

4D-1 手書き住所読取りにおける パターン処理と連携した住所知識処理方式

下村 秀樹* 福島 俊一* 森 義和**

*NEC情報メディア研究所 **NEC情報システムズ

1. はじめに

手書き住所の読取りは、郵便あて名区分機等で実用化されているが、いっそうの高性能化への要求がある。従来の一般的な手書き住所読取りでは、(1)住所記載領域切出し、(2)個別文字領域切出し、(3)文字認識、(4)住所知識処理、が逐次的に行われる。この流れのもとで読取り性能を高める一つの方針としては、住所知識処理以前の「パターン処理」が正解をもらさぬよう多くの切出し・文字認識候補を出し、住所知識処理でその候補の組合せを「検証」することが考えられる。「検証」とは、辞書や規則との照合に基づく読取り候補の絞り込みである。しかし、パターン処理で出力候補数を増加させると、正解含有率が増す反面、偶然検証規則を満たす（競合や誤読の原因となる）候補も増すという問題がある。また、手書き住所ではくずし字、文字の接触等があるため、処理速度やメモリといった実質的な制約のもとで、パターン処理だけによって正解含有率を高めることは難しい。

これらの問題に対して、我々は、(1)パターン処理から出力可能な詳細情報の積極的利用により住所知識処理の検証能力を高める、(2)検証でも読取り結果を確定できなかった原因部分に対して必要かつ詳細なパターン処理を住所知識処理からトップダウン的に再起動する、というアプローチが有効であると考え（図1）。本稿では、このようにパターン処理との連携を強化した住所知識処理の必要性和、実現例を述べる。

2. 住所知識処理とパターン処理の連携強化の必要性

2.1 パターン処理からの詳細情報の必要性

正解含有率を高めるためにパターン処理が多くの候補を出力すると、競合や誤読も増加する危険がある。したがって、住所知識処理に対しては、不適切な候補を除去する検証能力の強化が、よりいっそう求められる。

これに対しては、パターン処理で得られる情報、例えば、文字領域形状やサイズ、パターン処理中に得られる途中結果情報等を、より積極的に利用することが有効である。なお、この検証能力の強化は、次に述べるトップダウン処理要求の対象箇所との絞り込みにも有効である。

2.2 パターン処理のトップダウン的利用の必要性

より詳細なパターン処理が必要な箇所に対して、住所知識処理の中から再処理を要求する、「パターン処理のトップダウン利用」には、次のメリットがある。

(1)住所読取りに不要なパターン処理の回避

手書きの住所には、文字の接触、変形など、パターン処理が正解を出しにくいさまざまな要因がある。初めからこれらの要因をすべて考慮してパターン処理を行うと、処理時間の増加、不要な出力候補の増加を招く。これに対し、いったん住所知識処理を行えば、住所を読み取るために詳細なパターン処理が必要な箇所を絞り込めるので、不要なパターン処理を避けることができる。

(2)住所知識処理中の情報によるパターン処理の精密化

住所知識処理の途中では、読めない箇所に予測される文字や文字数、また候補となっている文字列の文字ピッチなど、さまざまな情報が得られる。パターン処理にこれらの情報をフィードバックすることによって、パターン処理をより詳細かつ精密に行うことができる。

3. パターン処理情報の積極的利用による検証

本章では、パターン処理情報の積極的利用による、読取り候補検証の強化の例について述べる。

3.1 ブロック形状による当該文字らしさ検証

町名の読取りでは、認識不良があっても、住所の階層構造からの制約によって、不良箇所の文字列を推測できる場合がある。図2の例（架空の住所）では、「宮前」が接触して切出し・認識に失敗したが、市名と町名の前後関係から、読めない領域は「宮前」をはじめとするいくつかの文字列に限定できる。このとき、領域のブロック形状（文字領域切出し処理から出力）を参照すれば、この部分に1文字あるいは3文字以上が入る可能性はかなり低いことがわかり、文字数として不適切な推測を除去することができる。

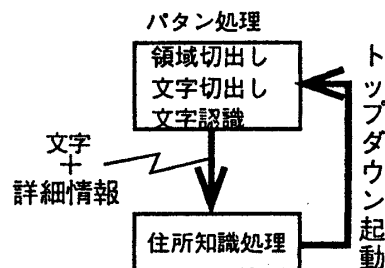


図1 住所知識処理とパターン処理の連携強化

なお、候補中の読取り不能位置を効率よく知るためには、文字列の一部が認識できていなくても、虫食い状態で効率よく候補を生成するアルゴリズムが必要である。これに対しては、「文字タグ法」というアルゴリズムを開発した[1]。

3.2 バタン特徴の類似性に基づく当該文字らしさ検証

例えば、町名部分の認識不良箇所が、住所知識処理により「一」のような単純な形状の文字と推測されたとする。しかし、もしそれに対応する記載ボタンが非常に複雑なものであれば、推測は適切でない。すなわち、認識不良領域のボタンの特徴と推測文字が持つそれとが大きく異なる場合、候補を絞り込むことができる。ボタンの特徴としては、ボタン処理の途中結果情報（例えば、領域の黒さ情報）、文字の画数などが考えられる。画数を利用した場合は、認識不良領域に対する認識候補文字の画数と推測文字の画数の比較という形で、比較的容易に実現できる。

4. 住所知識処理からの文字再検証

3章では、ボタン処理からの情報を利用した読取り候補の限定を述べた。この限定によっても候補が一意に絞れない場合、あるいは認識不良箇所が多く、結果に確信が持てない場合は、本章で述べるボタン処理の再起動が有効である。

4.1 町名部分の認識不良箇所の再切出し・再認識

住所知識処理の様々な検証でも認識不良が原因で結果を確定できない場合、その領域を再度切出し・認識すれば、正しい住所を読み取れる可能性がある。このとき住所知識処理は、該当領域が読まれるべき文字列を推測し、その候補情報をボタン処理に渡すことができる。したがって、再起動されたボタン処理は、最初よりも文字切出し・認識を精密に行うことができる。例えば、図2の「宮前」の接触については、検証で残った「宮前」をはじめとするいくつかの候補文字列を、住所知識処理からボタン処理に渡すことができる。

4.2 類似文字の競合判定

町名には「東町」「西町」「南町」「北町」などが多く現れ、認識系の性質によっては、これらを区別するキーとな

る文字（この場合では「東西南北」）の識別が難しい場合がある。図2では、「南」と「西」が競合した。認識系の弱点である類似文字が候補に複数個残った場合、認識候補順位だけで優劣を判断するのは危険である。これに対しては、特定の競合しやすい文字群だけを扱う認識系を用意し、それを住所知識処理から起動し判別する。

4.3 丁目番地部分のアラビア数字再切出し認識

丁目番地の数値は単純な図形の組合せで、かつ狭い領域に書かれるため、接触や入り組みを起こすことが多い。図2では、「27」が接触して切出し不良となっている。このような、アラビア数字の接触（比較的頻度が高い）に対して、ボタン処理と連携した次のような対処が考えられる。

まず、数字など丁目番地部分に使用される文字以外に認識された（あるいはリジェクトされた）領域は、接触や入り組みにより切出し不良が発生したと仮定する。そして、その部分をアラビア数字と見なして規則と照合し、規則をうまく満たす部分についてのみ、再度、切出しおよび認識を行おうというものである。規則を満たさない部分は再処理を行わないので、無駄なボタン処理を起動しなくて済む。また、町名のとくと違って認識不良箇所がどんな文字であるかまでを住所知識処理で予測することはできないが、町名を読み取ったあとであれば、丁目番地として許される数字の桁数はわかる。この情報をボタン処理部に渡すことでボタン処理を精密化できる。

5. おわりに

手書き住所の読取りについて、ボタン処理との連携を強化した住所知識処理と、その実施例を述べた。実装および効果の評価を行いたい。

謝辞

本稿で述べた内容の基本的なアイデア、ならびに実装に関して議論していただいた山内俊史氏(NEC)に感謝します。

参考文献

[1]福島他:手書き住所読取りのための町名検索アルゴリズム—住所文字タグ法—, 第50回情処全大, 4D-6 (予定)

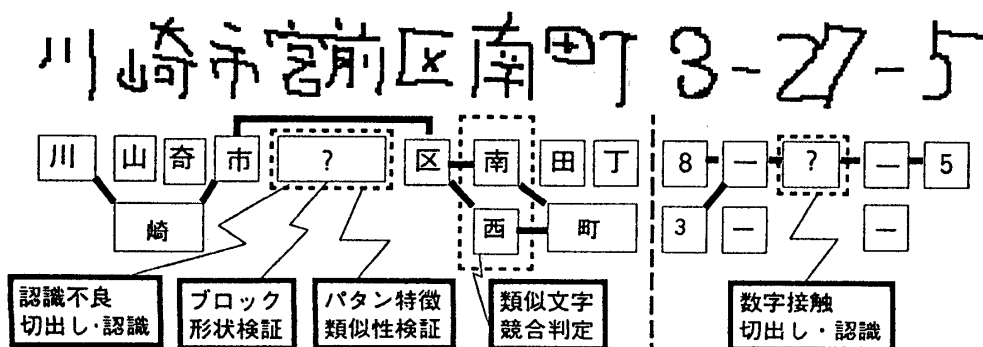


図2 バタン処理と連携した住所知識処理の例（記載住所は架空）