

解説

郵便物の欧文手書き住所認識技術

浜村 倫行 ((株) 東芝)

郵便区分処理自動化の歴史

郵便局では、毎日大量に集められる郵便物を宛先ごとに仕分けしなければならない。この作業を効率化するため、各国で自動化の取り組みがなされている。日本の場合、枠内に記載された郵便番号を読み取ることで自動化するシステムが1960年代から実用化されている。現在ではさらに住所を番地まで読み取り、配達する順番に並べるところまで自動化されていることをご存じだろうか？ 本稿では郵便物の仕分け自動化に使われている手書き住所認識技術を、欧米言語を中心に日本語住所も交えながら解説する。

図-1は現在の郵便区分機の外観である。東芝がスウェーデン郵政に納入した機体であり、全長は40mである。図の左端から投入された郵便物は住所を認識され、右奥の300個存在する区分箱に仕分けられる。処理能力は1秒間に10通以上にも上る。手書き住所の認識率は各社から公表されていないが、コンペティションなどによると漢字1文字の認識率は99.5%以上(ETL9Bデータベースにて)、1単語の認識率は95%以上(フランス語データベースRIMESにて)が達成されている。

コンペティションでは、認識できないもの=誤認識であるが、実運用で誤認識すると配達人のチェックでカバーするしかなく、配達遅延やコスト増大など影響が大きい。そのため、認識結果に自信がない場合は「分からない」という結果を返し、人手による区分に回すことが重要となる。



図-1 郵便区分機

日本の場合、初期の郵便区分機は、前述のとおり枠に記載された数字の認識機能のみであった。当時の日本では郵便番号は3桁または5桁であり、郵便番号の認識で自動化できるのは配達局レベルの仕分け(差立区分)のみであった。配達局とは宛先に近い郵便局のことで、配達局に集められた郵便物を配達人ごとに分ける作業(配達区分)、さらに配達人が配達ルートに沿って郵便物を並べる作業(道順組立)は人手で行われていた。その後漢字認識の研究が精力的に行われ、1980年代後半には住所の認識により配達人レベルの仕分けが可能となり、配達区分が自動化された。さらに郵便番号の7桁化、認識精度の向上、配達先レベルの認識の実現などにより、1990年代後半には道順組立も自動化された。世界的に見ると、郵便仕分け自動化への取り組みは各国でまちまちであり、日本のように道順組立まで自動化されている国もあれば、すべて人手で行われている国もある。

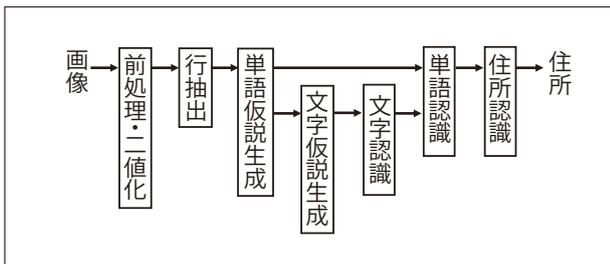


図-2 住所認識の流れ

住所認識の流れ

住所認識は多くの要素技術の複合体である。まず、住所認識全体の流れの例と各要素技術の概要を説明する。欧米言語を対象として説明をするが、多くの要素技術は日本語住所認識でも共通である。

図-2に一般的な住所認識の流れを示す。郵便物の画像が入力されると、ノイズ除去、郵便物位置切り出し、傾き補正などの各種前処理が施された後、文字部が1（黒画素）、背景が0（白画素）となるよう各ピクセルの輝度を二値に変換する二値化処理が行われる。背景が白で一様、文字は黒、と仮定できる場合は単純な固定しきい値処理でも十分であるが、明るさが変動する場合や背景が一様でない場合はしきい値の自動決定法である大津法¹⁾が広く用いられている。

次に、同一行に属する黒画素をグループ化する行抽出処理が行われる。行が傾かずに記載され、かつ行間が十分に離れている場合は、水平方向の黒画素数をカウントする処理（射影と呼ばれる）により黒画素の少ない位置を見つけ出し、そこで上下に分割することで行を抽出することができる。図-3の点線が行抽出結果の例である。行が傾いていたり、途中で曲がっていたり、また行間が狭く接触していたりする場合は工夫が必要であり、さまざまな方法が提案されている²⁾。

次に、行を単語に分割する単語仮説生成処理が行われる。単語間が十分に離れていない場合は、分割の仕方を一通りに定めることができないため、複数の分割候補が生成される。こうして生成される単語画像の候補を単語仮説と呼ぶ。図-3の実線の長方

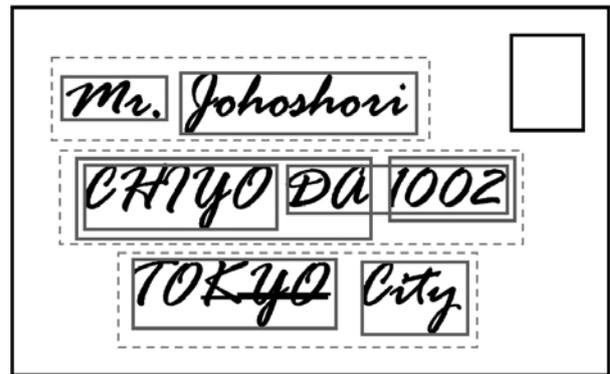


図-3 行抽出結果と単語仮説生成結果

形が単語仮説の例である。日本語の場合は単語間が離れていないため本処理は行われない。

さらに、各単語仮説（日本語の場合は各行）は文字に分割される。詳細は後述するが、単語分割と同様に文字分割も複数の候補が考えられ、複数の文字仮説が生成される。

生成された各文字仮説に対し、文字認識が行われる。文字認識は、一般に特徴抽出処理と識別処理に分けられる。特徴抽出処理では、文字仮説画像から「特徴量」が抽出される。特徴量とは、複数の数値の集まりであり、すなわち数値が d 個であれば d 次元のベクトルで表される。画像をセルに分けセル内の輝度値を平均するもの、平均・分散などの各種統計量を並べたもの、輝度勾配方向（輪郭方向）を数値化したもの、構造を抽出するものなど、さまざまな特徴量が提案されている。図-4は 4×4 のセルに分けている例である。

こうして得られた特徴ベクトル x をもとに、識別処理が行われる。識別処理にはサポートベクトルマシン（Support Vector Machine : SVM）、ニューラルネットワーク、部分空間法、疑似ベイズなどさまざまな一般のパターン認識手法を用いることができる³⁾。識別処理の結果は、一般に、各アルファベットにスコアが付与された形式で表すことができる。以後、この形式のことを「文字認識結果」と呼ぶことにする。図-4では文字認識結果をスコアの順にソートして表示している。

各文字仮説の文字認識結果を用いて、住所データ

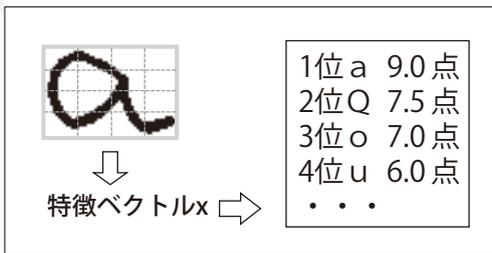


図-4 文字認識



図-5 手書き単語の例

ベースを利用しながら単語認識，住所認識が行われる。郵便物の手書き住所認識では，住所データベースの積極的な利用が肝要であり，特色でもある。以後は単語認識，住所認識について詳細に説明する。

手書き単語の認識

□手書き単語認識の難しさ

図-5は，フランスの住所に存在する単語である。何と書かれているか分かるだろうか？“adiene”，“aclieru”など，さまざまな読み方ができそうだが，正解は“adresse”である。1つ1つの文字が認識できても，文字の境目が分からないと単語としては認識できないことが分かるであろう。日本語でも，「明」と「日月」のように境目の違いから複数の読み方ができる例は多い。

しかし，おそらくフランス人は図-5の単語を読むことができる。それは“adiene”，“aclieru”という単語は住所に存在しないが“adresse”という単語は存在することを知っているためである。このように，単語を認識するためには単語の知識が重要であり，一般に「単語認識」とは単語辞書の中から正解単語を選び出す処理のことを指す。

単語認識には大きく2つの手法が存在する。1つは，個別の文字を認識してから単語認識を行おうとする手法であり，分析的手法（Analytic Approach）と呼ばれる。もう1つは，個別の文字を扱うことなく直接単語を読み取ろうとする手法であり，全体的

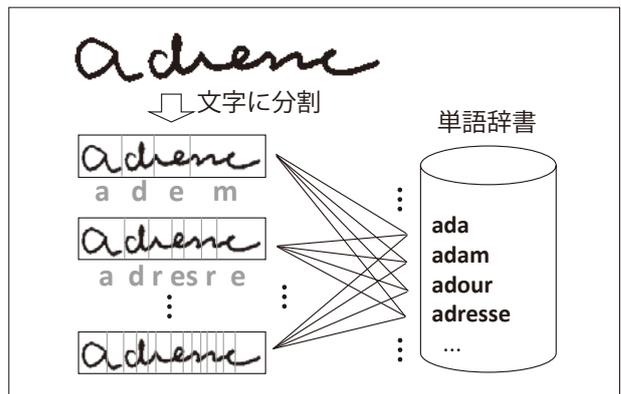


図-6 分析的手法

手法（Holistic Approach）と呼ばれる。以下，両手法について詳しく説明する。

□分析的手法

分析的手法では，個別の文字認識結果を利用して単語を認識する。個別の文字認識処理を実行するためには，単語画像を文字画像に分解する必要があるが，前述のとおり分解の仕方は一意に定まらない。そこで，図-6のように多数の分解候補を生成する。分解位置座標の候補は，輪郭形状の変化や黒画素の密度変化などを利用して決める。図-6では文字認識結果の1位候補のみを各文字仮説の下に示している。

これらの分解候補とその文字認識結果を，住所データベースから生成される単語辞書と照合し，全組合せの中で最も照合スコアの高い組合せをもって単語認識結果とする。照合スコアとしては，文字認識スコアの平均が広く用いられているが，近年は確率を計算する方法も提案され，認識精度向上が図られている⁴⁾。

□全体的手法

全体的手法は，個別の文字を認識するのではなく，直接単語を読み取ろうとする手法である。全体的手法では多くの場合，隠れマルコフモデル（Hidden Markov Model：HMM）と呼ばれる理論が用いられている。

HMMによる単語認識の一般的な流れを図-7に

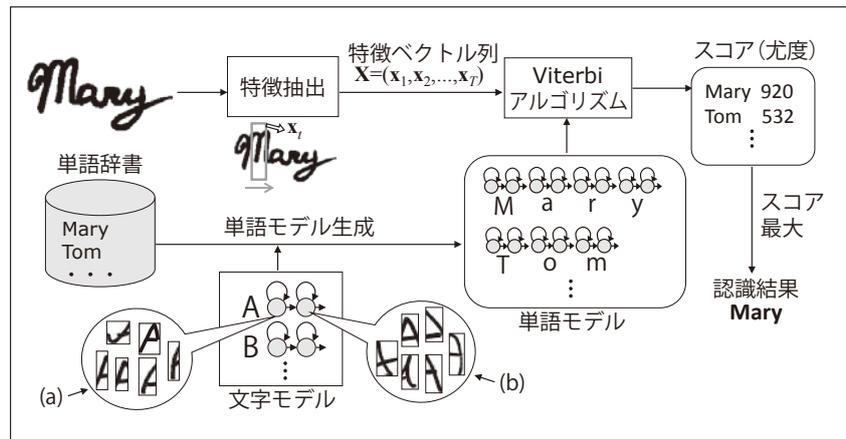


図-7 隠れマルコフモデル (HMM) による単語認識

示す。認識対象である単語画像が入力されると、まず特徴抽出処理が行われる。図のように、左から右へスライドする窓を考え、窓の内側の画像を順次切り出す。これらの画像の1つ1つから特徴量を抽出する。特徴量には、先に述べた文字認識における特徴量と同じものを用いることができる。左から t 番目の画像から抽出された特徴ベクトルを x_t と表し、 x_1 から x_T の全 T 個を並べたものを特徴ベクトル列 X とする。HMM による単語認識では、単語辞書内の各単語 w の「モデル」を考え、モデルから特徴ベクトル列 X が「生成される」と考え、生成される確率 $p(X|w)$ (尤度と呼ばれる) が最大となる単語を単語認識結果とする。

尤度を計算するために、各アルファベットを生成する「文字モデル」を利用する。文字モデルは図のように複数の丸印 (状態と呼ばれる) とそれをつなぐ矢印で表される。各状態には、図中 (a), (b) のような「文字の切れ端」が大量に記憶されている (実際には特徴空間上での切れ端の分布が記憶される)。矢印は状態遷移とその確率を表し、多くの場合図のように自分自身と隣の状態のどちらかにのみ遷移するモデルが用いられる。分布や状態遷移確率などのパラメータは、Baum-Welch アルゴリズムと呼ばれる方法により、正解の教示された大量の単語画像から学習することができる。

文字モデルを連結したものが「単語モデル」である。単語モデルの矢印に従って状態遷移しながら特徴ベクトル $x_1 \sim x_T$ が順に生成される確率が、求め

たい値である尤度 $p(X|w)$ である。この値は Viterbi アルゴリズムと呼ばれる方法で近似的に計算することができる。その大小関係により単語認識を行うことができる。

□ 両アプローチの比較

文字は黒画素の2次元配置で表現されるものであり、分析的手法は特徴抽出時に2次元配置情報を直接とらえることができるため、文字単体の識別能力では全体的手法に勝る。一方、全体的手法は文字の分割処理が不要であり、図-5のような分割位置が曖昧な対象に強い。このため、文字が複雑で文字単体の識別能力が重要になる日本語では分析的手法が、文字形状がシンプルで分割位置の曖昧な欧米言語では全体的手法が主流となっている。また、両アプローチは相補的であるため、併用するアプローチも提案されている。

手書き住所の認識

郵便区分機で最終的に必要な情報は、1つ1つの文字、単語の認識結果ではなく、住所である。住所は図-8 (a) のように木構造で表すことができる。すなわち、住所認識はこの木構造の末端ノード (図-8 (a) の例では street No の8つのノード) の中から正解を選び出す処理であり、木構造の探索問題とみなすことができる。

選び出すにあたり、すべてのノードを対象に単語

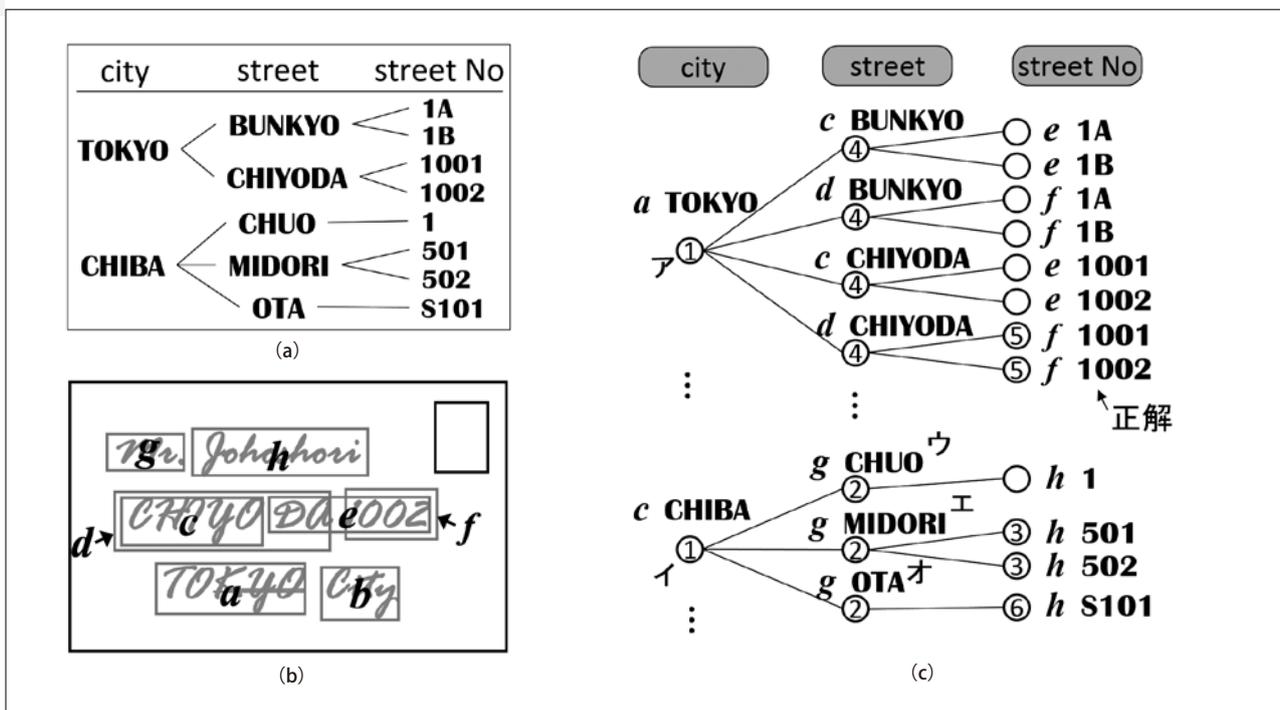


図-8 (a) 住所データベース, (b) 単語仮説, (c) 探索木

認識処理を行うのが理想的であるが、住所データベースの末端ノード数は数百万～数千万にまで及ぶことがあるため、膨大な処理時間がかかってしまう。制限時間内でいかに処理するかが重要となる。一般的には、上位階層から順に認識し、スコアの低いノードを枝刈りして絞り込むことで処理時間を削減する。図-8 (a) で説明すると、まず最上位階層である city の認識をし、たとえば CHIBA であると確定し（絞り込み）、次に CHIBA の下に属する CHUO, MIDORI, OTA のみを単語辞書として street の認識を行い、たとえば MIDORI と OTA の2候補に絞り込み、その下に属する street No を認識し…という具合である。

実際には探索対象は住所データベースではなく、図-8 (c) のように単語仮説と住所データベースをマージしたものとなる。住所データベースの各ノードに対応する単語仮説は複数考えられるためである。日本語の場合は単語仮説を用いることができないが、「上位階層の右隣」であることを何らかの記号で表現すれば同様の探索木で表すことができる。

上述したような上位階層から順に絞り込んでいく方法は「ビーム探索」と呼ばれる。一般に、ビーム

探索では、絞り込み度合の調節が必要である。絞り込み過ぎると正解を枝刈りしてしまう可能性が高まり、逆に絞り込みが弱すぎると制限時間内に末端ノードに到達できなくなる。制限時間の中でできるだけ絞り込みを弱くする調節が必要である。

しかし、文字認識などの探索前の処理にかかる時間が郵便物ごとにばらついている場合、探索に残される時間が変動するため、絞り込み度合をあらかじめ調節しておくことができない。そのため、近年は調節が不要である「最良優先探索」という探索法の適用が提案されている。最良優先探索とは、常に最もスコアの低いノードを探索していく方法であり、ビーム探索とは異なりスコアの低いノードの枝刈りは行われぬ。図-8 (c) に探索順序の例を示す。ノードの丸印内の数字が探索順序である。まず city 階層（ア、イ）が処理される。次に、それらの中でイがスコア最大であったとすると、その子ノード（ウ、エ、オ）が処理される。ここまではビーム探索と変わらない順序であるが、ビーム探索では次にウ、エ、オの3つを比較するのに対し、最良優先探索では未探索のノードすべて、つまりア、ウ、エ、オの4つを比較する。そして4つの中でエが最大で

あったとすると、その子ノードが処理される。ここでさらに、ビーム探索の場合は末端ノードに到達したら処理終了であるが、最良優先探索では制限時間内であれば次にスコアの高いノードの探索を進める。ビーム探索ではアを枝刈りした時点で正解到達が不可能であったところ、最良優先探索では図-8(c)のとおり制限時間内に4~6まで処理が進み、5で正解に到達できている。このように、制限時間の長さに応じ探索量が自動的に変わるため、ビーム探索のような調節が不要であることが分かる。

最良優先探索を適用する際の課題は、スコアの計算方法である。単語認識結果として得られるスコアは、そのままでは用いることができない。なぜなら、最良優先探索では階層の異なるノード、たとえばアとエを比較しなければならないからである。アを評価するには“a TOKYO”のスコア(たとえば50点とする)のみ参照すればよいが、エを評価するには“g MIDORI”のスコア(37点とする)だけでなくその親ノードである“c CHIBA”のスコア(83点とする)も考慮しなければならない。つまり、「50点 vs. 『83点 & 37点』」という比較が必要となる。単純に83点と37点の平均(60点)を用いることも考えられるが、より良いスコアとして事後確率を近似的に計算し利用するベイズ最良優先探索法(Bayesian Best-First Search : BB Search)が提案されている⁵⁾。スコア改善によりアを選択できればより速く正解に到達できる。

以上、制限時間を有効に使う方法を述べてきたので、時間を余らせると無駄になると思われるかもしれない。しかし、実は短い時間で処理を完了することも効果的である。これは、区分機における時間制約が、平均と最大の2つに分かれていることによる。平均処理時間は、区分機の処理能力とCPUコア数から定まるもので、たとえば1秒間に10通処理する区分機でCPUコアが4つであれば400ミリ秒となる。最大処理時間は、スキャナで画像取得してから区分箱に到達するまでの時間で定まるもので、一般に数百ミリ秒~数秒程度である。

平均より短い時間で処理できる郵便物があれば、

その分ほかの郵便物の制限時間を最大制約を超えない範囲で延ばすことができ、認識精度向上に寄与できる。たとえば活字で書かれた住所では、探索の途中で明らかに正解に到達したと判断できる場合があり、素早く探索を打ち切ることが可能である。また、逆に明らかに認識できないことが判断できた場合も、素早くリジェクト判定をすることが可能である。

今後の展望

住所認識を構成する各処理の精度、速度の向上は当然期待するところであるが、それだけではなくもう一步踏み込み、図-2の一般的な処理の流れを変えるようなブレークスルー技術に期待したい。たとえば、文字・単語認識では、先に仮説を生成する必要があるが、顔や人の検出技術のように仮説生成なしで位置検出できるようになれば大きな精度向上に結び付く可能性がある。また、現状では文字・単語認識にのみ機械学習手法が適用されているが、行抽出などにまで適用範囲を拡大できれば、精度向上はもちろんのこと開発コストの削減にもつながるであろう。

参考文献

- 1) 大津展之: 判別および最小2乗規準に基づく自動しきい値選定法, 信学論(D), Vol.63, No.4, pp.349-356 (Apr. 1980).
- 2) Ziaratban, M. and Faez, K.: Adaptive Script-independent Text Line Extraction, IEICE Transactions on Information and Systems, Vol.E94-D, No.4, pp.866-877 (Apr. 2011).
- 3) C. M. ビショップ: パターン認識と機械学習(上)(下), シュプリンガー・ジャパン(2007).
- 4) 浜村倫行, 赤木琢磨, 入江文平: 解析的単語認識における事後確率を用いた評価関数, 信学論(D), Vol.91, No.9, pp.2325-2333 (Sep. 2008).
- 5) Hamamura, T., Akagi, T. and Irie, B.: Bayesian Best-First Search for Pattern Recognition - Application to Address Recognition -, Proc. ICDAR2009, pp.461-465 (July 2009).
(2013年2月27日受付)

浜村 倫行 tomoyuki.hamamura@toshiba.co.jp

1999年東大大学院電子情報工学専攻修士課程修了。同年(株)東芝に入社。2012年東大大学院情報理工システム情報博士課程修了。博士(情報理工学)。文字認識技術の研究開発に従事。電子情報通信学会会員。