解説

文字認識技術とその応用

森　俊二　会津大学

はじめに

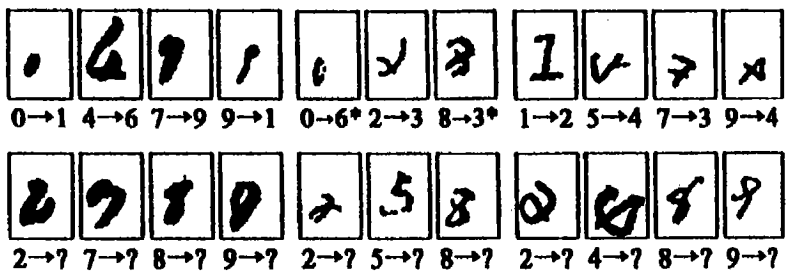
文字認識というテーマは古くて新しい．その意味するところは“できるようでなかなかできない”ということである．ここに文字認識の研究に携わる者が必ず感ずるものがある．これはいわゆる人工知能の研究者に共通するものであろう．私は本稿で，この研究者がいだくジレンマ，いらだちを読者に共感していただきたいと思っている．ある程度まではできるのであるが，その先となると何か巨大な壁が立ちはだかっているのではないかと思われる．そこで私の立場というのは，何か二重人格的である．文字は機械で認識できるのだと言っておきながら，実はそう簡単にはいかないと，片方では言っているのである．しかし，この二重人格的状況は事実であることを，これから本稿を読まれる方に理解していただきたいと思っている．実は，西田広文，山田博三両氏と共著で，“Optical Character Recognition”という本を書いた[1]．これは，600ページもある本であるが，それでも現在の文字認識の諸問題をすべて網羅しているわけではない．基礎的な，ある程度確立されているところを詳しく説明しているので，本稿でそれを引用する．そして，この本で取り扱うことのできなかった現代的問題についても，触れてみたい．

問題の設定

一言で，“文字”というけれども，その具体的対象は千差万別である．いわば，多次元の問題空間といったものを想定しなければならない．まず，典型的対象は手書きのアラビア数字である．少なくとも，日本人にとってその実用については，郵便番号を通じておなじみである．7桁数字の新方式は順調に普及しているようである．一体，機械の認識率はどの程度であるのか非常に興味あるところであるが，これは公表されていない．認識率は手書き文字の品質に強く依存している．普通に書けば，現在の技術レベルからすれば，100％近く読みとれるはずであると考えられる．機械はどのように文字を読めないか，間違うかを具体的に示したのが図-1である[2]．

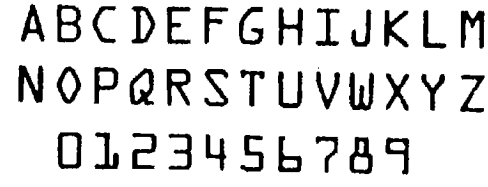
これは現在使用されている，郵便番号読みとり機の性能ではないが，かなりそれに近いものと想定される．全体としては98％の読みとり率が得られている．

さてここで，普通に書くということは一体どういうことなのか，実はこのことが文字認識の基本的問題なのである．文字の品質を定量的に述べることがきわめて難しい．図-1にあるように毛筆などで非常に太い文字を書いたり（a），使いふるしのボールペンでよく起こることであるが，文字のストローク（画）に切れがある場合（b），それに普通でない特異な形で書く（c），定性的に普通でない，のように文字を分類して述べられる．慣習的には，

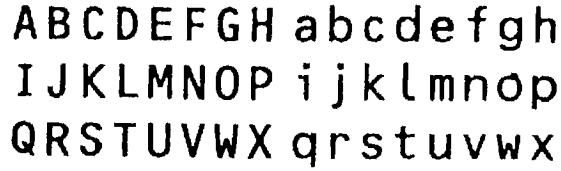


(a) 太文字　　(b) 画線のとぎれた文字　　(c) 特異な文字形

図-1　郵便番号として読みとり困難なデータ例[2]



(a) ISO OCR A



(b) ISO OCR B

図-2　標準OCR文字形

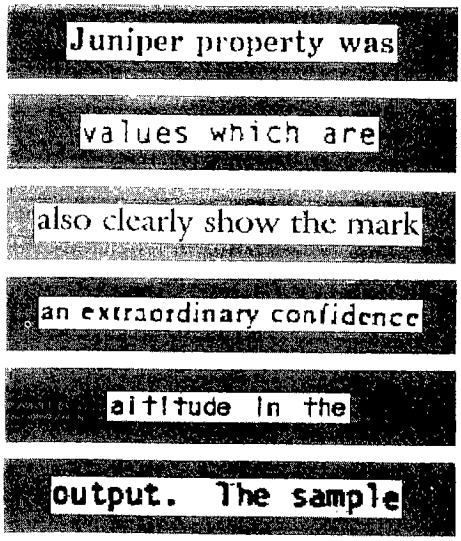


図-3　低品質文字データ例

制限された文字（constrained），loosely constrained，そして自由手書き文字（unconstrained），と定性的に分類されている．制限された文字とは機械に文字認識されやすいように加工された文字形または約束である．印刷文字としてこの典型が米国の提案によるISO OCR Aフォント文字形であり，一方ヨーロッパの規格としてはヨーロッパの提案したISO OCR Bフォントである．これらが図-2（a），（b）に示されている．ISO OCR Bフォントは，一見ほとんど形は普通のアルファベットと変わりがない．しかし，よく見ると決定的な違いが，実はある．それは文字幅が一様であるということである．本来，WとかMとかいう文字は横幅が広い，それが縮められている．これはなぜかというと，文字は単独に認識されるものではなく，単語，文として，文字列の認識が実際には問題になる．その時，このちょっとした制限がドラマティックに利いてくるのである．実はこの問題は先に述べた郵便番号の記入においても同じことであるので，それについて説明しよう．

郵便番号読みとりでは，実は決定的に機械に有利な条件を課しているのである．それは赤い文字枠内に数字を書くという約束事である．すなわち各文字列が確実に分離できるということである．これは専門的にはセグメントが容易であるという．もし，箱型の文字枠の代わりに，長い短形型の文字列枠の場合であると，文字と文字とが接触する可能性が出てくる．人間の場合，文字がお互いに接触することがあっても，別にどうということはない．極端なことをいうと，数字を重ねて書いても読めるのである．このセグメンテーションという機能が，人間にとってはきわめて自然な能力なのであるが，現在の文字認識技術にとっては，1つの大きな壁になっているのであり，正に現代的テーマとなっている．

さて，現在，印刷文字については，どの程度機械は読みとることができるのか，そして，何が問題か，これも読者にとって興味あるところであろう．実は英数字文字であるが，これについてはシステマティークな調査研究が公開されている[3]．

UNLVのISRIの実験である．同研究所では，世界でもトップレベルであると想定される6つの会社からOCRを購入し，大量の読みとり実験を行った．その結果，良質のデータに対しては，これらのOCRは99.77％から99.13％の読みとり率を示した．中間の質のデータについては，それらの読みとり率は99.27％から98.21％の範囲にあった．しかし，質の悪いデータについては97.01％から89.34％と，機械にあるばらつきが見えはじめ，全体としても読みとり率が落ちることが示されている．

ここで，問題になるのは，悪い質のデータとはどういうものなのか，ということである．その例が図-3に示されている．日本人でも，研究者，技術者なら問題なく読める文字データであるが，よく見ると問題点が分かる．それはたとえば，最初の行にある文字列においてp，rのストロークが切れていることである．その他4行目，5行目にあるように，文字の接触である．このような誤認識の原因が調べられて，文字の切れ，文字の接触，ノイズがそれぞれ52.1％，20.4％，3.4％と報告されている．また，4行目のデータでは，“extraordinary”の最初の“r”，“a”は，単独でそれらを認識することは，ほとんど不可能である．これはこの単語を知っているから，単語単位で文字を認識しているということになる．ここで，文字認識問題は自然言語情報とのかかわりであり，一般には文脈（コンテキスト）の問題が起こってくる．

一方日本語では，漢字，平仮名，片仮名，それに実際上は英数字を同時に認識しなければならない．特に漢字

はそれ自体，複雑で，偏とつくりがあり，それぞれがまた構造を持っていて，これは実際上，ヨーロッパ言語の単語に相当する．日本語のテキストを読みとるとなると，自然言語の援用が不可欠となってくる．さらにまた，問題がある．それは米国などで目下，大いに研究されているところであるが，チェックの読みとりである．チェックはさまざまに美しく印刷されていて，その上に文字を書き入れるということになっている．そうすると，複雑な背景から文字を切り出すという大きな問題が生じてくる．

このように，問題はさまざまであり，あるレベルまではできあがっているが，その先は，まだまだということ，非常に複雑な問題を抱えているということ，読めるといいながら，読めないという割り切れない状況がお分かりいただけたと思う．また，具体的に機械に与えられるのは，多くの場合，ドキュメントであり，表，図面，絵などを含んでいる．それをどう処理していくか，また分断されたテキストの各部分をどのように正しい順番で読んでいくのかという問題もある．

認識の手法と落し穴

文字認識の手法の基本として，テンプレート・マッチング，特徴抽出と特徴空間で識別，特徴抽出と対象文字の表現によるグラフマッチングがある．

さて，実はこれを説明して書いているうちに，初めの方ですでに枚数をオーバーフローしてしまった．そこで，ここでは基本となるテンプレート・マッチングと特徴抽出と落し穴について述べることにした．

テンプレート・マッチング

テンプレート・マッチングは最も原始的な方法であるが，またそれだけに基本的な手法ともいえる．世界最初のOCR特許は，この原理によるものである．具体的には相関法であり，入力文字画像を$f(x,y)$，i番目のテンプレートを$g_i(x,y)$とすると文字画像の正規化された相関，これを類似度$S(f)$は次のように定義される．

$$S^i(f)=\frac{\iint_R f(x,y)g_i(x,y)dxdy}{\sqrt{\iint_R f(x,y)^2dxdy}\sqrt{\iint_R g_i(x,y)^2dxdy}} \qquad (1)$$

ここでiは想定する文字集合のカテゴリー数をLとすると，$i=1,2,\ldots L$と置かれる．

この方法は原理的に印刷文字の認識に強味を持っているが，弱点も持っている．一番の問題は入力文字画像の位置ずれ，大きさ，スキューなどの幾何的変形（アフィン変換）に弱いということである．しかし濃度の変化，たとえば$f(x,y)\rightarrow Af(x,y)$となっても$S^i(f)$は不変となる．ともかく，相関法が役に立つためには，正規化といわれる前処理が非常に重要となる．この前処理を考える前に，普遍的な非常に重要な前処理がある．それはぼけ変換と呼ばれているもので，文字認識の世界ばかりでなく，コンピュータ・ビジョンの世界でも，基本的な手法として利用されている．直観的に標準文字をぼかしておけば，特に位置ずれに対して，認識率が向上することが期待される．あまりぼかしすぎると，識別の分解能力が落ちるが，理論的にも，このぼけ変換が効果のあることが立証され，実用されている．

さて正規化については，大きく2つの考え方がある．1つは原文字画像を，基準の位置，大きさ等に変換する直接的な方法と，いったん特徴を抽出し，その特徴空間上で，正規化を行うという方法である．この両者が等価であるための，特徴空間の構成は，モーメントを特徴軸とするもの，フーリエ展開をベースにするものに限られることが理論的に明らかとなっている．

テンプレート・マッチングを推し進めていくと，1つには画像平面上での文字濃度のマッチングではなく，画像平面上である特徴を抽出し，たとえば文字のストロークの方向のマッチングをとるという考え方が出てくる．(1)式で$f(x,y)$というスカラー量のかわりにベクトル量を導入することである．したがって，かけ算は内積となる．実はこれはマッチングにおける次元を増加されることになるが，かなり有効であることが確認されている．それゆえ，印刷文字だけではなく，正しく普通に書かれた手書き文字に対しても，このいわゆる方向マッチング法が有効であることが知られている．もちろん，この有効性も，文字の変形の程度による．しかし，この変形の程度がどのぐらいであるか，定量的には，まだ分かっていない．

さて，テンプレート・マッチングでは，1つのカテゴリーに対して，別に1つの標準パターンだけということはない．逆に，いろいろな変形した文字をなんとか読ませようと，複数の標準パターンを用意するという発想が自然と出てくる．しかし，これら標準パターンをどのように選べばよいのかということが問題となる．1つの標準パターンの場合，通常それが属するカテゴリーのデータ文字の平均パターンが選ばれる．そして手書き文字とはいえデータが多い場合，全体の平均図形は，確かに形のととのった，標準的な形となる．

さて，この問題を考えるには若干数学的な表現と概念が必要となる．まず，一般に$n\times m$の画像は$n\times m$個の要素を持つベクトルとして表現する．この場合大事なことは，メッシュを荒くとり，隣合うメッシュ（画素）相互の相関をできるだけ小さくするようにすることである．このためにもぼけ変換が使用される．こうしておいて，あるカテゴリーに属する文字画像全体を1つのベクトル集合と見なし，それを$n\times m$次元の空間でどのように分布するのかということを見るのである．たとえば，それが1つの高次元空間での平面に分布するとする．そうすると，類似度をとる操作は，1つの平均ベクトルの場合は入力ベク

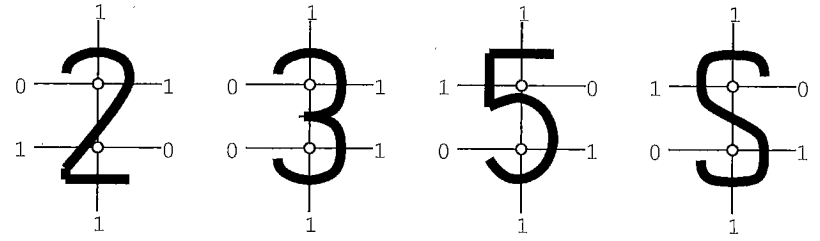


図-4

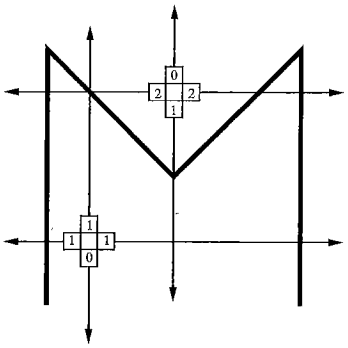


図-5

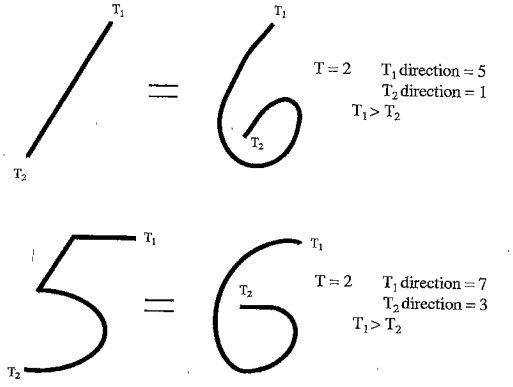


図-6

トルと標準ベクトルとのなす角度をθとすれば$\cos\theta$を値としてとるということになるが，一方，標準とすべきものが平面であるとすると，その平面への射影をとるということになり，明らかに，文字の変形に対して強力となる．そこで問題は，いかにしてこの平面を設定するかということになるが，これはこの文字集合を効率よく線形に表現する2つの直交ベクトルを選んでやればよいということになり，いわゆる共分数行列の固有値問題となり，容易に解ける．これは一般に標準とすべきものは，空間ということになり，部分空間法という名前がつけられている．また，ほぼ同様な手法として複合類似度法とも呼ばれている．これば歴史的にいえば，ASPET 70と呼ばれる高性能OCRの開発の際，飯島により考案されたものであり，部分空間法という名前はハワイ大学の渡辺により発表されたものであり，ほとんど同時期に独立に研究が行われた．

特徴抽出の落し穴

特徴抽出は文字認識の中心的問題であり，先にあげた本では200ページ余にわたって，この問題を取り扱っている．数学的な手法としては，関数展開法があり，モーメント，フーリエ展開がよく使用される．特にモーメントは幾何的変換に対して，不変な特徴を構成できたので，収束性に問題はあるにしても計算パワーの増大と共に，それも苦でなくなってきているので，今なお研究が行われている．

さて，ここでは直視的な手法について，具体的に述べる．簡単であり，しかも効果的手法として，ベル研究所の研究者によって考案されたゾンデ法というのがある．それが**図-4**に示されている．これは特に手書き数字の認識に適しており，当初，先に述べた郵便番号の文字枠内に2つの赤マークを印刷し，この周りに数字を書いていただくということも検討されたと聞いている．この図では白丸の位置から，探索用の線（ゾンデ）が突き出ており，これらを文字のストロークが横切ったかどうかで，1か0の値が各ゾンデに割り振られ，結局図に示すように1，0ベクトルが特徴ベクトルとして文字を表現することになる．そこで文字はわずか5次元のしかも1，0のベクトルで実に単純に表現されることになる．実際，これはOCRの当初において日本でも商用化されたのである．しかし，これば汎用性に欠けている．図の（d）に示されているように，文字“5”と“S”とでは区別がつかない．

この手法を一般化したのが，Glucksmanによって考案された，背景解析の手法であり，これが，**図-5**に示されている．先の探索用のゾンデを，各点で求めるということである．俄然情報量は増大する．しかし，4次のベクトル量それ自身は簡単である．しかも実際には，特定のパターンしか現れない．これを識別に持っていく一番簡単な方法は，特徴ベクトルのヒストグラムをとり，それを特徴空間とすることである．この手法は生国の米国よりも日本において大いに研究され，実用化もされた．特に電総研を中心として，この手法の一般化が行われ，場の効果法として発表された．NTTでも実用化研究が行われ，

実際に実用化された．しかし，この方法も弱点がある．それはストロークの直線性，曲がり具合などの幾何的特徴抽出には弱い．なんらかの幾何的特徴抽出の手法との併用が必要である．

さて，ここで幾何的特徴として，典型的なものは交差点，端点，それに端点でのストロークの方向がある．確かにこれは有効で，手書きの数字の場合，これらの特徴で十分と思われる．しかし，ここに落し穴がある．それが図-6に示されている．ここでは端点数が同じく2の場合であるが，その方向を特徴として加えると，“1”と“5”は容易に区別がつく．しかし，図の例では“1”と“6”，“5”と“6”はこれだけでは区別がつかないのである．この場合は，ストロークのなす凹凸の特性を見れば，確実に区別がつくのであるが，このような状況は随所に現れるのである．人間の無意識の能力とは不可思議である．人間の意識としての知力によって考えられた特徴は，無意識では分かっている文字の変形を，意識することがなかなかできないのである．基本的に特徴抽出とは，対象をできるだけ簡単に表現するためのものである．しかし，いろいろな場合を想定していくと，対象文字画像をできるだけ正確に表現することが必要になってくる．先に述べた凹凸の特徴でも，実はその凹凸の程度という，幾何的性質，すなわち曲率が重要な役割を演ずる場合がある．たとえば“1”と凹凸がゆるやかに書かれた“3”の区別などである．人間はしたがって，特徴抽出とは，情報を落すことではなく，情報を整理することであり，そこで対象の表現問題が提示されることになる．

さて，無理して文章を縮めたところ若干の余裕ができたので，最近私自身が共同研究者や学生とともに興味を持って研究している1つのテーマについて述べてみたい．それは曲線の精密な曲率を求めることである．人間は実にこの曲線の微妙な変化に敏感である．これをB-spline関数を用いて行っているわけであるが，その理由は，1つにはB-splineは区分的にではあるが連結関数であるので精密な曲線を求めることができるということ．したがって曲線近似を行う場合，ほとんど3次の多項式関数でよい．もう1つには，B-splineは非常にノイズに強いということである．CGでは，この曲線での表現であるので，その曲率は2次的である．しかし，OCRまた画像認識では，特徴抽出が問題である．そこで，Bezier曲線で作られた曲線をB-spline関数で近似することを考える．これは前者がいわゆる接続点がなくシームレスであるので，きわめて滑らかな人間の直観に合った曲率を得ることができるからである．しかし，人間では区別がつかないほどにこのBezier曲線に近似されたB-spline関数が，必ずしもBezier曲線と同じ曲率を与えてくれないという問題がある．この問題はある程度分かっているが，それにしても，人間は一体どのようにして微妙な曲線の滑らかさをいかにとらえているのか，この非常に原始的問題に対してもまだまだ技術は人間に及ばない．

文章の読みとり

特に英語はそうであるが，実際には人間は単語単位で読みとっているのではないかと思われるということから，ここ十年ぐらい前から，単語単位の読みとり技術が進んでいる．この場合，単語をまったく全体として，読みとる．すなわち単語内では文字がはなれていようがくっついていようがかまわないという高度な認識を狙ったものがあるが，これは結果的に膨大な数の新しい複合“文字”を考えることになりインプリメンテーションが大変になる．そこで一応，単語内での文字セグメンテーションは行い，たとえば読みとり不能文字があれば，それは一応読みとることのできた周辺の文字から推定するということが行われる．たとえば，エディターという単語は，“デ”の濁音のところがつぶれて，読みとり不能になることがよくある．そうするとエ？ィターという単語から？を推定することになる．エを頭にした文字の数は広辞苑によると128個ある．しかし，最後の文字がーである文字は14文字であり，大幅に減る．もしタカイを入れればエディターの一語のみに減る．このように状況によるが，辞書を活用することにより，かなり読みとり率の向上が見込まれることが知られている．なお，文字認識における自然言語処理の応用については，本学会誌に解説が書かれている[4]．

あとがき

私の手書きの原稿はきわめて，とてもマシンでは読めないと思って書いている．正直言って，この人間の能力は想像を絶する能力である．人間は正に総合力を持って読んでいるのであり，言語など，膨大な知識を基礎にしていることは間違いない．我々はまた，未来の未知の王国への入口に立っているのであるということを，つくづく感じざるを得ない．特に若い人にこの拙文が，なんらかの知的刺激となることを望んでやまない．

参考文献
1) Mori, S., Nishida, H. and Yamada, H.: Optical Character Recognition, Wiley (1999).
2) Tsutsumida, T., Matsui, T., Noumi, T. and Wakahara, T.: Results of IPTP Character Recognition Competitions and Studies on Multi-expert System for Handprinted Numeral Recognition, IEICE Trans. Inf. & Syst., Vol.E77-D, No.7, pp.801-809 (July 1994).
3) Nartker, T. A., Kanai, J. and Rice, S. V.: A Preliminary Report on OCR Problems in Less Document Conversion, Technical Report, TR-92-04, Information Science Research Institute of Nevada, Las Vegas (Apr. 1992).
4) 西野文人: 文字認識における自然言語処理, 情報処理, Vol.34. No.10, pp.1274-1280 (Oct. 1993).

(平成11年2月4日受付)