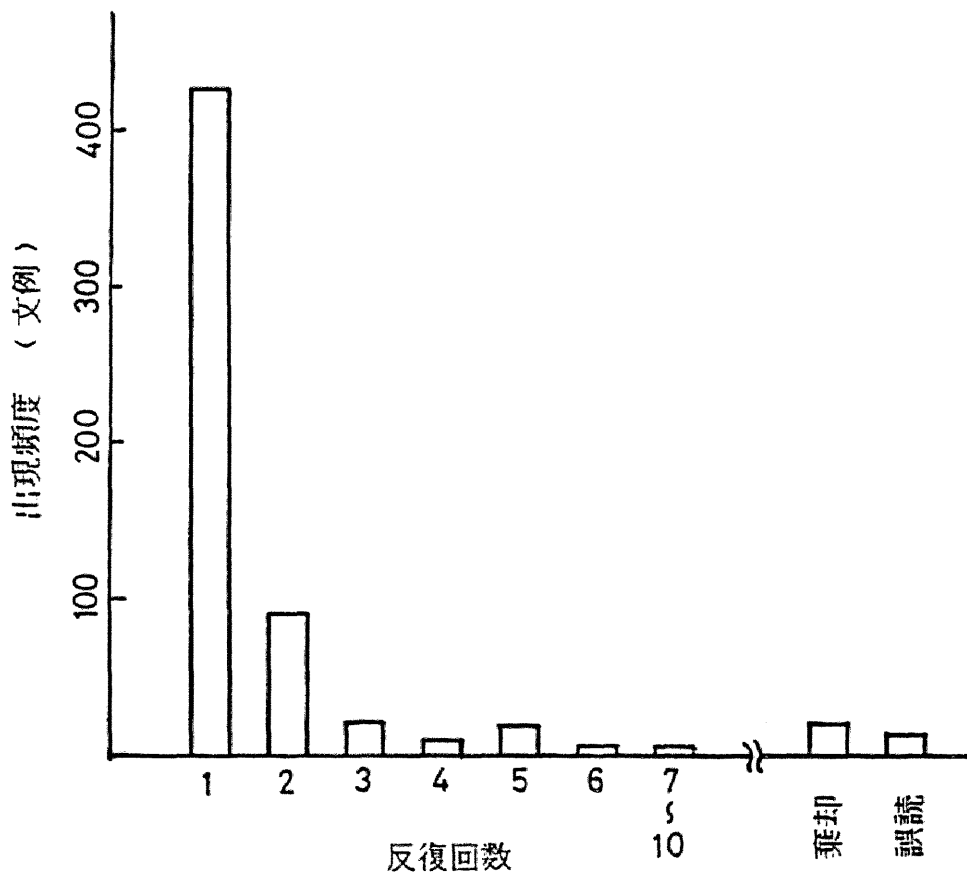


図6. 11 言語処理適用の反復回数

## 6.5 考察

### 6.5.1 分離有意文字処理との比較

言語情報を利用せずに分離文字を正しくセグメンテーションする手法として、5章で述べた分離有意文字処理がある。これは、あらかじめ分離しやすい文字をテーブルに登録しておき、この文字あるいは文字列が認識結果に出現した際には、文字の外接矩形の縦横比を利用して、それが1に近い方を選択する手法である。文字のセグメンテーション率の比較を表6.3に示す。分離有意文字処理の切り出し率が本手法に比較して低い原因は、この処理が第2種セグメンテーション不能文字に対しては適用できないこと、文字の外接矩形が正方形に近いかどうかの情報だけでは十分なセグメンテーションが不可能であることがあげられる。

表6.3 分離有意文字処理との比較

分離有意文字処理	97.3 %
本手法	98.8 %

### 6.5.2 縦書きへの応用

ここまでの実験では横書きの文字列に対して実験を行ってきたが、本手法は縦書きの文字列に対してもそのまま適用できる。実験によりそれを確認した。対象データは、データ2と同様の手法で縦方向に文字を配置したものであり、これをデータ3とする。データ3の例を図6.12に示す。データ3は500文例、5709文字から構成される。それに対する結果を表6.4に示す。言語処理がない場合の横書き文字列のセグメンテーション率は、縦書き文字列のそれに比較して低い値となっていることが分かる。その理由は、漢字には上下に分離するもの（「岩」→「山」+「石」）より左右に分離するもの（「明」→「日」+「月」）のほうが多い為である。文字の縦方向におけるセグメンテーション不能文字の存在を6.2.4と同様の処理により求めた結果、横方向のセグメンテーション不能文字の存在確率が9.7%に対し縦方向のそれは7.6%（第1種2.8%、第2種4.8%）であった。

平均印字速度を保証するために  
 装置構成が簡易である点で  
 信号電流を等価的に  
 従来実験的に得られていた

図6. 12 縦書き文字列（データ3）の例

表6. 4 縦書き文字列の場合の処理結果（データ3使用）

項目	言語処理	
	無し	有り
文字切り出し率	98.5%	99.4
文字認識率	99.1	99.8
文字列認識率	97.6	99.2

### 6.5.3 文字列の構造の階層的記述

文字列の構造は図6.13に示すように階層的に記述することが可能である。ここで、上位の階層と下位のそれとの関係はラティスで効率良く表現することが可能である。本手法では単語ラティスと候補文字ラティスの組み合わせにより、文字列を上位から下位にかけて表現することが可能となった。また、上位の知識を下位の情報の認識に利用するためには、ラティスの探索を行なうことにより実現される。更に、本手法のようにラティス間のインタラクションを考慮することにより、文節情報を文字の切り出しに利用するなど多段階の階層に渡って知識の相互利用が可能となる。

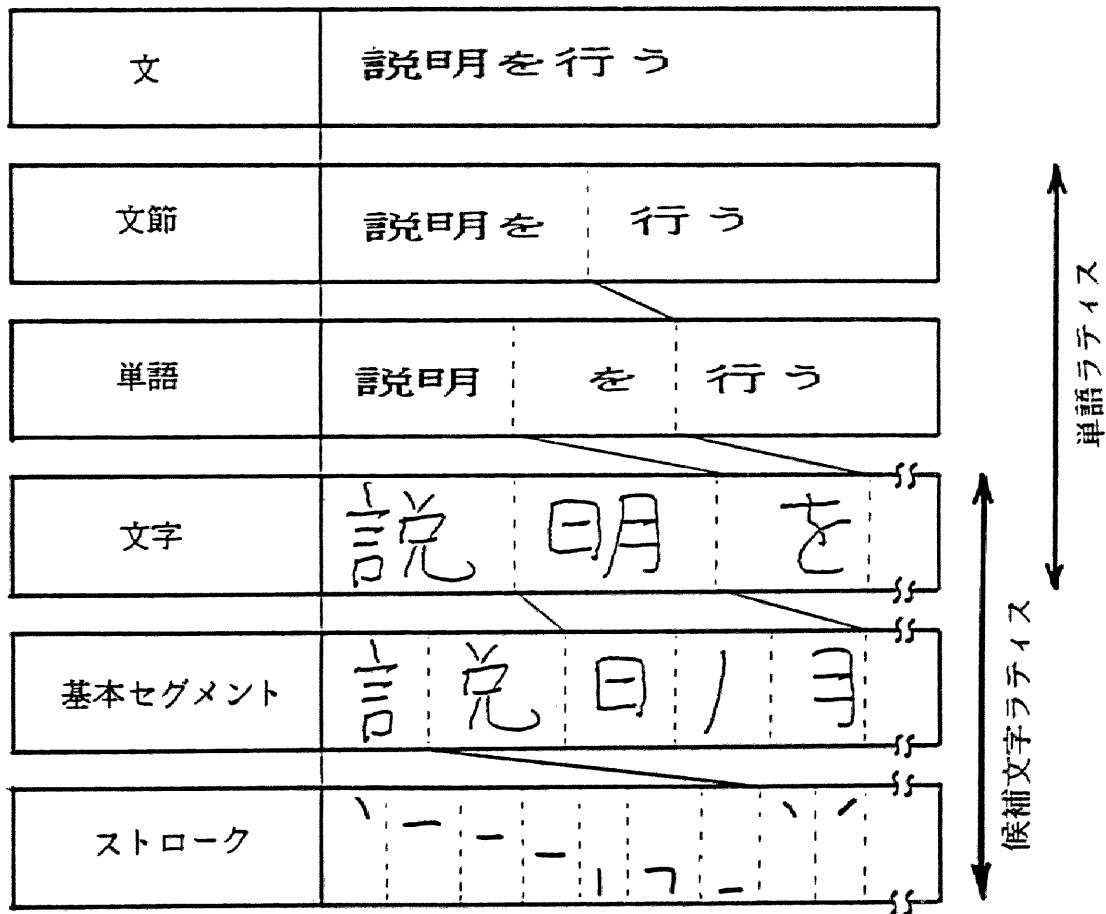


図6.13 文字列の階層性



#### 6. 5. 4 文字認識後処理法との関係

従来提案されている文字認識後処理法の多くは、基本的には文字認識の結果得られる候補文字の上位幾つかを並べ、その中から言語的に正しい文字列を選択する方式を取っている。つまり、文字認識結果の記述は図6. 14 (a) に示すようになる。一方、本手法では、基本セグメントの組み合わせから生成されるラティスの中から言語的に正しい文字列を選択しているため、文字認識結果の記述は候補文字ラティス(図6. 14 (b)) のようになる。この手法において、候補文字を枠内に筆記された個々の文字としたならば、従来の文字認識後処理法と等価な手法と考えられる。つまり、本手法は、言語的な処理はまだ不十分であるものの、従来の文字認識後処理法の文字認識部の記述をより拡張したものと考えられることができる。

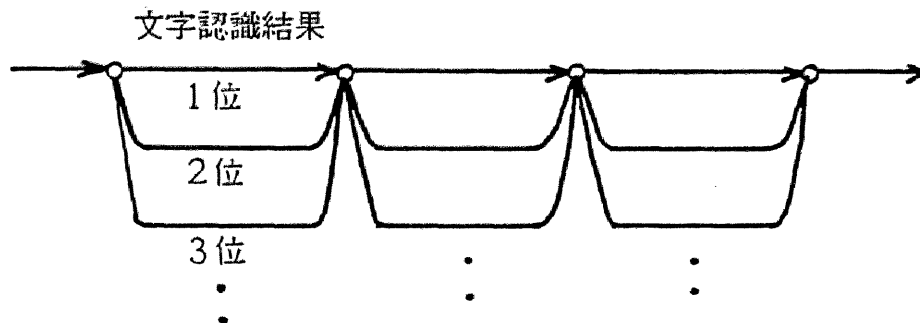


図6. 14 (a) 従来の文字認識後処理法における文字認識結果の記述

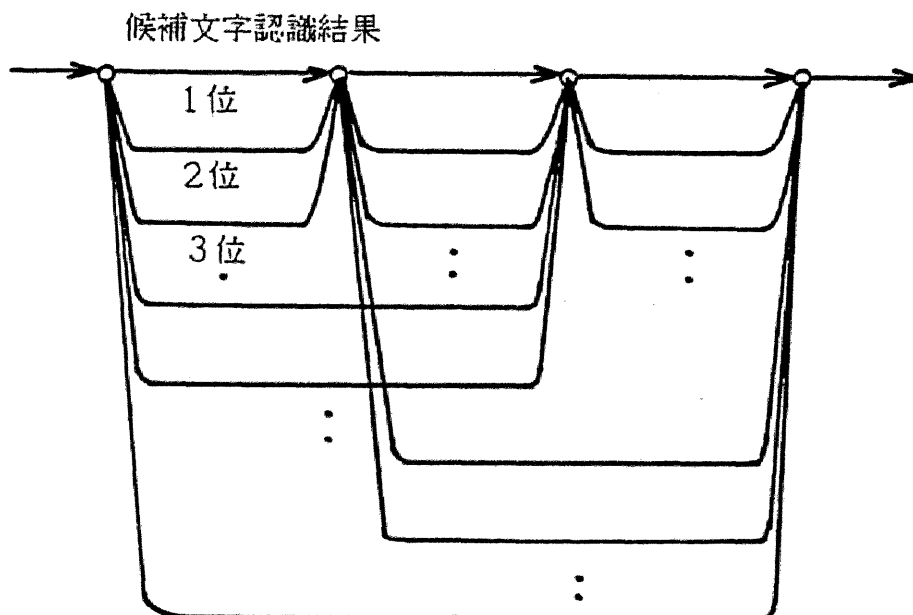


図6. 14 (b) 候補文字ラティス法の文字認識結果の記述

### 6.5.5 文字間隔が候補文字ラティス法に与える影響

候補文字ラティス法において、文字列の文字間隔が文字の認識に与える影響について調査した。文字間隔を-10%、0%、20%、40%、80%に変え文字列を合成した。文例はデータ2を作成した500文例中、先頭の100文例を使用した。その1例を図6.16に示す。候補文字ラティス法の文字重なり許容度Thは、5章の実験で使用した値と同様-0.15に設定した。但し、この調査では言語処理は省略した。結果を表6.5と図6.15に示す。文字間隔は、候補文字ラティス法に殆ど影響を与えないことを確認した。

表6.5 文字間隔と文字列の認識率

間隔	-10%	0	20	40	80
認識率	96.5%	96.8	96.7	96.6	96.9

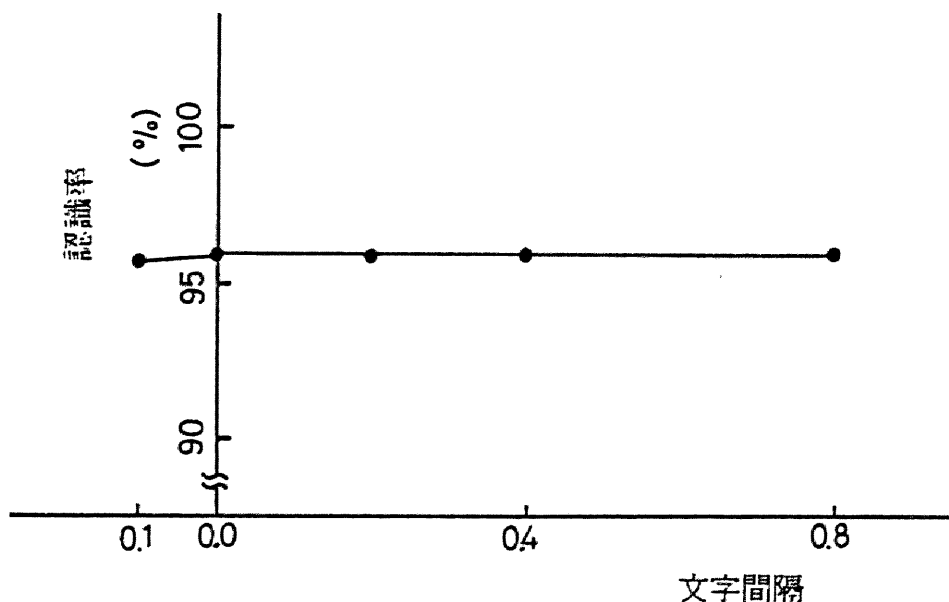
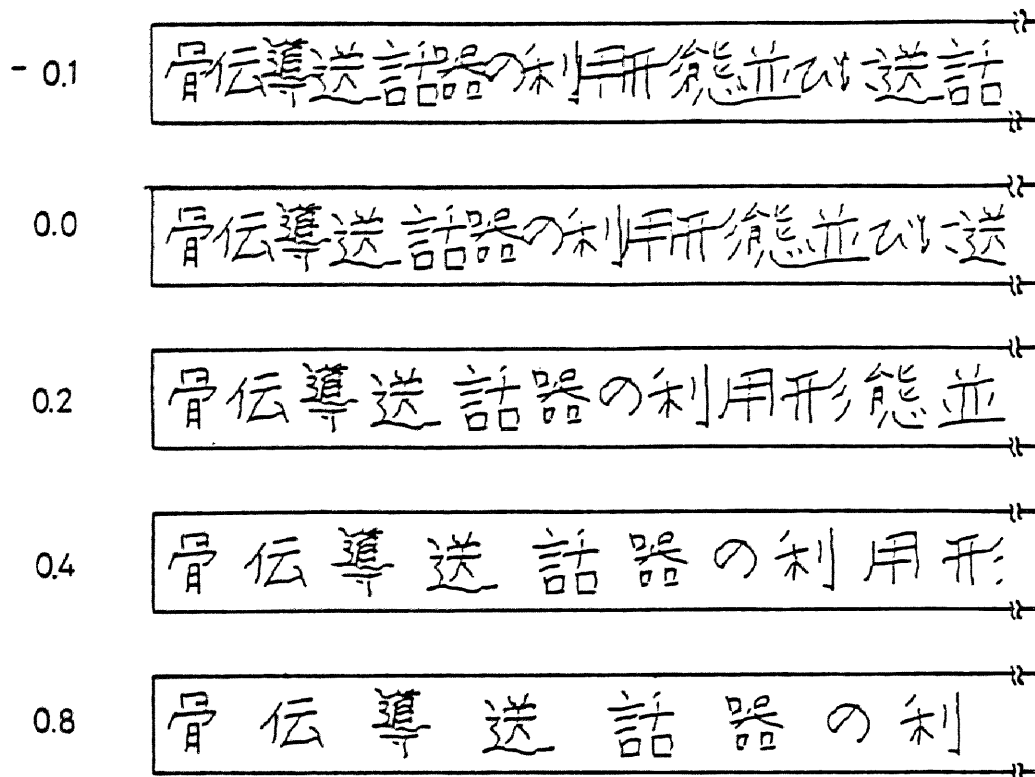


図6.15 文字間隔に対する文字列の認識率



間隔	認識結果
-0.1	骨伝導送話器の利用形態並びに送話
0.0	骨伝導送話器の利用形態並びに送話
0.2	骨伝導送話器の利用形態並びに送話
0.4	骨伝導送話器の利用形態並びに送話
0.8	骨伝導送話器の利用形態並びに送話

図6.16 文字間隔を変化させた場合の入力文字列と認識結果

### 6.5.6 候補文字認識における距離値の安定性について

候補文字ラティス中の候補文字には実文字（実際の文字）と非文字（文字以外の図形）が存在する（図6.17参照）。但し、ここでは文字とは常用漢字と平仮名のことをさす。もし非文字の距離値が実文字のそれに比較して十分大きな値となれば、非文字を棄却することが可能となる。非文字がラティス中から棄却できれば、候補文字ラティスが単純になり、より効率的なラティス探索が可能となる。そこで、実文字、非文字に関する、距離値の分布を調査した。使用データは、データ1の中で先頭から30文例を選出したものである。結果を表6.6と図6.18に示す。この結果から分かるように、距離値はあまり安定していないため、距離値によって実文字と非文字を分離することは困難であることを確認した。

但し、ここでは（距離値）＝（相異度）／（ストローク数）で定義する。





候補文字	文字の種類	認識結果	距離値
	実文字	地	0.20
	非文字	班	0.24
	非文字	せ	0.34
	非文字	作	0.39

図6.17 実文字と非文字の例

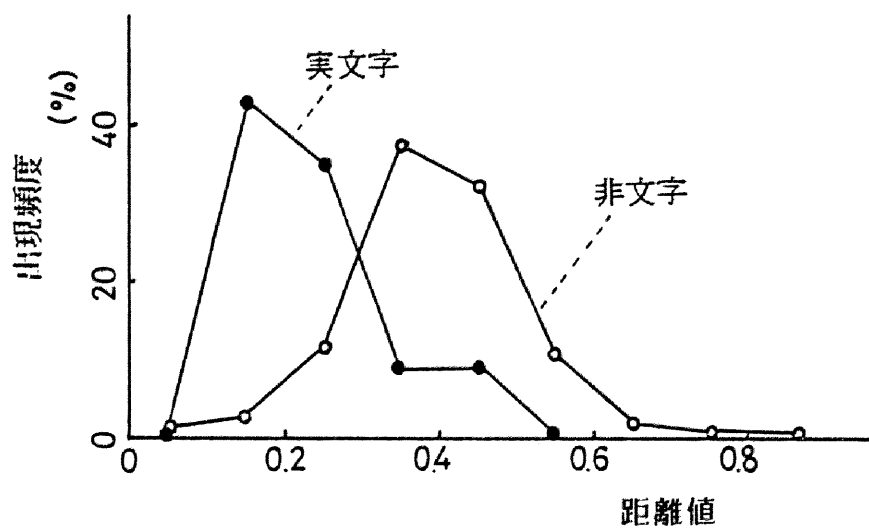


図6.18 距離値の頻度分布

表6.6 距離値の分布

	実文字	非文字
0.0~0.1	0.3%	0.0%
0.1~0.2	43.2	2.8
0.2~0.3	36.6	11.4
0.3~0.4	8.9	37.1
0.4~0.5	10.4	32.5
0.5~0.6	0.6	10.8
0.6~0.7	0.0	2.8
0.7~0.8	0.0	1.4
0.8~0.9	0.0	1.1
0.9~1.0	0.0	0.1
合計	100.0	100.0

## 6.6 結言

この章では、自由書式で筆記された文字列から文字を精度良くセグメンテーションし、同時に認識する手法について述べた。具体的には、文字のセグメンテーション、文字認識、言語処理を統合した認識系を構築し、言語情報を含む上位の知識を下位の処理に有効にフィードバックすることにより、手書き文字列から精度良く文字をセグメンテーションする手法を提案した。以下に結果をまとめる。

(1) 文字のセグメンテーション、文字認識、言語処理を統合した認識系を構築した。具体的には、文字を構成するセグメントと文字情報の結合に候補文字ラティス(2端子有向グラフ)表現を、文字情報と単語情報との結合には単語ラティス表現を提案した。

(2) 候補文字ラティス上の各ブランチの重み和がK番目に最短となる経路を第K次最短経路と呼ぶ。第K次最短経路を、Kを1から順次増加させながら、反復的に探索することにより、言語情報を文字のセグメンテーションと文字認識の両者に有効にフィードバックすることが可能となった。第K次最短経路の探索には動的計画法が有用であることを示した。

(3) 言語情報を導入したことにより、対象字種(常用漢字と平仮名)中9.7%ある分離文字を正しくセグメンテーションすることが可能となった。これにより、605文例の文字列データ(データ1とデータ2)に対する文字のセグメンテーション率は、言語情報を導入しない手法の97.0%から99.2%に向上した。また、本手法は文字の認識精度を向上させる文字認識の後処理効果もあり、文字の認識率は99.1%から99.8%に向上した。

(4) 本手法は縦書き文字列に対しても効果があり、500文例の文字列データ(データ3)に対して、文字のセグメンテーション率で99.4%を達成した。

この章で提案した自由書式で筆記された文字列の認識法では、文字認識部、言語処理部をモジュールとして扱っている。文字認識部では、文字図形を入力とし、認識結果の文字コードと、標準パターンと入力文字図形との間の相異度をN位まで出力する機能を持つ。言語処理部では、文章が日本語として正しいか否かを判定する機能を持っていて良い。そのため、各モジュールでより高度なアルゴリズムが開発されればそれを簡単に組み込むことが可能である。残された課題は、各モジュールの機能を拡

充させることにある。例えば、文字認識部では走り書きなどの低品質文字の認識アルゴリズムの導入が、また言語処理部では文章の意味までも考慮した処理の導入が考えられる。

## 第7章 文字・図形混在図面認識システム

### 7.1 緒言

本研究の最終的な目標は、筆記制限なく雑に手書きされた自由書式の文字・図形混在図面を自動的に認識するシステムの構築にある。マンマシンインタフェースの観点からすると、入力、認識、訂正、編集の各機能を含むシステムの実現が重要であるが、本研究では特に認識の部分だけに閉じて議論している。この章では、2～6章で検討してきた文字や図形の認識に関する基本技術を利用することにより実現した自由書式で筆記された文字・図形混在図面の、認識実験システムについて述べる。実験システムであるため、訂正や編集の方法についてはここでは議論しない。

文字と図形が混在した図面を認識するために、次の3段階から構成される処理を提案する。図7.1にこの処理の流れ図を示す。第1段階は、図面認識のための前処理にあたる。まず図面を構成する各部分図形を、文字領域と図形領域に分離する。また、図面中に書かれている文字には縦書きや横書きが存在している。そこで文字列の筆記方向を検出し、必要に応じて文字列に回転の正規化を加える。第2段階は、文字列の認識である。文字列の認識アルゴリズムは、本論文の5章、6章で詳細に述べた。前処理により分離され、更に正規化を加えられた文字列は、この文字列認識アルゴリズムをそのまま適用することにより、認識することが可能である。第3段階は、図形領域の認識である。図形領域の認識は、本論文の2章～4章で詳細に述べてきたので、それをそのまま適用することが可能である。

つまり、文字と図形が混在した図面の認識システムを構成するために、新規に必要な技術は、第1段階で示した文字と図形の分離処理と、文字列の正規化処理である。

この章では、まず7.2節で、図面認識システムのハードウェア構成について述べ、次に文字と図形が混在した図面の認識システムを構成するのに必要となる技術として、7.3節で文字と図形の自動分離法、7.4節で文字列の回転の正規化法[73]について述べる。更に、認識システムを実際に構成するためには、各部の処理速度についての検討が必要となる。特に、第2段階の文字列認識処理では5章で述べたように、多数の候補文字に対する認識処理が必要となり、処理量が膨大となり、認識システムを構成する上でのネックとなる。しかし、この箇所の処理は単純なマッチングの繰り返しであり、専用の処理装置(専用ハードウェア)で実現するのに適している。そこ



で、7.5節では、認識系を効率的に構築するために試作した専用処理装置について触れる。

本章で扱う範囲を図7.1に太枠で示す。

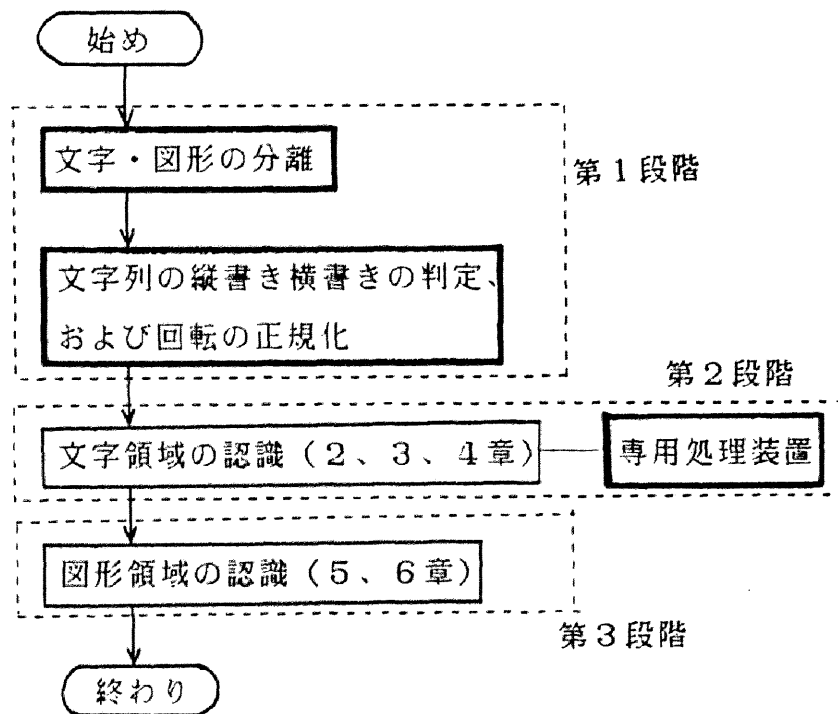


図7.1 処理の流れ

## 7.2 システム構成

2章～6章で行なった文字や図形の分析や、認識のシミュレーション実験は、全てデータゼネラル社のミニコン/MV8000で行なって来た。MV8000は1MIPSの処理速度を有する汎用ミニコンであるが、マルチユーザーの環境である。そのため、他のジョブの影響でタブレットからのデータに転送誤りが生じたりし、オンラインで認識システムを構成するには不適當である。この章では、図面中からの文字と図形の分離、文字列の認識、図形領域の認識をオンラインで統一的に実行できるようなハードウェアを構成した。

文字・図形混在図面の認識実験システムのハードウェア構成を図7.2に示す。中央処理装置としては、アポロ社のワークステーション/ドメインDN660を採用した。DN660はマルチウィンドーのディスプレイを持つワークステーションで、こ

のワークステーションに、文字や図形の筆記情報を採集するためのタブレットと、文字列認識を効率的に実現するための専用処理装置を接続している。DN660と専用処理装置との間では、GPIBの回線を利用したDMAの高速データ転送が可能となっている。転送速度は毎秒80Kbyte程度である。専用処理装置については7.5節で詳細に述べる。更に、DN660は、補助記憶装置として磁気ディスク167Mbyteと磁気テープ装置を有している。DN660とMV8000はGPIBおよびRS232Cの回線を用いて相互に接続されている。

図7.1の流れ図で示した処理の中で、第2段階の文字認識処理部は、試作の専用処理装置（ハードウェア）で実現した。一方、第1段階の前処理と第3段階の図形認識処理および第2段階の文字認識認識における専用処理装置の制御はDN660上のソフトウェアで実現した。

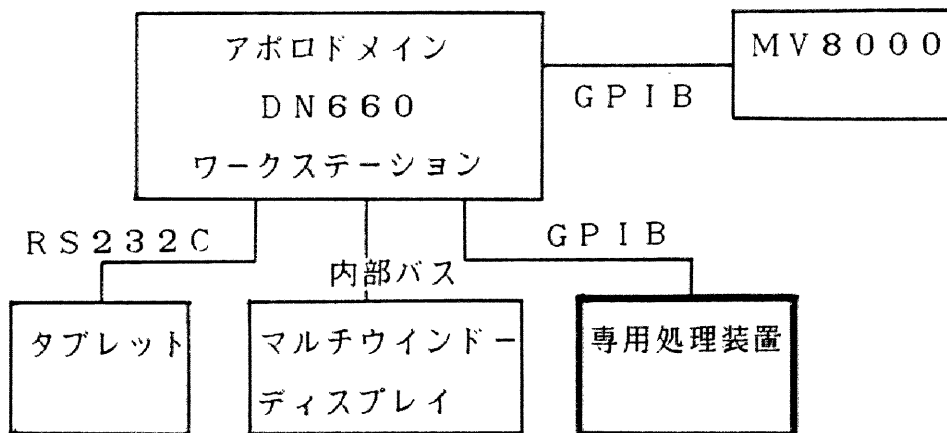


図7.2 認識システムの構成

### 7.3 文字と図形の分離

文字と図形が混在した図面を認識するため、前処理としてまず文字領域と図形領域を分離する必要がある。この節では、ストロークの大きさやストローク間の近接関係を利用した文字と図形の分離法を述べる。

### 7.3.1 処理対象

技術文書中によく出現するフローチャートやブロック図等の図面を処理対象とする。実験で使用するデータは、以下の①～③の条件で筆記された図面であり、タブレットから採集したものである。条件①②は筆記制限になるが、これはフローチャートやブロック図を筆記する際に大きな制限とはならない。データの収集数は10枚（5種類の図面を1名の筆記者が各2回筆記）であり、その1例を図7.3に示す。

- ①図面中の文字はだいたい同じ大ききで筆記する。
- ②文字領域と図形領域は重ならないように筆記する。
- ③図面中の文字列を筆記する際、縦がきと横書きの存在を許容する。

これらの図面を分析した結果、以下の性質を持つことを確認した。(1) 手書きのために、文字の大ききは変動する。しかし、その範囲は±20%程度である。(2) 図面中の図形部には、文字に比較して長いストロークが存在する。(3) 図形部は互いに連結している。しかし、手書き変形により、ストロークが本来連結している箇所互いに分離する場合がある。

上記の性質を考慮して、文字・図形分離法を検討する。

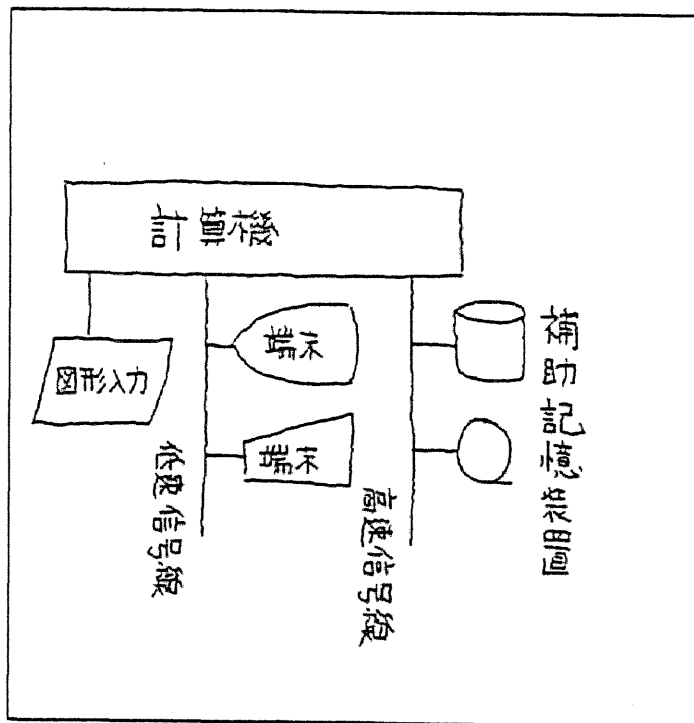


図7.3 文字・図形混在図面の例

### 7.3.2 ストロークの大きさに着目した文字と図形の分離

文字領域と図形領域で、それぞれを構成するストロークの大きさが異なることを利用して文字と図形を分離する手法を提案した。分離手法の流れを図7.4に示す。以下、順を追って各処理を説明する。

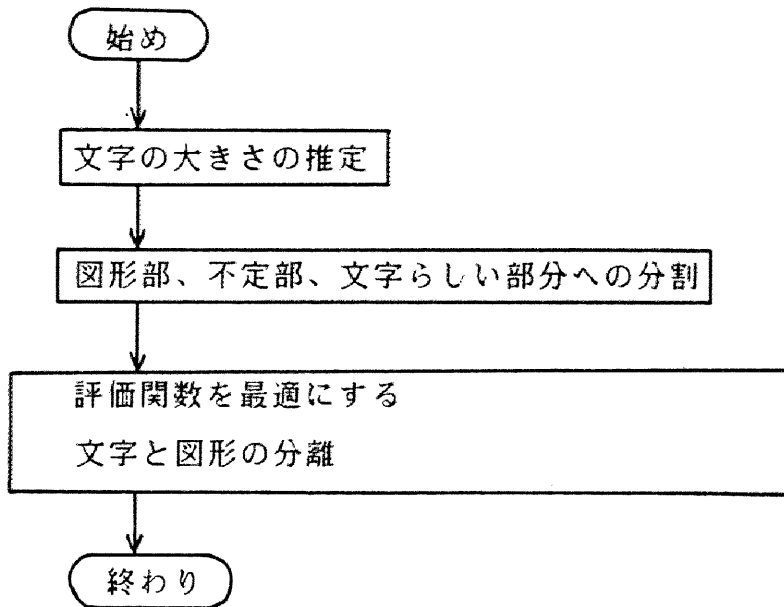


図7.4 文字と図形の分離法

### 7.3.3 文字の大きさの推定

認識対象とする図面は、文字領域と図形領域から構成されている。文字領域は多数の小さなストロークから構成される。一方、図形領域は種々の大きさのストロークが含まれるが、ストローク数は文字領域に比較して少ない。この性質を利用して図面中に含まれる文字の大きさを推定することが可能である。文字の大きさの推定法を以下に述べる。

あるストロークの座標値系列を  $\{(x_i, y_i), i=1, \dots, I\}$  とする。このストロークの大きさ  $S$  を、

$$S = \max_i \{ \max(x_i) - \min(x_i), \max(y_i) - \min(y_i) \} \quad (7.1)$$

で定義する。

図面中に含まれる文字に対して、ストロークの大きさに対するストロークの頻度分布を概念的に示した図が、図7.5(a)である。一方、図形部の頻度分布は図7.5(b)となる。図面中に含まれる全てのストロークに対する頻度分布はこの両者の頻度分布を足し合わせたものとなり、図7.5(c)のようになる。図7.5(c)の斜線部に示すように、この頻度分布の上部を切り出すことを考える。もし、図形部の分布が図7.5(b)に示すように一様分布であり、しかもその頻度が少ない場合には、この斜線部は図7.5(a)の分布の上部を切り出したものと同等になる。実際の図面において、一般にこの仮定は適用できる。この切り出した分布を利用して、それから文字の大きさを推定することが可能である。

文字と図形の混在した図面から文字の大きさを具体的に推定する手順を示す。まず、図面全体のストロークの大きさに対する頻度分布を作成する。次に、この分布から、面積で50%になるように分布の上部を切り出してくる。この部分のストロークの大きさの平均値を求める。図7.5(a)～(c)で示したように分布の上部を切り出し、その部分の平均値Mを求めれば、これは文字領域だけの上部を切り出して、その平均値を求めることとほとんど一致する。この平均値Mにパラメータ $\alpha$ を掛けた値を、文字の大きさの推定値とする。

ここで、パラメータ $\alpha$ の決定法について検討する。当用漢字1945字種と平仮名46字種を各5回ずつ筆記した合計9955文字の文字データを使用して以下の調査を行なった。各文字の大きさを1.0に正規化し、ストロークの大きさに対するストロークの頻度分布を調査した。その分布を図7.6に示す(図7.5(a)に対応)。分布の上部を面積でA%(図7.6斜線部)抽出して、それからストロークの大きさの平均値Mを求めた結果を表7.1に示す。Aが50%の時の平均値に $\alpha$ を掛けた値が1.0となるように $\alpha$ を決定した。つまり、 $\alpha$ として4.29を設定した。

次に、文字の大きさの推定値の誤差について検討する。実際の図面では文字の分布に加えて図形領域の分布が重畳されている。図形領域の分布が一様分布であると仮定すると、図面全体の分布の上部50%を切り出して来ることは、文字領域の分布で50%以上を切り出して来ることに相当する。ストロークの大きさの頻度分布の上部A%だけを利用することにより文字の大きさを推定することを考える。表7.1に、 $\alpha$ の値を一定にしてAの値を変化させた際の文字の大きさの推定値の変化を示す。この結果より、Aの値に拘らず( $40\% \leq A \leq 80\%$ )ストロークの大きさの頻度分布の

上部だけを利用することにより、文字の大きさは5%以内の精度で正しく推定されることが分かる。これにより、図面中の図形部の影響（Aの値の影響）を受けることなく文字の大きさを推定することが可能となる。

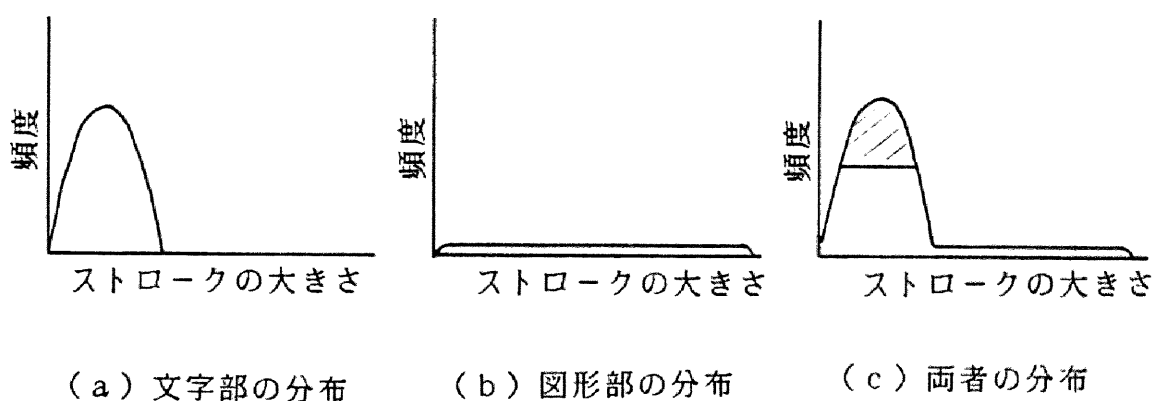


図7.5 ストロークの大きさに関する頻度分布

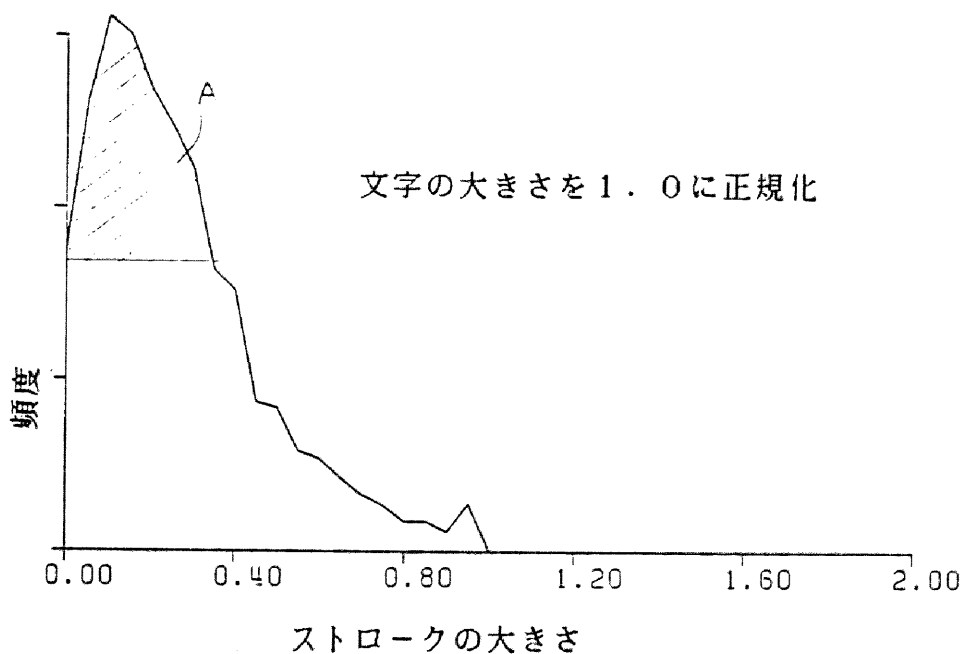


図7.6 ストロークの大きさに対する頻度分布（文字部分）

表 7. 1 文字の大きさの推定値

A (%)	平均値	文字の大きさの推定値	誤差 (%)
80	0.247	1.05	5
70	0.243	1.04	4
60	0.237	1.01	1
50	0.233	1.00	0
40	0.241	1.03	3

#### 7. 3. 4 ストロークの大きさによる文字と図形の分離

一般に図面中では、ストロークの大きさの大きいものは図形領域であり、小さいものは文字領域であることが多い。但し、ここでは領域の分離とはストロークを文字部分と図形部分に分類することである。文字と図形を自動分離するための予備実験として、ストロークの大きさだけによる文字と図形の分離を試みた。文字の大きさの推定値をもとにいき値を決定し、いき値以上の大きさを持つストロークを図形領域とする。

10枚の図面を対象に実験を行なった。その結果を図7. 7に示す。いき値の変化に対する、分離率、文字を図形と判定した分離誤り、図形を文字と判定した分離誤りを示す。但し分離率は、

$$\text{分離率} = \frac{\text{(正しく分離されたストロークの数)}}{\text{(総ストローク数)}} \quad (7.2)$$

で定義する。小さいいき値を設定すれば文字を図形に誤る頻度が高くなり、一方大きいいき値を設定すれば図形を文字に誤る頻度が高くなる。

しかし、この結果から分かるように、最適ないき値を設定してもストロークの大きさだけで文字と図形を分離する手法では、高い分離精度は得られない。

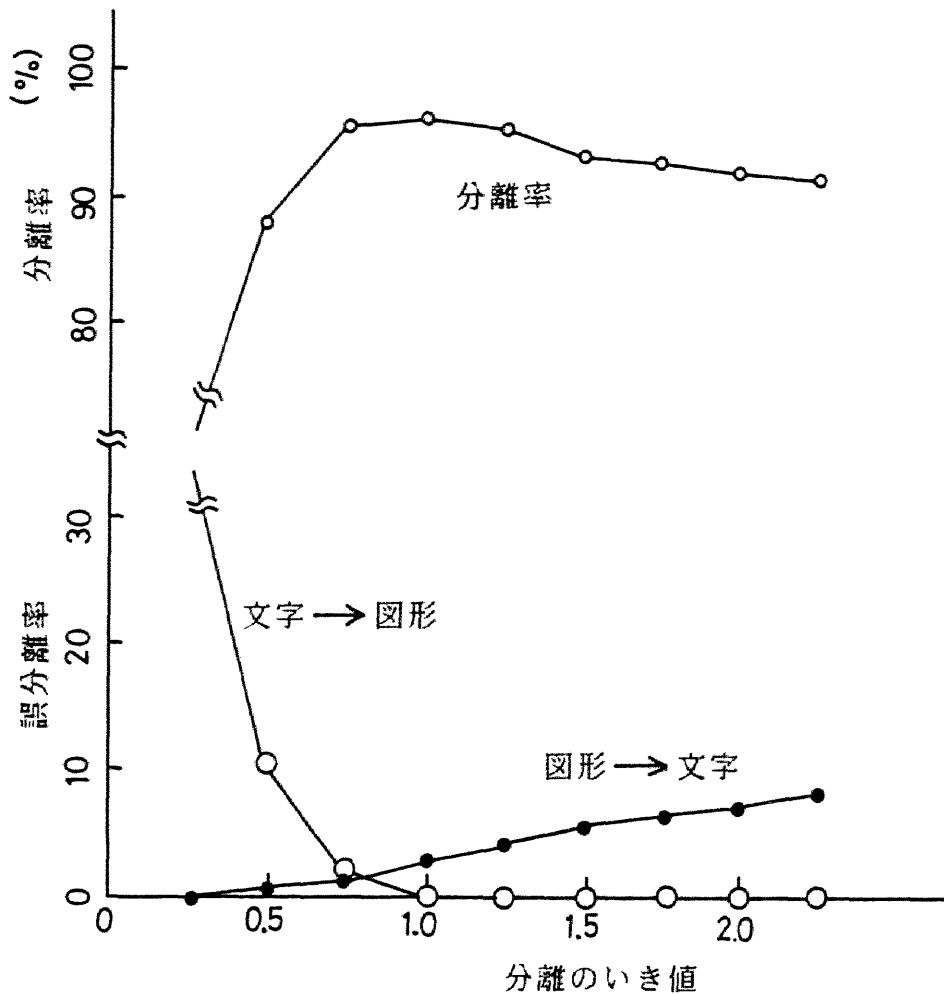


図7. 7 ストロークの大きさに着目した文字と図形の分離

### 7. 3. 5 ストロークの分類

前節で述べたように、ストロークの大きさだけで図面を文字領域と図形領域に分離することは困難である。そこで、ストロークの大きさと、ストローク相互の位置関係を利用した文字図形分離法を提案する。

まずストロークを

- ① 図形らしいストローク、
- ② 文字か図形か不明のストローク、
- ③ 文字らしいストローク

の3種類に分類する。これらの分類はストロークの大きさと、ストローク間の位置関係により決定する。以下に、各ストロークの分類基準を述べる。



① 図形らしいストローク (F (figure))

文字の大きさの推定値が正しければ、文字の大きさ以上の大きさを有するストロークは図形を構成するストロークとなる。ここでは、安全を見込んで、文字の大きさの推定値に  $(1 + \beta + \gamma)$  倍した値以上の大きさを持つストロークを、図形となるストロークとする。但し、 $\beta$  は文字の大きさの変動許容値である。7.3.1で述べたように文字の大きさの変動は  $\pm 20\%$  存在することから、 $\beta$  の値として 0.2 を選定する。また、 $\gamma$  は文字の大きさの推定値に対する誤差を見込んだ安全値である。7.3.3で述べたように推定値の誤差は  $\pm 5\%$  であるため、 $\gamma$  の値として 0.05 を選定する。

② 文字か図形か不確定のストローク (U (uncertain))

フローチャートやブロック図等の図形では、図形のストロークに連結もしくは近接しているストロークはやはり、図形であることが多い。そこで、ストロークの大きさは小さいものの、図形の近傍に位置するストロークを、文字か図形か不明のストロークとして分類する。具体的には以下で定義する自己ストローク間距離行列の要素の値が一定しきい値以下となるストロークを近傍ストロークとする。

③ 文字らしいストローク (C (character))

上記①②以外のストローク、つまり図形領域に近接せず、且つ大きさの小さいストロークを文字らしいストロークとする。

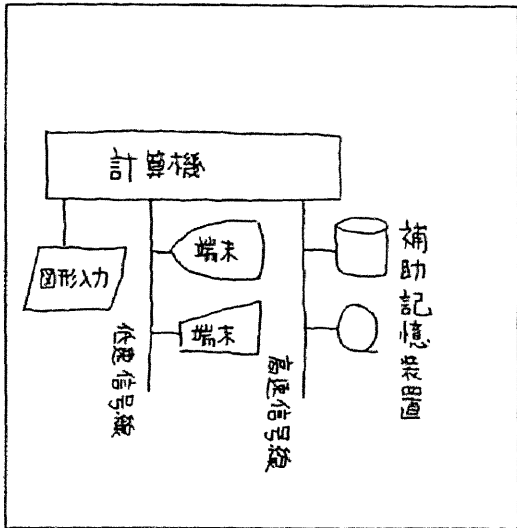
上記の分類処理を効率的に実行するため、ストローク間の距離を表現する自己ストローク間距離行列 D

$$D = \{d_{ij}\} \tag{7.3}$$

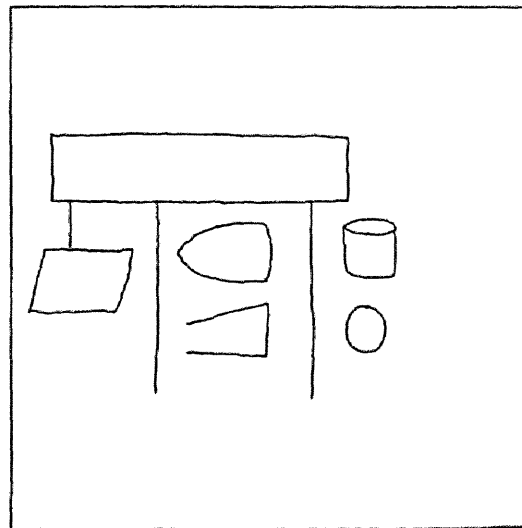
を定義する。 $d_{ij}$  は第  $i$  ストローク  $\{(x_{iu}, y_{iu}) \ u=1, \dots, U\}$  と第  $j$  ストローク  $\{(x_{jv}, y_{jv}) \ v=1, \dots, V\}$  の間の距離であり、以下の式により定義される。

$$d_{ij} = \min_{u,v} \left\{ \sqrt{(x_{iu} - x_{jv})^2 + (y_{iu} - y_{jv})^2} \right\} \tag{7.4}$$

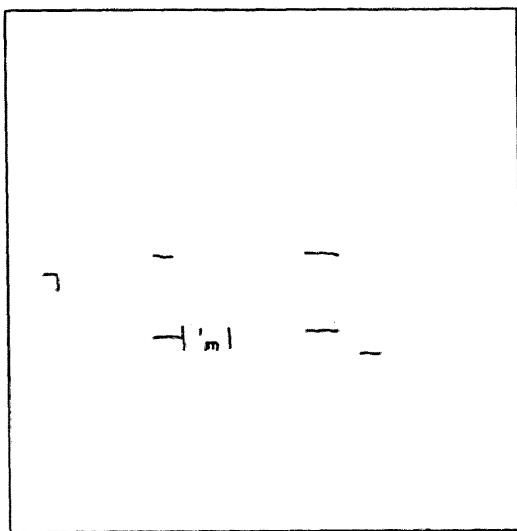
ストロークをこの3種類に分類した例を図7. 8に示す。(a)が入力図面、(b)が図形らしいストローク、(c)が不確定なストローク、(d)が文字らしいストロークである。



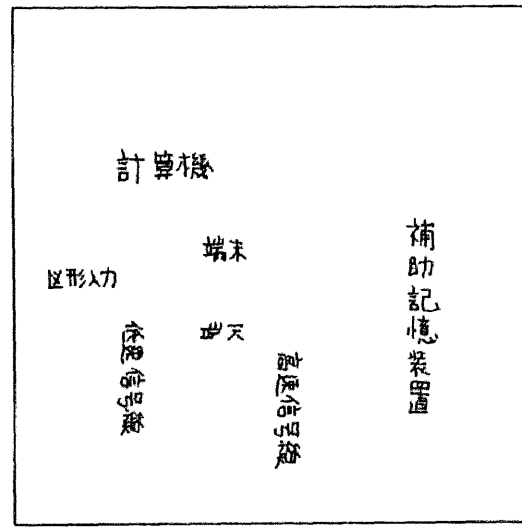
(a) 入力図形



(b) 図形らしい領域  
(図形らしいストローク)

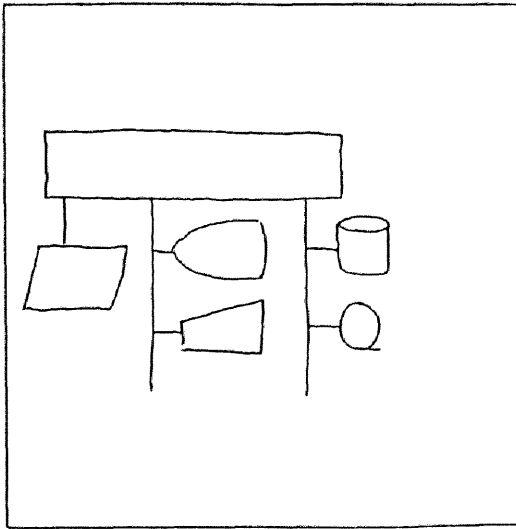


(c) 不確定領域  
(不確定ストローク)

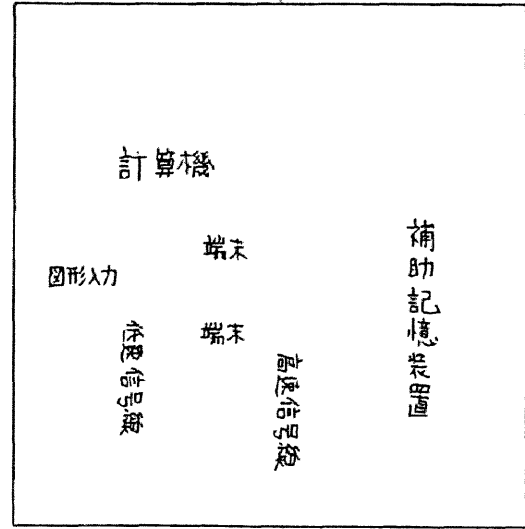


(d) 文字らしい領域  
(文字らしいストローク)

図7. 8(a)-(d) 文字と図形の分離の説明図



(e) 分離結果 (図形領域)



(f) 分離結果 (文字領域)

図 7. 8 (e),(f) 文字と図形の分離の説明図

### 7. 3. 6 評価関数による文字と図形の分離

前節までの処理により図面中のストロークを、図形部分、文字部分、不確定部分の三部分に分離した。ここでは、評価関数を導入し不確定部分を図形あるいは文字に最適に分離する手法について述べる。

対象としている図面では、文字を構成するストロークはストローク同士で隣接し合う確率が高い。これは図形を構成するストロークについても同様である。一方、文字を構成するストロークと図形を構成するストロークは互いに分離する傾向がある。ここではその特性を反映した評価関数  $T$  を導入する。

$$T = T_1 - T_2 \quad (7.5)$$

但し、

T 1 = 文字と図形との間の平均ストローク間距離

$$= \left( \frac{\sum_{i \in F} (\min_{j \in C} d_{ij})}{\sum_{i \in F} (i)} \right) + \left( \frac{\sum_{j \in C} (\min_{i \in F} d_{ij})}{\sum_{j \in C} (j)} \right) \quad (7.6)$$

T 2 = 図形内の平均ストローク間距離 + 文字内の平均ストローク間距離

$$= \left( \frac{\sum_{i \in F} (\min_{j \in F} d_{ij})}{\sum_{i \in F} (i)} \right) + \left( \frac{\sum_{j \in C} (\min_{i \in C} d_{ij})}{\sum_{j \in C} (j)} \right) \quad (7.7)$$

である。ここで、評価関数 T を最大とするように、不確定部分のストロークを文字あるいは図形のストロークに分類し、これを最終的な文字と図形の分離結果とする。

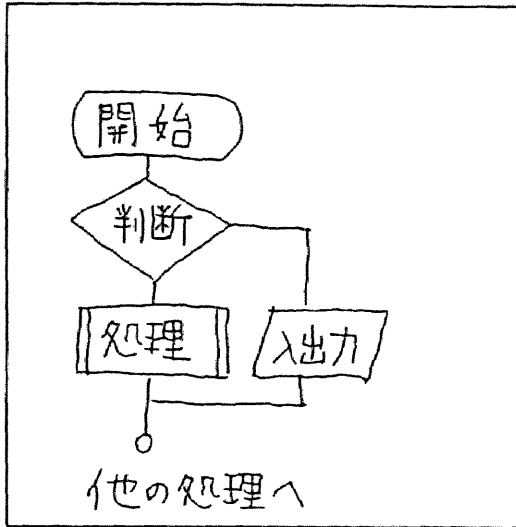
実際の計算は以下のように行なう。不確定なストロークの数が k であるとする、総当たり法で計算すると  $2^k$  回の T の計算が必要となる。この場合 k の値が増大すると膨大な計算量が必要となる。そこで計算の効率化のために、距離が遠いストロークは独立であると考えて、不確定部分のストロークをクラスター化して計算する。例えば不確定と分類されたストロークの数が 3 本で、内 1 本と 2 本に図面上で分離している場合を考える。この場合には 1 本と 2 本にクラスター化される。計算は表 7. 2 に示す場合について行なう。表中、F は不確定なストロークを図形として扱う場合を、C は文字として扱う場合を、\* はどちらか一方（文字でも図形でもどちらでも良い）に固定する場合を示す。総当たり法では全ての場合を計算するため以下のように 8 回 T を計算するが、クラスター化した場合には 6 回の T の計算で良い。この例ではあまり差が生じていないが、図 7. 8 に示すような実際の例では、不確定と分類されたストローク数は 13 であるので、総当たり法では 8192 ( $2^{13}$ ) 回、一方、クラスター化した場合には、9 個のクラスターに分かれ 34 ( $7 \times 2 + 2^2 + 2^4$ ) 回となり、計算量は大幅に減少される。

表 7. 2 評価関数 T を計算する回数

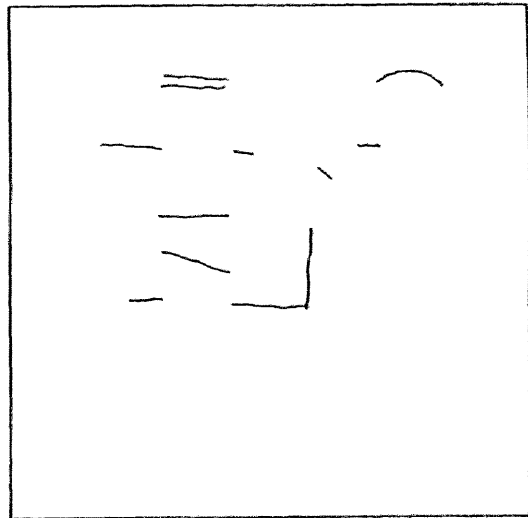
場合の数	総当たり法			クラスター化した場合		
	1	2	3	1	(2	3)
1	F	F	F	F	*	*
2	C	F	F	C	*	*
3	F	C	F	*	F	F
4	C	C	F	*	C	F
5	F	F	C	*	F	C
6	C	F	C	*	C	C
7	F	C	C			
8	C	C	C			

### 7. 3. 7 文字と図形の分離実験

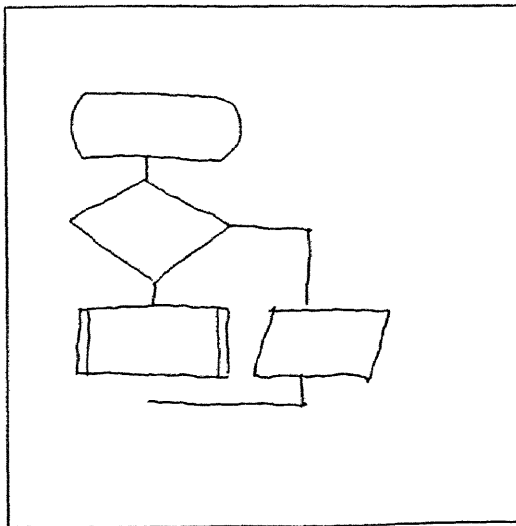
例 7. 2 に示したような図面 10 枚について文字図形の分離実験を行なった。評価関数 T として  $T_1 - T_2$  とした場合には分離率は 99. 4% であった。比較の目的で  $T = -T_2$  として実験を行なったところ分離率は 99. 1% であり、今回用いた評価関数が高い分離能力を有することを確認した。誤った例を図 7. 9 に示す。この例では、図面中のコネクタの記号（小丸）の大きさが小さいため文字領域と判定され、これに接続されている接続線は不確定なストロークとして分類されたものの、このコネクタに接触しているため、やはり文字領域として判定されたわけである。このような例では、ストロークの大きさとその接触（近傍）関係だけでは、正しく分類されない。これに対処するためには、更に高度な図形理解からのフィードバックが必要になると考えられる。



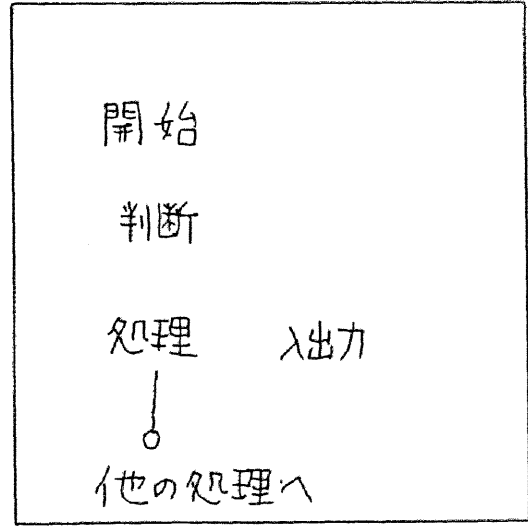
(a) 入力図形



(b) 不確定領域



(c) 分離結果 (図形領域)



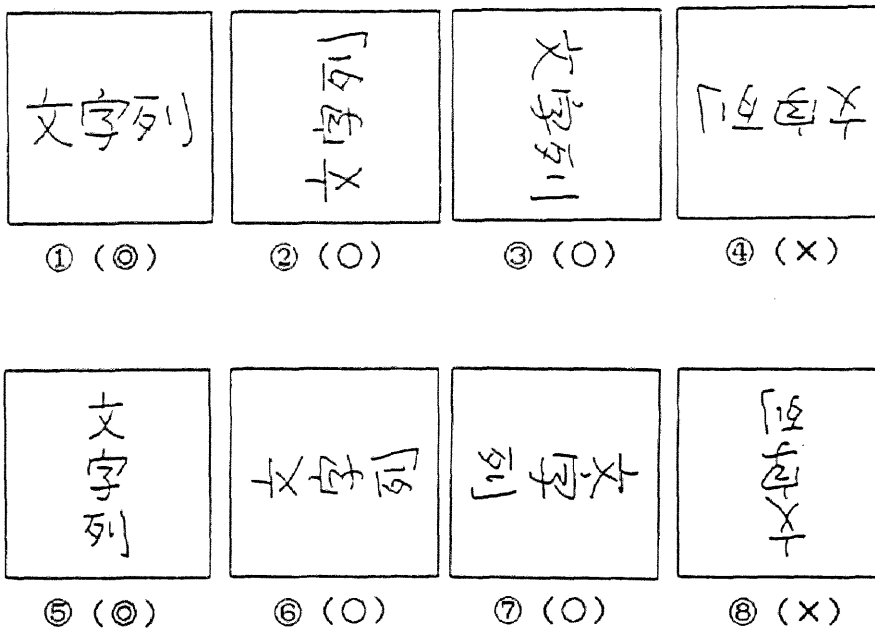
(d) 分離結果 (文字領域)

図 7. 9 誤分離の例

## 7.4 文字と図形の回転の正規化

### 7.4.1 文字列の方向の正規化

前節で図面中から文字列と図形を分離する手法を述べた。分離された文字列は種々の文字列方向（縦書き／横書き）および筆記方向（縦向き／横向き）により筆記されている。ここでは、それら文字列方向、筆記方向を自動的に判定する手法を述べる。本手法により、図7.10に示すような8種類の文字列を、それぞれ分類することが可能となる。但し、この8種類のうち逆方向に筆記される場合は実際には殆どない。



但し、(◎) (○) は一般図面によく出現するが、  
(×) はほとんど出現しない。

図7.10 8種類の文字列（文字列方向、筆記方向による変形）

#### (1) 文字列方向の判定

文字列を構成するストロークを構成する座標点系列を  $\{(X_i, Y_i) \mid i=1, \dots, N\}$  とする。これらの点の分布で最も広がりの大きい方向、つまり主軸（第1固有ベクトル）

ル) を文字列の方向とする。

文字列方向 ( $L_x$ ,  $L_y$ ) は以下の式により与えられる。

$$\begin{cases} L_x = \sqrt{(1/2 + (S_{11} - S_{22}) / (2 \cdot \sqrt{(\Delta)}))} & (7.8) \\ L_y = \left\{ (\sqrt{(\Delta)} - (S_{11} - S_{22})) / (2 \cdot S_{12}) \right\} \cdot L_x & (7.9) \end{cases}$$

但し、

$$S_{11} = \sum_{i=1}^N (X_i - X_m)^2 \quad (7.10)$$

$$S_{22} = \sum_{i=1}^N (Y_i - Y_m)^2 \quad (7.11)$$

$$S_{12} = \sum_{i=1}^N (X_i - X_m) \cdot (Y_i - Y_m) \quad (7.12)$$

$$\Delta = (S_{11} - S_{22})^2 + 4 \cdot S_{12} \cdot S_{12} \quad (7.13)$$

$$X_m = \left\{ \sum_{i=1}^N X_i \right\} \cdot \frac{1}{N} \quad (7.14)$$

$$Y_m = \left\{ \sum_{i=1}^N Y_i \right\} \cdot \frac{1}{N} \quad (7.15)$$

である。

## (2) 筆記方向の判定

ここでは、文字や図形を構成するストロークの方向成分に関する性質および手書きの筆記方向特性を利用して、文字の筆記方向を判定する手法を述べる。

文字や図形を構成するストロークには一般に縦横成分が多い。更に、縦ストロークはペンを上から下へ移動させ、横ストロークはペンを左から右へ移動させて筆記することが多い。そこで、入力ストロークを微小線素に分割し、方向  $\theta$  に関する各線素の頻度分布 (ストローク方向分布)  $h(\theta)$  を導出する。入力ストロークが正面方向から筆記された場合には、 $h(\theta)$  には上から下方向 ( $\theta = 3/2\pi$ ) と、左から右方向 ( $\theta = \pi/2$ ) と、



$\theta=0$ ) とに 2 峰性のピークが立つ。

このストロークの方向分布  $h(\theta)$  を用いて文字列の筆記方向の検出を行うことが可能である。2 峰性のピークを見いだす相関関数の値  $f_1 \sim f_4$  を以下のように定義する。 $f_1$  から  $f_4$  までの値の中で最大の関数値を取る関数を選択し、それに対応した方向を検出結果とする。但し、 $g(\theta)$  を図 7.14(b') に示す。

$$\textcircled{1} \quad f_1 = \int_0^{2\pi} h(\theta)g(\gamma)d\theta \quad (\text{検出角 } 0 \text{ 度}) \quad (7.16)$$

$$\textcircled{2} \quad f_2 = \int_0^{2\pi} h(\theta)g(\gamma + 1/2 \pi)d\theta \quad (\text{検出角 } 90 \text{ 度}) \quad (7.17)$$

$$\textcircled{3} \quad f_3 = \int_0^{2\pi} h(\theta)g(\gamma + \pi)d\theta \quad (\text{検出角 } 180 \text{ 度}) \quad (7.18)$$

$$\textcircled{4} \quad f_4 = \int_0^{2\pi} h(\theta)g(\gamma + 3/2 \pi)d\theta \quad (\text{検出角 } 270 \text{ 度}) \quad (7.19)$$

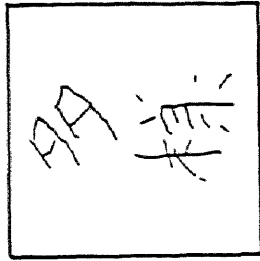
### (3) 実験結果

文章中からランダムに抽出した  $N$  文字から構成される文字列に対して、図 7. 10 で示した 8 種類の方向の手書き文字列を人工的に合成して、方向の判定実験を行なった。各  $N$  に対して 1600 個の文字列で実験を行なった。結果を表 7. 3 に示す。誤り判定の例を図 7. 11 に示す。3 文字以上から構成される文字列では正しく判定されることが確認された。

前節で分離した文字列の 1 例に対して、文字列の方向を正規化した結果を図 7. 12 に示す。

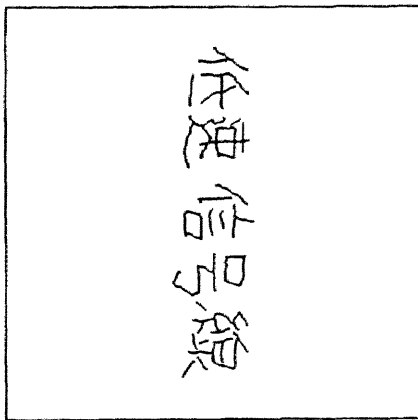
表 7. 3 方向の判定実験

N	判定率
2	99.6%
3	100.0%
それ以上	100.0%

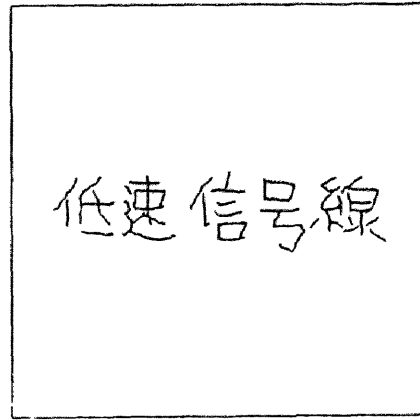


⑥を①と判定

図7. 1 1 誤判定の例



(a) 文字領域の分離結果



(b) 回転正規化結果

$$F1 = 11$$

$$F2 = 2$$

$$F3 = 10$$

$$F4 = 20 \quad \text{---} \rightarrow 270 \text{度の回転}$$

図7. 1 2 文字列の方向の正規化

#### 7. 4. 2 文字と図形の回転の正規化

前節では、文字列の筆記の方向は縦向きか横向きしか扱わなかった。しかし、ストローク方向ヒストグラムを利用する筆記方向の検出は更に発展させて、任意の方向か

ら筆記された文字列や図形を、正面から筆記された状態に変換する回転の正規化を可能にする。この節では、文字や図形の回転の正規化法について述べる。

従来の文字図形の認識手法では、文字や図形をタブレットから手書きで入力する際、タブレットに対し正面方向から筆記するという条件があり、条件を満たさなければ、文字や図形は回転して入力されるため正しく認識はできなかった。タブレットに対し任意の方向からの入力を許容すれば、この方式の操作性は大きく向上する。ここでは、文字や図形を構成するストロークには縦横成分が多いという性質および手書きの筆記方向特性を利用した文字や図形の回転正規化を行う前処理法を提案する。シミュレーション実験により良好な結果が得られた。

### (1) 回転変動と認識率

予備調査として、4章で述べた手法を用い、回転変動した手書き図形（フローチャートやブロック図100枚を回転したもの）に対する認識実験を行った。図7.13に回転変動と認識率の関係を示す。回転変動を±3度以内になれば、認識率の低下はほとんどみられない。ここでは±3度以内への回転正規化を目標とする。

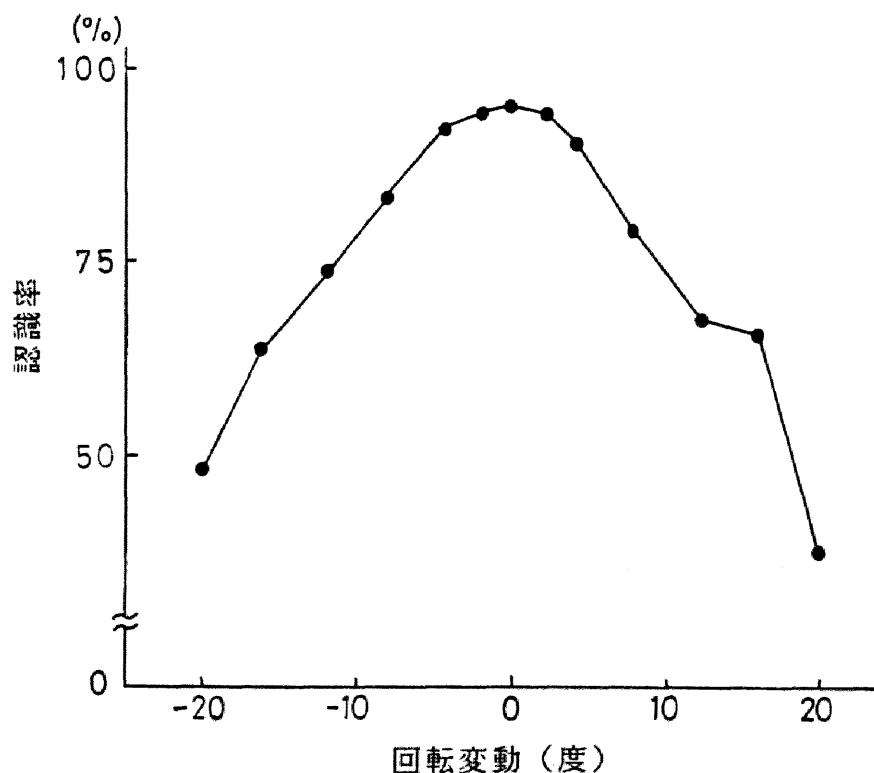


図7.13 回転変動と認識率

## (2) ストロークの方向分布

前節で述べたと同様に、入力ストロークを微小線素に分割し、方向 $\theta$ に関する各線素の頻度分布、つまりストロークの方向分布  $h(\theta)$  を導出する。入力ストロークが正しく正面方向から筆記された場合には、 $h(\theta)$  には上から下方向( $\theta=3/2\pi$ )と、左から右方向( $\theta=0$ )とに2峰性のピークが立つ。一般には、ピーク的位置は回転変動に比例して変位する。図7.14(a), 図7.15(a)に示す回転した図形と文字に対する  $h(\theta)$  をそれぞれ図7.14(b), 図7.15(b)に示す。

## (3) 回転方向の検出

$h(\theta)$  を用いて文字や図形の回転方向 $\gamma'$ の検出を行う。2峰性のピークを相関演算により見いだす関数  $f(\gamma)$  を定義する。 $f(\gamma)$  を最大にする $\gamma$ が検出方向 $\gamma'$ である。

$$f(\gamma) = \int_0^{2\pi} h(\theta)g(\gamma + \theta)d\theta \quad (7.20)$$

但し、 $g(\theta)$  は図7.14(b)に示す関数である。図7.14(a), 図7.15(a)に対する $f(\gamma)$ をそれぞれ図7.14(c), 図7.15(c)に示す。

回転の正規化は、原パターンを( $-\gamma'$ )回転させることにより、実現できる。

(4) 実験結果 回転した文字や図形3600枚(一例を図7.14(a), 図7.15(a)に示す。)を使用してシミュレーション実験を行った。±3度以内の精度で回転の正規化が成された率(正規化率)は99%であった。本手法の導入により、回転に対して自由な書式で手書きされた文字や図形の認識が可能となる。

### 7.4.3 文字列認識および図形認識

文字と図形の分離処理(7.3参照)により図面から分離された図形領域は、そのまま2~4章で述べた図形認識アルゴリズムにより認識することが可能である。一方、分離された文字領域については、文字列の方向の正規化処理(7.4.1参照)により正面方向からの筆記された縦書きあるいは横書きの文字列に変換することができる。この文字列データは5~6章で述べた文字列認識アルゴリズムにより認識することが可能である。

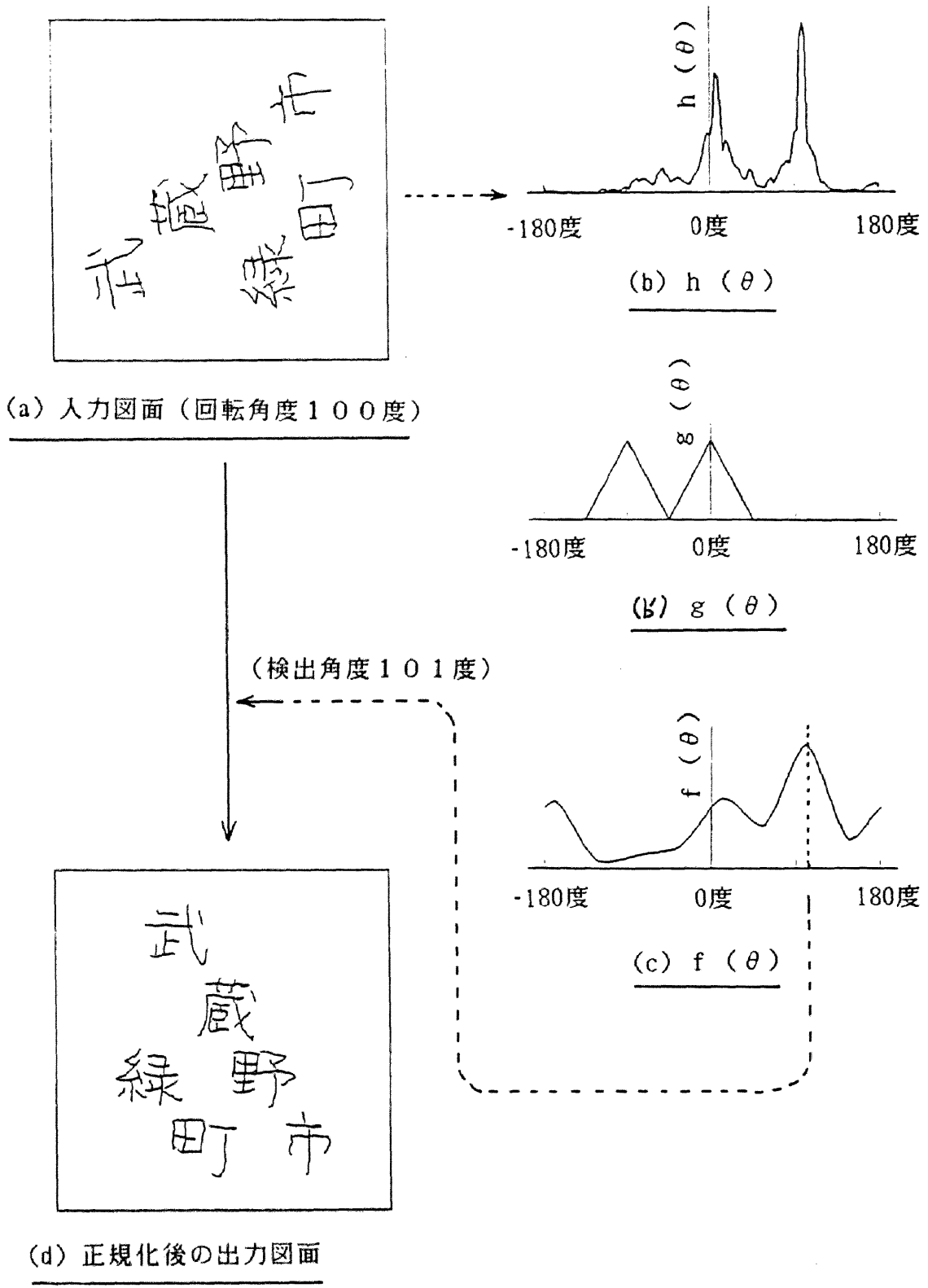
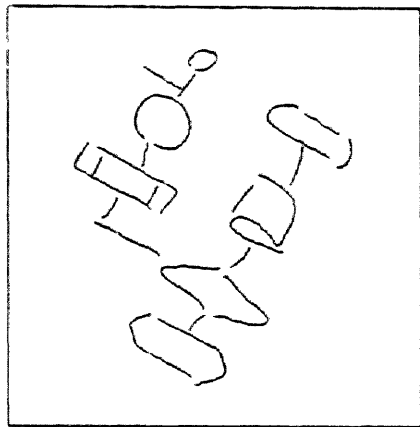
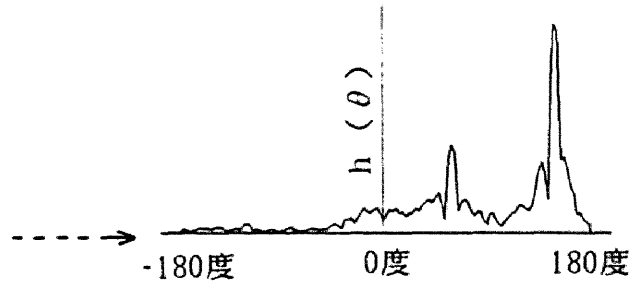


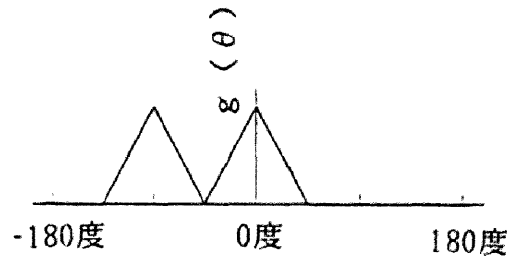
図7. 14 文字列の回転の正規化



(a) 入力図面 (回転角度 150度)

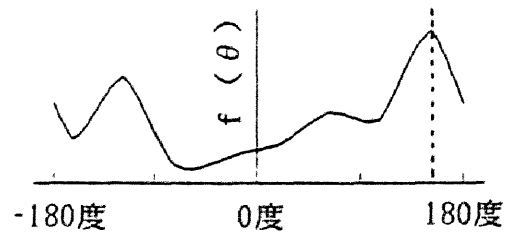


(b)  $h(\theta)$

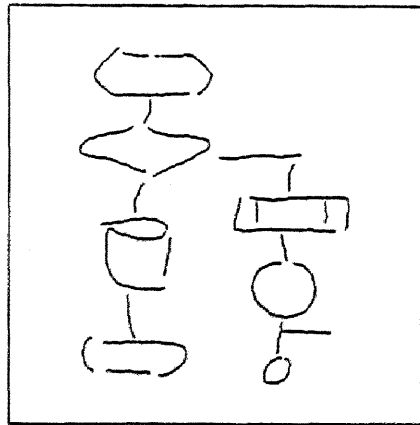


(b')  $g(\theta)$

(検出角度 152度)



(c)  $f(\theta)$



(d) 正規化後の出力図面

図7. 15 図形の回転の正規化

図7. 8に示した入力図面を、本認識システムにより認識した結果を図7. 16に示す。

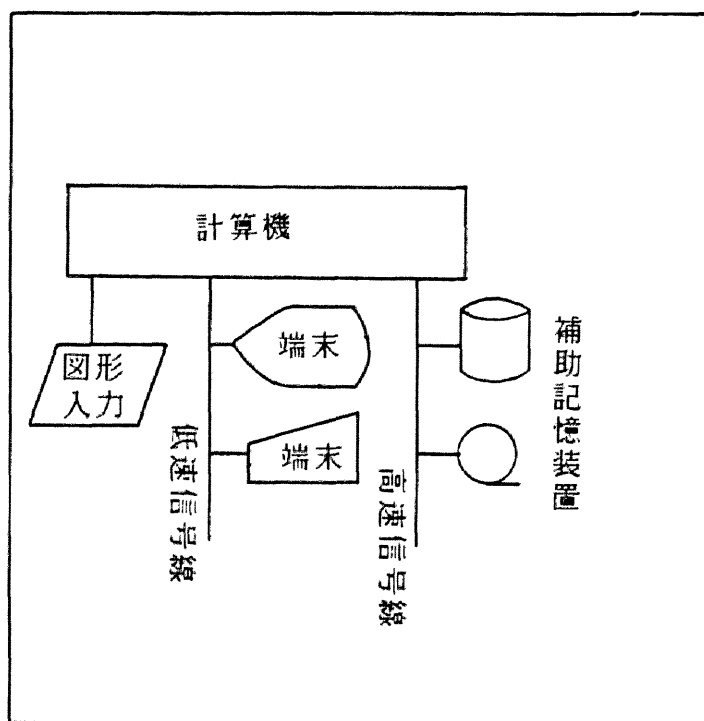


図7. 16 認識結果

### 7. 5 文字列認識専用処理装置

この章で提案した認識システムの第1段階から第3段階までのすべてをDN660のソフトウェアで実施し、その計算時間を検討した。図7. 3に示した図面に対しては、第1段階に30秒、第2段階に540秒、第3段階に110秒の計算時間がかかる。計算時間は入力図面によって異なるが、一般に第2段階の処理に計算時間の多くが費やされる。そこで第2段階の文字列認識処理を高速に実現するための専用処理装置を試作した。ハードウェア化した認識アルゴリズムは、5、6章で提案した「候補文字ラティス法」である。候補文字ラティス法の計算時間が多くなる理由は、文字を高精度に切り出す目的で、文字切り出しの多数の可能性から生成される候補文字の全てに対して文字認識処理を適用するためである。

本認識専用処理装置は、単に認識システムを高速化するのみでなく、円滑な文字列認識実験あるいは、大量データを利用した文字の分析を可能にする。本専用処理装置

では、処理の高速化を図るため、16台の演算ユニット（演算部プロセッサ）をパラレルに構成する。この節では、まず各処理の概要を述べ、試作したプロセッサについて述べる。

### 7.5.1 認識処理の装置化

候補文字ラティス法の処理手順を以下の①～⑨に示す。この処理手順は、文字列の前処理（②～⑤）、候補文字ラティスの生成（⑥⑦）、ラティスの探索（⑧）、の3段階に大分類できる。各部の処理量を分析する。入力文字列によって処理量は異なるので、ここでは5文字から構成される「情報処理に」を例に取って説明する。この例では、候補文字が49文字生成された。つまり、5文字を認識するのに49文字の文字認識処理が必要となる。その結果、DN660では、前処理に1秒、候補文字ラティスの生成に120秒、ラティスの探索などその他の処理に1秒（言語処理は除く）の計算時間がかかるという結果となった。

そこで、処理量は多いが処理内容は単純な「前処理」と「候補文字ラティス生成処理」を専用処理装置に割り当て、処理の制御が複雑な「ラティスの探索」は汎用計算機で行なう構成を実現する。

ここで専用処理装置で行なう箇所を括弧で示す。

- ①座標値入力：データを読み込む
- ②【文字列の大きさの正規化】：文字列の幅が一定になるように拡大縮小を行なう
- ③【特徴点抽出】：ストロークを等間隔の点系列に変換する
- ④【基本セグメントへの分割】：文字列を偏やつくりなどの細かい図形に分解する
- ⑤【候補文字の生成】：基本セグメントを組み合わせて、候補文字を生成する
- ⑥【マッチング】：候補文字の大きさを2次モーメントで正規化、  
標準パターンとの相異度計算、その結果のソーティングを行なう
- ⑦【候補文字ラティスの生成】：全ての候補文字のストローク番号と、認識結果、  
相異度を表に登録する
- ⑧ラティスの探索：相異度の和が最小となる文字系列を探索する。
- ⑨結果出力：探索した文字系列を出力する。

### 7.5.2 アルゴリズム上での処理の高速化



本認識処理をハードウェア化するにあたり、認識アルゴリズムを見直し、認識アルゴリズムの本質を変更しない範囲で、アルゴリズムの最適化を図った。以下に改良点を述べる。

(1) 2、3、5、9、17点近似のストローク間マッチング法

ストローク形状を少ない情報量で最適に表現するためには、長いストロークは多数の近似点数で、短いストロークは少数の近似点数で、と点近似するような手法が望ましい。しかし、近似点数を可変にする方法では、入力パターンと標準パターンとの間で距離を計算する際、異なった点で近似されたストローク間で、マッチングを取る必要が生じる。そのため、DPマッチング、あるいはその都度ストロークを同じ点数で近似しなおす、等の処理が必要となるが、これらの処理はいずれも極めて計算量が多い。ここでは近似の点数を2、3、5、9、17にして、点の間引きにより点数の整合が取れる点近似法を採用した(図7.17参照)。

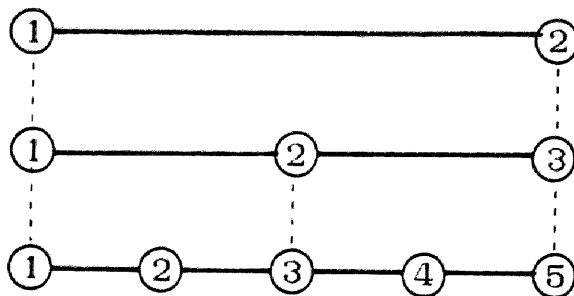


図7.17 点の対応付け

(2) 簡易DPマッチング

ストローク間の相異度の計算にDPマッチングを採用する目的は、異なる点数から構成されるストローク間の相異度を求めること、変形を点の対応を変えることによって吸収することである。前者は、上記(1)の手法で解決される。後者については、実際の点の対応のずれは1点以下が多いことを考え、ここではストローク間の距離を以下のdで定義する。

$$d = \sum_i \min (g(i-1, i), g(i, i), g(i+1, i))$$

但し、 $g(i, j)$  は一方のストロークの点  $i$  と、他方のストロークの点  $j$  との間のユークリッド距離である。この相異度尺度  $d$  を、ここでは簡易 DP マッチングと呼ぶ。

### 7.5.3 専用処理装置のハードウェア構成

処理の高速化を実現するために、本専用処理装置では演算部を並列に構成した。本装置はシステム制御部と16台の演算部プロセッサから構成されている。ハードウェアのブロック図を図7.18に示す。また、本専用処理装置の主な仕様を表7.4に、演算部プロセッサ1台分の仕様を表7.5に示す。その写真を図7.19に示す。

文字列認識処理で最も処理量の多い箇所は、候補文字の認識処理である。候補文字の認識処理では、候補文字と標準辞書（パターン）とのマッチングを多数行なう必要がある。本専用装置では、標準辞書を16台の演算部プロセッサに分担して格納し、16台で同時にマッチング処理を行なうことが可能である。

本装置の動作順序を簡単に説明する。本装置はドメインDN660とGPIBで接続されている。まず初期設定として、標準辞書をDN660から標準パターンメモリーに転送する。標準パターンメモリーは最大1万6千文字分の容量を有している。常用漢字の字種は約2千字であるが、残りのメモリーは変形した文字を辞書に登録する際に利用できる。次にIPLにより、標準辞書を各演算部プロセッサに分割する。認識動作では、入力した文字列をまず1台の演算部プロセッサで文字列の大きさの正規化、特徴点抽出、基本セグメントへの分割、候補文字の生成を行なう。生成された候補文字のデータは他の15台の演算部プロセッサに転送され、自分を含めた16台の演算部プロセッサにより、標準辞書の各文字と候補文字との間の相異度が計算される。制御部プロセッサは、各演算部プロセッサから出力される相異度のソーティングを行ない、候補文字認識結果を計算する。その結果はGPIBの回線を利用してDN660に転送される。DN660は、その結果をもとに文字列認識結果を計算する。

本装置は、単に認識のデモシステムを構成するだけでなく、認識アルゴリズムの改良や、手書き文字列の分析に利用できるように、ある程度の汎用性を持たせてある。パラメータの変更も容易に行なえるようになっている。これらは、DN660からのコマンドにより制御が可能である。コマンドの主な仕様を表7.6に示す。

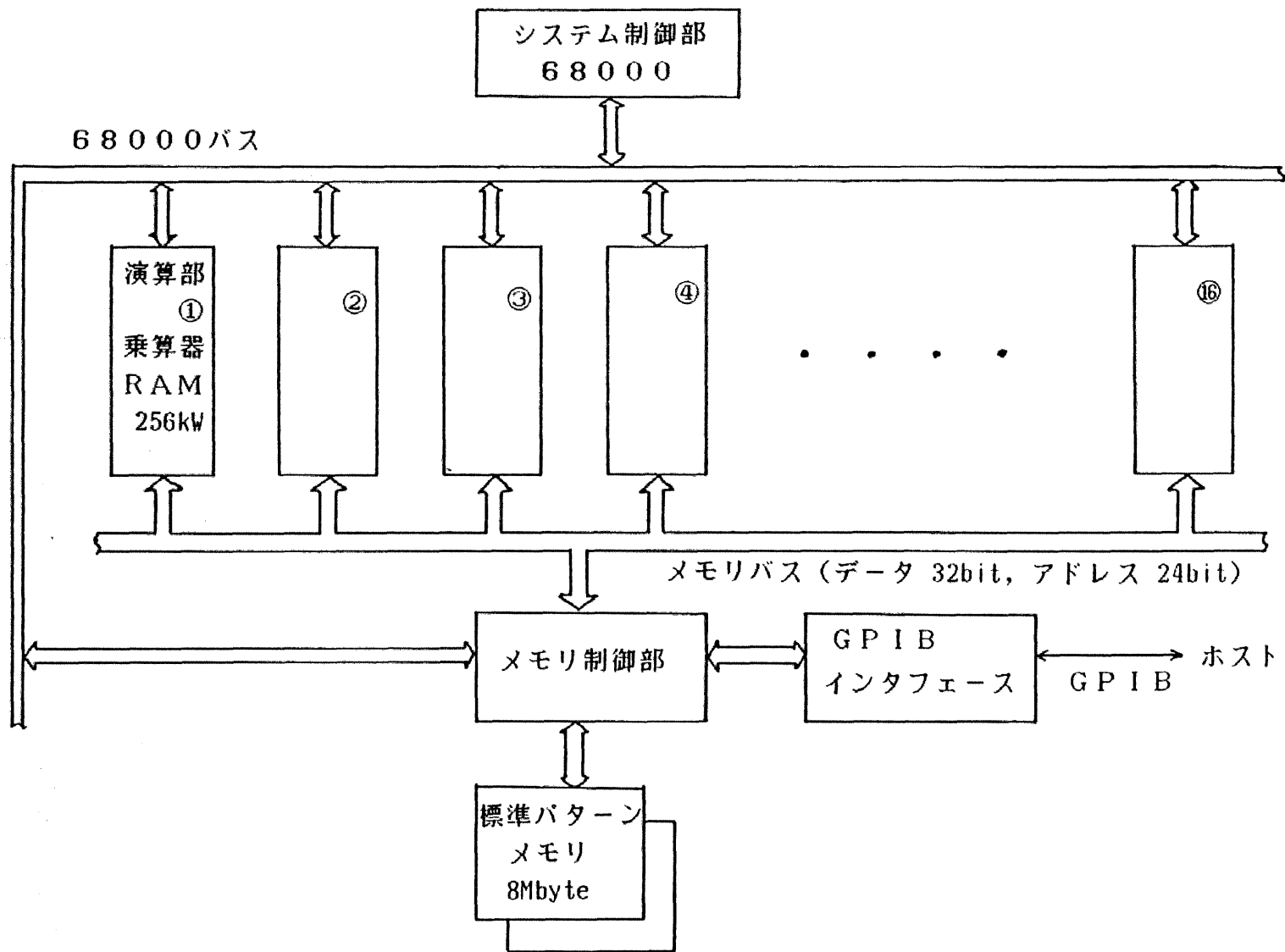
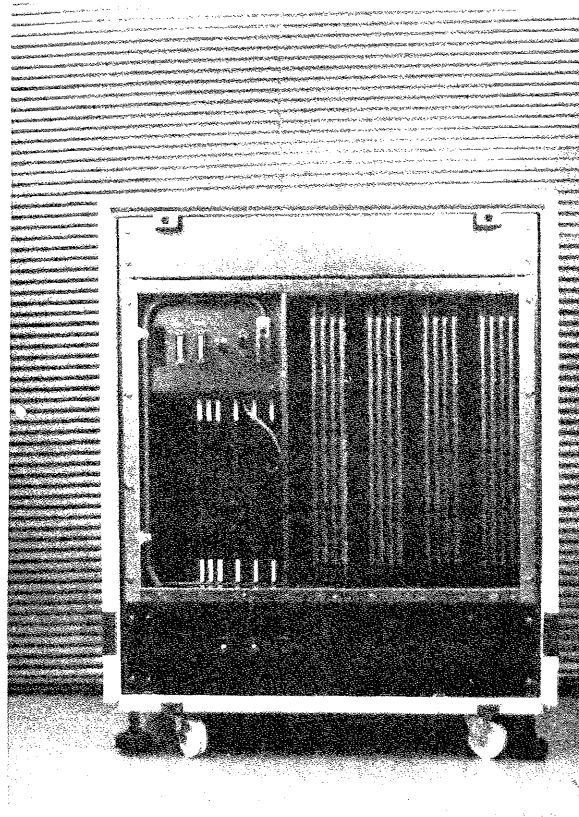
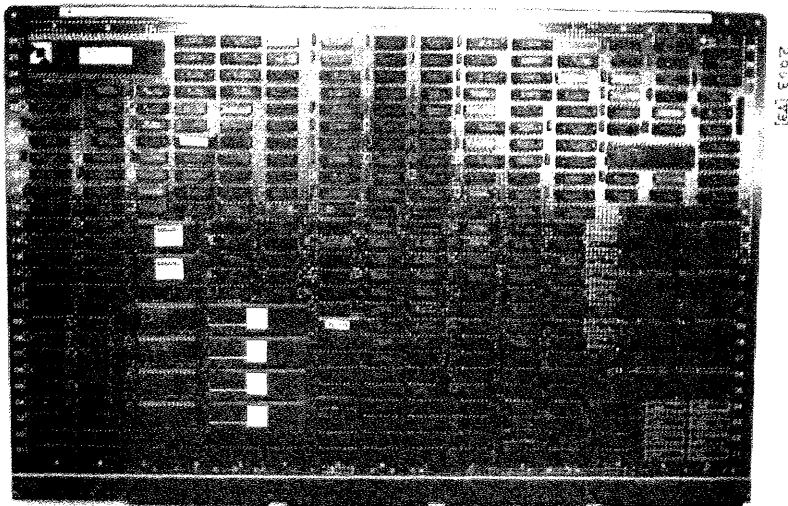


図7. 18 専用処理装置のブロック図



(a) 全体像



(b) 演算部プロセッサ

図7.19 専用処理装置

表7.4 専用処理装置の主な仕様

システム制御部	CPU	68000
	語長	16Bit
	サイクルタイム	750nS
演算部プロセッサ	16台をパラレルに動作させる	
標準パターンメモリ	構成	32Bit, 1WORD構成
	容量	8Mbyte
インターフェイス	GPIB1	DMA (入力データ/結果のデータ)
	GPIB2	プログラム転送 (コマンド/ステータス)

表7.5 演算部プロセッサの主な仕様

CPU	2903を4個 (ビットスライスプロセッサ)	
演算語長	16Bit並列	
プログラム語長	64Bit	
レジスタ	汎用レジスタ	16Bit, 32個
	インデックス	16Bit, 16個
内部メモリ	プログラムメモリ	64Bit, 8KW
	ワークメモリ	16Bit, 16KW
	辞書メモリ	16Bit, 256KW
サイクルタイム	250nS	

表7.6 コマンドの主な仕様

標準パターンの転送	標準パターンを転送 標準パターン用メモリーは約1万6千文字分の容量を持つ
IPL	プロセッサの初期化コマンド、 各演算部への標準パターンの分割、 マイクロプログラムのローディング
パラメータ設定	以下に示す処理上のパラメータを設定できる。 ①文字列の大きさの正規化：大きさ ②特徴点抽出：刻みの間隔 ③基本セグメントへの分割：分割の大きさ ④候補文字の生成：生成の条件（大きさなど） ⑤マッチング： 標準パターン（漢字、平仮名、片仮名、英数字）の選択 例えば、「漢字」AND「平仮名」 距離尺度の設定（二乗距離 または 絶対値距離） （単純距離 または 簡易DP距離） ⑥候補文字ラティスの生成： ラティスの各ブランチの深さの設定 つまり、候補文字の認識結果を何位まで取るかの設定 ⑦ストロークのマッチング処理だけを行なう機能の設定
処理起動	入力データを転送と処理装置の起動
結果転送指定	ホスト計算機に結果が転送
ユニットメモリREAD	演算ユニットのメモリーの内容をホスト計算機に転送する
エラーREAD	処理時のエラー情報（例えばメモリーのエラー等）を転送
ハードウェアRESET	電源投入時の状態にRESET

#### 7.5.4 専用処理装置の能力

汎用計算機（1MIPS）に比較して、本専用処理装置により処理能力は飛躍的に向上した。一例として、5文字から構成される文字列「情報処理に」を入力し、その処理時間を測定した。実測結果を表7.7に示す。この結果から分かるように処理の大半は候補文字の認識処理（マッチング処理）に費やされている。本処理装置では、このマッチング処理を16台の演算プロセッサで完全に並列に実行している。1MIPSの汎用計算機との単純な比較では、このマッチング部分については約55倍の認識速度が達成された。1台の演算プロセッサの能力は約3.5MIPSとなる。前処理と、コマンドの転送処理等を加えた場合には、約52倍の処理速度が達成された。

表7.7 専用処理装置の処理時間（例「情報処理に」の場合）

	本専用処理装置	汎用計算機（1MIPS）
コマンド転送 [入力データの転送] [結果の転送]	0.200S	0.0S
前処理 [文字列の大きさの正規化] [特徴点抽出] [基本セグメントへの分割] [候補文字の生成]	0.014S	1.0S
候補文字の認識 [マッチング] [候補文字ラティスの生成]	2.152S	120.0S
合計	2.366S	121.0S

## 7.6 結言

この章では、自由書式で筆記された文字・図形混在図面の、認識実験システムについて検討した。文字と図形が混在した図面を認識するために、3段階から構成される処理を提案した。第1段階では、図面認識のための前処理として、まず図面を文字領域と図形領域に分離し、次に文字列の方向の正規化を行なう。第2段階では、5章～6章で提案した認識アルゴリズムを用いて文字列の認識を行なう。第3段階は、2章～4章で提案したアルゴリズムを用いて図形領域の認識を行なう。この章では、文字と図形が混在した図面の認識システムを構成するために新規に必要な技術である文字と図形の分離処理と、文字列の正規化処理について詳細に述べた。更に、高速な認識実験システムを構成するために、文字列認識の専用処理装置（専用ハードウェア）を試作したので、それについて述べた。

以下に、結果をまとめる。

(1) 図面中の文字や図形を構成するストロークの統計的な性質を明らかにした。この性質を利用して文字と図形を自動的に分離する手法を提案した。この分離法は、ストロークの大きさ情報およびストローク間の近接関係を利用したもので、2段階から構成される。まず、入力ストロークを、文字らしい領域、図形らしい領域、文字か図形か不確実な領域に分類する。次に、文字らしさあるいは図形らしさを表現する評価関数を導入して、この評価関数を最大にするように不確実な領域を文字あるいは図形に判定する。本手法により、精度99.4%で、文字と図形を分離することが可能となった。

(2) 文字列の方向（縦方向／横方向）と筆記方向（縦書き／横書き）を検出し、正規化する手法について述べた。文字列の方向については、筆記点の分布を計算し、分布の広がり最大の方向を文字列の方向とする。一方、筆記方向については、ストロークの方向分布を利用して検出する。文字や図形を構成するストロークには一般に縦横成分が多く、更に縦ストロークはペンを上から下へ移動させ、横ストロークはペンを左から右へ移動させて筆記することが多い。この性質を利用し、ストロークの方向に関する頻度分布のピーク位置により筆記方向を検出する。筆記方向が検出できれば、回転処理により文字列を正規化することが可能である。提案の手法により、3文字以上から構成される文字列に対して、100%の精度で文字列の方向と筆記方向を検出できることを確認した。



(3) 認識実験システムを効率的に構築するために専用処理装置を試作した。本論文で述べた文字列認識アルゴリズムでは、セグメンテーションと文字認識を融合してセグメンテーション精度を向上させているため、多数の文字認識（候補文字認識）回数が必要になる。そのため、汎用計算機上でソフトウェアで実現したのでは膨大な計算時間が必要となる。これに対処するために16台の演算装置を並列に実行する専用のハードウェアを構成した。これにより、文字列認識を高速に実現することが可能になり、1MIPSの計算機上での実行に比較して約50倍の候補文字認識速度が実現できた。

残された課題を以下に述べる。

(1) 今回構成した文字・図形混在認識システムは、実験システムであり、実際の図面作成の目的には、不備な点が多い。例えば、誤認識された結果の訂正機能や、編集機能などは、不可欠である。これらの機能を追加して、文字・図形入力システムを構成する必要がある。また、手書きのオンライン入力システムをマンマシンインタフェースの観点から評価することは今後の課題である。

(2) 今回提案した文字と図形の分離手法では、図面中で文字と図形の接触を許していない。筆記制限を更に解除し、文字と図形の接触を許容する文字と図形の分離手法の開発が望まれる。

(3) 認識対象の拡大は重要である。図形領域については、フローチャートのような規則の強い図形以外に、規則の弱いイラスト等のラフスケッチまでも認識すること、また文字列認識については、画数の固定で丁寧に筆記された文字列から、より一般の筆記に近い走り書き文字列を認識することなどがあげられる。

本章では、文字・図形混在図面認識の為の実験システムを構築した。ここで提案したシステムや手法により、自由書式で筆記されたフローチャートやブロック図などの文字・図形混在図面を、高速かつ精度良く認識するシステムを構成できる見通しが得られた。

## 第 8 章 結 論

本論文は、手書きの文字や図形をオンライン認識する手法について述べたものであり、著者が昭和55年以来N T T 武蔵野電気通信研究所で行なってきた研究をまとめたものである。

手書きの文字や図形の認識を利用した計算機への文字図形情報の入力、将来のマシンインタフェースとして極めて有望である。しかし、本研究以前の認識技術は、文字部分の認識に関しては、固定枠内に筆記された文字を1文字だけ認識する技術であり、図形部分の認識に関しては、各シンボルは定められた画数や筆順で筆記する必要がある、更にシンボル間のセグメンテーションの指示を人間が行なう必要があった。そのため、手書き文字図形認識による、文字図形情報の入力は扱いにくいものであり、その応用範囲も限られたものであった。本論文では認識対象を自由な書式で筆記された文字列や、図形を含む図面等に拡大し、その範囲で議論を行なった。本論文の主眼点は、文字や図形を筆記する際の書式に関する制限あるいは画数や筆順に関する制限を解除することと、それと同時に高精度な認識系を実現することにある。本論文は8章から構成され、まず図形部分の認識に関する基本技術の検討(2、3、4章)、次に文字部分の認識に関する基本技術の検討(5、6章)、最後に文字・図形の混在図面の認識システム(7章)から組み立てられている。以下に本研究で得られた結果を要約する。

第1章では、機械による手書き文字・図形認識について概説するとともに、これらの研究の動向を述べ、本論文の目的を明確にした。特に、文字に加えて図形をも高精度に認識することの必要性や、筆記制限を緩和することの重要性を述べた。

第2章では、基本図形(フローチャート等の図面を構成する個々のシンボル)の筆記特性に関する分析、およびその認識手法について検討した。図形を認識する際には文字の場合と異なり、決まった画数と筆順は存在しない。これに対処するため、入力ストローク列に対応した標準ストローク列を自動的に生成し、入力図形と標準図形との間で整合をとる画数・筆順自由整合法を提案した。この手法により、1種類の標準シンボルを登録するだけで、任意の画数と筆順で筆記された基本図形を認識することが可能となった。その結果720個の基本図形に対して97.3%の認識率が達成された。また、ストロークの方向と位置を用いたDPマッチングをストローク間の相異

度尺度とするパターン間整合法を提案した。DPマッチングを使用しない場合の認識率95.0%に比較し、本手法では97.3%の認識率が得られ、DPマッチングが手書き変形に強いことを示した。

第3章では、基本図形（シンボル）群とそれらを接続する直線群が複合して構成される手書き図形（例えば、フローチャート、ブロック図、論理回路図など）を対象に、そこからシンボルや直線を自動的にセグメンテーションし同時に認識する手法、つまり自由書式で筆記された図形の認識法について検討した。この手法では、まず画数・筆順自由整合法により入力図形中から各シンボルに対する候補図形を抽出する。この手法により99.6%の候補図形抽出率が得られた。次に候補図形をラティス（2端子グラフ）で表現し、ラティスを探索することによりシンボルのセグメンテーションと認識とを同時に実現する。この手法を候補ラティス法と名付けた。この手法を利用すれば、図形全体として最適な解釈を認識結果とすることができ、セグメンテーションの高精度化が達成された。100枚の図形データに対して89.8%の認識率を得ることができた。候補ラティス法によりシンボルの切り出し位置を指示する筆記時の制限が解除された。

第4章では、シンボル間の接続ルールを知識として利用する高精度な手書き図形認識について検討した。第3章の手法で得られた認識結果について、シンボル間の接続関係を記述し、接続関係が予め登録した接続ルールに矛盾する場合にはそれを修正する手法を提案した。この手法の導入により、包含関係を持つシンボル（フローチャート等に含まれるシンボルには、あるシンボルが他のシンボルの一部となるような包含関係をもつシンボルが多数ある。）のセグメンテーションや、形状が類似したシンボルの認識に対して、良好な結果が得られることを示した。接続ルールの導入により89.8%の図形の認識率は96.1%に向上した。

第5章では、白紙紙面上あるいは罫線上に自由書式で筆記された文字列を認識する手法を検討した。手書きの場合文字ピッチの変動が大きく、左右の文字が接触する場合もある。更に、日本語には偏とつくり等、左右に分離する文字が多数存在する。そのため、単に図形的なまとまりだけで文字をセグメンテーションするわけには行かない。そこで、文字のセグメンテーションと個々の文字の認識を融合した候補文字ラティス法を提案し、その手法を評価した。この手法により、自由書式で手書きされた日本語文字列から文字をセグメンテーションし認識することが可能となった。その結果、105個の自由書式文字列に対して96.3%の文字列認識率が達成された。これに

より、文字を枠内に筆記する必要がなくなることを示した。

第6章では、言語処理、文字認識、文字のセグメンテーションの上位概念から下位概念までを統合した文字列認識の為の認識系を提案し、文字認識や言語処理が文字のセグメンテーション処理に与える効果について述べた。文字認識の情報だけを文字のセグメンテーションに利用した候補文字ラティス法(5章の手法)では、文字が偏とつくり等に分離しそれらが別の文字になる場合(この文字を分離有意文字と名付けた)には正しく認識されない場合が多い。例えば、「説明」が「説日月」と誤認識される。常用漢字と平仮名の全字種を分析したところ、分離有意文字は9.7%存在することを確認した。これらの分離有意文字に対処するため、言語情報を文字のセグメンテーションに効果的にフィードバックする手法を提案した。本手法により、文字のセグメンテーションの高精度化が実現でき、文字のセグメンテーション率は、言語情報を導入しない場合の97.0%から99.2%に向上した。更に、本手法は文字認識の後処理的な効果もあり、文字認識率は99.1%から99.8%に向上した。

第7章では、文字・図形の混在した図面の認識システムについて述べた。そこで必要となる、文字・図形の分離手法、文字列の方向の正規化手法を検討し、更に文字列認識処理を高速化するために試作した専用処理装置について述べた。まず、ストロークの長さや形状に関する性質を利用した文字・図形の分離手法を提案した、本手法により、99.4%の文字図形分離率を得ることができた。次に、筆記時の筆点の方向特性を利用した文字列の方向の正規化手法を提案した。更に、第5章で述べた文字列認識処理では、種々のセグメンテーションの可能性から生成される多数の候補文字を認識するため、多くの計算量が必要となる。処理を効率的に行なうため16台の演算装置を並列に設置した専用処理装置を試作し、1MIPSのミニコンの約50倍の処理速度を実現した。

本研究により、以下に示す2つの大きな成果が得られた。

第1の成果は、オンライン手書き手書き文字・図形の認識における筆記制限の緩和である。筆記制限の中で筆記者に最も負担を感じさせるものが、画数や筆順の制限と、固定された書式による制限であるという立場から、これらの制限を緩和した認識手法を提案した。この手法により、筆記制限が少ない認識系を実現することができた。本研究により解除した具体的な筆記制限は、図形部分の認識においては各基本図形をシステムが定めた筆順・画数で筆記すること、基本図形ごとに切り出し位置を指示する

ことであり、文字部分の認識においては文字の枠内筆記の制限等である。これらの筆記制限の解除は筆記者の精神的かつ肉体的な疲労を低減するのみならず、この技術が文字認識の応用範囲の拡大に果たす役割は大きい。第1の成果は工学的に非常に価値の高いものである。

第2の成果は、文字や図形を高精度に認識するために知識の利用の有効性を示し、その具体的な実現手法を提供したことである。人間は、文字を読み取る際に種々の知識を利用する。単独の文字を見ただけでは読み取れない文字も文章として見れば読むことができる。また、図形的なまとまりとして見ればどこが文字の切れ目か分からない文字列も、文字の概念を導入することにより切れ目ははっきりする。例えばアラビア文字を知らない人間はアラビア文字のセグメンテーションが出来ないことから理解できる。知識の利用は、文字だけでなく、人間がフローチャートや更にはイラストのような不定形状の図形を認識する際にも同様になされている。これらの処理は人間は殆ど無意識に行なっているが、これらの処理を機械上で実現することは困難なことであった。本論文ではこれらの知識の利用を汎用的かつ統一的に実現する手法を提案した。これにより、文字切り出しにおける文字認識の利用、文字列認識における言語情報の利用、図形認識におけるシンボル間の接続ルール利用などが可能となり、文字や図形の機械認識における上位概念利用の有効性を具体的に示すことが可能となった。第2点目の成果は認知科学の分野と関係するとともに、人工知能の立場からも興味ある内容である。

本研究を通じて明らかになった事実は、手書きのマンマシンインターフェースを実現する際に多数の貢献をするものと確信する。

残された課題を以下に示す。

(1) まず、認識対象の拡大である。第1章で述べたように、本論文では図形領域についてはフローチャートのような規則の強い図形(定形図形)に限ったが、より規則の弱いイラストなどのラフスケッチ(不定形図形)の認識は困難な課題である。

(2) 文字部分については、丁寧に筆記された文字列(筆順は自由、画数は登録方式)を対象としたが、一般の筆記に近い走り書き文字列の認識は重要な課題である。

(3) 更に、本論文では認識システムとしては、実験システムに留まっているが、編集機能や訂正機能を追加した文字図形入力システムを構成し、手書き入力システムをマンマシンインターフェイスの観点から考察することなどが必要である。

本論文の成果が、手書き文字・図形認識の発展に寄与し、真に情報処理の自動化や機械化に役立つこと、またパターン認識の原理的研究の一つとなることを期待する。

## 謝 辞

本研究遂行には、多数の方々からの御支援と御協力を賜わった。本研究の遂行にあたり、多大の御理解と御支援を頂いたNTT（日本電信電話株式会社）基礎研究所長・加藤嘉則博士、基礎研究所・情報通信基礎研究部長・島田禎晋博士、情報通信基礎研究部統括・塚本克治博士、東京工科大学教授（前 情報通信基礎研究部長）畔柳功芳博士に厚く感謝します。著者のNTT入社当時の研究室長であり、討論を通して研究の進め方を教示して頂いたセコム株式会社セコムテクニカルセンター・セコム研究所長（前 画像通信研究部長）橋本新一郎博士に深謝します。また、情報通信基礎研究部・第三研究室長・増田功博士、ならびにATR視聴覚機構研究所・視覚研究室長（前 NTT基礎研究所・主幹研究員）梅田三千雄博士には、本研究の開始当初から現在に至るまで長期間に渡って、限りない御指導と御助言を賜わり、心から感謝の意を表わします。

本論文のまとめるにあたり、数々の御懇切な御教授を賜わり、日頃からも御指導を頂く名古屋大学工学部・福村晃夫教授に心から感謝の意を表わします。また、本論文に関して適切な御批判と御指導を賜わった名古屋大学工学部・鳥脇純一郎教授、吉田雄二教授に厚く感謝します。更に、豊橋技術科学大学・本多波雄学長には、著者が名古屋大学在学中に、研究の方法など多くのことを学ばせて頂きました。三重大学工学部・三宅康二教授、木村文隆助教授、鶴岡信治助手、ならびに聖徳女子短期大学・吉村ミツ教授は、著者が大学在学中から現在に至るまで、パターン認識と文字認識に関する多くの知識を著者に伝え、本研究遂行上に多大な影響を与えました。特に、吉村ミツ教授には、本論文をまとめる際にも種々の懇切丁寧な注意や指導を頂きました。ここに記して厚く御礼を申し上げます。

また、NTT基礎研究所・情報通信基礎研究部総括補佐・小高和己博士、同研究部・第三研究室主任研究員・若原徹博士には、直接の上司として、本研究の開始当初からあらゆる面に於いて御指導、御激励を賜わりました。第三研究室研究主任・新谷幹夫氏には本論文の第6章で使用した言語処理部の開発で協力して頂きました。同研究室主任研究員・高橋時市郎氏、NTT電子機構技術研究所・機構技術研究部主任研究員・目黒真一氏には本研究を進めるにあたり適切な助言を頂きました。更に、基礎研究所担当部長・内藤誠一郎博士、第三研究室主任研究員・萩田紀博博士、同主任研究員・秋山照雄博士、ならびに第三研究室の諸氏には、日頃から有益な討論をして頂き

ました。ここに厚く感謝致します。最後に、文字や図形データの収集、およびアンケートに御協力して頂いた所内の諸氏に厚く感謝します。



## 参考文献

- (1) 橋本編著：“文字認識概論”，電気通信協会。
- (2) 佐藤、棟上：“自由手書き図面理解の基礎実験”，12回画像工学コンファレンス，7-3，1981。
- (3) S. Shimizu, S. Nagata, S. Inoue, M. Yoshida: "Logic Circuit Processing System", ICPR, pp.717-719, 1982.
- (4) 恒川、吉野、森、岡崎：“手書き図面読み取り装置”，コンピュータビジョン，25-1，1983。
- (5) 全、恩田、小沢：“手書き図面における図形部抽出の一方法”，信学総全大、1423，1983。
- (6) 安居院、飯塚、金子：“自由手書き論理回路図の認識”，信学総全大、1430，1983。
- (7) 名倉、末永：“文字認識機能を導入したファクシミリベースの手書き図面清書法”，信学技報、IE81-12，1981。
- (8) 林、吉田、福村：“高速テンプレートマッチング法と気象図への応用”，PRL78-90，1979。
- (9) 谷口、遠藤、田町：“ミニコンによる天気図認識システム”，電気学会、研究会資料、IP79-37，1979。
- (10) 青山、棟上：“印刷楽譜の自動読み取り”，PRL82-5，1982。
- (11) T.L.Diamond: "Devices for reading handwritten characters", in Proc. Eastern Computer Conf., pp.232-237, 1957 .
- (12) L.D.Harmon: "Handwritten reader recognizes whole word", Electronics, August, 24, pp.29-31, 1962 .
- (13) M.Eden: "Handwriting and pattern recognition", IRE Trans. Information theory, August, pp.160-166, 1962.
- (14) I.F.Sutherland: "Sketchpad, a man-machine graphical communication system", SJCC, pp.329-346, 1963.
- (15) T.Marrill et al.: "CYCLOPS-1: A second generation recognition system", in Proc. FJCC, pp.27-33, 1963.
- (16) R.M.Brown: "On-line computer recognition of handprinted characters", IEEE

- Trans. Electron Comput., Vol13, pp.750-752, 1964.
- (17)W.Teitelman:"Real time recognition of handdrawn characters", in Proc. FJCC, pp.559-575, 1964.
- (18)P.Mermelstein:"Experiments on computer recognition of connected handwritten word", Information and Control, Vol7, pp.255-270, 1964.
- (19)G.F.Groner:"Real time recognition of handprinted text", in Proc. FJCC, pp.591-609, 1966.
- (20)G.F.Groner et al.:"On-line computer classification of handprinted Chinese characters as a translation aid", IEEE Trans. Electron. Comput., Vol.EC-16, pp.856-860, 1967.
- (21)C.C.Li et al.:"On recognition of handwriting Chinese characters", in Proc. 1st Princeton Conf. on Information Science and Systems, pp.235-239, 1967.
- (22)W.R.Nugent:"The on-line recognition of cursive writing using geometric-topological invariants of stroke succession", 1st Annual IEEE Computer Conf., pp.145-148, 1967.
- (23)G.M.Miller:"On-line recognition of hand-generated symbols", in Proc. FJCC, pp.399-412, 1969.
- (24)S.Hanaki et al., "Real time recognition of handprinted characters - Japanese character 'Katakana' recognition", NEC Research and Development, No.14, pp.80-86, 1969.
- (25)萩原、奥川、柏岡："手書き過程による文字の識別性について"、信学研資、A68-221, IT68-23, 1968.
- (26)藤崎、永井："手書き数字のオンライン認識の一方法"、信学総全大、124, 1971.
- (27)H.Fujisaki et al.:"On-line recognition of hand-written numerals", Annual Report of Engineering Research Institute, Faculty of Engineering, University of Tokyo, Vol.30, 1971
- (28)柏岡、岩岡："手書き漢字のオンライン識別"、信学研資、A71-119, 1972.
- (29)V.M.Powers:"Pen direction sequence in character recognition", Pattern Recognition, Vol15, pp.291-302, 1973.

- (30)古川、三田：“手書き漢字のオンライン識別”、信学研資、PRL73-22, 1973.
- (31)荒川、増田：“手書き英数字のオンライン認識”信学研資、PRL72-68, 1972.
- (32)寺井、中田：“手書き漢字・片仮名のオンライン実時間認識”、信学論(D)、Vol.56-D, No.5, pp.312-319, 1973.
- (33)寺井、中田：“オンライン手書き文字認識を用いた原稿校正システム”、情報処理、Vol.15, No.6, pp.419-427, 1974.
- (34)牧野、木沢：“グラフィックディスプレイを用いた漢字認識”、信学総全大、1297, 1973.
- (35)荒川、増田：“手書き文字のオンライン認識”、信学論(D)、Vol.J59-D, No.11, pp.809-816, 1976.
- (36)藤原、池田、富永、清野：“接線ベクトル列を用いたオンライン手書き文字の認識”、情報処理、Vol.17, No.3, pp.191-199, 1976.
- (37)小高、荒川、増田：“ストロークの点近似による手書き仮名、漢字のオンライン認識”、信学論(D)、Vol.J63-D, No.2, pp.153-160, 1980.
- (38)小高、若原、増田：“筆順に依存しないオンライン文字認識アルゴリズム”、信学論(D)、Vol.J65-D, No.6, pp.679-686, 1982.
- (39)W.C.Lin:“Machine recognition and plotting of hand-sketched line figures”, IEEE, Trans. of SMC-8, pp.52-57, 1981.
- (40)加藤 他：“相対位置行列を用いた実時間手書き文字認識”、信学総全大、1308, 1979.
- (41)北原、磯道：“方向余弦DPマッチングによる手書き常用漢字オンライン認識システム”、信学論(D)、Vol.J64-D, No.11, pp.1013-1020, 1981.
- (42)E.F.Yhap et al.:“An on-line Chinese character recognition system”, IBM.Res.Develop., Vol.25, No.3, pp.187-195, 1981.
- (43)高橋 他：“オンライン手書き文字認識システムを用いた日本語文章作成システムについて”、情処全大、11-6, 1981.
- (44)C.C.Tappert:“Cursive script recognition by elastic matching”, IBM Res. Develop., 26, 6, pp.291-302, 1973.
- (45)横山、北川、河内：“個人との適応性を考慮したオンライン手書き文字認識”、信学総全大、1349, 1982.
- (46)吉田、迫江：“スタックDPマッチングによるオンライン手書き文字認識”信学

技報、PRL83-29, pp.1-8, 1983.

(47)若原、小高、梅田：“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)、J66-D, 5, pp.593-600, 1983.

(48)若原、梅田：“ストローク結合規則を用いたオンラインくずし字分類”,信学論(D), J67-D,N011, pp.1285-1292, 1984.

(49)T.Wakahara,M.Umeda:”On-Line Cursive Script Recognition Using Stroke Linkage Rules”,ICPR,pp.1065-1068,1984.

(50)M.Nakagawa,T.Manabe,K.Aoki,Y.Ikeda,N.Takahashi:”Online handwritten character recognition as Japanese input method”,ICTP,pp.191-196, 1983.

(51)O.Kato, H.Iwase, M.Yoshida, J.Tanahashi:”Interactive hand-drawn input system”, PRIP, pp.544-549, 1982.

(52)村瀬, 若原, 梅田:”タブレット入力による手書き線図形認識アルゴリズムの検討”,信学技報,PRL81-69,pp.17-24, 1982.

(53)村瀬, 若原, 梅田:”タブレット入力による手書き線図形認識の検討”,信学全大, 1296,5-281, 1982.

(54)村瀬, 若原, 梅田:”接続ルールを考慮した候補ラティス法による手書きフローチャートのオンライン認識”,信学技報,PRL82-71,pp.47-54, 1983.

(55)村瀬, 若原, 梅田:”手書きフローチャート認識における候補ラティス法の改良”,信学全大,1429, 5-300, 1983.

(56)若原, 村瀬, 梅田:”タブレット入力による手書き文字・図形認識”,TV学会技報, IPA64-3, pp.43-48, 1983.

(57)村瀬, 若原, 梅田:”候補ラティス法による手書きフローチャートのオンライン認識”,信学論(D),J66-D,6,pp.675-682, 1983.

(58) H.MURASE,T.WAKAHARA,M.UMEDA:”Online Recognition Algorithm for Hand-Sketched Flowchart by Candidate Lattice Method”,System・Computer・Controls,Vol.14,No.3,pp.37-46, 1983 (文献58の翻訳)。

(59)村瀬, 若原, 梅田:”接続ルールを導入した候補ラティス法によるオンライン手書き線図形認識”,信学論(D),J67-D,3,pp.273-280, 1984.

(60)村瀬, 若原, 梅田:”接続ルールを用いたトップダウン処理によるオンライン手書き線図形認識”,研究実用化報告、第34巻、第1号, pp.97-109, 1985.

(61) H.Murase, T.Wakahara:”Online Hand-Sketched Figure Recognition”, The

Journal of the Pattern Recognition Society, Vol.19, No.2, pp.147-160, 1986.

(62) J.F. Jarvis: "The line drawing editor; Schematic diagram editing using pattern recognition techniques", Computer Graphics and Image Processing, Vol.6, No.5, pp.452-484, 1977.

(63) 富本 他: "オンライン手書き漢字・ひらがなの認識"、信学技報、PRL82-37, 1982.

(64) P.J. Ye, H. Hugli, F. Pellandini: "Techniques for chinese character recognition with reduced writing constraints", ICPR, pp.1043-1045, 1984.

(65) ゆる木 他: "階層分析法によるオンライン文字認識"、信学技報、PRL84-17, 1984.

(66) 広瀬 他: "オンライン手書き文字認識装置"、シャープ技報、Vol.28, pp.69-74, 1984.

(67) 佐藤: "走り書き文字のオンライン認識の試み"、信学総全大、89, 1983.

(68) 児島: "隣接線分構造解析法によるオンライン手書き図形認識"、情処全大、31回、6G-D, 1985.

(69) 矢野 他: "線分解法によるフローチャートシンボル認識"、信学総全大、782, 1984.

(70) 影沢: "タブレットを用いた文章図形編集システム"、信学総全大、1550, 1984.

(71) 横山 他: "高度マンマシンシステム(1)~(7)"、第30回情報処全大、5F1-1~5F-7, 1985.

(72) 村瀬, 若原, 梅田: "候補文字ラティス法による枠無し筆記文字列のオンライン認識", 信学論(D), J68-D, 4, pp.765-772, 1985.

(73) 村瀬, 若原, 梅田: "ストロークの方向分布を利用した文字・図形の回転正規化", 信学会情報・システム部門全大, 86, 1-86, 1983.

(74) 村瀬, 若原, 梅田: "候補文字ラティス法によるオンライン手書き文字列の認識", 信学技報, PRL84-13, pp.67-76, 1984.

(75) 村瀬, 若原, 小高: "文字認識・言語処理を導入した手書き文字列からの文字切出し", 信学会情報・システム部門全大, 54, 1-54, 1985.

(76) 村瀬, 新谷, 若原: "知識を用いた枠無し手書き文字列の階層的認識", 信学技報, SP86-34(PRU86-27), pp.57-66, 1986.

- (77)村瀬, 新谷, 若原, 梅田:”言語情報を導入した手書き文字列からの文字の切り出しと認識” 信学論(D), J69-D, 9, pp.1292-1301, 1986.
- (78)村瀬, 木村, 吉村, 三宅:”パターン整合法における特性核の改良とその手書き平仮名文字認識への応用”, 信学論(D), J64-D, 3, pp.276-283, 1981.
- (79)村瀬, 木村, 吉村, 三宅:”パターン類似度法による手書き平仮名文字認識の実験”, 信学技報, PRL79-2, pp.9-18, 1979.
- (80)村瀬, 木村, 吉村, 三宅:”変動吸収特性核を用いた手書き平仮名文字認識”, 信学技報, PRL79-87, pp.7-12, 1980.
- (81)村瀬, 吉村, 三宅:”パターン類似度法を用いた手書き平仮名文字認識の一実験”, 信学全大, 1333, 5-322, 1979.
- (82)村瀬, 杉浦, 吉村, 三宅:”手書き平仮名文字認識における複合類似度的手法の応用効果”, 電気関係学会東海支部連合大会, 378, 1978.
- (83)村瀬, 吉村, 木村, 三宅:”マッチング法による平仮名文字認識—特徴数減少の実験”, 電気関係学会東海支部連合大会, 319, 1979.
- (84)R.Bellman 著、渡辺茂訳:”グラフとアルゴリズム”、共立出版.
- (85)鹿野, 好田:”会話音声の機械認識における言語処理”, 信学論(D), J61-D, 4, pp.253-260, 1978.
- (86)T.C.Hu著、伊理正夫訳:”整数計画法とネットワークフロー”、培風館.
- (87)浦昭二:”データ構造”、共立出版.
- (88)Alan.C.Shaw:”A Formal Picture Description Scheme as a Basis for Picture Processing Systems”, Information and Control, Vol.14, pp.9-52, 1969.
- (89)M.Berthod, O.Faugeras:”Using Context in the Global Recognition of a Set of Objects: an Optimization approach”, Information Processing 80, IFIP, pp.695-698, 1980.
- (90)K.C.Hayes.Jr:”Reading Handwritten Words Using Hierarchical Relaxation”, CGIP, Vol.14, pp.344-364, 1980.
- (91)C.C.Tappert:”Adaptive On-line Handwriting Recognition”, ICPR, pp.1004-1007, 1974.
- (92)J.M.Kurtzberg and C.C.Tappert:”Segmentation Procedure for Handwritten Symbols and Words”, IBM Technical Disclosure Bulletin, vol25.No7B, Dec, pp.3848-3852, 1982.

- (93)馬場口, 塚本, 相原:”手書き日本語文字列からの文字切り出しの基礎実験”,信学論(D)、J68-D, 12, pp.2123-2131. 1985.
- (94)T.Akiyama,S.Naito,I.Masuda:”A method of character extraction from format-unknown document images”,ICTP, pp.85-90, 1983.
- (95)R.G.Casey,G.nagy:”Recursive Segmentation and Classification of Composite Character Patterns”,Proc. of 6-th ICPR, pp.1023-1026, 1982.
- (96)吉野, 青木:”構文的手法を用いての手書き筆記体英単語の認識”,信学技報, PRL 82-47, 1982.
- (97)小川, 谷口:”弛緩法によるストローク分解と文字列のセグメンテーション”,信学技法 PRL 80-5, 1980.
- (98)藤田, 酒井, 長谷, 米田, 吉田:”図書目録カードの読み取り実験”,信学技法, PRL 83-57, 1983.
- (99)庭田, 中村, 南:”欧文テキスト画像を対象とした文字領域の抽出”,情報処理全大 26, 2B-3, 883, 1983.
- (100) 山本, 山田, 舟久保, 安田:”文字認識による英文の文献入力システム”,情報処理全大,28,1L-6, 885, 1984.
- (101)K.Kubota,O.Iwaki,H.Arakawa:”Document Understanding System”,ICPR, pp.612-614, 1984.
- (102)W.Doster:”Different States of a Document’s Content on its way from The Gutenbergian World to the Electronic World”,ICPR, pp.872-874, 1984.
- (103) 大田, 森, 坂井:”漢字パターン列の特徴パラメータによる検索—名刺画像における姓名の場合—”,信学論,J64-D,N011, pp.997-1004, 1981.
- (104) 山田, 西村, 豊田:”英字新聞の文字の切り出し” 情報処理全大,28,1N-5, 961, 1984.
- (105) 新谷, 梅田:”文字認識における複合後処理法の能力評価”,信学論(D),J68-D,5, pp.1118-1124, 1985.
- (106) 杉村, 斎藤:”文字接続情報を用いた読み取り不能文字の判定処理—文字認識への応用—”,信学論(D),J68-D,N0.1, pp.64-71, 1985.
- (107) 池田, 大田, 上野:”手書き原稿認識における語いおよび構文の検定”,情報処理学会論文誌,VOL26,N0.5, 1985.
- (108) 大田, 鈴木, 池田:”手書き日本語文認識における文字列利用の一方式”, 信学

論(D), J68-D, NO.3, pp.330-336, 1985.

(109) 長田, 牧野, 日高: "日本語文脈情報を用いた文字認識", 信学論(D), J67-D, NO.4, pp.520-527, 1984.

(110) 金田一、池田編: "学研国語大辞典", 学習研究社.

(111) R.Bellman, R.Kalaba: "On kth Best Policies", J.Soc.Indust.Appl.Math., Vol.8, No.4, pp.582-588, 1960.

(112) Lindsay, Norman: "Human Information Processing (Second Edition)", Sect.1, pp.2-55, Academic Press.



## 付録

- A. シンボル辞書（JIS-C6270）
- B. 文字の標準辞書
- C. 分離有意文字の一覧
- D. 言語処理における接続規則の一覧

```

----- SYMBOL DICTIONARY -----
+-----+
| Representation format: |
| |
| DEFINE_SYMBOL |
|     NAME="symbol name" ("list of labels") |
|     "label":=("stroke type (LINE or ARC)" |
|               , "starting point coordinate" |
|               , "ending point coordinate" |
|               [, "middle point coordinate"]) |
|     PARAM. *="parameter" |
| END_DEFINE_SYMBOL |
+-----+

```

```

DEFINE_SYMBOL
    NAME="PROCESS" (A, B, C, D)
    A:=(LINE, (0, 2), (3, 2))
    B:=(LINE, (3, 2), (3, 0))
    C:=(LINE, (3, 0), (0, 0))
    D:=(LINE, (0, 0), (0, 2))
    PARAM. WIDTH_LIMIT= 5.0
    PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
    NAME="DECISION" (A, B, C, D)
    A:=(LINE, (0, 1), (2, 0))
    B:=(LINE, (2, 0), (4, 1))
    C:=(LINE, (0, 1), (2, 2))
    D:=(LINE, (2, 2), (4, 1))
    PARAM. WIDTH_LIMIT= 1.5
    PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
    NAME="PREPARATION" (A, B, C, D, E, F)
    A:=(LINE, (4, 12), (20, 12))
    B:=(LINE, (4, 0), (20, 0))
    C:=(LINE, (0, 6), (4, 12))
    D:=(LINE, (0, 6), (4, 0))
    E:=(LINE, (24, 6), (20, 12))
    F:=(LINE, (24, 6), (20, 0))
    PARAM. WIDTH_LIMIT= 1.5
    PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
    NAME="PREDEFINED PROCESS" (A, B, C, D, E, F)
    A:=(LINE, (0, 0), (24, 0))
    B:=(LINE, (0, 0), (0, 12))
    C:=(LINE, (0, 12), (24, 12))
    D:=(LINE, (24, 0), (24, 12))
    E:=(LINE, (2, 0), (2, 12))
    F:=(LINE, (22, 0), (22, 12))
    PARAM. WIDTH_LIMIT= 1.5
    PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
    NAME="MANUAL OPERATION" (A, B, C, D)
    A:=(LINE, (0, 12), (4, 0))
    B:=(LINE, (4, 0), (20, 0))
    C:=(LINE, (0, 12), (24, 12))
    D:=(LINE, (24, 12), (20, 0))
    PARAM. WIDTH_LIMIT= 2.0
    PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
    NAME="AUXILIARY OPERATION" (A, B, C, D)
    A:=(LINE, (0, 0), (14, 0))
    B:=(LINE, (0, 0), (0, 14))
    C:=(LINE, (0, 14), (14, 14))
    D:=(LINE, (14, 14), (14, 0))
    PARAM. WIDTH_LIMIT= 1.0
    PARAM. HIGHT_LIMIT= 1.0
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="MERGE" (A, B, C)
    A:=(LINE, (0, 12), (16, 12))
    B:=(LINE, (8, 0), (0, 12))
    C:=(LINE, (8, 0), (16, 12))
    PARAM. WIDTH_LIMIT= 2.0
    PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="EXTRACT" (A, B, C)
    A:=(LINE, (0, 0), (16, 0))
    B:=(LINE, (0, 0), (8, 12))
    C:=(LINE, (16, 0), (8, 12))
    PARAM. WIDTH_LIMIT= 2.0
    PARAM. HIGHT_LIMIT= 0.3
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="COLLATE" (A, B, C, D, E, F)
    A:=(LINE, (0, 0), (16, 0))
    B:=(LINE, (0, 0), (8, 12))
    C:=(LINE, (16, 0), (8, 12))
    D:=(LINE, (8, 12), (0, 24))
    E:=(LINE, (8, 12), (16, 24))
    F:=(LINE, (0, 24), (16, 24))
    PARAM. WIDTH_LIMIT= 5.0
    PARAM. HIGHT_LIMIT= 0.3
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="SORT" (A, B, C, D, E, )
    A:=(LINE, (0, 12), (16, 12))
    B:=(LINE, (8, 0), (0, 12))
    C:=(LINE, (8, 0), (16, 12))
    D:=(LINE, (0, 12), (8, 24))
    E:=(LINE, (8, 24), (16, 12))
    PARAM. WIDTH_LIMIT= 5.0
    PARAM. HIGHT_LIMIT= 0.4
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="MANUAL INPUT" (A, B, C, D)
    A:=(LINE, (0, 0), (24, 0))
    B:=(LINE, (0, 0), (0, 8))
    C:=(LINE, (0, 8), (24, 12))
    D:=(LINE, (24, 12), (24, 0))
    PARAM. WIDTH_LIMIT= 2.0
    PARAM. HIGHT_LIMIT= 0.4
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
    NAME="INPUT/OUTPUT" (A, B, C, D)
    A:=(LINE, (0, 0), (6, 12))
    B:=(LINE, (6, 12), (24, 12))
    C:=(LINE, (24, 12), (18, 0))
    D:=(LINE, (0, 0), (18, 0))
    PARAM. WIDTH_LIMIT= 2.0
    PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="ONLINE STRAGE" (A, B, C, D)
    A:=(ARC, (6, 12), (6, 0), (0, 6))
    B:=(ARC, (24, 12), (24, 0), (18, 6))
    C:=(LINE, (6, 12), (24, 12))
    D:=(LINE, (6, 0), (24, 0))
    PARAM. WIDTH_LIMIT= 2.0
    PARAM. HIGHT_LIMIT= 0.2
    PARAM. R1=0.20
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="OFFLINE STRAGE" (A, B, C, D)
    A:=(LINE, (0, 12), (16, 12))
    B:=(LINE, (8, 0), (0, 12))
    C:=(LINE, (8, 0), (16, 12))
    D:=(LINE, (6, 4), (10, 4))
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="DOCUMENT" (A, B, C, D, E)
    A:=(LINE, (0, 3), (0, 15))
    B:=(LINE, (0, 15), (24, 15))
    C:=(LINE, (24, 15), (24, 3))
    D:=(ARC, (0, 3), (12, 3), (6, 0))
    E:=(ARC, (12, 3), (24, 3), (18, 6))
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="PUNCHED CARD" (A, B, C, D, E)
    A:=(LINE, (0, 0), (0, 8))
    B:=(LINE, (24, 0), (24, 12))
    C:=(LINE, (4, 12), (24, 12))
    D:=(LINE, (0, 0), (24, 0))
    E:=(LINE, (0, 8), (4, 12))
    PARAM. R5=0.25
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="DECK OF CARDS" (A, B, C, D, E, F, G, H, I, J)
    A:=(LINE, (0, 0), (0, 9))
    B:=(LINE, (24, 0), (24, 12))
    C:=(LINE, (3, 12), (24, 12))
    D:=(LINE, (0, 0), (24, 0))
    E:=(LINE, (0, 9), (3, 12))
    F:=(LINE, (3, 12), (9, 18))
    G:=(LINE, (9, 18), (30, 18))
    H:=(LINE, (30, 18), (30, 6))
    I:=(LINE, (30, 6), (24, 0))
    J:=(LINE, (24, 12), (30, 18))
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
NAME="FILE OF CARDS" (A, B, C, D, E, F, G)
A:=(LINE, (0, 0), (0, 9))
B:=(LINE, (24, 0), (24, 12))
C:=(LINE, (3, 12), (24, 12))
D:=(LINE, (0, 0), (24, 0))
E:=(LINE, (0, 9), (3, 12))
F:=(LINE, (20, 14), (26, 14))
G:=(LINE, (26, 14), (26, 8))
PARAM. HIGHT_LIMIT= 0.5
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
NAME="PUNCHED TAPE" (A, B, C, D, E, F)
A:=(LINE, (0, 2), (0, 10))
B:=(LINE, (24, 2), (24, 10))
C:=(ARC, (0, 10), (12, 10), (6, 8))
D:=(ARC, (12, 10), (24, 10), (18, 12))
E:=(ARC, (0, 2), (12, 2), (6, 0))
F:=(ARC, (12, 2), (24, 2), (18, 4))
PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
NAME="MAGNETIC TAPE" (A, B, C)
A:=(LINE, (6, 0), (15, 0))
B:=(ARC, (6, 12), (6, 0), (0, 6))
C:=(ARC, (6, 12), (6, 0), (12, 6))
PARAM. HIGHT_LIMIT= 0.5
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
NAME="MAGNETIC DRUM" (A, B, C, D, E)
A:=(LINE, (3, 0), (15, 0))
B:=(LINE, (3, 12), (15, 12))
C:=(LINE, (15, 0), (15, 12), (18, 6))
D:=(LINE, (15, 0), (15, 12), (12, 6))
E:=(LINE, (3, 0), (3, 12), (0, 6))
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
NAME="MAGNETIC DISK" (A, B, C, D, E)
A:=(LINE, (0, 3), (0, 15))
B:=(LINE, (12, 3), (12, 15))
C:=(ARC, (0, 15), (12, 15), (6, 18))
D:=(ARC, (0, 15), (12, 15), (6, 12))
E:=(ARC, (0, 3), (12, 3), (6, 0))
PARAM. WIDTH_LIMIT= 3.0
PARAM. HIGHT_LIMIT= 0.5
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
NAME="CORE" (A, B, C, D, E, F)
A:=(LINE, (0, 0), (0, 14))
B:=(LINE, (0, 14), (14, 14))
C:=(LINE, (14, 14), (14, 0))
D:=(LINE, (0, 0), (14, 0))
E:=(LINE, (0, 12), (14, 12))
F:=(LINE, (2, 0), (2, 14))
PARAM. HIGHT_LIMIT= 0.5
PARAM. R1=0.20
END_DEFINE_SYMBOL

```

```

DEFINE_SYMBOL
    NAME="DISPLAY" (A, B, C, D, E)
    A:=(LINE, (15, 20), (30, 20))
    B:=(LINE, (15, 0), (30, 0))
    C:=(ARC, (30, 20), (30, 0), (35, 10))
    D:=(ARC, (0, 10), (15, 20), (5, 16))
    E:=(ARC, (0, 10), (15, 0), (5, 4))
    PARAM. WIDTH_LIMIT= 1.5
    PARAM. HIGHT_LIMIT= 0.3
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="FLOW LINE" (A)
    A:=(LINE, (0, 0), (1, 1))
    PARAM. WIDTH_LIMIT= 1.0
    PARAM. HIGHT_LIMIT= 1.0
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="PARALLEL MODE" (A, B)
    A:=(LINE, (0, 3), (20, 3))
    B:=(LINE, (0, 0), (20, 0))
    PARAM. WIDTH_LIMIT= 1.0
    PARAM. HIGHT_LIMIT= 0.1
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="COMMUNICATION LINK" (A, B, C)
    A:=(LINE, (0, 2), (14, 0))
    B:=(LINE, (14, 0), (10, 4))
    C:=(LINE, (10, 4), (24, 2))
    PARAM. WIDTH_LIMIT= 1.0
    PARAM. HIGHT_LIMIT= 0.1
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="CONNECTOR" (A, B)
    A:=(ARC, (5, 10), (5, 0), (0, 5))
    B:=(ARC, (5, 10), (5, 0), (10, 5))
    PARAM. HIGHT_LIMIT= 0.5
    PARAM. R1=0.25
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="TERMINAL" (A, B, C, D)
    A:=(LINE, (4, 8), (24, 8))
    B:=(LINE, (4, 0), (24, 0))
    C:=(ARC, (4, 8), (4, 0), (0, 4))
    D:=(ARC, (24, 8), (24, 0), (28, 4))
    PARAM. WIDTH_LIMIT= 1.0
    PARAM. HIGHT_LIMIT= 0.2
END_DEFINE_SYMBOL

DEFINE_SYMBOL
    NAME="COMMENT" (A, B, C)
    A:=(LINE, (0, 12), (6, 12))
    B:=(LINE, (0, 12), (0, 0))
    C:=(LINE, (0, 0), (6, 0))
    PARAM. WIDTH_LIMIT= 3.0
    PARAM. HIGHT_LIMIT= 0.5
END_DEFINE_SYMBOL

```

## 辞書登録した文字

## 画数別のカテゴリー数

画数	平仮名	常用漢字(含,当用漢字)	画数変動文字	合計
1画	11個	2(2)個	0個	13個
2	20	12(12)	1	33
3	21	30(30)	2	53
4	12	66(64)	4	82
5	8	94(89)	5	107
6	4	110(106)	8	122
7	0	141(138)	12	153
8	0	189(178)	12	201
9	0	177(170)	19	196
10	0	199(190)	21	220
11	0	195(179)	14	209
12	0	200(190)	5	205
13	0	146(138)	6	152
14	0	106(101)	10	116
15	0	103(98)	5	108
16	0	69(67)	0	69
17	0	34(31)	0	34
18	0	34(32)	1	35
19	0	20(17)	1	21
20	0	11(11)	0	11
21	0	5(5)	0	5
22	0	2(2)	0	2
23	0	1(1)	0	1
1~23	76	1946(1851)	126	2148
全カテゴリー数				2148個

標準文字

通し番号 JIS漢字コード

(画数, 画数内での通し番号)

一	乙	く	し	つ	て	の	ひ	へ	る	ろ	ん	ろ	九	七
1 306C (1, 1)	2 3235 (1, 2)	3 242F (1, 3)	4 2437 (1, 4)	5 2444 (1, 5)	6 2446 (1, 6)	7 244E (1, 7)	8 2452 (1, 8)	9 2458 (1, 9)	10 2468 (1, 10)	11 246D (1, 11)	12 2473 (1, 12)	13 243D (1, 13)	14 3055 (2, 1)	15 3057 (2, 2)
十	人	丁	刀	二	入	八	又	了	カ	子	い	う	え	こ
16 303D (2, 3)	17 3F4D (2, 4)	18 437A (2, 5)	19 4561 (2, 6)	20 4673 (2, 7)	21 467E (2, 8)	22 482E (2, 9)	23 4874 (2, 10)	24 4E38 (2, 11)	25 4E4F (2, 12)	26 3B52 (2, 13)	27 2424 (2, 14)	28 2426 (2, 15)	29 2428 (2, 16)	30 2423 (2, 17)
す	ち	と	ぬ	ね	み	め	ゆ	よ	ら	リ	れ	わ	そ	へ
31 2439 (2, 18)	32 2441 (2, 19)	33 2448 (2, 20)	34 244C (2, 21)	35 244D (2, 22)	36 245F (2, 23)	37 2461 (2, 24)	38 2466 (2, 25)	39 2468 (2, 26)	40 2469 (2, 27)	41 246A (2, 28)	42 246C (2, 29)	43 246F (2, 30)	44 2438 (2, 31)	45 245A (2, 32)
ひ	下	干	丸	久	及	弓	己	工	口	才	三	山	士	子
46 2454 (2, 33)	47 323C (3, 1)	48 3433 (3, 2)	49 345D (3, 3)	50 3557 (3, 4)	51 355A (3, 5)	52 355D (3, 6)	53 384A (3, 7)	54 3929 (3, 8)	55 387D (3, 9)	56 3A4D (3, 10)	57 303D (3, 11)	58 3033 (3, 12)	59 304E (3, 13)	60 3052 (3, 14)
勺	女	小	上	丈	刃	寸	夕	千	川	大	土	亡	凡	万
61 3C58 (3, 15)	62 3D77 (3, 16)	63 3E2E (3, 17)	64 3E55 (3, 18)	65 3E66 (3, 19)	66 3F4F (3, 20)	67 4D23 (3, 21)	68 4D3C (3, 22)	69 4D69 (3, 23)	70 4D6E (3, 24)	71 4267 (3, 25)	72 455A (3, 26)	73 4834 (3, 27)	74 485E (3, 28)	75 487E (3, 29)
与	孔	水	あ	お	か	け	さ	せ	に	は	ま	む	も	や
76 4D3F (3, 30)	77 3926 (3, 31)	78 3F65 (3, 32)	79 2422 (3, 33)	80 242A (3, 34)	81 2428 (3, 35)	82 2431 (3, 36)	83 2435 (3, 37)	84 2438 (3, 38)	85 2448 (3, 39)	86 244F (3, 40)	87 245E (3, 41)	88 246D (3, 42)	89 246E (3, 43)	90 2464 (3, 44)
を	ふ	べ	そ	づ	く	で	じ	び	引	円	王	火	化	介
91 2472 (3, 45)	92 2455 (3, 46)	93 2459 (3, 47)	94 243E (3, 48)	95 2445 (3, 49)	96 243D (3, 50)	97 2447 (3, 51)	98 2438 (3, 52)	99 2453 (3, 53)	100 3D7A (4, 1)	101 315F (4, 2)	102 3226 (4, 3)	103 325D (4, 4)	104 323D (4, 5)	105 327D (4, 6)
刈	牛	凶	斤	区	欠	月	犬	元	幻	戸	五	午	互	公
106 3422 (4, 7)	107 358D (4, 8)	108 3627 (4, 9)	109 3654 (4, 10)	110 3668 (4, 11)	111 3767 (4, 12)	112 376E (4, 13)	113 3824 (4, 14)	114 3835 (4, 15)	115 3836 (4, 16)	116 394D (4, 17)	117 395E (4, 18)	118 3961 (4, 19)	119 395F (4, 20)	120 3978 (4, 21)
孔	今	止	氏	支	尺	手	収	少	升	冗	心	仁	水	井
121 3926 (4, 22)	122 3A23 (4, 23)	123 395F (4, 24)	124 3961 (4, 25)	125 3859 (4, 26)	126 3C5C (4, 27)	127 3C6A (4, 28)	128 3C7D (4, 29)	129 3E2F (4, 30)	130 3E23 (4, 31)	131 3E69 (4, 32)	132 3F34 (4, 33)	133 3F4E (4, 34)	134 3F65 (4, 35)	135 3968 (4, 36)
切	双	太	丹	中	帛	天	斗	内	日	反	比	匹	父	夫
136 4D5A (4, 37)	137 415D (4, 38)	138 424D (4, 39)	139 493D (4, 40)	140 4366 (4, 41)	141 4424 (4, 42)	142 4537 (4, 43)	143 454D (4, 44)	144 4662 (4, 45)	145 487E (4, 46)	146 4831 (4, 47)	147 4866 (4, 48)	148 4824 (4, 49)	149 4963 (4, 50)	150 4957 (4, 51)
不	仏	分	文	片	方	乏	木	毛	奴	友	予	六	永	込
151 4954 (4, 52)	152 4A28 (4, 53)	153 4A2E (4, 54)	154 4A38 (4, 55)	155 4A52 (4, 56)	156 4A7D (4, 57)	157 4B33 (4, 58)	158 4C5A (4, 59)	159 4C53 (4, 60)	160 4C68 (4, 61)	161 4D27 (4, 62)	162 4D3D (4, 63)	163 4F38 (4, 64)	164 314A (4, 65)	165 3976 (4, 66)
氷	辺	き	た	な	心	ほ	は	と	そ	ふ	こ	ず	ち	圧
166 4938 (4, 67)	167 4A55 (4, 68)	168 242D (4, 69)	169 243F (4, 70)	170 244A (4, 71)	171 2455 (4, 72)	172 2458 (4, 73)	173 2451 (4, 74)	174 2449 (4, 75)	175 243E (4, 76)	176 2457 (4, 77)	177 2434 (4, 78)	178 243F (4, 79)	179 2442 (4, 80)	180 3033 (5, 1)
以	永	央	可	加	外	且	刊	甘	丘	旧	去	巨	叫	玉
181 3D4A (5, 2)	182 314A (5, 3)	183 3178 (5, 4)	184 3244 (5, 5)	185 3243 (5, 6)	186 333D (5, 7)	187 336E (5, 8)	188 342A (5, 9)	189 3445 (5, 10)	190 3556 (5, 11)	191 356C (5, 12)	192 356E (5, 13)	193 357D (5, 14)	194 3528 (5, 15)	195 364C (5, 16)
句	兄	穴	玄	古	広	功	甲	巧	号	込	左	札	冊	四
196 3667 (5, 17)	197 3738 (5, 18)	198 376A (5, 19)	199 383C (5, 20)	200 3845 (5, 21)	201 592D (5, 22)	202 3879 (5, 23)	203 3943 (5, 24)	204 382A (5, 25)	205 3966 (5, 26)	206 397E (5, 27)	207 3938 (5, 28)	208 3E25 (5, 29)	209 397D (5, 30)	210 394D (5, 31)



市 史 仕 司 矢 示 失 写 主 囚 出 処 召 申 世

211 3854 (5, 32) 212 3848 (5, 33) 213 3845 (5, 34) 214 3844 (5, 35) 215 4070 (5, 36) 216 3828 (5, 37) 217 3834 (5, 38) 218 3840 (5, 39) 219 3867 (5, 40) 220 3870 (5, 41) 221 3050 (5, 42) 222 3068 (5, 43) 223 3824 (5, 44) 224 3830 (5, 45) 225 4324 (5, 46)

生 正 石 斥 占 他 打 代 台 庁 田 奴 冬 尼 白

226 4038 (5, 47) 227 4035 (5, 48) 228 4050 (5, 49) 229 4040 (5, 50) 230 4064 (5, 51) 231 4236 (5, 52) 232 4247 (5, 53) 233 4265 (5, 54) 234 4266 (5, 55) 235 4423 (5, 56) 236 4544 (5, 57) 237 4558 (5, 58) 238 4557 (5, 59) 239 4674 (5, 60) 240 4772 (5, 61)

半 犯 皮 必 氷 付 布 払 平 丙 辺 弁 母 包 北

241 4836 (5, 62) 242 4848 (5, 63) 243 4868 (5, 64) 244 4820 (5, 65) 245 4839 (5, 66) 246 4955 (5, 67) 247 4958 (5, 68) 248 4827 (5, 69) 249 4837 (5, 70) 250 4834 (5, 71) 251 4855 (5, 72) 252 4858 (5, 73) 253 4860 (5, 74) 254 4871 (5, 75) 255 4844 (5, 76)

本 末 未 民 矛 目 由 右 用 幼 立 令 礼 好 字

256 4856 (5, 77) 257 4876 (5, 78) 258 4824 (5, 79) 259 4831 (5, 80) 260 4837 (5, 81) 261 4850 (5, 82) 262 4833 (5, 83) 263 3126 (5, 84) 264 4051 (5, 85) 265 4044 (5, 86) 266 4829 (5, 87) 267 4861 (5, 88) 268 4869 (5, 89) 269 3825 (5, 90) 270 3874 (5, 91)

巡 迅 存 は ほ が げ ぶ ぶ さ せ 扱 安 衣 芋

271 3064 (5, 92) 272 3857 (5, 93) 273 4238 (5, 94) 274 2450 (5, 95) 275 2450 (5, 96) 276 2420 (5, 97) 277 2432 (5, 98) 278 2456 (5, 99) 279 2457 (5, 100) 280 2436 (5, 101) 281 2430 (5, 102) 282 3037 (6, 1) 283 3032 (6, 2) 284 3001 (6, 3) 285 3072 (6, 4)

因 印 羽 宇 汚 仮 回 会 灰 各 汗 気 机 危 企

286 3078 (6, 5) 287 3075 (6, 6) 288 3128 (6, 7) 289 3127 (6, 8) 290 3178 (6, 9) 291 3236 (6, 10) 292 3273 (6, 11) 293 3271 (6, 12) 294 3325 (6, 13) 295 3346 (6, 14) 296 3440 (6, 15) 297 3524 (6, 16) 298 3479 (6, 17) 299 2460 (6, 18) 300 3468 (6, 19)

吉 休 吸 朽 共 仰 曲 刑 血 件 光 交 向 行 考

301 3548 (6, 20) 302 3559 (6, 21) 303 3550 (6, 22) 304 3560 (6, 23) 305 3626 (6, 24) 306 3644 (6, 25) 307 3644 (6, 26) 308 3734 (6, 27) 309 3760 (6, 28) 310 3760 (6, 29) 311 3877 (6, 30) 312 3872 (6, 31) 313 3872 (6, 32) 314 3954 (6, 33) 315 3940 (6, 34)

好 后 江 合 再 在 糸 死 至 旨 寺 字 自 耳 次

316 3825 (6, 35) 317 3821 (6, 36) 318 3836 (6, 37) 319 3867 (6, 38) 320 3446 (6, 39) 321 3857 (6, 40) 322 3865 (6, 41) 323 3860 (6, 42) 324 3864 (6, 43) 325 3850 (6, 44) 326 3878 (6, 45) 327 3874 (6, 46) 328 3828 (6, 47) 329 3824 (6, 48) 330 3821 (6, 49)

式 芝 守 朱 州 舟 充 巡 旬 如 匠 色 尽 迅 成

331 3830 (6, 50) 332 3847 (6, 51) 333 3868 (6, 52) 334 3868 (6, 53) 335 3823 (6, 54) 336 3822 (6, 55) 337 3830 (6, 56) 338 3864 (6, 57) 339 3850 (6, 58) 340 4721 (6, 59) 341 3822 (6, 60) 342 3827 (6, 61) 343 3854 (6, 62) 344 3857 (6, 63) 345 4022 (6, 64)

西 舌 先 全 早 争 壮 存 多 宅 団 池 地 竹 虫

346 4036 (6, 65) 347 4085 (6, 66) 348 4068 (6, 67) 349 4134 (6, 68) 350 4167 (6, 69) 351 4168 (6, 70) 352 4164 (6, 71) 353 4238 (6, 72) 354 4238 (6, 73) 355 4270 (6, 74) 356 4344 (6, 75) 357 4353 (6, 76) 358 4347 (6, 77) 359 4350 (6, 78) 360 4366 (6, 79)

仲 兆 伝 吐 当 同 式 肉 任 年 伐 帆 妃 百 伏

361 4367 (6, 80) 362 4378 (6, 81) 363 4541 (6, 82) 364 4547 (6, 83) 365 4576 (6, 84) 366 4631 (6, 85) 367 4675 (6, 86) 368 4679 (6, 87) 369 4724 (6, 88) 370 4727 (6, 89) 371 4832 (6, 90) 372 4841 (6, 91) 373 4852 (6, 92) 374 4834 (6, 93) 375 4874 (6, 94)

米 忙 每 名 有 羊 吏 西 列 劣 老 灯 近 迎 孝

376 4846 (6, 95) 377 4838 (6, 96) 378 4868 (6, 97) 379 4836 (6, 98) 380 4020 (6, 99) 381 4059 (6, 100) 382 4078 (6, 101) 383 4836 (6, 102) 384 4873 (6, 103) 385 4874 (6, 104) 386 4837 (6, 105) 387 4874 (6, 106) 388 3581 (6, 107) 389 3756 (6, 108) 390 3827 (6, 109)

廷 尿 返 邦 防 ほ た ぶ き 亜 位 医 井 壺 応

391 4466 (6, 110) 392 4722 (6, 111) 393 4856 (6, 112) 394 4822 (6, 113) 395 4849 (6, 114) 396 2450 (6, 115) 397 2440 (6, 116) 398 2456 (6, 117) 399 2422 (6, 118) 400 3021 (7, 1) 401 3040 (7, 2) 402 3065 (7, 3) 403 3347 (7, 4) 404 3060 (7, 5) 405 3172 (7, 6)

花 何 我 快 改 貝 戒 角 完 肝 含 汽 希 岐 忌

406 3256 (7, 7) 407 3237 (7, 8) 408 3268 (7, 9) 409 3277 (7, 10) 410 3272 (7, 11) 411 3320 (7, 12) 412 3270 (7, 13) 413 3351 (7, 14) 414 3430 (7, 15) 415 3442 (7, 16) 416 3452 (7, 17) 417 3525 (7, 18) 418 3475 (7, 19) 419 3474 (7, 20) 420 3477 (7, 21)

技 却 究 求 狂 局 近 均 吟 君 形 系 芸 迎 决

421 353B (7. 22) 422 355I (7. 23) 423 3566 (7. 24) 424 356I (7. 25) 425 3636 (7. 26) 426 3649 (7. 27) 427 366I (7. 28) 428 365I (7. 29) 429 3663 (7. 30) 430 372F (7. 31) 431 374I (7. 32) 432 374F (7. 33) 433 375D (7. 34) 434 375E (7. 35) 435 375B (7. 36)

見 言 吳 孝 更 攻 抗 坑 谷 告 克 困 佐 災 材

436 382B (7. 37) 437 3840 (7. 38) 438 3852 (7. 39) 439 3827 (7. 40) 440 393B (7. 41) 441 3936 (7. 42) 442 3933 (7. 43) 443 3923 (7. 44) 444 432B (7. 45) 445 397D (7. 46) 446 386E (7. 47) 447 3824 (7. 48) 448 3834 (7. 49) 449 3852 (7. 50) 450 365D (7. 51)

作 私 志 伺 兕 似 車 社 寿 秀 住 初 助 序 床

451 386E (7. 52) 452 3864 (7. 53) 453 3856 (7. 54) 454 3847 (7. 55) 455 387B (7. 56) 456 3877 (7. 57) 457 3C56 (7. 58) 458 3C52 (7. 59) 459 3C77 (7. 60) 460 3D2B (7. 61) 461 3D3B (7. 62) 462 3D5B (7. 63) 463 3D75 (7. 64) 464 3D7B (7. 65) 465 3E32 (7. 66)

肖 抄 条 状 身 臣 伸 辛 凶 吹 声 赤 折 走 足

466 3E53 (7. 67) 467 3E35 (7. 68) 468 3E72 (7. 69) 469 3E70 (7. 70) 470 3F4B (7. 71) 471 3F43 (7. 72) 472 3F2D (7. 73) 473 3F49 (7. 74) 474 3F5E (7. 75) 475 3F8J (7. 76) 476 403C (7. 77) 477 4056 (7. 78) 478 405E (7. 79) 479 4176 (7. 80) 480 422D (7. 81)

即 東 村 妥 体 对 沢 扱 但 男 冲 町 沈 弟 低

481 422B (7. 82) 482 422B (7. 83) 483 423C (7. 84) 484 4245 (7. 85) 485 424E (7. 86) 486 425D (7. 87) 487 4274 (7. 88) 488 4272 (7. 89) 489 4322 (7. 90) 490 434B (7. 91) 491 322B (7. 92) 492 442E (7. 93) 493 444D (7. 94) 494 446F (7. 95) 495 4463 (7. 96)

廷 呈 努 投 豆 尿 忍 妊 壳 伯 麦 拔 判 坂 伴

496 446E (7. 97) 497 446B (7. 98) 498 455B (7. 99) 499 456A (7. 100) 500 4626 (7. 101) 501 4722 (7. 102) 502 4726 (7. 103) 503 4725 (7. 104) 504 4764 (7. 105) 505 476C (7. 106) 506 477E (7. 107) 507 4834 (7. 108) 508 483D (7. 109) 509 3464 (7. 110) 510 463C (7. 111)

否 批 尾 扶 兵 别 返 芳 邦 防 忘 妨 坊 没 妙

511 485D (7. 112) 512 4863 (7. 113) 513 487B (7. 114) 514 485E (7. 115) 515 483E (7. 116) 516 484C (7. 117) 517 4856 (7. 118) 518 4827 (7. 119) 519 482E (7. 120) 520 484B (7. 121) 521 483A (7. 122) 522 483B (7. 123) 523 4837 (7. 124) 524 4857 (7. 125) 525 4E2F (7. 126)

役 余 抑 采 乱 卵 里 利 良 冷 励 劳 泳 延 学

526 4C72 (7. 127) 527 403E (7. 128) 528 405E (7. 129) 529 4D6B (7. 130) 530 407D (7. 131) 531 407J (7. 132) 532 4E24 (7. 133) 533 407B (7. 134) 534 4E49 (7. 135) 535 4E64 (7. 136) 536 4E65 (7. 137) 537 4F2B (7. 138) 538 314B (7. 139) 539 3154 (7. 140) 540 335B (7. 141)

季 享 邪 述 阻 邨 迭 迫 附 委 依 育 雨 泳 英

541 352B (7. 142) 542 357D (7. 143) 543 3C5B (7. 144) 544 3D52 (7. 145) 545 414B (7. 146) 546 4527 (7. 147) 547 4533 (7. 148) 548 4777 (7. 149) 549 496D (7. 150) 550 305I (7. 151) 551 304D (7. 152) 552 306B (7. 153) 553 312B (7. 154) 554 314B (7. 155) 555 3157 (7. 156)

易 沿 炎 延 往 欧 押 殴 卸 河 果 晒 佳 画 芽

556 3057 (7. 157) 557 316B (7. 158) 558 316A (7. 159) 559 3164 (7. 160) 560 317D (7. 161) 561 3224 (7. 162) 562 3227 (7. 163) 563 3225 (7. 164) 564 3237 (7. 165) 565 324F (7. 166) 566 324C (7. 167) 567 3247 (7. 168) 568 3242 (7. 169) 569 325B (7. 170) 570 326A (7. 171)

怪 劾 扩 学 岳 官 岩 岸 季 奇 祈 宜 泣 糾 拒

571 327B (7. 172) 572 332F (7. 173) 573 334B (7. 174) 574 335B (7. 175) 575 3359 (7. 176) 576 3437 (7. 177) 577 3464 (7. 178) 578 345F (7. 179) 579 352B (7. 180) 580 3477 (7. 181) 581 3527 (7. 182) 582 353B (7. 183) 583 3563 (7. 184) 584 355A (7. 185) 585 3577 (7. 186)

居 扱 供 協 享 況 京 金 苦 具 空 屈 茎 径 券

586 356F (7. 187) 587 3572 (7. 188) 588 3627 (7. 189) 589 362B (7. 190) 590 357C (7. 191) 591 3637 (7. 192) 592 357E (7. 193) 593 3662 (7. 194) 594 366C (7. 195) 595 3677 (7. 196) 596 3675 (7. 197) 597 367E (7. 198) 598 3754 (7. 199) 599 3752 (7. 200) 600 3774 (7. 201)

肩 弦 固 呼 幸 効 肯 拘 国 刻 妻 刷 参 使 始

601 382A (7. 202) 602 3839 (7. 203) 603 3847 (7. 204) 604 3846 (7. 205) 605 392C (7. 206) 606 387A (7. 207) 607 394E (7. 208) 608 3934 (7. 209) 609 3977 (7. 210) 610 398F (7. 211) 611 3844 (7. 212) 612 387E (7. 213) 613 387E (7. 214) 614 384B (7. 215) 615 386F (7. 216)

姉 枝 刺 社 事 侍 治 寔 者 舍 邪 若 取 受 周

616 385D (7. 217) 617 385E (7. 218) 618 3849 (7. 219) 619 3863 (7. 220) 620 3876 (7. 221) 621 387B (7. 222) 622 3C23 (7. 223) 623 3C42 (7. 224) 624 3C54 (7. 225) 625 3C4B (7. 226) 626 3C59 (7. 227) 627 3C63 (7. 228) 628 3C6B (7. 229) 629 3C75 (7. 230) 630 3C7E (7. 231)

叔 631 3047 (8, 82)	述 632 3052 (8, 83)	所 633 306A (8, 84)	承 634 3E95 (8, 85)	招 635 3E37 (8, 86)	松 636 3E9E (8, 87)	沼 637 3E42 (8, 88)	昇 638 3E3A (8, 89)	垂 639 3F62 (8, 90)	炊 640 3F66 (8, 91)	枢 641 3F75 (8, 92)	性 642 4020 (8, 93)	青 643 4044 (8, 94)	制 644 4029 (8, 95)	姓 645 402B (8, 96)
征 646 402C (8, 97)	昔 647 40NE (8, 98)	析 648 404F (8, 99)	拙 649 405B (8, 100)	阻 650 414B (8, 101)	宗 651 3021 (8, 102)	卒 652 4234 (8, 103)	拓 653 4273 (8, 104)	卓 654 426E (8, 105)	担 655 4334 (8, 106)	知 656 434E (8, 107)	注 657 4360 (8, 108)	忠 658 4359 (8, 109)	宙 659 436B (8, 110)	抽 660 435A (8, 111)
長 661 443B (8, 112)	直 662 443E (8, 113)	坪 663 445A (8, 114)	定 664 446A (8, 115)	底 665 446E (8, 116)	邸 666 4521 (8, 117)	抵 667 4471 (8, 118)	的 668 452A (8, 119)	迭 669 4533 (8, 120)	典 670 4535 (8, 121)	店 671 4539 (8, 122)	東 672 456C (8, 123)	到 673 457E (8, 124)	毒 674 4577 (8, 125)	突 675 4640 (8, 126)
届 676 464F (8, 127)	乳 677 4670 (8, 128)	念 678 4730 (8, 129)	波 679 474B (8, 130)	拜 680 4752 (8, 131)	杯 681 4755 (8, 132)	拍 682 476F (8, 133)	泊 683 4771 (8, 134)	迫 684 4777 (8, 135)	板 685 4844 (8, 136)	版 686 4847 (8, 137)	非 687 4873 (8, 138)	肥 688 485E (8, 139)	彼 689 4850 (8, 140)	泌 690 4857 (8, 141)
表 691 4930 (8, 142)	苗 692 4944 (8, 143)	府 693 495C (8, 144)	附 694 496B (8, 145)	怖 695 4950 (8, 146)	武 696 4970 (8, 147)	侮 697 496E (8, 148)	服 698 497E (8, 149)	沸 699 492B (8, 150)	物 700 4A2A (8, 151)	併 701 4A3B (8, 152)	並 702 4A42 (8, 153)	步 703 4A52 (8, 154)	法 704 4A21 (8, 155)	放 705 4A7C (8, 156)
宝 706 4A75 (8, 157)	抱 707 4A7A (8, 158)	奉 708 4A7H (8, 159)	房 709 4B3C (8, 160)	肪 710 4B43 (8, 161)	牧 711 4B52 (8, 162)	奔 712 4B5B (8, 163)	妹 713 4B65 (8, 164)	枚 714 4B67 (8, 165)	味 715 4C23 (8, 166)	明 716 4C40 (8, 167)	命 717 4C3F (8, 168)	免 718 4C4B (8, 169)	茂 719 4C50 (8, 170)	盲 720 4C55 (8, 171)
門 721 4C67 (8, 172)	夜 722 4C6B (8, 173)	油 723 4C70 (8, 174)	林 724 4E53 (8, 175)	例 725 4E63 (8, 176)	炉 726 4F27 (8, 177)	和 727 4F42 (8, 178)	逆 728 3E55 (8, 179)	建 729 377A (8, 180)	限 730 3942 (8, 181)	孤 731 394B (8, 182)	郊 732 3959 (8, 183)	泉 733 4074 (8, 184)	送 734 4177 (8, 185)	退 735 4260 (8, 186)
追 736 444B (8, 187)	逃 737 4E2B (8, 188)	迷 738 4E42 (8, 189)	郎 739 4F3A (8, 190)	哀 740 3025 (9, 1)	胃 741 305F (8, 2)	為 742 3059 (9, 3)	戕 743 3052 (8, 4)	姻 744 3075 (9, 5)	榮 745 314B (9, 6)	映 746 3147 (9, 7)	疫 747 3156 (9, 8)	屋 748 3230 (9, 9)	音 749 323B (9, 10)	科 750 324A (9, 11)
架 751 3E40 (9, 12)	海 752 3324 (9, 13)	界 753 3326 (8, 14)	悔 754 327B (9, 15)	皆 755 3327 (9, 16)	革 756 3357 (8, 17)	活 757 336B (8, 18)	括 758 3367 (9, 19)	卷 759 342C (9, 20)	冠 760 3427 (9, 21)	皇 761 3944 (9, 22)	看 762 3447 (8, 23)	紀 763 352A (8, 24)	既 764 347B (9, 25)	軌 765 3E30 (9, 26)
客 766 3552 (9, 27)	逆 767 3555 (8, 28)	急 768 355E (8, 29)	級 769 3569 (9, 30)	狭 770 3639 (9, 31)	峽 771 362E (8, 32)	軍 772 3733 (8, 33)	計 773 3757 (9, 34)	係 774 373B (8, 35)	型 775 373F (8, 36)	契 776 3740 (9, 37)	桌 777 3829 (8, 38)	研 778 3825 (8, 39)	建 779 377A (9, 40)	限 780 3642 (8, 41)
故 781 384E (8, 42)	孤 782 384B (8, 43)	枯 783 384F (8, 44)	弧 784 384C (9, 45)	後 785 3865 (8, 46)	厚 786 387C (8, 47)	郊 787 3954 (9, 48)	香 788 3961 (8, 49)	荒 789 3953 (8, 50)	紅 790 394B (9, 51)	侯 791 3875 (9, 52)	恒 792 3831 (8, 53)	拷 793 3565 (9, 54)	恨 794 3A25 (9, 55)	查 795 3A3A (9, 56)
砂 796 3A30 (8, 57)	碎 797 3A55 (9, 58)	昨 798 3A72 (8, 59)	咲 799 3A69 (8, 60)	削 800 3A6F (9, 61)	思 801 3B57 (8, 62)	指 802 3B5B (9, 63)	姿 803 3B51 (8, 64)	施 804 3B5C (8, 65)	持 805 3B70 (9, 66)	室 806 3C3C (9, 67)	首 807 3C73 (9, 68)	狩 808 3C60 (8, 69)	秋 809 3C29 (9, 70)	拾 810 3126 (9, 71)
臭 811 3020 (8, 72)	重 812 3045 (9, 73)	柔 813 3040 (8, 74)	祝 814 304B (9, 75)	春 815 3055 (9, 76)	俊 816 3053 (8, 77)	盾 817 3062 (8, 78)	叙 818 307E (8, 79)	昭 819 3E3C (8, 80)	乘 820 3E6B (8, 81)	城 821 3E6B (8, 82)	淨 822 3E7H (8, 83)	食 823 3E2B (9, 84)	神 824 3F40 (9, 85)	信 825 3F2E (9, 86)
侵 826 3F2F (8, 87)	津 827 4445 (9, 88)	帥 828 3F63 (8, 89)	是 829 4027 (9, 90)	星 830 4031 (9, 91)	政 831 402F (9, 92)	牲 832 4037 (9, 93)	省 833 3E4A (8, 94)	窳 834 4060 (9, 95)	淺 835 4075 (8, 96)	宣 836 406B (8, 97)	專 837 406C (8, 98)	洗 838 4C7B (8, 99)	染 839 4D77 (8, 100)	泉 840 4C74 (8, 101)

前	祖	草	送	莊	奏	相	則	促	俗	待	退	耐	怠	胎
841 4130 ( 9.102)	842 4144 ( 9.103)	843 4170 ( 9.104)	844 4177 ( 9.105)	845 4171 ( 9.106)	846 4155 ( 9.107)	847 4164 ( 9.108)	848 4227 ( 9.109)	849 4225 ( 9.110)	850 4227 ( 9.111)	851 4254 ( 9.112)	852 4180 ( 9.113)	853 4251 ( 9.114)	854 4255 ( 9.115)	855 4258 ( 9.116)
炭	單	胆	段	茶	昼	衷	柱	勅	珍	追	訂	貞	帝	点
856 4334 ( 9.117)	857 4331 ( 9.118)	858 4340 ( 9.118)	859 4344 ( 9.120)	860 4363 ( 9.121)	861 4368 ( 9.122)	862 4367 ( 9.123)	863 4360 ( 9.124)	864 4430 ( 9.125)	865 4441 ( 9.126)	866 4448 ( 9.127)	867 4478 ( 9.128)	868 4457 ( 9.129)	869 4468 ( 9.130)	870 4540 ( 9.131)
度	奴	逃	峙	独	南	派	背	肺	烟	癸	飛	阜	美	秒
871 4558 ( 9.132)	872 4550 ( 9.133)	873 4628 ( 9.134)	874 4360 ( 9.135)	875 4648 ( 9.136)	876 4666 ( 9.137)	877 4749 ( 9.138)	878 4758 ( 9.139)	879 4758 ( 9.140)	880 4824 ( 9.141)	881 4827 ( 9.142)	882 4874 ( 9.143)	883 4850 ( 9.144)	884 4877 ( 9.145)	885 4863 ( 9.146)
品	負	赴	風	封	柄	变	便	保	胞	某	冒	盆	迷	面
886 4944 ( 9.147)	887 4968 ( 9.148)	888 4968 ( 9.149)	889 4977 ( 9.150)	890 4975 ( 9.151)	891 4941 ( 9.152)	892 4951 ( 9.153)	893 4958 ( 9.154)	894 4950 ( 9.155)	895 4926 ( 9.156)	896 4937 ( 9.157)	897 4941 ( 9.158)	898 4957 ( 9.159)	899 4942 ( 9.160)	900 4940 ( 9.161)
約	勇	幽	洋	要	律	柳	厘	郎	院	陷	郭	郡	降	除
901 4973 ( 9.162)	902 4026 ( 9.163)	903 4029 ( 9.164)	904 404E ( 9.165)	905 4057 ( 9.166)	906 4E27 ( 9.167)	907 407E ( 9.168)	908 4E52 ( 9.169)	909 4F34 ( 9.170)	910 3121 ( 9.171)	911 3459 ( 9.172)	912 3354 ( 9.173)	913 3734 ( 9.174)	914 3E5F ( 9.175)	915 3070 ( 9.176)
陣	造	速	孫	逐	通	庭	述	途	逢	浮	陞	運	案	院
916 3F58 ( 9.177)	917 4224 ( 9.178)	918 422E ( 9.179)	919 4239 ( 9.180)	920 4360 ( 9.181)	921 4440 ( 9.182)	922 4460 ( 9.183)	923 447E ( 9.184)	924 4553 ( 9.185)	925 4629 ( 9.186)	926 4862 ( 9.187)	927 4845 ( 9.188)	928 4F22 ( 9.189)	929 3046 (10. 3)	930 3121 (10. 2)
員	益	悅	宴	桜	翁	恩	荷	家	夏	華	蚊	害	格	核
931 3077 (10. 3)	932 3157 (10. 4)	933 3159 (10. 5)	934 3163 (10. 6)	935 3A78 (10. 7)	936 3227 (10. 8)	937 3238 (10. 8)	938 3259 (10. 10)	939 3248 (10. 11)	940 3246 (10. 12)	941 3254 (10. 13)	942 3263 (10. 14)	943 3352 (10. 15)	944 3344 (10. 16)	945 3348 (10. 17)
株	陷	起	記	歸	鬼	飢	虐	宮	拳	胸	恐	脅	恭	訓
946 3374 (10. 18)	947 3459 (10. 19)	948 3527 (10. 20)	949 3520 (10. 21)	950 3522 (10. 22)	951 3534 (10. 23)	952 3532 (10. 24)	953 3554 (10. 25)	954 3550 (10. 26)	955 3573 (10. 27)	956 3638 (10. 28)	957 3632 (10. 29)	958 3630 (10. 30)	959 3633 (10. 31)	960 3731 (10. 32)
郡	惠	軒	兼	劍	儉	原	庫	個	悟	娛	高	校	航	候
961 3732 (10. 33)	962 3749 (10. 34)	963 382E (10. 35)	964 3773 (10. 36)	965 3775 (10. 37)	966 3770 (10. 38)	967 3836 (10. 39)	968 4048 (10. 40)	969 3844 (10. 41)	970 3867 (10. 42)	971 3864 (10. 43)	972 3862 (10. 44)	973 3938 (10. 45)	974 3852 (10. 46)	975 3875 (10. 47)
耕	降	貢	岡	骨	根	差	竣	座	栽	宰	財	劑	索	殺
976 3940 (10. 48)	977 395F (10. 49)	978 3957 (10. 50)	979 3964 (10. 51)	980 3970 (10. 52)	981 3A20 (10. 53)	982 3A39 (10. 54)	983 3A36 (10. 55)	984 3A42 (10. 56)	985 3A4F (10. 57)	986 3A48 (10. 58)	987 3A62 (10. 59)	988 3A5E (10. 60)	989 3A77 (10. 61)	990 3B26 (10. 62)
蚕	殘	紙	師	脂	時	疾	射	借	弱	酒	殊	珠	修	從
991 3B30 (10. 63)	992 3844 (10. 64)	993 3866 (10. 65)	994 3855 (10. 66)	995 3869 (10. 67)	996 387E (10. 68)	997 3C40 (10. 69)	998 3C40 (10. 70)	999 3C5A (10. 71)	1000 3C65 (10. 72)	1001 3C72 (10. 73)	1002 3C60 (10. 74)	1003 3E5E (10. 75)	1004 3D20 (10. 76)	1005 3E56 (10. 77)
純	殉	淮	書	徐	除	消	称	笑	祥	将	症	辱	真	振
1006 3063 (10. 78)	1007 305E (10. 79)	1008 3054 (10. 80)	1009 3071 (10. 81)	1010 3079 (10. 82)	1011 3070 (10. 83)	1012 3E43 (10. 84)	1013 3E4E (10. 85)	1014 3E50 (10. 86)	1015 3E40 (10. 87)	1016 3E20 (10. 88)	1017 3E49 (10. 89)	1018 3E28 (10. 90)	1019 3E3F (10. 91)	1020 3E56 (10. 92)
針	娠	浸	陣	衰	粹	畝	席	隻	扇	素	租	倉	桑	搜
1021 3F48 (10. 93)	1022 3F31 (10. 94)	1023 3F58 (10. 95)	1024 3F58 (10. 96)	1025 3F64 (10. 97)	1026 3F68 (10. 98)	1027 4026 (10. 99)	1028 4044 (10.100)	1029 4049 (10.101)	1030 4070 (10.102)	1031 4147 (10.103)	1032 4145 (10.104)	1033 4152 (10.105)	1034 372E (10.106)	1035 4150 (10.107)
造	速	息	孫	帶	泰	託	恥	致	值	畜	逐	秩	朕	通
1036 4224 (10.108)	1037 422E (10.109)	1038 4229 (10.110)	1039 4239 (10.111)	1040 4253 (10.112)	1041 4259 (10.113)	1042 4277 (10.114)	1043 4351 (10.115)	1044 4357 (10.116)	1045 4340 (10.117)	1046 435F (10.118)	1047 4360 (10.119)	1048 43E1 (10.120)	1049 415F (10.121)	1050 4440 (10.122)

庭	遞	哲	展	徒	途	島	討	党	凍	透	桃	唐	倒	洞
1051 4W6D (10, 123)	1052 447E (10, 124)	1053 452F (10, 125)	1054 453B (10, 126)	1055 454C (10, 127)	1056 455J (10, 128)	1057 456J (10, 129)	1058 4624 (10, 130)	1059 455E (10, 131)	1060 456D (10, 132)	1061 4629 (10, 133)	1062 456D (10, 134)	1063 455E (10, 135)	1064 455D (10, 136)	1065 4539 (10, 137)
特	箇	能	納	惱	破	馬	配	俳	倍	梅	畔	班	般	秘
1066 4643 (10, 138)	1067 463F (10, 139)	1068 473D (10, 140)	1069 473C (10, 141)	1070 473A (10, 142)	1071 474B (10, 143)	1072 474F (10, 144)	1073 475B (10, 145)	1074 475D (10, 146)	1075 475C (10, 147)	1076 475F (10, 148)	1077 484R (10, 149)	1078 4E49 (10, 150)	1079 453E (10, 151)	1080 4E6B (10, 152)
疲	被	姪	俵	病	浜	敏	浮	粉	紛	陞	勉	捕	浦	砲
1081 486B (10, 153)	1082 486F (10, 154)	1083 493J (10, 155)	1084 493B (10, 156)	1085 494Z (10, 157)	1086 494D (10, 158)	1087 495Z (10, 159)	1088 495Z (10, 160)	1089 4A34 (10, 161)	1090 4A36 (10, 162)	1091 4A45 (10, 163)	1092 4A59 (10, 164)	1093 4A6J (10, 165)	1094 4E3A (10, 166)	1095 4E24 (10, 167)
峰	倣	紡	剖	埋	脈	眠	娘	耗	紋	容	浴	流	留	旅
1096 4A7E (10, 168)	1097 4A6F (10, 169)	1098 4B4Z (10, 170)	1099 4B3B (10, 171)	1100 4B64 (10, 172)	1101 4C2I (10, 173)	1102 4C3Z (10, 174)	1103 4C3C (10, 175)	1104 4E57 (10, 176)	1105 4E66 (10, 177)	1106 4D46 (10, 178)	1107 4D6J (10, 179)	1108 4E2E (10, 180)	1109 4E3J (10, 181)	1110 4E39 (10, 182)
料	倫	淚	烈	連	恋	浪	朗	逸	陰	教	鄉	健	險	週
1111 4E4J (10, 183)	1112 4E5I (10, 184)	1113 4E5E (10, 185)	1114 4E7E (10, 186)	1115 4F2Z (10, 187)	1116 4E7B (10, 188)	1117 4F3Z (10, 189)	1118 4F2F (10, 190)	1119 3D6F (10, 191)	1120 3I2Z (10, 192)	1121 3E3E (10, 193)	1122 3E3F (10, 194)	1123 37I2 (10, 195)	1124 3B5I (10, 196)	1125 3E3B (10, 197)
進	隨	逮	陳	都	陶	陪	部	猛	郵	遊	陸	隆	陵	惡
1126 3F4A (10, 198)	1127 3F6F (10, 199)	1128 4Z6J (10, 200)	1129 4444 (10, 201)	1130 4554 (10, 202)	1131 4E2B (10, 203)	1132 476E (10, 204)	1133 4974 (10, 205)	1134 4C54 (10, 206)	1135 4D39 (10, 207)	1136 4D3J (10, 208)	1137 4E2E (10, 209)	1138 4E34 (10, 210)	1139 4E4D (10, 211)	1140 3D2D (11, 1)
移	異	尉	域	逸	陰	液	貨	菓	械	郭	掛	渴	乾	慚
1141 3D5C (11, 2)	1142 3D5B (11, 3)	1143 3D5J (11, 4)	1144 3D6E (11, 5)	1145 3D6F (11, 6)	1146 3I2Z (11, 7)	1147 3I5E (11, 8)	1148 3E5F (11, 9)	1149 3E5B (11, 10)	1150 3E2B (11, 11)	1151 3E54 (11, 12)	1152 3E5D (11, 13)	1153 3E69 (11, 14)	1154 3A2E (11, 15)	1155 3A2R (11, 16)
貫	患	眼	寄	基	規	偽	菊	脚	球	救	許	虛	魚	御
1156 345E (11, 17)	1157 343E (11, 18)	1158 346E (11, 19)	1159 347J (11, 20)	1160 347D (11, 21)	1161 3E2C (11, 22)	1162 3E3B (11, 23)	1163 3E46 (11, 24)	1164 3E5J (11, 25)	1165 3E6E (11, 26)	1166 3E5F (11, 27)	1167 3E7E (11, 28)	1168 3E7E (11, 29)	1169 3E7E (11, 30)	1170 3E6E (11, 31)
強	教	鄉	菌	偶	掘	揭	啓	經	健	險	現	康	黃	控
1171 3E2F (11, 32)	1172 3E3E (11, 33)	1173 3E3F (11, 34)	1174 3E5D (11, 35)	1175 3E7E (11, 36)	1176 372I (11, 37)	1177 3747 (11, 38)	1178 373C (11, 39)	1179 375D (11, 40)	1180 377Z (11, 41)	1181 3E3J (11, 42)	1182 3E3D (11, 43)	1183 3E2F (11, 44)	1184 3E2E (11, 45)	1185 3E3E (11, 46)
黑	混	紺	婚	祭	菜	細	採	濟	彩	齋	產	慘	視	執
1186 397E (11, 47)	1187 3A2E (11, 48)	1188 3A3D (11, 49)	1189 3A27 (11, 50)	1190 3A57 (11, 51)	1191 3A5B (11, 52)	1192 3A59 (11, 53)	1193 3A4E (11, 54)	1194 3A5I (11, 55)	1195 3A4C (11, 56)	1196 3A5B (11, 57)	1197 3E3R (11, 58)	1198 3E34 (11, 59)	1199 3E6E (11, 60)	1200 3C3E (11, 61)
捨	斜	赦	糶	寂	授	終	習	週	洪	宿	肅	淑	術	庶
1201 3C4E (11, 62)	1202 3C5D (11, 63)	1203 3C4F (11, 64)	1204 3C6J (11, 65)	1205 3C64 (11, 66)	1206 3C7B (11, 67)	1207 3D24 (11, 68)	1208 3D2C (11, 69)	1209 3D3E (11, 70)	1210 3D4Z (11, 71)	1211 3D49 (11, 72)	1212 3D4D (11, 73)	1213 3D4F (11, 74)	1214 3D5I (11, 75)	1215 3D6E (11, 76)
商	唱	章	涉	紹	訟	常	情	剩	進	深	紳	推	醉	崇
1216 3E2E (11, 77)	1217 3E27 (11, 78)	1218 3E4F (11, 79)	1219 3E44 (11, 80)	1220 3E5Z (11, 81)	1221 3E59 (11, 82)	1222 3E6F (11, 83)	1223 3E7D (11, 84)	1224 3E6A (11, 85)	1225 3F4A (11, 86)	1226 3F3C (11, 87)	1227 3F4Z (11, 88)	1228 3E64 (11, 89)	1229 3E6C (11, 90)	1230 3E7Z (11, 91)
清	盛	責	惜	雪	設	接	船	旋	組	粗	措	窓	巢	掃
1231 4D3E (11, 92)	1232 4D3E (11, 93)	1233 4D5E (11, 94)	1234 4D4B (11, 95)	1235 4D6E (11, 96)	1236 4D5F (11, 97)	1237 4D5C (11, 98)	1238 4I2E (11, 99)	1239 4D7B (11, 100)	1240 4I4B (11, 101)	1241 4I4E (11, 102)	1242 4I3C (11, 103)	1243 4I5B (11, 104)	1244 4I6E (11, 105)	1245 4I5D (11, 106)
側	族	率	逮	袋	第	脫	探	淡	斷	窒	著	鳥	張	帳
1246 4Z2E (11, 107)	1247 4Z3Z (11, 108)	1248 4E2B (11, 109)	1249 4Z5I (11, 110)	1250 4Z5E (11, 111)	1251 4Z6B (11, 112)	1252 4Z2E (11, 113)	1253 4Z3E (11, 114)	1254 4Z3B (11, 115)	1255 4Z47 (11, 116)	1256 4Z6Z (11, 117)	1257 4Z7B (11, 118)	1258 4Z3B (11, 119)	1259 4Z2E (11, 120)	1260 4Z2I (11, 121)

頂	彫	陳	停	笛	軫	添	都	盜	陶	悼	堂	動	得	豚
1261 443A (11.122)	1262 442B (11.123)	1263 444A (11.124)	1264 446A (11.125)	1265 452E (11.126)	1266 453E (11.127)	1267 453F (11.128)	1268 455A (11.129)	1269 457D (11.130)	1270 462B (11.131)	1271 456B (11.132)	1272 463E (11.133)	1273 453D (11.134)	1274 464D (11.135)	1275 465A (11.136)
軟	粘	腦	婆	敗	排	培	陪	舶	販	票	描	貧	婦	符
1276 467D (11.137)	1277 473A (11.138)	1278 473E (11.139)	1279 474C (11.140)	1280 475A (11.141)	1281 475B (11.142)	1282 475D (11.143)	1283 476E (11.144)	1284 477E (11.145)	1285 484E (11.146)	1286 493C (11.147)	1287 484J (11.148)	1288 454F (11.149)	1289 495B (11.150)	1290 456A (11.151)
部	副	閑	偏	崩	訪	望	麻	密	務	猛	問	野	詠	唯
1291 497A (11.152)	1292 497B (11.153)	1293 494A (11.154)	1294 465D (11.155)	1295 497B (11.156)	1296 482C (11.157)	1297 483E (11.158)	1298 486B (11.159)	1299 4C2B (11.160)	1300 4C3B (11.161)	1301 4C5A (11.162)	1302 4C6A (11.163)	1303 4C5E (11.164)	1304 4E7E (11.165)	1305 4C2B (11.166)
郵	庸	欲	翌	理	陸	略	粒	隆	涼	獾	陵	累	暉	詠
1306 4D3B (11.167)	1307 4D47 (11.168)	1308 4D5F (11.169)	1309 4D6E (11.170)	1310 4D7D (11.171)	1311 4E2E (11.172)	1312 4E2C (11.173)	1313 4E3B (11.174)	1314 4E3A (11.175)	1315 4E4B (11.176)	1316 4E4A (11.177)	1317 4E4D (11.178)	1318 4E5F (11.179)	1319 515F (11.180)	1320 515B (11.181)
過	階	遇	遂	墮	隊	達	遲	道	遍	陽	廊	握	偉	飲
1321 3E21 (11.182)	1322 3E2C (11.183)	1323 3E7B (11.184)	1324 3E6B (11.185)	1325 424A (11.186)	1326 426E (11.187)	1327 432B (11.188)	1328 435B (11.189)	1329 4E3B (11.190)	1330 4A57 (11.191)	1331 4D5B (11.192)	1332 4F2D (11.193)	1333 3D2E (12. 1)	1334 3D4E (12. 2)	1335 3D7B (12. 3)
雲	運	營	詠	越	援	奧	溫	過	賀	繪	開	階	街	覺
1336 314D (12. 4)	1337 313F (12. 5)	1338 314A (12. 6)	1339 315B (12. 7)	1340 315B (12. 8)	1341 3167 (12. 9)	1342 317C (12. 10)	1343 323B (12. 11)	1344 3261 (12. 12)	1345 326C (12. 13)	1346 332B (12. 14)	1347 332B (12. 15)	1348 332C (12. 16)	1349 333B (12. 17)	1350 335D (12. 18)
割	間	寒	堪	閑	敢	棺	款	換	喚	喜	期	貴	幾	揮
1351 336A (12. 19)	1352 345E (12. 20)	1353 342B (12. 21)	1354 342E (12. 22)	1355 3457 (12. 23)	1356 343A (12. 24)	1357 343D (12. 25)	1358 343E (12. 26)	1359 343B (12. 27)	1360 342D (12. 28)	1361 346E (12. 29)	1362 347C (12. 30)	1363 352E (12. 31)	1364 347E (12. 32)	1365 347B (12. 33)
棄	棋	欺	喫	給	距	曉	極	勤	筋	琴	遇	景	輕	敬
1366 347E (12. 34)	1367 347D (12. 35)	1368 3E3D (12. 36)	1369 3E4A (12. 37)	1370 3E6B (12. 38)	1371 3E77 (12. 39)	1372 3E47 (12. 40)	1373 3E4B (12. 41)	1374 3E5D (12. 42)	1375 3E5A (12. 43)	1376 3E57 (12. 44)	1377 3E7B (12. 45)	1378 374A (12. 46)	1379 375A (12. 47)	1380 374B (12. 48)
結	檢	堅	卷	減	湖	雇	港	硬	絞	項	慌	詐	最	裁
1381 376B (12. 49)	1382 3821 (12. 50)	1383 377B (12. 51)	1384 3777 (12. 52)	1385 383A (12. 53)	1386 385D (12. 54)	1387 385B (12. 55)	1388 3941 (12. 56)	1389 3945 (12. 57)	1390 394A (12. 58)	1391 396D (12. 59)	1392 383C (12. 60)	1393 383E (12. 61)	1394 3847 (12. 62)	1395 385E (12. 63)
策	酢	散	齒	詞	紫	滋	軸	濕	煮	集	就	衆	順	循
1396 3A7E (12. 64)	1397 3F5D (12. 65)	1398 3E3E (12. 66)	1399 387E (12. 67)	1400 386C (12. 68)	1401 3867 (12. 69)	1402 3C2E (12. 70)	1403 3C3A (12. 71)	1404 3C3E (12. 72)	1405 3C51 (12. 73)	1406 3D3B (12. 74)	1407 3D22 (12. 75)	1408 3D3D (12. 76)	1409 3D57 (12. 77)	1410 3D5B (12. 78)
暑	勝	証	象	燒	掌	粧	詔	晶	焦	硝	場	疊	植	殖
1411 3D8B (12. 79)	1412 3E21 (12. 80)	1413 3E5A (12. 81)	1414 3E5D (12. 82)	1415 3E4E (12. 83)	1416 3E3B (12. 84)	1417 3E53 (12. 85)	1418 3E5B (12. 86)	1419 3E3D (12. 87)	1420 3E47 (12. 88)	1421 3E4B (12. 89)	1422 3E6C (12. 90)	1423 3E7E (12. 91)	1424 3F2E (12. 92)	1425 3F2B (12. 93)
木	診	尋	遂	隨	晴	婿	稅	絕	然	善	訴	疎	創	裝
1426 3F3B (12. 94)	1427 3F47 (12. 95)	1428 3F5E (12. 96)	1429 3F6B (12. 97)	1430 3F6F (12. 98)	1431 4D3E (12. 99)	1432 4C3B (12.100)	1433 4D47 (12.101)	1434 4D6A (12.102)	1435 413E (12.103)	1436 4131 (12.104)	1437 414A (12.105)	1438 414E (12.106)	1439 414F (12.107)	1440 417E (12.108)
喪	葬	測	屬	尊	惰	墮	隊	貸	替	達	短	彈	遲	着
1441 415E (12.109)	1442 417E (12.110)	1443 422C (12.111)	1444 423D (12.112)	1445 423A (12.113)	1446 424E (12.114)	1447 424A (12.115)	1448 426E (12.116)	1449 425F (12.117)	1450 425B (12.118)	1451 432E (12.119)	1452 433B (12.120)	1453 434E (12.121)	1454 435B (12.122)	1455 435E (12.123)
貯	朝	超	脹	痛	提	堤	程	渡	答	等	登	湯	統	筒
1456 437B (12.124)	1457 442B (12.125)	1458 443E (12.126)	1459 4431 (12.127)	1460 444E (12.128)	1461 447E (12.129)	1462 446B (12.130)	1463 447B (12.131)	1464 454F (12.132)	1465 457E (12.133)	1466 457B (12.134)	1467 455D (12.135)	1468 457E (12.136)	1469 457D (12.137)	1470 457B (12.138)

痘	塔	道	童	鈍	癩	買	媒	博	飯	番	晚	蠻	悲	費
1471 4577 (12, 139)	1472 4563 (12, 140)	1473 4638 (12, 141)	1474 4638 (12, 142)	1475 465F (12, 143)	1476 4751 (12, 144)	1477 4763 (12, 145)	1478 476E (12, 146)	1479 476E (12, 147)	1480 4853 (12, 148)	1481 4856 (12, 149)	1482 4855 (12, 150)	1483 4854 (12, 151)	1484 4851 (12, 152)	1485 4871 (12, 153)
備	箏	評	富	普	復	幅	遍	補	募	報	貿	棒	帽	傍
1486 4877 (12, 154)	1487 492E (12, 155)	1488 493E (12, 156)	1489 4959 (12, 157)	1490 4961 (12, 158)	1491 497C (12, 159)	1492 497D (12, 160)	1493 4957 (12, 161)	1494 4954 (12, 162)	1495 4967 (12, 163)	1496 4973 (12, 164)	1497 4987 (12, 165)	1498 4960 (12, 166)	1499 4959 (12, 167)	1500 4935 (12, 168)
滿	無	愉	遊	雄	裕	猶	陽	葉	搖	揚	落	絡	痢	硫
1501 487E (12, 169)	1502 4C35 (12, 170)	1503 4C78 (12, 171)	1504 4D37 (12, 172)	1505 4D3A (12, 173)	1506 4D35 (12, 174)	1507 4D31 (12, 175)	1508 4D58 (12, 176)	1509 4D55 (12, 177)	1510 4D49 (12, 178)	1511 4D48 (12, 179)	1512 4D6E (12, 180)	1513 4D5D (12, 181)	1514 4E21 (12, 182)	1515 4E32 (12, 183)
量	罌	裂	廊	惑	腕	灣	違	遠	隔	遣	艇	愛	暗	意
1516 4E4C (12, 184)	1517 4E5D (12, 185)	1518 4E76 (12, 186)	1519 4F2D (12, 187)	1520 4F47 (12, 188)	1521 4F53 (12, 189)	1522 4F51 (12, 190)	1523 3D63 (12, 191)	1524 3173 (12, 192)	1525 3356 (12, 193)	1526 382F (12, 194)	1527 447A (12, 195)	1528 3D26 (13, 1)	1529 3D45 (13, 2)	1530 3D55 (13, 3)
違	遠	園	塩	煙	鉛	虞	暇	嫁	禍	雅	解	塊	慨	該
1531 3D63 (13, 4)	1532 3173 (13, 5)	1533 316D (13, 6)	1534 3176 (13, 7)	1535 316C (13, 8)	1536 3174 (13, 9)	1537 3E73 (13, 10)	1538 3D28 (13, 11)	1539 3274 (13, 12)	1540 3252 (13, 13)	1541 326D (13, 14)	1542 3272 (13, 15)	1543 327A (13, 16)	1544 333A (13, 17)	1545 333A (13, 18)
隔	較	樂	滑	漢	感	幹	奮	寬	義	詰	業	禁	愚	群
1546 3356 (13, 19)	1547 3353 (13, 20)	1548 335A (13, 21)	1549 336A (13, 22)	1550 3441 (13, 23)	1551 3436 (13, 24)	1552 3439 (13, 25)	1553 3428 (13, 26)	1554 3432 (13, 27)	1555 3541 (13, 28)	1556 354D (13, 29)	1557 3648 (13, 30)	1558 3658 (13, 31)	1559 3672 (13, 32)	1560 3752 (13, 33)
傾	繼	携	傑	絹	獻	遣	源	鼓	誇	碁	鉉	催	歲	載
1561 3738 (13, 34)	1562 3751 (13, 35)	1563 3748 (13, 36)	1564 376E (13, 37)	1565 3828 (13, 38)	1566 3825 (13, 39)	1567 382F (13, 40)	1568 3838 (13, 41)	1569 385D (13, 42)	1570 3858 (13, 43)	1571 3858 (13, 44)	1572 3858 (13, 45)	1573 3845 (13, 46)	1574 385D (13, 47)	1575 385C (13, 48)
債	罪	榨	試	詩	資	飼	嗣	雌	辞	慈	愁	酬	準	署
1576 3844 (13, 49)	1577 3861 (13, 50)	1578 3871 (13, 51)	1579 386E (13, 52)	1580 386D (13, 53)	1581 3871 (13, 54)	1582 3874 (13, 55)	1583 384C (13, 56)	1584 3873 (13, 57)	1585 3C2D (13, 58)	1586 387C (13, 59)	1587 3D25 (13, 60)	1588 3D37 (13, 61)	1589 3D6D (13, 62)	1590 3D7D (13, 63)
昭	傷	獎	詳	蒸	角	飾	新	寢	慎	睡	數	勢	誠	聖
1591 3E48 (13, 64)	1592 3D7D (13, 65)	1593 3E29 (13, 66)	1594 3E5C (13, 67)	1595 3E78 (13, 68)	1596 3F28 (13, 69)	1597 3E7E (13, 70)	1598 3F37 (13, 71)	1599 3F32 (13, 72)	1600 3F35 (13, 73)	1601 3F67 (13, 74)	1602 3F74 (13, 75)	1603 4D2A (13, 76)	1604 4D3F (13, 77)	1605 4D38 (13, 78)
跡	節	撰	戰	踐	禪	塑	想	僧	統	賊	損	滯	滝	嘆
1606 4D57 (13, 79)	1607 4D61 (13, 80)	1608 4D5D (13, 81)	1609 4D6F (13, 82)	1610 4128 (13, 83)	1611 4135 (13, 84)	1612 413A (13, 85)	1613 4158 (13, 86)	1614 414E (13, 87)	1615 4233 (13, 88)	1616 4231 (13, 89)	1617 4238 (13, 90)	1618 425A (13, 91)	1619 426C (13, 92)	1620 4332 (13, 93)
暖	置	稚	痴	蓄	腸	跳	貨	艇	鉄	電	殿	塗	働	督
1621 4348 (13, 94)	1622 4356 (13, 95)	1623 4355 (13, 96)	1624 4354 (13, 97)	1625 435F (13, 98)	1626 4432 (13, 99)	1627 4437 (13, 100)	1628 4442 (13, 101)	1629 447A (13, 102)	1630 4534 (13, 103)	1631 4545 (13, 104)	1632 4542 (13, 105)	1633 4549 (13, 106)	1634 452F (13, 107)	1635 4644 (13, 108)
農	搬	煩	頌	微	福	腹	墓	豊	飽	幕	夢	盟	滅	預
1636 474D (13, 109)	1637 4842 (13, 110)	1638 4851 (13, 111)	1639 4852 (13, 112)	1640 4879 (13, 113)	1641 4A21 (13, 114)	1642 4A22 (13, 115)	1643 4A68 (13, 116)	1644 4A2D (13, 117)	1645 4B3D (13, 118)	1646 4B58 (13, 119)	1647 4C34 (13, 120)	1648 4C43 (13, 121)	1649 4E47 (13, 122)	1650 4D42 (13, 123)
誉	腰	溶	裸	雷	酪	裏	虜	鈴	零	廉	路	楼	話	賄
1651 4D4D (13, 124)	1652 3978 (13, 125)	1653 4D4F (13, 126)	1654 4D67 (13, 127)	1655 4D68 (13, 128)	1656 4D6F (13, 129)	1657 4E22 (13, 130)	1658 4E3A (13, 131)	1659 4E66 (13, 132)	1660 4E6D (13, 133)	1661 4E77 (13, 134)	1662 4F29 (13, 135)	1663 4F3D (13, 136)	1664 4F43 (13, 137)	1665 4F45 (13, 138)
隱	醇	際	障	遭	適	維	隱	馱	演	歌	寡	箇	概	閣
1666 3123 (13, 139)	1667 395A (13, 140)	1668 3A5D (13, 141)	1669 3E63 (13, 142)	1670 4178 (13, 143)	1671 452C (13, 144)	1672 3D5D (14, 1)	1673 3123 (14, 2)	1674 3158 (14, 3)	1675 3163 (14, 4)	1676 324E (14, 5)	1677 3249 (14, 6)	1678 3255 (14, 7)	1679 3255 (14, 8)	1680 3355 (14, 9)

閑	管	慣	旗	疑	漁	境	銀	驅	語	誤	構	綱	酵	豪
1681 3458 (14, 30)	1682 3449 (14, 31)	1683 3437 (14, 32)	1684 3474 (14, 33)	1685 3533 (14, 34)	1686 3578 (14, 35)	1687 3620 (14, 36)	1688 3654 (14, 37)	1689 3666 (14, 38)	1690 3860 (14, 39)	1691 3850 (14, 40)	1692 3930 (14, 41)	1693 3948 (14, 42)	1694 3954 (14, 43)	1695 3968 (14, 44)
穀	酷	獄	魂	際	察	雜	算	酸	誌	磁	漆	種	需	銃
1696 3972 (14, 45)	1697 3973 (14, 46)	1698 3976 (14, 47)	1699 3932 (14, 48)	1700 3950 (14, 49)	1701 3921 (14, 50)	1702 3928 (14, 51)	1703 3938 (14, 52)	1704 3940 (14, 53)	1705 3867 (14, 54)	1706 3927 (14, 55)	1707 3937 (14, 56)	1708 3967 (14, 57)	1709 3978 (14, 58)	1710 3946 (14, 59)
緒	彰	障	靜	精	製	誓	說	錢	銃	漸	總	遭	層	增
1711 3067 (14, 60)	1712 3934 (14, 61)	1713 3963 (14, 62)	1714 4045 (14, 63)	1715 4039 (14, 64)	1716 4030 (14, 65)	1717 4040 (14, 66)	1718 4062 (14, 67)	1719 4320 (14, 68)	1720 4120 (14, 69)	1721 4132 (14, 70)	1722 4160 (14, 71)	1723 4178 (14, 72)	1724 4166 (14, 73)	1725 4170 (14, 74)
像	憎	態	奪	端	誕	嫡	徵	適	滴	摘	稻	銅	德	讀
1726 4170 (14, 75)	1727 4170 (14, 76)	1728 4256 (14, 77)	1729 4325 (14, 78)	1730 4330 (14, 79)	1731 4342 (14, 80)	1732 4364 (14, 81)	1733 4427 (14, 82)	1734 4520 (14, 83)	1735 4529 (14, 84)	1736 4526 (14, 85)	1737 3070 (14, 86)	1738 4630 (14, 87)	1739 4641 (14, 88)	1740 4649 (14, 89)
認	寧	髮	罰	戕	碑	鼻	漂	腐	復	聞	暮	慕	墨	膜
1741 4727 (14, 90)	1742 4728 (14, 91)	1743 4831 (14, 92)	1744 4839 (14, 93)	1745 4836 (14, 94)	1746 4864 (14, 95)	1747 4921 (14, 96)	1748 4934 (14, 97)	1749 4965 (14, 98)	1750 4923 (14, 99)	1751 4938 (14, 100)	1752 4968 (14, 101)	1753 4969 (14, 102)	1754 4947 (14, 103)	1755 4920 (14, 104)
漫	慢	鳴	銘	綿	模	網	誘	樣	踊	領	僚	綠	歷	曆
1756 4921 (14, 105)	1757 4870 (14, 106)	1758 4944 (14, 107)	1759 4943 (14, 108)	1760 4949 (14, 109)	1761 4947 (14, 110)	1762 4956 (14, 111)	1763 4936 (14, 112)	1764 4940 (14, 113)	1765 4959 (14, 114)	1766 4946 (14, 115)	1767 4930 (14, 116)	1768 4950 (14, 117)	1769 4972 (14, 118)	1770 4971 (14, 119)
練	漏	遺	雌	熟	連	線	選	遷	墜	踏	導	遺	慰	影
1771 4970 (14, 120)	1772 4933 (14, 121)	1773 3064 (14, 122)	1774 3973 (14, 123)	1775 3047 (14, 124)	1776 3065 (14, 125)	1777 4070 (14, 126)	1778 4124 (14, 127)	1779 4108 (14, 128)	1780 4446 (14, 129)	1781 4627 (14, 130)	1782 4633 (14, 131)	1783 3064 (15, 1)	1784 3056 (15, 2)	1785 3146 (15, 3)
銳	閱	謁	緣	橫	億	課	餓	確	監	飲	緩	器	輝	儀
1786 3154 (15, 4)	1787 3150 (15, 5)	1788 3154 (15, 6)	1789 3167 (15, 7)	1790 3223 (15, 8)	1791 3227 (15, 9)	1792 3250 (15, 10)	1793 3260 (15, 11)	1794 3340 (15, 12)	1795 3446 (15, 13)	1796 3437 (15, 14)	1797 3448 (15, 15)	1798 3467 (15, 16)	1799 3533 (15, 17)	1800 3537 (15, 18)
戲	窮	緊	勳	慶	劇	擊	潔	樞	稿	撮	贊	暫	賜	質
1801 3534 (15, 19)	1802 3567 (15, 20)	1803 3658 (15, 21)	1804 3720 (15, 22)	1805 3744 (15, 23)	1806 3760 (15, 24)	1807 3762 (15, 25)	1808 3769 (15, 26)	1809 3822 (15, 27)	1810 3946 (15, 28)	1811 3923 (15, 29)	1812 3937 (15, 30)	1813 3943 (15, 31)	1814 3972 (15, 32)	1815 3941 (15, 33)
趣	熟	遵	潤	諸	賞	衝	囑	審	震	穗	請	線	選	遷
1816 3971 (15, 34)	1817 3047 (15, 35)	1818 3065 (15, 36)	1819 3061 (15, 37)	1820 3074 (15, 38)	1821 3050 (15, 39)	1822 3057 (15, 40)	1823 3070 (15, 41)	1824 3939 (15, 42)	1825 3940 (15, 43)	1826 4966 (15, 44)	1827 4041 (15, 45)	1828 4070 (15, 46)	1829 4124 (15, 47)	1830 4128 (15, 48)
潛	葳	諾	談	駐	鑄	調	澄	潮	墜	締	敵	徹	撤	踏
1831 4078 (15, 49)	1832 4222 (15, 50)	1833 4274 (15, 51)	1834 4340 (15, 52)	1835 4373 (15, 53)	1836 4372 (15, 54)	1837 4434 (15, 55)	1838 4021 (15, 56)	1839 4420 (15, 57)	1840 4446 (15, 58)	1841 4479 (15, 59)	1842 4528 (15, 60)	1843 4530 (15, 61)	1844 4531 (15, 62)	1845 4627 (15, 63)
導	熱	輩	賠	箱	範	般	罷	標	賓	膚	敷	賦	舞	憤
1846 4633 (15, 64)	1847 4720 (15, 65)	1848 4754 (15, 66)	1849 4765 (15, 67)	1850 4822 (15, 68)	1851 4847 (15, 69)	1852 4857 (15, 70)	1853 4860 (15, 71)	1854 4938 (15, 72)	1855 4950 (15, 73)	1856 4966 (15, 74)	1857 4957 (15, 75)	1858 4964 (15, 76)	1859 4971 (15, 77)	1860 4930 (15, 78)
噴	墳	幣	弊	緋	舖	暴	撲	摩	魅	默	憂	養	窠	履
1861 4920 (15, 79)	1862 4927 (15, 80)	1863 4930 (15, 81)	1864 4940 (15, 82)	1865 4954 (15, 83)	1866 4950 (15, 84)	1867 4930 (15, 85)	1868 4950 (15, 86)	1869 4960 (15, 87)	1870 4925 (15, 88)	1871 4958 (15, 89)	1872 4928 (15, 90)	1873 4950 (15, 91)	1874 4952 (15, 92)	1875 4974 (15, 93)
慮	察	輪	靈	論	還	誕	避	絳	隣	緯	衛	憶	穩	懷
1876 4938 (15, 94)	1877 4940 (15, 95)	1878 4958 (15, 96)	1879 4960 (15, 97)	1880 4940 (15, 98)	1881 3954 (15, 99)	1882 4942 (15, 100)	1883 4972 (15, 101)	1884 4925 (15, 102)	1885 4959 (15, 103)	1886 3050 (16, 1)	1887 3152 (16, 2)	1888 3221 (16, 3)	1889 3234 (16, 4)	1890 3278 (16, 5)



壞	獲	館	還	憾	機	橋	凝	薰	怠	激	憲	賢	興	鏞
1891 3275 (116, 61)	1892 3340 (116, 71)	1893 3458 (116, 81)	1894 3454 (116, 91)	1895 3438 (116, 101)	1896 3521 (116, 111)	1897 3636 (116, 121)	1898 3645 (116, 131)	1899 3730 (116, 141)	1900 3746 (116, 151)	1901 3763 (116, 161)	1902 3778 (116, 171)	1903 3820 (116, 181)	1904 3810 (116, 191)	1905 3850 (116, 201)
衡	狠	錯	諮	儒	樹	獸	縱	孃	錠	親	薪	銓	整	績
1906 3955 (116, 211)	1907 3926 (116, 221)	1908 3978 (116, 231)	1909 3970 (116, 241)	1910 3974 (116, 251)	1911 3979 (116, 261)	1912 3943 (116, 271)	1913 3944 (116, 281)	1914 3956 (116, 291)	1915 3978 (116, 301)	1916 3946 (116, 311)	1917 3945 (116, 321)	1918 3956 (116, 331)	1919 3929 (116, 341)	1920 3921 (116, 351)
薦	操	濁	壇	築	頭	燈	糖	篤	曇	燃	濃	薄	縛	繁
1921 4126 (116, 361)	1922 4160 (116, 371)	1923 4279 (116, 381)	1924 4345 (116, 391)	1925 4358 (116, 401)	1926 4620 (116, 411)	1927 4575 (116, 421)	1928 4570 (116, 431)	1929 4646 (116, 441)	1930 4656 (116, 451)	1931 4733 (116, 461)	1932 4738 (116, 471)	1933 4778 (116, 481)	1934 4778 (116, 491)	1935 4849 (116, 501)
避	奮	壁	絳	膨	謀	藥	輸	諭	融	謠	擁	賴	隣	隸
1936 4872 (116, 511)	1937 4893 (116, 521)	1938 4849 (116, 531)	1939 4825 (116, 541)	1940 4844 (116, 551)	1941 4845 (116, 561)	1942 4874 (116, 571)	1943 4822 (116, 581)	1944 4821 (116, 591)	1945 4838 (116, 601)	1946 4858 (116, 611)	1947 4848 (116, 621)	1948 4854 (116, 631)	1949 4854 (116, 641)	1950 4850 (116, 651)
鍊	錄	赫	轄	環	犧	擬	謹	謙	嚴	講	購	懇	擦	謝
1951 4825 (116, 661)	1952 4831 (116, 671)	1953 3952 (117, 1)	1954 3960 (117, 2)	1955 3944 (117, 3)	1956 3938 (117, 4)	1957 3938 (117, 5)	1958 3960 (117, 6)	1959 3920 (117, 7)	1960 3931 (117, 8)	1961 3956 (117, 9)	1962 3958 (117, 10)	1963 3925 (117, 11)	1964 3924 (117, 12)	1965 3955 (117, 13)
爵	醜	縮	償	礁	績	鮮	織	霜	燥	鍛	聽	膳	優	翼
1966 3956 (117, 14)	1967 3939 (117, 15)	1968 3940 (117, 16)	1969 3978 (117, 17)	1970 3940 (117, 18)	1971 4055 (117, 19)	1972 4121 (117, 20)	1973 4121 (117, 21)	1974 4174 (117, 22)	1975 4167 (117, 23)	1976 4343 (117, 24)	1977 4430 (117, 25)	1978 4625 (117, 26)	1979 4625 (117, 27)	1980 4783 (117, 28)
覽	療	齡	穫	額	觀	簡	顏	騎	驗	蘭	顛	鎖	瞬	織
1981 4077 (117, 29)	1982 4145 (117, 30)	1983 4170 (117, 31)	1984 3947 (118, 1)	1985 3958 (118, 2)	1986 3945 (118, 3)	1987 3948 (118, 4)	1988 3949 (118, 5)	1989 3933 (118, 6)	1990 3933 (118, 7)	1991 4874 (118, 8)	1992 3932 (118, 9)	1993 3937 (118, 10)	1994 3926 (118, 11)	1995 3929 (118, 12)
職	繕	礎	駮	贈	題	徵	鎮	尉	難	藩	覆	癖	翻	曜
1996 3926 (118, 13)	1997 4136 (118, 14)	1998 4143 (118, 15)	1999 4178 (118, 16)	2000 4223 (118, 17)	2001 4258 (118, 18)	2002 4428 (118, 19)	2003 4443 (118, 20)	2004 4621 (118, 21)	2005 4671 (118, 22)	2006 4840 (118, 23)	2007 4824 (118, 24)	2008 4448 (118, 25)	2009 4850 (118, 26)	2010 4048 (118, 27)
濫	離	糧	臨	類	髓	韻	願	鏡	縲	警	鷄	鯨	靈	識
2011 4074 (118, 28)	2012 4825 (118, 29)	2013 4846 (118, 30)	2014 4857 (118, 31)	2015 4860 (118, 32)	2016 3971 (118, 33)	2017 3124 (119, 1)	2018 3968 (119, 2)	2019 3640 (119, 3)	2020 3728 (119, 4)	2021 3759 (119, 5)	2022 3755 (119, 6)	2023 3710 (119, 7)	2024 3928 (119, 8)	2025 3921 (119, 9)
髓	瀨	臟	爆	譜	簿	霧	麗	鄉	音	議	競	響	梟	護
2026 3977 (119, 10)	2027 4025 (119, 11)	2028 4221 (119, 12)	2029 4774 (119, 13)	2030 4968 (119, 14)	2031 4860 (119, 15)	2032 4838 (119, 16)	2033 4866 (119, 17)	2034 3641 (119, 18)	2035 3944 (120, 1)	2036 3625 (120, 2)	2037 3641 (120, 3)	2038 3710 (120, 4)	2039 4228 (120, 5)	2040 3984 (120, 6)
讓	釀	籍	騰	欄	艦	顧	魔	踴	露	驚	襲	鑑	屯	厄
2041 3979 (120, 7)	2042 3978 (120, 8)	2043 4052 (120, 9)	2044 4620 (120, 10)	2045 4075 (120, 11)	2046 3947 (121, 1)	2047 3950 (121, 2)	2048 4862 (121, 3)	2049 4076 (121, 4)	2050 4228 (121, 5)	2051 3643 (122, 1)	2052 3931 (122, 2)	2053 3935 (122, 3)	2054 4858 (124, 6)	2055 4271 (124, 5)
凹	皿	汁	仙	凸	缶	肌	朴	妄	杉	把	戾	拐	昆	肢
2056 3174 (125, 103)	2057 3826 (125, 104)	2058 3943 (125, 105)	2059 4067 (125, 106)	2060 4640 (125, 107)	2061 3940 (126, 119)	2062 4829 (126, 120)	2063 4851 (126, 121)	2064 4051 (126, 122)	2065 3979 (127, 151)	2066 4744 (127, 152)	2067 4061 (127, 153)	2068 3920 (127, 154)	2069 3928 (127, 155)	2070 3989 (127, 156)
尚	齊	泥	披	泡	抹	岬	杵	垣	挾	洪	甚	挑	亭	洞
2071 3930 (128, 194)	2072 4046 (128, 195)	2073 4525 (128, 196)	2074 4864 (128, 197)	2075 4822 (128, 198)	2076 4875 (128, 199)	2077 4028 (129, 200)	2078 4848 (129, 201)	2079 3940 (129, 190)	2080 3634 (129, 191)	2081 3939 (129, 192)	2082 3953 (129, 193)	2083 4228 (129, 194)	2084 4228 (129, 195)	2085 4228 (129, 196)
棧	酌	宵	唇	逝	栓	挿	俸	竜	涯	殼	喝	溪	堂	崎
2086 3837 (110, 211)	2087 3920 (110, 212)	2088 3920 (110, 213)	2089 3930 (110, 214)	2090 4042 (110, 215)	2091 4072 (110, 216)	2092 4156 (110, 217)	2093 4870 (110, 218)	2094 4835 (110, 219)	2095 3936 (111, 194)	2096 3940 (111, 195)	2097 3965 (111, 196)	2098 3940 (111, 197)	2099 3958 (111, 198)	2100 3941 (111, 199)

蛇	据	曹	眺	釣	偵	猫	瓶	堀	悠	渦	隅	傘	棚	塚
2101 375B 131.2001	2102 377B 131.2011	2103 418C 131.2021	2104 422F 131.2031	2105 446D 131.2041	2106 446S 131.2051	2107 472D 131.2061	2108 485S 131.2071	2109 485S 131.2081	2110 462R 131.2091	2111 315Z 132.1961	2112 5679 132.1971	2113 4531 132.1981	2114 431R 132.1991	2115 431D 132.2001

搭	棟	扉	霽	墀	猿	靴	褐	頑	嫌	溝	漠	鉢	遮	塾
2116 45EE 132.2011	2117 45E7 132.2021	2118 456Z 132.2031	2119 4F37 132.2041	2120 44ED 132.2051	2121 316E 133.1451	2122 3724 133.1461	2123 396C 133.1471	2124 3466 133.1481	2125 3713 133.1491	2126 354Z 133.1501	2127 477B 133.1511	2128 45C3 133.1521	2129 3C87 134.1101	2130 3D4E 134.1131

馱	漬	僕	稼	湍	纒	槽	褒	壕	磨	矯	濯	頻	襟	癒
2131 424C 134.1141	2132 445Z 134.1151	2133 484D 134.1161	2134 5254 135.1041	2135 333S 135.1051	2136 46C0 135.1061	2137 416S 135.1071	2138 482B 135.1081	2139 5E6C 135.6B1	2140 4861 135.631	2141 3E3R 137.321	2142 427S 137.331	2143 4851 137.341	2144 5E6F 138.341	2145 477E 138.351

藻	朝	羅
2146 4174 139.191	2147 474E 139.201	2148 435S 139.211

F	f <sub>1</sub>	f <sub>r</sub>	F	f <sub>1</sub>	f <sub>r</sub>	F	f <sub>1</sub>	f <sub>r</sub>
暗	→ 日	音	勘	→ 甚	力	級	→ 糸	及
姻	→ 女	因	堪	→ 土	甚	給	→ 糸	合
韻	→ 音	員	棺	→ 木	官	許	→ 言	午
映	→ 日	央	肝	→ 月	干	吟	→ 口	今
詠	→ 言	永	艦	→ 舟	監	駟	→ 馬	区
馭	→ 馬	尺	鑑	→ 金	監	靴	→ 革	化
橫	→ 木	黃	岐	→ 山	支	訓	→ 言	川
欧	→ 区	欠	機	→ 木	幾	群	→ 君	羊
加	→ 力	口	紀	→ 糸	己	計	→ 言	十
嫁	→ 女	家	規	→ 夫	見	鯨	→ 魚	京
課	→ 言	果	記	→ 言	己	結	→ 糸	吉
蚊	→ 虫	文	軌	→ 車	九	嫌	→ 女	兼
塊	→ 土	鬼	輝	→ 光	軍	獻	→ 南	犬
械	→ 木	戒	騎	→ 馬	奇	謙	→ 言	兼
絵	→ 糸	会	議	→ 言	義	軒	→ 車	干
概	→ 木	既	喫	→ 口	契	弦	→ 弓	玄
格	→ 木	各	詰	→ 言	吉	現	→ 王	見
較	→ 車	交	脚	→ 月	却	枯	→ 木	古
轄	→ 車	害	吸	→ 口	及	娛	→ 女	吳
株	→ 木	朱	球	→ 王	求	誤	→ 言	吳

F	fe	fr	F	fe	fr	F	fe	fr	
功	→	工	力	紙	→	糸	氏		
効	→	交	力	肢	→	月	支		
好	→	女	子	脂	→	月	旨		
校	→	木	交	詞	→	言	司		
硬	→	石	更	詩	→	言	寺		
紅	→	糸	工	試	→	言	式		
絞	→	糸	交	誌	→	言	志		
鉦	→	金	広	賜	→	貝	易		
腰	→	月	要	時	→	日	寺		
紺	→	糸	甘	辞	→	舌	辛		
砂	→	石	少	軸	→	車	由		
細	→	糸	田	執	→	幸	丸		
材	→	木	才	射	→	身	寸		
財	→	貝	才	斜	→	余	斗		
坂	→	土	反	謝	→	言	射		
崎	→	山	奇	取	→	耳	又		
錯	→	金	昔	珠	→	王	朱		
始	→	女	台	終	→	糸	冬		
姉	→	女	市	縱	→	糸	從		
枝	→	木	支	銃	→	金	充		
						縮	→	糸	宿
						純	→	糸	屯
						所	→	戸	斤
						緒	→	糸	者
						諸	→	言	者
						助	→	且	力
						叙	→	余	又
						昭	→	日	召
						松	→	木	公
						硝	→	石	肖
						礁	→	石	焦
						紹	→	糸	召
						訟	→	言	公
						証	→	言	正
						詔	→	言	召
						詳	→	言	羊
						鐘	→	金	童
						城	→	土	成
						錠	→	金	定
						囑	→	口	屬

分離有意文字 ( 61/266 ~ 120/266 )

F	f <sub>l</sub>	f <sub>r</sub>	F	f <sub>l</sub>	f <sub>r</sub>	F	f <sub>l</sub>	f <sub>r</sub>
植	→ 木	直	線	→ 糸	泉	彈	→ 弓	單
觸	→ 角	虫	銑	→ 金	先	談	→ 言	炎
紳	→ 糸	申	鮮	→ 魚	羊	知	→ 矢	口
針	→ 金	十	繕	→ 糸	善	恥	→ 耳	心
吹	→ 口	欠	粗	→ 米	且	柱	→ 木	主
炊	→ 火	欠	組	→ 糸	且	鑄	→ 金	壽
睡	→ 目	垂	訴	→ 言	斥	駐	→ 馬	主
錘	→ 金	垂	双	→ 又	又	張	→ 弓	長
枢	→ 木	区	槽	→ 木	曹	町	→ 田	丁
姓	→ 女	生	相	→ 木	目	眺	→ 目	兆
晴	→ 日	青	臟	→ 月	蔵	脹	→ 月	長
精	→ 米	青	統	→ 糸	壳	調	→ 言	周
誠	→ 言	成	孫	→ 子	系	勅	→ 束	力
請	→ 言	青	村	→ 木	寸	鎮	→ 金	真
静	→ 青	争	馱	→ 馬	太	坪	→ 土	平
析	→ 木	斤	胎	→ 月	台	鈞	→ 金	勺
績	→ 糸	責	胎	→ 月	台	堤	→ 土	是
切	→ 七	刀	諾	→ 言	若	締	→ 糸	帝
絕	→ 糸	色	短	→ 矢	豆	艇	→ 舟	廷
栓	→ 木	全	誕	→ 言	延	訂	→ 言	丁
			鍛	→ 金	段			

分離有意文字 ( 121/266 ~ 180/266 )

F	f <sub>2</sub>	f <sub>1</sub>	F	f <sub>2</sub>	f <sub>1</sub>	F	f <sub>2</sub>	f <sub>1</sub>
的	白	勺	納	糸	內	譜	言	普
鉄	金	失	破	石	皮	賦	貝	武
吐	口	土	杯	木	不	粉	米	分
奴	女	又	肺	月	市	紛	糸	分
桃	木	兆	媒	女	某	柄	木	丙
棟	木	東	梅	木	每	砲	石	包
灯	火	丁	舶	舟	白	胞	月	包
糖	米	唐	爆	火	暴	訪	言	方
統	糸	充	畑	火	田	坊	土	方
討	言	寸	鉢	金	本	妨	女	方
動	重	力	板	木	反	棒	木	奉
胴	月	同	版	片	反	紡	糸	方
銅	金	同	畔	田	半	肪	月	方
読	言	売	販	貝	反	謀	言	某
鈍	金	屯	晚	日	免	堀	土	屈
軟	車	欠	妃	女	己	翻	番	羽
如	女	口	妃碑	女石	己卑	埋	土	里
認	言	忍	姬	女	臣	妹	女	未
燃	火	然	標	木	票	味	口	未
粘	米	占	評	言	平	岬	山	甲

分離有意文字 (181/266 ~ 240/266)

F	f <sub>l</sub>	f <sub>r</sub>	F	f <sub>l</sub>	f <sub>r</sub>
妙	→ 女	少	齡	→ 齒	令
眠	→ 目	民	練	→ 糸	東
娘	→ 女	良	鍊	→ 金	東
明	→ 日	月	炉	→ 火	戶
銘	→ 金	名	話	→ 言	舌
鳴	→ 口	鳥	賄	→ 貝	有
紋	→ 糸	文			
野	→ 里	予			
約	→ 糸	勺			
訊	→ 言	尺			
誘	→ 言	秀			
欲	→ 谷	欠			
絡	→ 糸	各			
理	→ 王	里			
略	→ 田	各			
粒	→ 米	立			
料	→ 米	斗			
糧	→ 米	量			
林	→ 木	木			
鈴	→ 金	令			

分離有意文字 ( 241/266 ~ 266/266 )

(1) 使用した品詞 (特殊記号も含む)

接頭語, 接尾語, 助数詞, 造語成分,  
名詞, 形式名詞, 代名詞, 連体詞, 副詞,  
接続詞, 感動詞, 自動詞, 他動詞, 形容詞, 形容動詞, 助動詞,  
格助詞, 接続助詞, 副助詞, 係り助詞, 終助詞,  
間投助詞, 並列助詞,  
準体助詞, 補助動詞, 補助形容詞, 連語, 記号, 慣用句,  
  
数字, 1文字名詞, 平仮名, 片仮名, 漢字, 読点, 句点  
文節, 準文節, 片仮名列, 文末, 空記号

(2) 使用した品詞群シンボル

活用語 = { 自動詞, 他動詞, 形容詞, 形容動詞, 助動詞,  
補助動詞, 補助形容詞 }

用言 = { 自動詞, 他動詞, 形容詞, 形容動詞, 助動詞,  
補助動詞, 補助形容詞 }

体言 = { 名詞, 形式名詞, 代名詞, 1文字名詞 }

助詞 = { 格助詞, 接続助詞, 副助詞, 係り助詞, 終助詞,  
間投助詞, 並列助詞, 準体助詞 }

五段活用動詞 = { 自動詞 (五段活用), 他動詞 (五段活用) }

上一段活用動詞 = { 自動詞 (上一段活用), 他動詞 (上一段活用) }

下一段活用動詞 = { 自動詞 (下一段活用), 他動詞 (下一段活用) }

カ変動詞 = { 自動詞 (カ変), 他動詞 (カ変) }

サ変動詞 = { 自動詞 (サ変), 他動詞 (サ変) }

音便形を持つ五段活用動詞 = { 自動詞 (くが、な、ば、ま) 行五段活用 }



，他動詞（くが、な、ば、ま）行五段活用）}

音便形を持たない五段活用動詞 = 五段活用動詞から

’音便形を持つ五段活用動詞’を除いたもの

音便形を持たない用言 = 用言から’音便形を持つ五段活用動詞’を除いたもの

形容動詞以外の用言 = { 自動詞，他動詞，形容詞，  
補助動詞，補助形容詞 }

形容動詞以外の活用語 = { 自動詞，他動詞，形容詞，助動詞，  
補助動詞，補助形容詞 }

助詞群A = { 格助詞 { から、と、の } }

副助詞 { ほど、ばかり、だけ、のみ、など、きり、まで } }

助詞群B = { が、を } を除く格助詞

助詞群C = { が } を除く格助詞

助動詞群A = { 助動詞 { する、させる、れる、られる } }

助動詞群B = { 助動詞 { ようだ、そうだ、だ } }

### ( 3 ) 使用した接続規則

以下の記法に従う

- $A + B \implies C$  ; AとBが接続してCになる。

- $A_1, A_2, \dots, A_n + B \implies C_1, C_2$

これは、 $A_1 + B \implies C_1$

$A_2 + B \implies C_1$

•

•

$A_n + B \implies C_1$

$A_1 + B \implies C_2$

$A_2 + B \implies C_2$

•

•

$A_n + B \implies C_2$

と等価である。

- A, A<sub>1</sub>, A<sub>2</sub>, B, C<sub>1</sub>, C<sub>2</sub> は以下のいずれかである。

- 品詞等 (品詞、品詞群シンボル)

- $A \{ x_1, x_2, \dots, x_n \}$ 。

これは品詞等Aの  $x_1, x_2, \dots, x_n$  という要素。

- A (活用型)。

これは品詞等Aの中で括弧内の活用型のものを示す。

- $[ A_1 \cdot A_2 \cdot \dots \cdot A_n ]$  (活用型)。

これは  $A_1$  (活用型),  $A_2$  (活用型), ...,  $A_n$  (活用型) と等価。

接続規則を次頁から示す。但し、規則は接続する品詞等に対応して1 2タイプに分類した。

## ①助動詞に接続する規則

[ 五段活用動詞・形容詞・形容動詞・

助動詞{ない, たい, だ, ようだ, そうだ, ます, です, た} ](未然) +  
助動詞{う} ==> 文節, 準文節

[ 上一段活用動詞・下一段活用動詞・カ変 ](未然), サ変動詞(未然) +  
助動詞{させる} ==> 文節, 準文節

五段活用動詞(未然), サ変動詞(未然) + 助動詞{せる} ==> 文節, 準文節

[ 用言・助動詞 ](連用), 助動詞{なさ} + 助動詞{そうだ}  
==> 文節, 準文節

[ 自動詞・助動詞群A・助動詞{た, たい, た, だ, ない} ](連体) +  
助動詞{そうだ} ==> 文節, 準文節

[ 音便形を持つ用言・助動詞 ](連用) + 助動詞{た} ==> 文節, 準文節

音便形を持つ五段活用(連用) + 助動詞{だ} ==> 文節, 準文節

体言, 助詞群A, 格助詞{を, が} + 助動詞{だ} ==> 文節, 準文節

体言, 助詞, [ 形容動詞・助詞群B ](語幹),

[ 形容詞・助動詞{ない, たい, らしい} ](終止) + 助動詞{です}  
==> 文節, 準文節

[ 自動詞(未然), 形容詞・形容動詞・助動詞群B・  
助動詞{ない, たい, らしい} ](連用) + 助動詞{ない}  
==> 文節, 準文節

[ 五段活用動詞・上一段活用動詞・下一段活用動詞・  
カ変・助動詞{ます}・助動詞群A ](未然), サ変動詞(未然)  
+ 助動詞{ぬ} ==> 文節, 準文節

[ 五段活用動詞・助動詞{ます}・サ変動詞・カ変 ](終止),  
[ 上一段活用動詞・下一段活用動詞・カ変・助動詞群A ](未然),  
サ変動詞(未然) + 助動詞{まい}  
=> 文節, 準文節

[ 自動詞・助動詞群A ](連用) + 助動詞{ます} => 文節, 準文節

体言, 活用語(連体) + 助動詞{みたいだ} => 文節, 準文節

[ 上一段活用動詞・下一段活用動詞・カ変(未然), サ変動詞 ](未然) +  
助動詞{よう} => 文節, 準文節

[ 用言・助動詞{たい, ない, たい}・助動詞群A ](連体),  
準体助詞{の}, 格助詞{の}, 連体詞 + 助動詞{ようだ}  
=> 文節, 準文節

[ 自動詞・形容詞・助動詞{ない, たい, た, ぬ} ](終止), 体言, 副詞,  
形容動詞(語幹), 助詞 + 助動詞{らしい} => 文節, 準文節

[ 上一段活用動詞・下一段活用動詞・カ変・助動詞群A ](未然) +  
助動詞{られる} => 文節, 準文節

[ 五段活用動詞・サ変動詞 ](未然) + 助動詞{れる} => 文節, 準文節

[ 自動詞・助動詞群A ](連用) + 助動詞{たい} => 文節, 準文節

## ②格助詞に接続する規則

体言，助詞群A { 'から' を除く }，接続助詞 { て }，  
[ 音便形を持たない五段活用動詞・助動詞 { た，だ } ] (連用)  
+ 格助詞 { から } ==> 文節，準文節

体言，副助詞，係り助詞 {こそ，しも，も}，  
並列助詞 { たり，と，とか，やら } + 格助詞 { が }  
==> 文節，準文節

体言，助詞群A，格助詞 { へ }，活用語 (終止)，助詞，副詞 + 格助詞 { って }  
==> 文節，準文節

体言，助詞群A + 格助詞 { で } ==> 文節，準文節

体言，助詞群A，格助詞 { へ }，活用語 (終止)，副助詞，並列助詞，  
格助詞，副詞 + 格助詞 { と } ==> 文節，準文節

体言，助詞群A + 格助詞 { として } ==> 文節，準文節

体言，[ 自動詞・助動詞群A ] (連用)，助詞群A + 格助詞 { に }  
==> 文節，準文節

体言，[ 自動詞・助動詞群A ] (終止)，接続助詞 { から } + 格助詞 { には }  
==> 文節，準文節

体言，副詞，並列助詞 { たり } + 格助詞 { の }  
==> 文節，準文節

体言，準体助詞 { の }，副助詞 { など，だけ } + 格助詞 { へ }  
==> 文節，準文節

体言，助詞群A，[ 用言・助動詞群A・助動詞 { た } ] (連体)  
+ 格助詞 { より } ==> 文節，準文節

体言，準体助詞，助詞群A { 'と，きり' を除く }，並列助詞 { と } +  
格助詞 { を } ==> 文節，準文節

### ③接続助詞に接続する規則

[用言・助動詞](終止) + 接続助詞(から) ==> 文節, 準文節

活用語(終止) + 接続助詞(が) ==> 文節, 準文節

活用語(終止) + 接続助詞(けど) ==> 文節, 準文節

活用語(終止) + 接続助詞(けれど) ==> 文節, 準文節

活用語(終止) + 接続助詞(けれども) ==> 文節, 準文節

活用語(終止) + 接続助詞(し) ==> 文節, 準文節

活用語(終止), 形容詞(連用), 助動詞(た)(終止) + 接続助詞(って)  
==> 文節, 準文節

自動詞(連用) + 接続助詞(つ) ==> 文節, 準文節

自動詞(連用) + 接続助詞(つつ) ==> 文節, 準文節

[音便形を持たない用言・助動詞](連用) + 接続助詞(て) ==> 文節, 準文節

[音便形を持たない用言・助動詞](連用) + 接続助詞(ても)  
==> 文節, 準文節

音便形を持つ五段活用(連用), 助動詞(ない)(終止) + 接続助詞(で)  
==> 文節, 準文節

音便形を持つ五段活用(連用), 助動詞(ない)(終止) + 接続助詞(でも)  
==> 文節, 準文節

活用語(終止) + 接続助詞(と) ==> 文節, 準文節

助動詞(た) + 接続助詞(ところが) ==> 文節, 準文節

助動詞(た), 自動詞(終止) + 接続助詞(ところで) ==> 文節, 準文節

自動詞(終止) + 接続助詞(ところを) ==> 文節, 準文節

[ 自動詞・形容詞・助動詞群A・助動詞{た, ない, たい} ](終止),  
[ 形容動詞・助動詞群B ](語幹), 体言, 助詞群B, 準体助詞,  
副助詞{まで, だけ}, 副詞 + 接続助詞{どころか} ==> 文節, 準文節

活用語(連体) + 接続助詞{ので} ==> 文節, 準文節

活用語(連体) + 接続助詞{のに} ==> 文節, 準文節

活用語(假定) + 接続助詞{ば} ==> 文節, 準文節

助動詞{た}(連体) + 接続助詞{まま} ==> 文節, 準文節

活用語(連体) + 接続助詞(ものから) ==> 文節, 準文節

活用語(連体) + 接続助詞(もので) ==> 文節, 準文節

活用語(連体) + 接続助詞(ものの) ==> 文節, 準文節

活用語(連体) + 接続助詞(ものを) ==> 文節, 準文節

名詞, 副詞, [形容詞・形容動詞](語幹), 形容詞(連体),  
[自動詞・助動詞群A](連用) + 接続助詞(ながら) ==> 文節, 準文節

準体助詞(の), 活用語(連体) + 接続助詞(くせに) ==> 文節, 準文節

#### ④副助詞に接続する規則

形容動詞以外の用言(連体), 形容動詞(語幹), 名詞, 助動詞(だ)(連体),  
助詞群A, 格助詞(が), 副助詞(でも, のみ, ばかり)  
+ 副助詞(か) ==> 文節, 準文節

体言, 活用語(連体), 助動詞(た)(連体) + 副助詞(きり) ==> 文節, 準文節

体言, 活用語(連体), 連体詞 + 副助詞(くらい) ==> 文節, 準文節

体言, 活用語(連体), 連体詞 + 副助詞(くらい) ==> 文節, 準文節

活用語(連用), 体言, 助詞群B, 格助詞(を),  
副助詞(のみ, だけ, など, か) + 副助詞(さえ) ==> 文節, 準文節

体言, 自動詞(連体), [形容詞・形容動詞](連用), 助詞群C,  
副助詞(など, のみ, だけ, まで), 並列助詞(か)  
+ 副助詞(しか) ==> 文節, 準文節

体言, 助詞群C + 副助詞(すら) ==> 文節, 準文節

活用語(連体), 名詞, 助詞群B + 副助詞(だけ) ==> 文節, 準文節

体言, 準体助詞, 助詞群A, 助詞群C, 接続助詞(て, で)  
+ 活用語(連用), 副詞 + 副助詞(でも) ==> 文節, 準文節



助数詞 + 副助詞{として} ==> 文節, 準文節

体言 + 副助詞{とて} ==> 文節, 準文節

活用語(終止), 形容動詞以外の用言(連用), 形容動詞(連用),  
接続助詞{て}, 副助詞{たり}, サ変動詞(語幹), 格助詞{がを除く}  
体言 + 副助詞{など} ==> 文節, 準文節

活用語(終止), 活用語(命令), 助詞, 体言 + 副助詞{などと} ==> 文節, 準文節

活用語(終止), 活用語(命令), 助詞, 体言 + 副助詞{などは} ==> 文節, 準文節

活用語(終止), 活用語(命令), 助詞, 体言 + 副助詞{などが} ==> 文節, 準文節

活用語(終止), 活用語(命令), 助詞, 体言 + 副助詞{なども} ==> 文節, 準文節

体言, 助詞 + 副助詞{なんて} ==> 文節, 準文節

活用語(連体), 名詞, 助詞群C + 副助詞{のみ} ==> 文節, 準文節

活用語(連体), 名詞, 助詞群B, 副助詞{ほど, など} + 副助詞{ばかり}  
==> 文節, 準文節

助数詞 + 副助詞{ずつ} ==> 文節, 準文節

体言, 準体助詞, 活用語(終止) + 副助詞{ほど} ==> 文節, 準文節

体言, 副詞, [用言・助動詞群A・助動詞{ない, た}](連用), 格助詞,  
副助詞{くらい, ずつ, など, ばかり, ほど, やら, たり},  
接続助詞{て} + 副助詞{まで} ==> 文節, 準文節

[音便形を持たない用言・助動詞{だ, そうだ}・助動詞群A](連用) +  
副助詞{たり} ==> 文節, 準文節

#### ㊤係り助詞に接続する規則

体言，助詞群B + 係り助詞{こそ} ==> 文節，準文節

活用語(連用) + 係り助詞{こそすれ} ==> 文節，準文節

体言 + 係り助詞{って} ==> 文節，準文節

体言，助詞 + 係り助詞{とは} ==> 文節，準文節

体言，活用語(連用)，助詞群A，助詞群B，形容動詞(連用) + 係り助詞{は}  
==> 文節，準文節

体言，活用語(連用)，助詞群A，助詞群B，格助詞{を}，形容動詞(連用)  
+ 係り助詞{も} ==> 文節，準文節

#### ㊥終助詞に接続する規則

活用語(終止) + 終助詞{けれど} ==> 文節，文末の文節，準文節

活用語(終止) + 終助詞{けれども} ==> 文節，文末の文節，準文節

### ㊦ 並列助詞に接続する規則

活用語(連体), 形容詞(連用), 体言, 副詞, 格助詞, 副助詞, 形容動詞(語幹)  
+ 並列助詞{か} ==> 文節, 準文節

[ 音便形を持たない用言・助動詞{だ, そうだ}・助動詞群A ](連用)  
+ 並列助詞{たり} ==> 文節, 準文節

音便形を持つ五段活用(連用) + 並列助詞{だり} ==> 文節, 準文節

体言, 準体助詞 + 並列助詞{と} ==> 文節, 準文節

体言, 活用語(終止) + 並列助詞{とか} ==> 文節, 準文節

体言, 準体助詞 + 並列助詞{に} ==> 文節, 準文節

体言, 活用語(終止) + 並列助詞{の} ==> 文節, 準文節

体言, 準体助詞 + 並列助詞{や} ==> 文節, 準文節

体言, 準体助詞, 活用語(連体) + 並列助詞{やら} ==> 文節, 準文節

### ㊧ 準体助詞に接続する規則

体言, 活用語(連体), 連体詞 + 準体助詞{の} ==> 名詞, 準文節

### ㊨ 自動詞活用語に接続する規則

用言(連用) + 自動詞活用語 ==> 文節

補助用言(連用) + 自動詞活用語 ==> 文節

接続詞 + 自動詞活用語 ==> 文節

体言 + 自動詞活用語 ==> 文節

#### ⑩空記号に接続する規則

連体詞 + 空記号 ==> 文節

副詞 + 空記号 ==> 文節

接続詞 + 空記号 ==> 文節

感動詞 + 空記号 ==> 文節

用言(連用) + 空記号 ==> 文節

用言(終止) + 空記号 ==> 文節, 文末の文節

用言(連体) + 空記号 ==> 文節

補助用言(連用) + 空記号 ==> 文節

補助用言(終止) + 空記号 ==> 文節, 文末の文節

補助用言(連体) + 空記号 ==> 文節

記号 + 空記号 ==> 助数詞

記号 + 空記号 ==> 文節

片仮名列 + 空記号 ==> 名詞

助動詞(終止) + 空記号 ==> 文末の文節

### ㊦その他

接頭語 + 名詞 ==> 名詞

名詞 + 名詞 ==> 名詞

名詞 + 接尾語 ==> 名詞

活用語(連体) + 形式名詞 ==> 文節, 準文節

連体詞 + 命令 ==> 文節

数字 + 助数詞 ==> 数字

記号 + 記号 ==> 記号

片仮名 + 片仮名 ==> 片仮名列

片仮名列 + 片仮名 ==> 片仮名列

数字 + 数字 ==> 数字, 名詞

数字 + 助数詞 ==> 名詞, 文節

### ㊧追加

文末の文節 + 句点 ==> 文節

文節 + 自動詞活用語 ==> 文節

体言 + 格助詞 ==> 文節

名詞, 形式名詞, 代名詞 + 空記号 ==> 文節

## 著者文献リスト

(論文、研究会、全国大会、特許、賞罰)

### 論文

- (1) 村瀬, 木村, 吉村, 三宅: "パターン整合法における特性核の改良とその手書き平仮名文字認識への応用", 信学論(D), J64-D, 3, pp.276-283, 1981.
- (2) 村瀬, 若原, 梅田: "候補ラティス法による手書きフローチャートのオンライン認識", 信学論(D), J66-D, 6, pp.675-682, 1983.
- (3) 村瀬, 若原, 梅田: "接続ルールを導入した候補ラティス法によるオンライン手書き線図形認識", 信学論(D), J67-D, 3, pp.273-280, 1984.
- (4) 村瀬, 若原, 梅田: "候補文字ラティス法による枠無し筆記文字列のオンライン認識", 信学論(D), J68-D, 4, pp.765-772, 1985.
- (5) 村瀬, 若原, 梅田: "接続ルールを用いたトップダウン処理によるオンライン手書き線図形認識", 研究実用化報告、第34巻、第1号, pp.97-109, 1985.
- (6) H.Murase, T.Wakahara: "Online Hand-Sketched Figure Recognition", The Journal of the Pattern Recognition Society, Vol.19, No.2, pp.147-160, 1986.
- (7) 村瀬, 新谷, 若原, 梅田: "言語情報を導入した手書き文字列からの文字の切り出しと認識" 信学論(D), J69-D, 9, pp.1292-1301 1986.

### 研究会

- (1) 村瀬, 木村, 吉村, 三宅: "パターン類似度法による手書き平仮名文字認識の実験", 信学技報, PRL79-2, pp.9-18, 1979.
- (2) 村瀬, 木村, 吉村, 三宅: "変動吸収特性核を用いた手書き平仮名文字認識", 信学技報, PRL79-87, pp.7-12, 1980.
- (3) 村瀬, 若原, 梅田: "タブレット入力による手書き線図形認識アルゴリズムの検討", 信学技報, PRL81-69, pp.17-24, 1982.
- (4) 村瀬, 若原, 梅田: "接続ルールを考慮した候補ラティス法による手書きフローチャートのオンライン認識", 信学技報, PRL82-71, pp.47-54, 1983.
- (5) 若原, 村瀬, 梅田: "タブレット入力による手書き文字・図形認識", TV 学会技報, IPA64-3, pp.43-48, 1983.
- (6) 村瀬, 若原, 梅田: "候補文字ラティス法によるオンライン手書き文字列の認識", 信学技報, PRL84-13, pp.67-76, 1984.
- (7) 村瀬, 新谷, 若原: "知識を用いた枠無し手書き文字列の階層的認識", 信学技報, SP 86-34(PRU86-27), pp.57-66, 1986.

## 全国大会、その他

- (1) 村瀬, 吉村, 三宅: "パターン類似度法を用いた手書き平仮名文字認識の一実験", 信学全大, 1333, 5-322, 1979.
- (2) 村瀬, 杉浦, 吉村, 三宅: "手書き平仮名文字認識における複合類似度的手法の応用効果", 電気関係学会東海支部連合大会, 378, 1978.
- (3) 村瀬, 吉村, 木村, 三宅: "マッチング法による平仮名文字認識-特徴数減少の実験", 電気関係学会東海支部連合大会, 319, 1979.
- (4) 村瀬, 若原, 梅田: "タブレット入力による手書き線図形認識の検討", 信学全大, 1296, 5-281, 1982.
- (5) 村瀬, 若原, 梅田: "手書きフローチャート認識における候補ラティス法の改良", 信学全大, 1429, 5-300, 1983.
- (6) 村瀬, 若原, 梅田: "ストロークの方向分布を利用した文字・図形の回転正規化", 信学会情報・システム部門全大, 86, 1-86, 1983.
- (7) 村瀬, 若原, 小高: "文字認識・言語処理を導入した手書き文字列からの文字切出し", 信学会情報・システム部門全大, 54, 1-54, 1985.
- (8) H.MURASE, T.WAKAHARA, M.UMEDA: "Online Recognition Algorithm for Hand-Sketched Flowchart by Candidate Lattice Method", System・Computer・Controls, Vol.14, No.3, pp.37-46, 1983 (信学論(D), J66-D, 6, pp.675-682の翻訳).

## 特許関係

- (1) 村瀬, 若原: "手書き線図形処理方式"、特許出願 昭56-74012
- (2) 村瀬, 若原, 梅田: "手書き線図形認識方法"、特許出願 昭57-64162
- (3) 村瀬, 若原, 梅田: "手書き線図形認識方法"、特許出願 昭57-64163
- (4) 村瀬, 若原, 梅田: "手書き線図形認識方法"、特許出願 昭58-36330
- (5) 村瀬, 若原, 梅田: "オンライン手書き文字列認識方式"、  
特許出願 昭59-15167
- (6) 高橋, 村瀬: "キーボード"、実用新案出願 昭58-112320
- (7) 村瀬, 若原: "ボタンマッチング方式"、特許出願中

## 賞罰

- (1) 篠原記念学術奨励賞、電子通信学会、昭和61年3月25日受賞.