

言語情報を利用した手書き文字列からの 文字切り出しと認識

正員 村瀬 洋[†] 正員 新谷 幹夫[†]
正員 若原 徹[†] 正員 小高 和己[†]

Segmentation and Recognition of Hand-Written Character String using Linguistic Information

Hiroshi MURASE[†], Mikio SHINYA[†], Toru WAKAHARA[†] and
Kazumi ODAKA[†], *Members*

あらまし 本論文は、自由書式で筆記された文字列から文字を精度良く切り出すと同時に認識する手法について述べたものである。文字の切り出し、文字認識、言語処理を統合した認識系を構築し、言語情報などの上位の知識を文字認識や文字の切り出しに有効にフィードバックすることにより、切り出しの高精度化と認識率の向上を実現した。処理を以下に示す。①まず入力文字列をセグメントに分割する。次にセグメントの組み合わせをブランチとする2端子グラフ(候補文字ラティス)で文字列を記述する。②そのラティスの探索により文字認識と文字切り出しを融合して、文字系列の候補を選出する。③その文字系列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、これを認識結果とする。文字系列の反復的な探索には動的計画法を採用した。本手法をオンライン文字列認識に適用した結果、言語情報を利用しない従来の切り出し法に比較して、切り出し率は97.0%から99.2%に向上し、文字の認識率は99.1%から99.8%に向上し、本手法の有効性を確認した。

1. ま え が き

日本語情報処理の発展に伴い、手書き文字認識による簡便な文字入力法が期待されている。その際、より円滑なマンマシンインターフェースを実現するため、筆記時の各種の制限を緩和することが不可欠である。特に、文字の枠内筆記の制限を緩和して、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が目ざされている。その実現のためには、手書き文字列から個々の文字を精度良く切り出す技術を確立する必要がある。

従来の文字の切り出し法には、文字を囲む矩形枠を利用する手法^{(1),(2)}等があるが、これらの手法では文字間が接触したような文字列には適用できない。一方、文字認識の結果を文字切り出しにフィードバックする手法^{(3)~(6)}が提案されている。しかし、日本語文章には「明」(「日」+「月」)のように、偏とつくり

分離し、偏とつくり自身が文字となる場合が多数含まれるため、文字認識の結果だけから文字を一意に切り出すことはできない⁽³⁾。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切り出して認識しているのは、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^{(7)~(9)}、複数の文字を単位に照合する手法⁽¹⁰⁾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フィードバックさせる手法ではなかった。

本論文では、文字の切り出し、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切り出しに有効にフィードバックする手法を提案する。同様の考え方は連続音声認識⁽¹¹⁾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

[†] NTT 電気通信研究所, 武蔵野市
NTT Electrical Communications Laboratories, Musashino-shi,
Tokyo, 180 Japan

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。

2. 知識を導入した文字切り出しの考え方

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切り出すことができる。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、文字ごと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、表1に示すように切り出しに有効な知識を3段階に分類し、その役割を説明する。

(a) 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切り出しは可能である。

(b) この例では、漢字が偏とつくりに分離するため、図形的なまとまりを考慮するだけでは文字の切り出しは困難である。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で、つまり文字認識の導入により文字の切り出しは可能である。

(c) この例では、漢字が偏とつくりに分離しか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、単語処理の導入により文

字の切り出しが可能である。

このように、精度の良い文字の切り出し法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。

2.2 文字認識を利用した切り出し法の限界

左右に分離した文字や、文字間が接触する文字をも切り出すことができるように、従来、文字認識を切り出しに利用する手法が提案されている。この手法は2.1(b)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、2.1(c)で述べたような限界を持つ。ここでは1例として、各文字の切り出しと認識を同時に実現する候補文字ラティス法^[3]を取り上げ、文字認識を利用した切り出し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(2.1(c)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、それがどのように切り出し誤りをするかを調査することにより行った。候補文字ラティス法は本論文で提案する手法に包含されている。該当箇所は3.1であり、手法の簡単な説明はそこで行う。

2.2.1 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名の筆記者に筆記させ、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。

2.2.2 切り出し不能文字の分析

1文字が複数の文字に分離することにより切り出し誤り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、切り出し不能文字は2148字種中、207字種(9.7%)存在することを確認した。

切り出し不能文字を、切り出し誤りの発生する原因により、以下の2種類に分類する。

(1) 第1種切り出し不能文字: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切り出し誤りが発生する文字。第1種切り出し不能文字は122字種(5.7%)存在した。その例を図1(a)に示す。

(2) 第2種切り出し不能文字:上記以外の理由で文字切り出し誤りの発生する文字。例えば、手書き変形により文字が二つの図形に分離し、それぞれを文字として誤認識することで、切り出し誤りが発生する文字

表1 手書き文字列の文字切り出し

| | 文字列例 | 切り出しに必要な知識 |
|---|------------------|------------|
| ① | 日本の | 図形のまとまり |
| ② | 高維青友し 1文字 1文字 | 文字形状(文字認識) |
| ③ | 言糸田な 形容動詞 | 言語情報(文節処理) |

などがある。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、本論文で述べる文字切り出し法では、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種切り出し不能文字は85字種(4.0%)存在し、その例を図1(b)に示す。

全字種中9.7%存在する切り出し不能文字は、文字

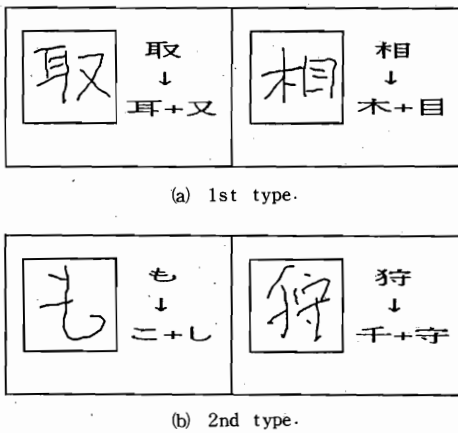


図1 切り出し不能文字の例
Fig.1 Examples of characters that are unable to be segmented.

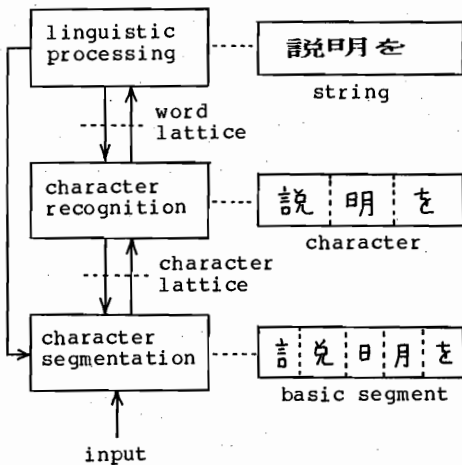


図2 ブロック図
Fig.2 Block diagram.

認識の情報だけを利用している従来手法では、本質的に正しく切り出されず、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。

3. 言語情報を利用した文字の切り出しと認識

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、文字切り出し、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。文字切り出し層と、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。ラティスとは切り出し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2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切り出し位置と、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。

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では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、罫線の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⁽³⁾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。)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3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¹⁰⁾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

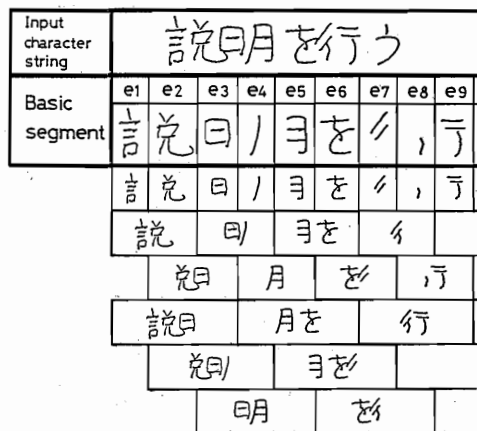


図3 候補文字の生成
 Fig.3 Candidate character generation.

表2 累積分類率

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

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2に示す。

(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3に対する
 候補文字ラティスのグラフ表現を図4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,

$$F = \sum_{\substack{\text{候補文字} \\ \in \text{経路}}} (\text{候補文字} \\ \text{の相異度}) \times (\text{候補文字を構成する} \\ \text{基本セグメント数})$$

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4の例では, 第1次認識候補は「説」「日」「月」
 「を」「行」「う」となる。

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

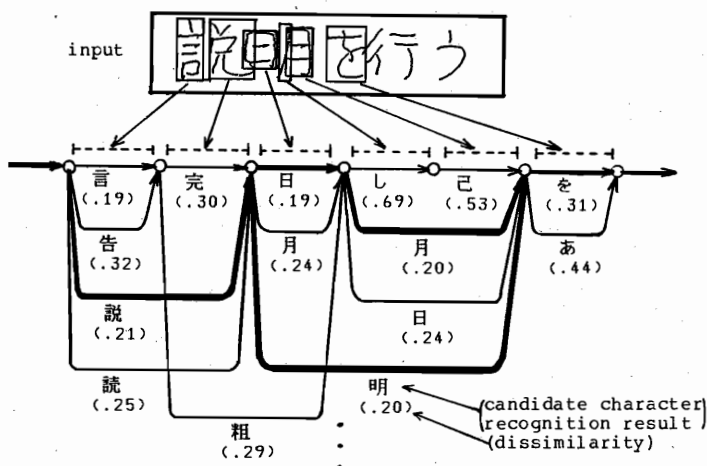


図4 候補文字ラティス
 Fig.4 Candidate character lattice.

イ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、学研国語大辞典を基に作成したもので、約8万語からなる。「説明を」に対する単語ラティスの例を図5に示す。

(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いながらグラフを探索する。接続規則として約130項目が登録されている。接続規則の1例を以下に示す。

- ・ 名詞+接尾語 → 名詞
- ・ 名詞+格助詞{と} → 文節
- 等

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し、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5の例(「説明を」)は言語的に許容されるが、図4の例に対する最短経路では「説日月」の箇所が許容されない。

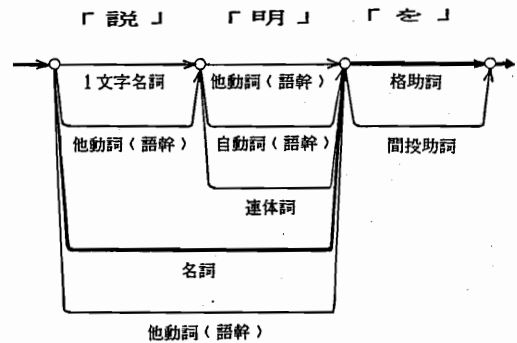


図5 単語ラティス
Fig.5 Word lattice.

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、照合範囲を単位として得ることができる。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、文字認識の候補と切り出し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3.4に述べる。

この文字列に対して再び3.2で示した言語処理を適用し、言語的に矛盾が無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10より大となる文字列では、認識を棄却する。

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¹³⁾。この探索は本論文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昇べき

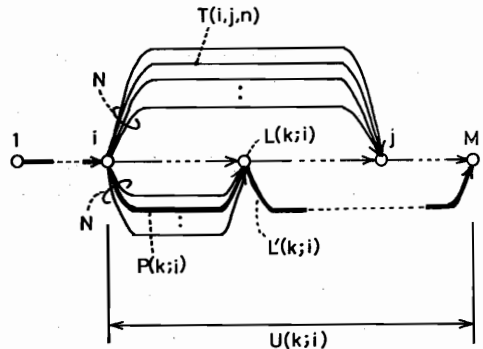


図6 第K次最短経路の探索
Fig.6 Searching for Kth minimum cost path.

の順に並んでいるとする(図6参照)。

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(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)*で表し、*n*の値を*P(k; i)*で表し、*K'*の値を*L'(k; i)*で表す。

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

で*K*番目に最小な*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。

・ステップ1

初期設定として次の代入を実行する。

$$i = M - 1$$

$$U(1; M) = 0$$

$$U(2; M) = \text{無量大}$$

...

$$U(k; M) = \text{無量大}$$

・ステップ2

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

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

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

ここで、右辺のMIN(*k; ...*)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。

$$L(k; i) = j'$$

$$L'(k; i) = k''$$

$$P(k; i) = n'$$

つまり、*U(k; i)*, *L(k; i)*, *L'(k; i)*, *P(k; i)*は、*j*が(*i+1*)から*M*までの*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。

図4のラティスに対する第1次から第5次までの認識候補を以下に示す。

第1次認識候補: 「説」「日」「月」「を」

第2次認識候補: 「説」「明」「を」

第3次認識候補: 「言」「完」「日」「月」「を」

第4次認識候補: 「読」「日」「月」「を」

第5次認識候補: 「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。

4. 評価実験

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文列(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7(a)に示す。

(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(2.2.1 参照)。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 7 (b) に示す。

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切り出し率(文字

切り出し率)、切り出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 3 に示す。但し、文字切り出し率、文字認識率はそれぞれ、

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

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

により与えられる。

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、言語処理が文字切り出し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。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切り出しができた例を図 8 (a) に、正しく文字認識できた例を図 8 (b) に、正しく文字切り出しできなかった例を図 8 (c) に示す。切り出し誤りの原因は、今回の実験では言語検定を比較的ゆるく設定したため、図 8 (c) のような文字列も言語的に許容されたことによる。

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切り出し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 9 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 4 の入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 秒かかると、この例では 2 回の反復がなされた。但し、実験は 1 MIPS のミニコン(FORTRAN 使用)で行った。

詳細に検討を続ける

安定でしかも量産に向く

(a) data 1.

誤動作防止に必要な受信

平均印字速度を保証する

(b) data 2.

図 7 データ 1, 2 の例
Fig.7 Examples of data 1, 2.

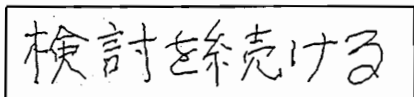
表 3 実験結果

| 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 |

5. 考 察

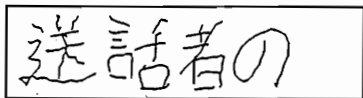
5.1 従来手法(分離有意文字処理)との比較

言語情報を利用せずに分離文字を正しく切り出す手法として、文献3の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く、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



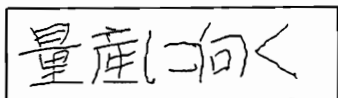
Correct answer : 「検討を続ける」
 1st candidate : 「検討を糸売ける」
 2nd candidate : 「検討を糸売けそ」
 3rd candidate : 「検討を続ける」→ STOP
 (correct)

(a) Correctly segmented character string.



Correct answer : 「送話者の」
 1st candidate : 「迷話者の」
 2nd candidate : 「迷言舌者の」
 3rd candidate : 「送話者の」→ STOP
 (correct)

(b) Correctly recognized character string.



Correct answer : 「量産に向く」
 1st candidate : 「量産し拘く」
 2nd candidate : 「量産し拘ん」
 10th candidate : 「量産し招く」→ STOP
 (misrecognition)

(c) Missegmented character string.

図8 認識結果の例
 Fig.8 Examples of recognition result.

に近い方を選択する手法である。文字の切り出し率の比較を表4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種分離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切り出しが不可能であることがあげられる。

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来たが、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10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5に示す。言語処理がない場合の横書き文字列の切り出し率は、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切り出し不能文字の存在

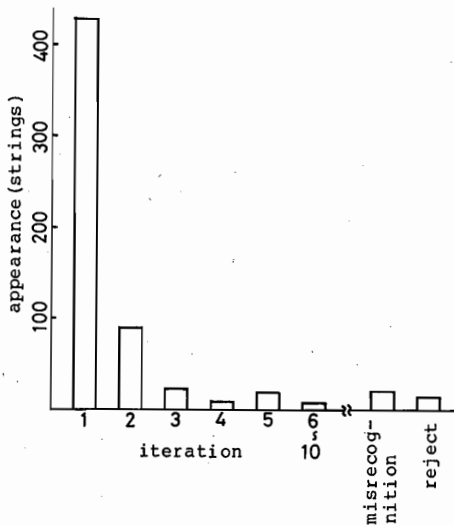


図9 反復回数の出現頻度
 Fig.9 Appearance of the iteration number.

表4 従来手法の文字切り出し率(データ1使用)

| | |
|----------|-------|
| 分離有意文字処理 | 97.3% |
| 本手法 | 98.8% |

を2と同様の処理により求めた結果、横方向の切り出し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%、2.2.2参照）であった。

5.3 本手法の限界

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なっている。例えば「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今回は特に対策を設けなかったが、これらに対しては例外処理などが必要となる。

本手法は、罫線上に自由ピッチで筆記された文字列から文字を切り出し認識することが可能である。しかし、文字の大きさが極端に変わる文字列の場合に、そのまま本手法を適用することは困難である。また、縦書き文字列、横書き文字列、図形等の混在した図面の認識には、前処理としてこれらを分類する手法が必要となる。

実験で使用したデータは、自由書式ではあるものの個々の文字品質は高い文字から構成されている。本手法を、品質の低い文字から構成される文字列にそのまま適用すれば、個々の文字認識精度が低いため、文字列の第1次認識候補の正当率の低下が予想される。

6. む す び

本論文では、自由書式で筆記された文字列から文字

を精度良く切り出し、同時に認識する手法について述べた。具体的には、文字の切り出し、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切り出す手法を提案した。以下に結果をまとめる。

(1) 文字の切り出し、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（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、言語情報を文字切り出し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次最短経路の探索は動的計画法が利用できることを示した。

(3) 言語情報を導入したことにより、文字字種中9.7%ある分離文字を正しく切り出す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）に対する文字切り出し率は、言語情報導入しない手法の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、文字切り出し率で99.4%を達成した。

本手法は文字認識部、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れば良い。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。例えば、文字認識部では低品質文字のオンラインあるいはOCR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。今後は、各モジュールの機能を拡充し、本手法を走り書き文字列のような低品質な文字列に適用して行く予定である。

謝辞 日頃御指導頂く畔柳前情報通信基礎研究部長、塚本統括、増田第三研究室長に深謝します。研究を進める上で有益な議論をして頂いた視覚情報処理研究グ



図10 データ3の例
Fig.10 Examples of data 3.

表5 縦書き文字列の場合の処理結果

| 項目 | データ名 | 言語処理 | |
|---------|------|-------|------|
| | | 無し | 有り |
| 文字切り出し率 | データ3 | 98.5% | 99.4 |
| 文字認識率 | データ3 | 99.1 | 99.8 |

ループの梅田主幹研究員，データ収集に協力して頂いた第三研究室の諸氏に感謝致します。

文 献

- (1) 馬場口，塚本，相原：“手書き日本文字列からの文字切り出しの基礎的実験”，信学論(D)，J68-D，12，pp.2123-2131(昭60-12)。
- (2) 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(D)，J64-D，11，pp.997-1004(昭56-11)。
- (3) 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(昭60-04)。
- (4) T. Akiyama, S. Naito and I. Masuda: "A method of character extraction from format-unknown document images", ICTP, pp. 85-90 (1983)。
- (5) R. G. Casey and G. Nagy: "Recursive Segmentation and Classification of Composite Character Patterns", Proc. of 6-th ICPR, pp. 1023-1026 (1982)。
- (6) 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，p. 885(昭59)。
- (7) 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(昭60-05)。
- (8) 杉村，斎藤：“文字連接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，1，pp. 64-71(昭60-01)。
- (9) 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，4，pp. 520-527(昭59-04)。
- (10) 大田，鈴木，池田：“手書き日本文認識における文字列利用の一方式”，信学論(D)，J68-D，3，pp. 330-336(昭60-03)。
- (11) 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(昭53-04)。
- (12) 小高，若原，増田：“筆順に依存しないオンライン手書き文字認識アルゴリズム”，信学論(D)，J65-D，6，pp. 679-686(昭59-06)。
- (13) R. Bellman and R. Kalaba: "On kth Best Policies", J. Soc. Indust. Appl. Math., 8, 4, pp. 582-588 (1960)。

(昭和61年2月25日受付)



村瀬 洋

昭53名大・工・電子卒。昭55同大学院修士課程了。同年日本電信電話公社入社。以来，文字・図形認識の研究に従事。現在，NTT基礎研究所情報通信基礎研究部第三研究室研究主任。昭60年度篠原記念学術奨

励賞受賞。



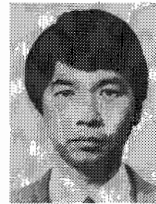
新谷 幹夫

昭54早大・理工・応物卒。昭56同大学院修士課程了。在学中，視覚系の心理物理的研究に従事。同年電電公社武蔵野電気通信研究所入所。以来，文字認識の研究，画像生成の研究に従事。応用物理学会会員。



若原 徹

昭50東大・工・物理工学卒。昭52同大学院修士課程了。同年日本電信電話公社入社。以来，文字認識の研究に従事。現在，NTT基礎研究所情報通信基礎研究部第三研究室主任研究員。工博。



小高 和己

昭47工学院大・工・電子卒。昭50千葉大・大学院修士課程了。同年日本電信電話公社武蔵野電気通信研究所入所。以来，オンライン手書き文字認識の研究・実用化に従事。現在，NTT基礎研究所情報通信基礎研究部総括補佐担当主幹研究員。工博，IEEE，AVIRG各会員。