

# 手書き筆記体からの文字切り出し

小泉 和也<sup>1</sup>・西村 広光<sup>2</sup>・富川 武彦<sup>2</sup>

1 情報学部研究生メディア学科(s048501@cce.kanagawa-it.ac.jp)

2 情報メディア学科({nishimura,tomikawa}@ic.kanagawa-it.ac.jp)

## Hand-written Cursive Character Segmentation

Kazuya KOIZUMI<sup>1</sup>, Hiromitsu NISHIMURA<sup>2</sup>, Takehiko TOMIKAWA<sup>2</sup>

### Abstract

The purpose of this paper is the character segmentation of handwritten cursive script written in English. Many of current methods segment one character to many small regions. A number of high performance segmentation methods have been proposed in cooperating with character recognition. But, most of application needs high performance even if character recognition results are not used. Here, our method does not use any results of character recognition and segments on cursive script using some specific feature of writing. In order to evaluate our proposed method, we have constructed a cursive handwritten script database. As a result of the evaluation, about 75% of characters could be segmented at appropriate points and about 45% of cursive words could be segmented at correct number of characters.

**Key Words:** hand-written character, segmentation, cursive character, character recognition

### 1. はじめに

手書き筆記体文字を認識する際には、続き書きされた文字を一文字ずつ切り出して認識する方法が一般的である。現在最も広く利用されている手法は、多重切断仮説を利用して認識する手法である<sup>1)</sup>。また現在、手書きオフライン文字認識では単文字英数字文字の認識率は約97%であるが、手書き筆記体文字の認識率は約80%である。認識性能低下の原因として、文字同士が接触している場合に、正しく一文字ずつを認識することができないことがあげられる。そのため、文字認識の精度向上には各文字の切り出しが不可欠だといえる。

複数の切断候補を探索し、複数の切断領域を組み合わせながら認識を行う多重切断仮説を利用した認識手法が報告されている<sup>1)</sup>。この手法を利用すれば、理論的には過剰切断があっても認識には影響を与えることはないが、認識速度の面から考えると、過剰な切断仮説は認識時の組み合わせパターンの指数関数的増加を生み、認識速度低下につながるため、可能な限り過剰切断を行わないようにする必要がある。そこで本研究では英字筆記体に着目し、過剰切断を抑制しながら高精度な文字を切り

出す方法について検討を行った。

また、手書き筆記体文字のデータベースとして、大規模かつ安価で利用できるようなものが無いため、本研究では、手書き筆記体文字を採取し、データベースを構築し実験に利用した。

### 2. 筆記体サンプルの収集

文字認識や文字切り出しの研究を行う際の問題点として、サンプルが非常に少なく、存在しても解像度が低い等が挙げられる。また、CEDAR CDROM database[1]<sup>2)</sup>という300ppi 8-bit, 300ppi 1-bitのサンプルを有償で収録してあるものもあるが、高価な使用料が必要である。この点から、教育機関のようなところが筆記体文字のサンプルデータを収集して、無償で利用可能なデータベースを作成し、同一のデータベースでの評価を可能とすることは今後の文字認識分野の研究発展に大きく寄与できるものと考えられる。そのため本研究ではサンプルの収集から始めることにした。

本章では、単語の切り出しを行うためのサンプルの収集につ

いて概要を説明する。以降はデータベース作成のための切り出しを単語の切り出し、単語から文字を切り出す場合は文字の切り出しと表すこととする。

## 2.1. 対象となる単語の選別

本研究において、対象単語の選別は後述の2つの考え方に基いている。これは特殊な単語のみを収集することを避けることと、頻出単語を採取することでデータベースの有用性を高めようとしたためである。

1つ目はより使われやすい単語を調べるために、ニュース記事に着目し、日常生活で多く利用されている単語を収集しようと考えた。データ採取には「Newsの英単語をマスターしよう<sup>3)</sup>」というツールを利用した。これは、ニュースサイトを巡回して頻出の単語を探し、それを英単語の問題として出題するというものである。この機能を利用して、ニュースサイトにおける頻出英単語の調査を行った。

2つ目は、採取した単語が時事に偏ることを避けるため、頻出単語のうち、動詞のみを選別した。本研究で選出した頻出動詞45単語を表1に示す。

表1 45単語の表

arrest	draw	practice	remain
appear	except	provide	reserve
ask	field	put	use
attack	fight	move	run
break	focus	need	schedule
charge	force	notice	seek
cut	give	include	shoot
deal	hold	interest	think
debate	judge	issue	trade
decide	keep	race	unite
delay	leave	rate	warn
die			

## 2.2. 手書き筆記体サンプル画像の収集

手書き筆記体文字のサンプルを収集する際、日本人のみとなってしまうため、本研究において文字切り出しの対象となるサンプルは日本人の手書きによる筆記体とすることとした。

まず、サンプルの書き方を説明する。収集したサンプルを切

り分けやすいように縦2cm、横8cmの単語の書くスペースを限定して書かせることとした。筆記サンプルを個別に領域に分けて筆記させたのは、データ採取の容易さを求めただけでなく、各単語を筆記する際に小さすぎる領域にした場合に、隣接した文字が過剰に接触することを避けるためでもある。このスペースはA4のコピー用紙1枚あたりに縦7個、横3個の計21個を用意した。被験者には、このシートに1列1単語を書かせることとした。すなわち、1枚のシートに3種類の単語を7回ずつ筆記させることとした。

なお、筆記シートは単語切り出しを容易にするため枠を設けた透明シートを重ねて、固定位置、固定の大きさに筆記させることとしている。

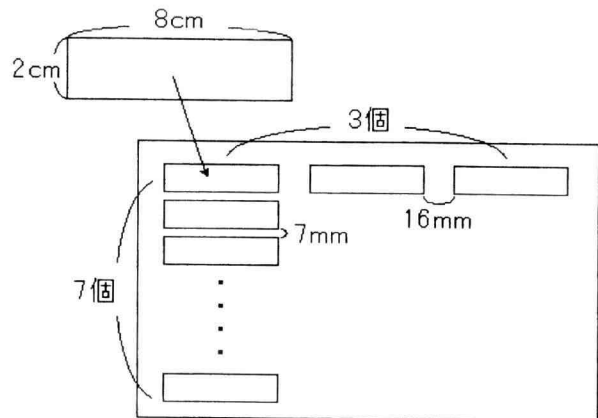


図1 用紙の規格

筆記に使用するペンは、PILOT SUPER プチ<中>耐水性を用いることにした。筆記するペンを1種に特定したのは、耐水性ペンを使うことでかすれの抑制をし、太さのあるペンで書くことで文字の太さ・大きさを揃えるようにした。

読み込みは、オムロン社製HS600というシートフィーダの着いたものを利用し、150dpiの解像度で読み込むこととした。オートシートフィーダのスキャナを利用することで膨大な記録用紙をまとめて読み込むことが可能であるが、原稿送り機能は高精度に行うことができないため、原稿が傾く場合があるが、ここではシートフィーダの調整を行い、最小限の傾きに抑えられるようにして読み込みを行うこととした。

読み込んだ記録用紙1枚のデータを単語ごとに切り出すシステムを構築し、単語切り出しを行った。この単語切り出しは、記録用紙に場所や大きさの制限を与えているため、特定領域を切り出す比較的単純な処理で実現できる。

単語切り出しは、次の2段階の処理によって実現した。1段階目の処理として、1種類ずつの単語に分けるために1列ごとに切り分ける。1つの画像に対し1種類単語が7個縦に並んでいる画像ができる。次に、2段階目の処理で、1段階目で切り分けた画像を1つの単語ごとに切り分ける。こうして1種類の単語に対して7つの単語に分けることができる。

切り出した単語は濃淡画像で、縦 152pixel×横 520pixel で保存することとした。これはサンプルデータの画像から1単語ごとに切り分けた際サンプル収集で利用したテンプレートのサイズと実際のサンプル画像の状態から1単語に切り分けた際、文字部分が切れてしまうことがないであろうサイズを探した結果である。

### 3. 文字切り出しシステムの概要

本章では、構築した文字切り出しのシステムについて説明する。なお、このシステムで用いる切り出す対象である入力ファイル形式は、PGMのASCIIデータとし256階調の濃淡画像を利用することとした。

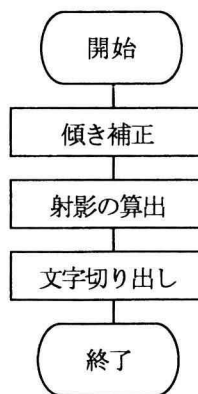


図2 全体の流れ

図2はおおまかな全体の流れである。まず切り出したい単語の画像を入力し、文字領域が判断しやすいように濃度の2値化をする。この2値化は入力画像にノイズが入らないように作成したものであるため、手動で濃度値の半分値の閾値を決定して行っている。次に傾き補正を行い、射影の算出をした後に文字分割と進む。最後に、実行結果の切り出し後の画像ファイルと連結領域数を書き出す。

次節からは、システムの大きな部分である傾き補正、射影の

算出、文字分割とそれを行うときの分割数の検査について説明する。

#### 3.1. 傾き補正

文字を切り出す際に後の3.3節で説明しているが、文字と文字が離れるように縦に白い切り出し線を引いて離している。このとき、切り出し線が文字部分の本体を切り出さないように、また文字と文字が繋がっている連結線の部分で切り出すようにしなければならない。しかし、切り出す目安として3.2節を用いているため、文字が傾いた状態だと文字と文字の連結線が切れない恐れが出てくる。そのため、文字を直立させる状態にしておく必要がある<sup>9)</sup>。

傾き補正を行う上で2分探索法を利用した。これを用いて、初期状態を補正角度 $0^\circ$ として $-30^\circ \sim +30^\circ$ までの範囲で、3.2節のような縦方向の射影を算出し、射影の値の最大になる角度を最適角度とする方法をとった。補正角度の範囲は横に単語が書かれていること、さらにサンプル収集の段階で枠を決めてサンプルをとったので、枠の対角線以上傾くことは無いと考えられることから、傾き補正の限界を $\pm 30^\circ$ とした。手順は以下ようになる。

- (1)  $-30^\circ$  のときの射影と  $+30^\circ$  のときの射影を求め、射影の最大値が大きいものを選択する。
- (2) 次に、調べた角度の半分角度の射影をとり、射影の最大値を比較し、大きいものを選択する。  
はじめの場合、 $-30^\circ \sim 30^\circ$  であるので次に検査する角度は  $0^\circ$  となり、先に選択した射影の最大値と比較し、大きいものを選択する。
- (3) (2) を繰り返し、選択したものと調べた角度との射影が等しくなった場合に、探索完了とする。

以上の手順によって求めた補正角度から角度補正を行う。このときの例として図3、図4を示す。このとき、ただ画像を回転させたのでは入力ファイルのサイズから文字がはみ出てしまう。単純に画像サイズを大きくすることで解決することもできるが、他の画像サイズとの均整がとれなくなるため好ましくないと考えた。そこで、本研究で行う傾き補正では、対象画像を補正角度だけ傾いている平行四辺形と考え、それを長方形に補正するというように横軸方向のみの画素移動で傾き補正を行う。

こうすることにより、入力したときの画像サイズを変えることなく、傾き補正をすることができる。このときに利用した式を次に示す。

$$\begin{aligned} M_x &= S_x - S_y \times \tan \theta \\ M_y &= S_y \end{aligned} \quad (1)$$

補正前の座標 ( $S_x, S_y$ ) , 補正後の座標 ( $M_x, M_y$ )



図 3 傾き補正前

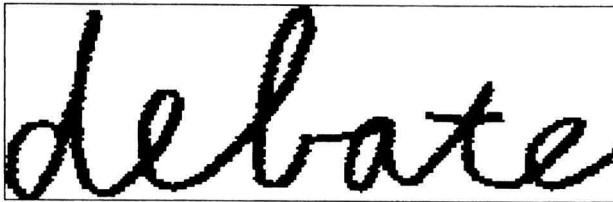


図 4 傾き補正後

### 3.2. 射影

文字の切り出しを行う際に、文字が書かれている部分である文字領域と文字と文字がつながっている部分であるストローク領域を判断し、ストローク領域の部分で切り出しを行わなければならない。この文字領域とストローク領域の判断として射影を用いることとした<sup>9)</sup>。

本研究では、射影を求める際に、単純に黒画素を射影するのではなく、文字の上輪郭と下輪郭の差で求めることとした。文字の上輪郭とは縦方向の走査線を上から順に探索していき、最初に黒画素を検出した座標をつないだもので、図 6 のようになる。同じように、今度は画像の下から上に黒い画素を探していくのが下輪郭とする。これも図で表すと図 7 のようになる。黒い画素とは文字が書かれている部分が黒画素で表されているので、その部分のところを指している。図 6、図 7 において、灰色で示した領域は文字の黒画素領域、黒で示した箇所は文字の黒画素の輪郭を表している。

得られた射影と射影を微分したものをを用いて 3.3 節の文字分割で利用する。

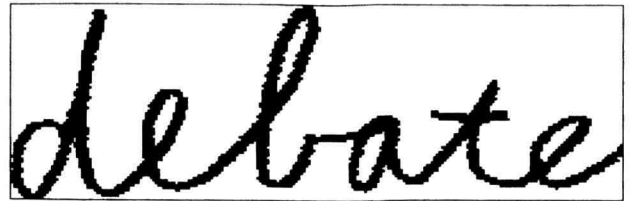


図 5 サンプル図

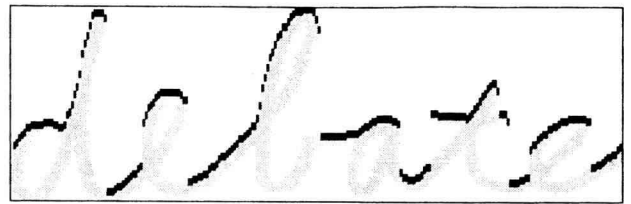


図 6 上輪郭

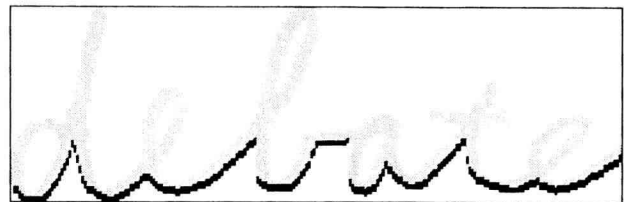


図 7 下輪郭

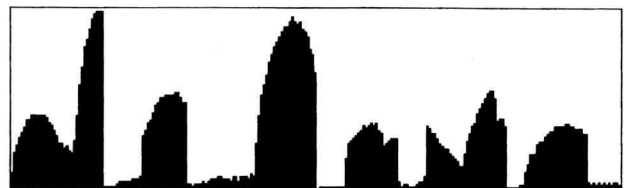


図 8 上輪郭・下輪郭を利用した射影

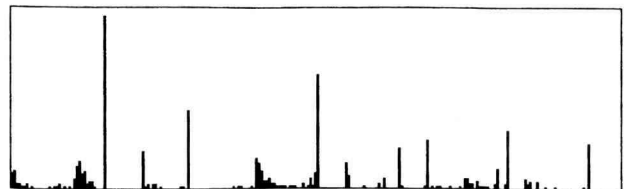


図 9 微分した射影

### 3.3. 文字分割

本研究で文字分割を行う際には、縦方向の垂直な直線で分割することとした。分割を行う際には、ストローク方向と垂直な線分を利用したほうが高精度な文字分割ができることがあるが、英文字筆記体の場合には、横方向に文字が並んでいるため、縦方向の直線で文字を分割しても結果に大きな差が現れないと考えたためである。

3.1 節で述べたように、文字分割を行う際に文字同士を分けるために縦に白い切り分け線を引いている。次から用いている図では白色であると見えないため見つけやすいよう黒色か灰色の色を付けている。この切り分け線を引く基準は微分した射影を見て、射影の変化が文字で使われている線の太さの2倍よりも大きければ引いていくことにした。文字の線の太さは画像の各画素列の黒画素数の平均値を取ったものとした。しかし、このままでは切り分け線が多数引かれてしまう。そこで本来の単語が書かれていることをふまえると、文字の左側に切り分け線を入れることである程度違和感なく分割できると考え、切り分け線の数を少なくすることができる。手順としては次のようになる。

- (1) 微分した射影から左より1列ずつ変化量を比較していき、変化が大きいところで切り分け線を引くことができるようになる。

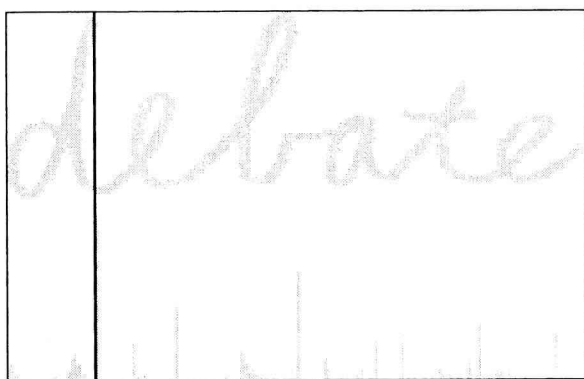


図 10 文字の切り出し1段階目

- (2) 切り分け線を引く判定として微分前の射影を見て、射影が大きくなっている手前まで切り出し線を移動させる。

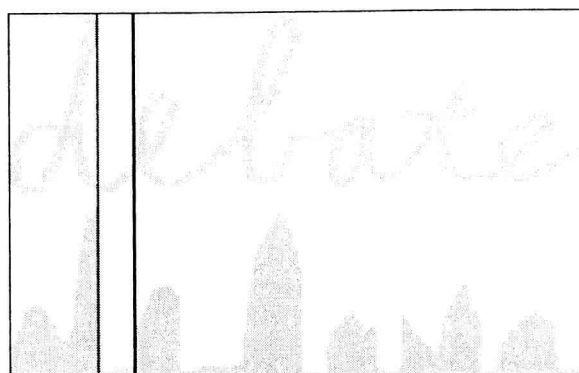


図 11 文字の切り出し2段階目

- (3) (1) と (2) を繰り返して、文字ごとに切り出す。

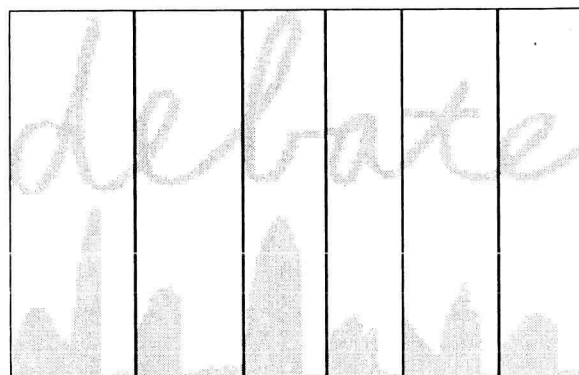


図 12 文字切り出し完了状態

(2) の処理だけでも文字を切り出すことは可能であるが、ここまでの方法だけでは、切り出せない文字があった。そこで切り出し精度を上げるため(1)の手順を導入した。

文字分割を行う際に、数 pixel の領域を持つ微小連結領域が発生してしまうことがある。微小連結領域が発生してしまうと、連結領域数が増えてしまい本来の連結領域数にならないばかりか、連結領域数が増えることによって認識の処理段階での認識率が落ちてしまう恐れもある。微小連結領域に切断することを回避するため、切り出し線を決定する前に回りの数 pixel の連結領域を調べ、微小連結領域が発生してしまう場合には、微小連結領域が発生しない位置まで切り出し線を移動させた。

### 3.4. 分割数の検査

デジタル画像は色情報を画素ごとに独立して持っている。この色情報を使い、同じ色情報を持っている画素が連結している

ものには同じラベルを付加する。これをラベリングという。<sup>9)</sup> ラベリングを利用することで複数の画素を連結した画素の集合として検出することが可能となる。

本研究においては文字を切り分け後の分割数を調べることや処理後の出力ファイルにラベルごとに色を付け、視覚的な判断を容易にするため、またラベルごとの面積を求めるために利用している。このラベルごとの画素数を求めることで、3.3節に書かれている連結領域の大きさを調べることができる。

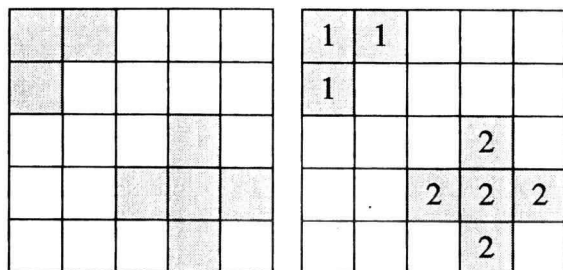


図 13 ラベリング

#### 4. 切り出し結果

本章では、3章で説明した流れで行った文字切り出しの結果をもとに、人の目視による評価と、単語の文字数と文字切り出しの分割数について評価、検討を行う。

##### 4.1. 評価方法

単語の切り出し処理をしてある45サンプルを連結領域ごとに色付けし、目視しやすいようにしたものを被験者に目視で判定させた。判定は、「切り出し結果を1文字ごとにアルファベットとして読むことができるか」という基準で行った。判定においては、1つの連結領域をそれぞれ独立に見るのではなく、複数の連結領域を合わせて一文字を構成できる場合にも良いと判定させた。

図14はその評価例である。縦に引かれている灰色の線は本提案手法で求めた切り分け線を表し、文字の下にある横に引いている①と②の線は人間が1文字と判断する範囲を図で表したものである。①の灰色の線のように切り分け線ごとに1文字と判断することもでき、②の黒い線のように2つ以上の切り分け領域から1文字と判断しても良い。評価は3段階に分けるものとし、「○：良い」「△：判断困難」「×：悪い」に分類させ評価した。次に説明する3つのものは、各評価の例である。

図14の場合は②のように3つの領域に分けて読み取ると

「c」、「u」、「t」と読むことができる。これより、各文字がはっきりと判るため「○：良い」と判断することができる。

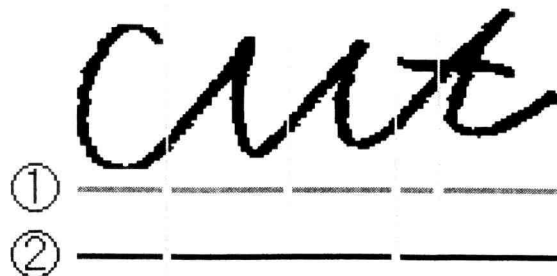


図 14 判断例：良い

図15の場合は灰色の下線のように分けることができるが、「o」、「e」はその文字通りに判断できるが、「m」、「v」は文字が切断されすぎていている。このため、「△：判断困難」と判断することができる。



図 15 判断例：判断困難

図16の場合も図15と同じように行い、「h」、「o」はその文字通りに判断できるが、「l」、「d」が接触してしまっており、その通りの文字として見るができないため、「×：悪い」と判断することができる。

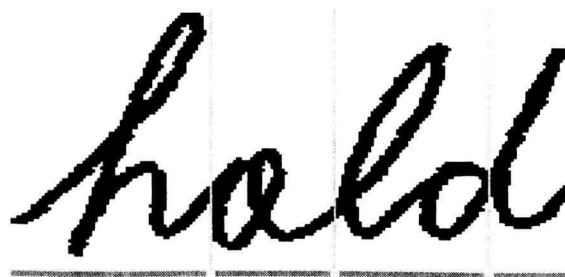


図 16 判断例：悪い

以上の例に従って評価を行った。評価対象文字数は3178文字で、本学の日本人の学部4年生14名で評価を行った。

## 4.2. 評価結果

4.1の評価方法に従って評価を行った結果を表2に記す。表2の数値は評価を行った45単語に使われている文字数の母数3178に対する割合を表したもので、評価を行った14人の評価結果の平均値を取っている。

表2 評価の結果

評価	○ 「良い」	△ 「判断困難」	× 「悪い」
割合(%)	74.17	7.99	17.84

またパターンを切り出す数が適切であるかの評価として、文字数に対する切り出し後の連結領域数を調べたものを図17に記す。適切であるというのは単語の文字数と連結領域数が同じであるほど良いが、100%未満であると切り出す数が少なくなるので、この場合は1番悪い状態となる。表では同じである場合を100%として表している。

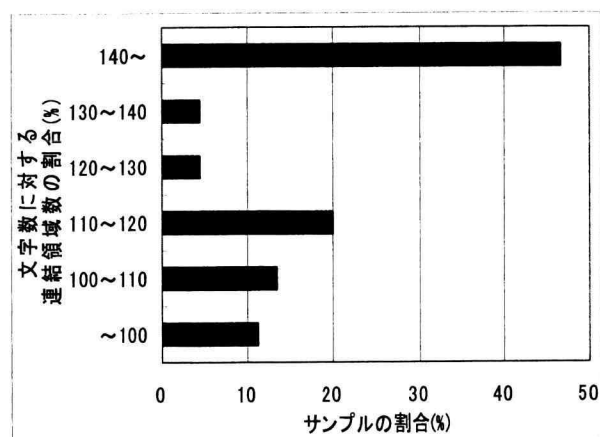


図17 文字数に対する連結領域数の割合

## 5. まとめ

評価実験から70%以上の英文字筆記体については文字ごとに切り出すことに成功していると言えるが、単語の文字数に対する分割数の割合を見ると100%以上が大半であることから、1文字を過剰に分割していることがわかる。過剰分割率が120%くらいであれば認識においてあまり問題はないが、140%を超えてしまうと認識率が落ちてしまうといわれている。また単語の文字数によっても1つ多く分割されただけで分割率

が大きく変わってしまうことがあるため、今後実際に認識システムと組み合わせての性能評価を行う必要がある。

さらに連結領域数を数える際に、「i」、「j」のような点が入っているものは点のみで1つの連結領域となるため、実際の文字数よりも大きくなる。

## 5.1. 文字切り出しにおける問題点

このシステムで文字の切り出しを行う際に、上輪郭と下輪郭の射影を用いているが、これによる問題点を挙げると以下のものがある。

- (1) 「t」が2つ続いて書いてあるときやそれに似たような射影が大きいところにある場合、文字連結部分を切り出すことができない。  
「tt」のように規則性がある文字列については、それら2文字を組み合わせたものを、1つのパターンとして認識システムにおいて認識できるようにすることも考えられるが、ひとつのストロークが別なストロークと規則性を持たずに接触する場合もあり、今後さらに検討を進める必要がある。

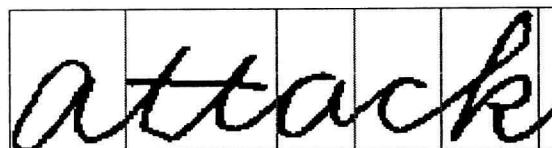


図18 問題点1

- (2) 「m」、「n」、「r」、「u」、「v」、「w」というような文字部分も文字連結部分と似ているところ、つまり文字の途中で上輪郭、下輪郭の差が大きく出やすい文字がある場合はその場所ごとに切り出されてしまい、過剰分割が起きる。

しかし、これらは文字の形状情報を利用せずには的確な文字切り出しを行うのは困難であるため、過剰切断を完全に抑制することは難しいと考えられる。

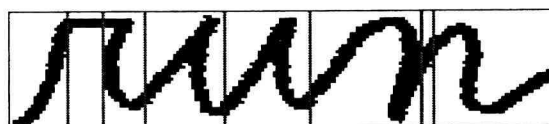


図19 問題点2

- (3) 「i」、「j」等、文字に点が入っている文字の場合、上輪郭と下輪郭の差をとると、点の位置から下輪郭の位置までを考慮してしまう問題がある
- このような場合に、小さな連結領域をあらかじめ削除してから文字切り出し処理を行うことも考えられるが、ストロークのかすれや、不連続なストロークと点を区別することが難しく、さらに検討を進める必要がある。

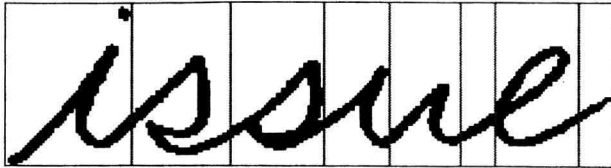


図 20 問題点 3

また、画像の 2 値化処理を行う際にサンプル画像の状態が非常に良いものだけを対象としたため、単純な濃度値の中間から 2 値化することでもあまり問題はないが、ノイズがある画像の場合は切り出しが困難となってしまう。

## 5.2. 今後の改良点

本システムでは英文字筆記体特有の特徴をほとんど利用していない。そこで、ストローク方向などの推定、英文字筆記体特有の文字連結部分の形状などの特徴を利用した切り出し方法の検討を行うことによって精度を向上することができると考えられる。

また、このまま特徴を利用しない場合でもいくつかの改良の余地はあると考える。

傾き補正に関して、現在は 1 単語の全体を見て補正をしているが、画像のベクトル化等の手法により 1 文字ずつ傾き補正することが可能であると考えられる。

「i」、「j」のような点のある文字は射影を求めるときに問題が起こる可能性があるため、ラベリングを利用して連続領域の面積を求めることにより点のような面積の小さな領域を画像から消去することができる。これは 3.3 節で述べたように、数 pixel の微小領域を求めることができることで可能だと考えることができる。また、点を消したところで多くの場合は消しても認識可能であるため問題ないと言われている。

文字を分割する際に縦方向の白い直線で分割しているが、も

っとも線の細い部分を判断し、最適な方向の線分で分割することで精度向上が実現できると考える。

文字と文字の幅という点からも文字部分と文字連結部分の幅は書かれる文字や書く人の癖の影響により特定することは困難であるが、より多くのサンプルを取ることやそれによる切り出し方法の検討を行うことで過剰な切り出しを抑制することができると考えられる。

## 謝辞

サンプル収集、実験の評価に協力してくれた富川研究室諸氏、ならびに電気電子工学科学学生諸氏に心から感謝致します。

## 参考文献

- 1) H. Fujisawa, Y. Nakano and K. Kurino: Segmentation Methods for Character Recognition: Proc. IEEE, Vol.80 [7], pp.1079-1092(1992)
- 2) CECDR CDROM database, <http://www.cedar.buffalo.edu/Databases/CDROM1/>
- 3) Sunny RALA Online, <http://www.george24.com/~mascaras/>
- 4) 安居院猛, 中嶋正之 画像情報処理, 森北出版 (1991)
- 5) H. YAMADA and Y. NAKANO.: Cursive Handwritten Word Recognition Using Multiple Segmentation Determined by Contour Analysis: IEICE Trans INF. & SYST. Vol. E79-D, pp.464-470(1996)
- 6) 田村秀行, コンピュータ画像処理入門, 総研出版 (1985)