

文字方向および行方向に依存しない オンライン手書き枠なし文字列認識

小沼 元輝 中川 正樹

東京農工大学大学院工学研究科

〒184-8588 東京都小金井市中町 2-24-16

E-mail: motoki@hands.ei.tuat.ac.jp

あらまし 本報告では、筆記制限の緩和を目的に、文字方向および行方向に依存しない文字列認識手法を提案する。ペン入力においてユーザの発想を妨げないためには、できる限り筆記制限を排除する必要がある。手書き枠なし文字列認識における既存の研究では、横書きの文字列を認識対象としたものがほとんどであった。本手法では、ペン先座標の時系列情報を利用して文字の方向と行の方向を推定し、その仮定に基づいて、文脈処理を統合した既存の枠なし手書き文字認識手法を適応することにより、文字方向と行方向に依存しない枠なし手書き文字列認識を実現した。本手法は、縦書きや横書き、そして、斜め書きの混在にも原理的に対応可能である。これは、タブレット PC や対話型電子白板などでの自由な筆記認識に有効であると考えている。

キーワード オンライン認識, 手書き文字認識, 枠なし認識, 筆記方向自由認識

An On-line Writing-box-free Line-direction Free and Character-orientation Free Recognition System for Handwritten Text

Motoki Onuma Masaki Nakagawa

Graduate School of Technology, Tokyo University of Agriculture and Technology,

2-24-16 Naka-cho, Koganei-shi, Tokyo, 184-8588 Japan

E-mail: motoki@hands.ei.tuat.ac.jp

Abstract This paper describes a writing-box-free on-line handwritten text recognition method that is liberated from constraints on writing direction (line direction) and character orientation. This method estimates the line direction and character orientation using the time sequence information of pen-tip coordinates and employs a writing-box-free recognizer with context processing combined assuming the line direction and character orientation. The method can theoretically cope with mixture of vertical, horizontal and skewed lines with arbitrary character orientations. It is expected useful for tablet PC's or interactive electronic whiteboards.

Keyword on-line recognition, handwritten character recognition, writing-box-free recognition, writing-direction-free recognition

1. はじめに

近年、PDA の液晶画面の大型化や対話型電子白板の登場、そして、Microsoft 社による Tablet PC の発表、また、Anoto ペンや e-pen などの紙に書く手書きの電子化技術の実用化により、筆記枠制限を課さないオンライン手書き文字列認識に対する新しい需要が開けつつある。筆記枠のなかに文字を一字ずつ筆記していく単一文字認識（以後、単文字認識）だけではなく、筆記枠のない入力面への手書き文字列認識（以後、枠な

し文字列認識）にもニーズが寄せられ、我々の技術も製品として利用されている[1].

しかしながら、現在発表されている枠なし文字列認識技術における既存の研究では、横書きの文字列を認識対象としたものがほとんどである[1, 2].

これに対して、我々は、行方向を緩和する手法を提案し、縦書きや横書き、斜め書きの混在を認識できる手法を提案した[3]. しかし、対話型電子白板でしばしば筆記される行も文字も傾いた文字列には対応できて

いなかった。

本報告では、上記手法を見直し、文字方向に対する制限を除くとともに、先の手法の問題点も改善して、より包括的に文字方向・行方向制限を緩和する枠なし文字列認識を提示する。

ここで述べる方式では、現在まで研究されてきた高精度の単文字認識を変更なく組み込むことができるため、より自然で、かつ、高性能な手書き文字入力を実現することができる。

2. 文字方向と行方向の定義

ここではまず用語の定義を行う。ストロークとは、ペンダウンからペンアップまでに採集されるペン先座標（筆点座標）の系列を意味し、文字方向とは文字の上辺から底辺への方向を指し、行方向とは文字列における書き始めから文字が連なる方向を指す（図1）。行方向は我々の一般的理解と同じであるが、文字方向は通常の筆記では下向きになるので注意されたい。改行、行方向の変化点、一定間隔以上の空白などで分離される文字列を文字列構成要素と呼び、それぞれが行方向を持つ（図2）。文字方向と行方向は独立である。

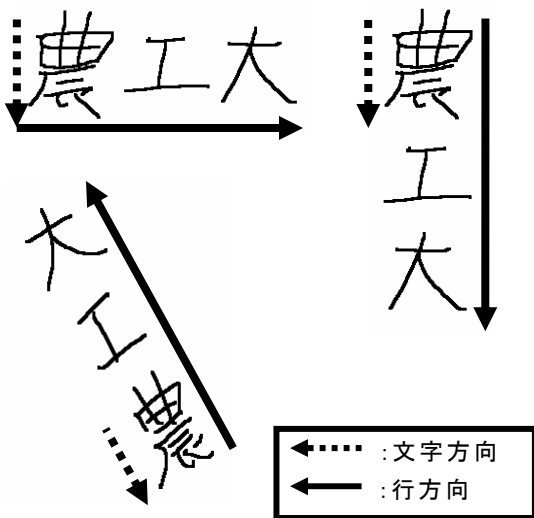


図1. 文字方向と行方向

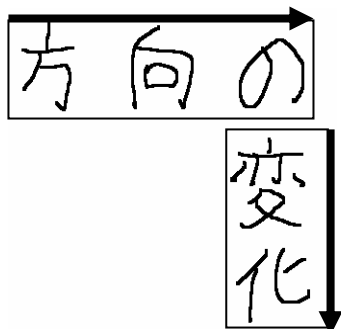


図2. 文字列構成要素と行方向

3. 認識処理の構成

行方向・文字方向に依存しない文字列認識は、文字列要素への分解、各文字列要素における文字方向と行方向の推定および仮定、推定・仮定された方向での文字列認識、最終的な評価を伴い、その処理の流れは図3に従う。

本章では、以下、これらの処理の順に提示する。

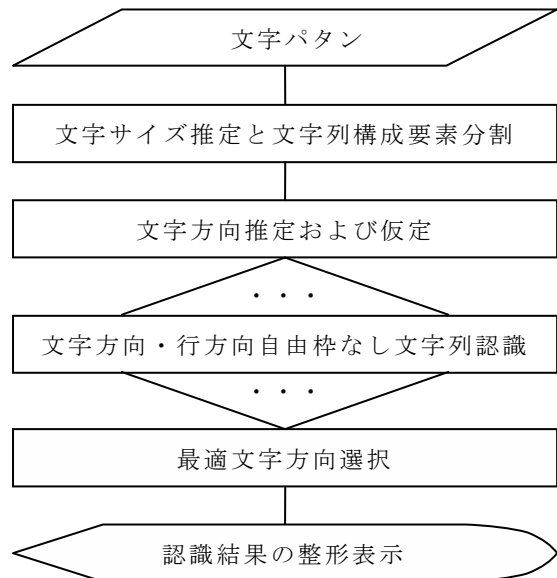


図3. 処理の流れ

3.1. 文字サイズ推定と文字列構成要素分割

文字サイズの推定と文字列構成要素への分離を、図4の処理により行う。

(1) 文字サイズ推定処理

画面上に記入されたストローク列から、文字サイズ（外接矩形を正方形としたときの辺の長さ）を推定する。ここで推定された文字サイズは、改行・文字列間隔検出、仮分割処理および認識処理に用いられる。

文字サイズ推定処理では、画面上の全てのストロークに対して、その外接矩形の長いほうの辺の長さ（長辺長）の分布から文字サイズを推定する（図5）。

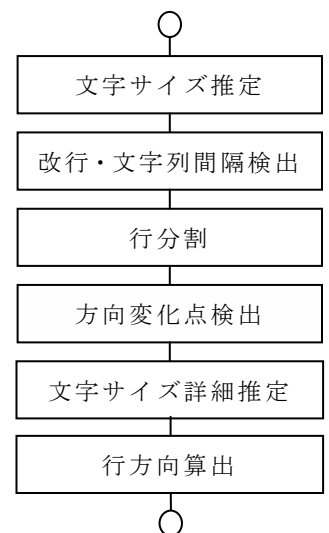


図4. 処理の流れ (3.1)

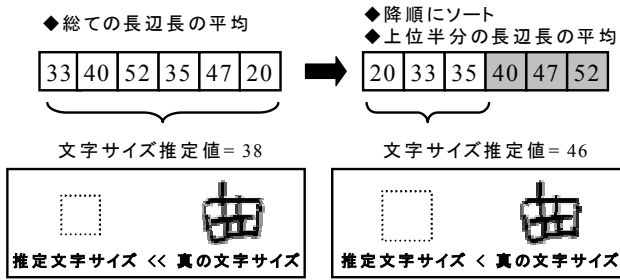


図 5.文字サイズ推定処理例：“曲”（総画数 6）

(2) 改行・文字列間隔検出処理

上記，文字サイズ推定処理で求められた推定文字サイズをもとに，改行検出および文字列間隔の検出をクラスタリングにより行う。

ストローク間隔（前のストロークの終点から次のストロークの始点への距離）には，文字列内の小さい遷移と文字列間の大きな遷移の 2 種類があり得る．文字サイズと比較して，ある基準より大きいものが含まれる場合は，改行があるものと仮定する．そして，ストローク間隔のクラスタリングを行うことで，文字列内遷移と区別して改行を検出する（図 6）。

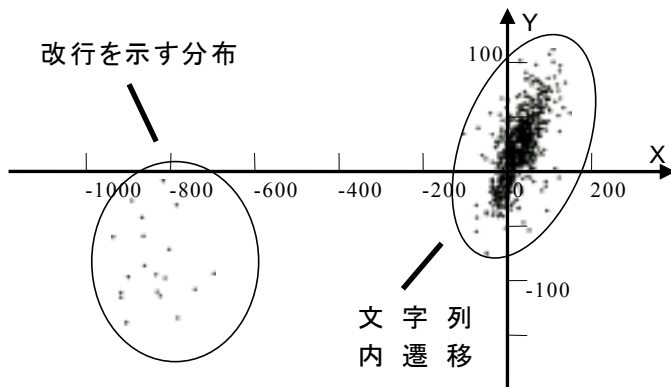


図 6. ストローク間隔の遷移のクラスタリング

(3) 行分割処理

検出した改行・文字列間隔を用いて，画面上のストローク列を文字列構成要素に分割する。

(4) 方向変化点検出処理

簡条書きなどにおいて，横書きで書いた見出しの後に縦書きの行を続けて書く場合がある．また，図などに阻まれた画面上で方向を変えて続けて書くといった場合には，改行や文字列間隔を検出するだけでは行として分割ができない（図 7）．したがって，行方向の推定が難しくなる．

そこで，行方向が途中で急激に変化している箇所を特定することで，文字列の分割や行方向の推定を正確に行うことができる．方向変化点の検出のためには，

対象となる文字列の始点と終点を結ぶ直線からの最遠点を求めることで実現できる．また，この手法を再帰的に用いることで，複数の行方向変化に対応できる．これらの方向変化点により，文字列構成要素はさらに短い文字列構成要素に分割される（図 8）。

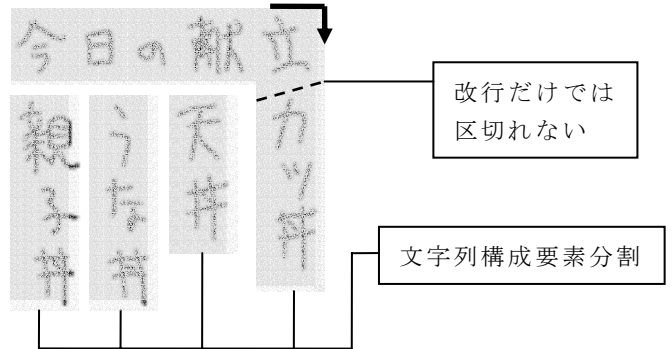


図 7. 改行・文字列間隔で分割できない例

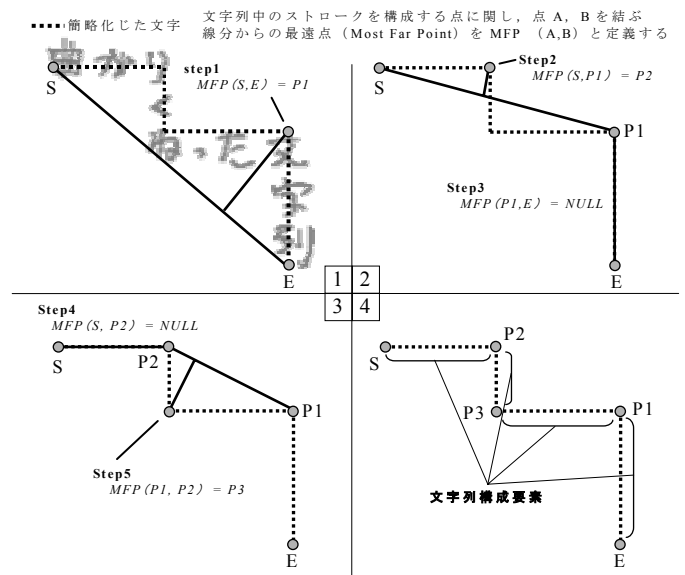


図 8. 方向変化点検出処理

(5) 文字サイズ詳細推定処理

文字列構成要素ごとで文字サイズが変動することを考慮して，文字列構成要素ごとに文字サイズを詳細に推定し直す．方法は文字サイズ推定処理と同じであるが，それを文字列構成要素ごとに適用することで，文字サイズ推定の精度を向上させる．

(6) 行方向算出

文字列構成要素ごとに，最初のストロークの始点座標と最終ストロークの終点座標から行方向を求める．

3.2. 文字方向推定および仮定

文字方向の推定および仮定を、図9の処理により行う。

(1) 文字方向推定処理

文字を記入する際、ストローク間の運筆を除いた筆点座標の変位は、右方向か下方向への移動が顕著に見られる。

図9. 処理の流れ (3.2)

これは、日本語の、特に漢字

が、右方向あるいは下方向に筆記されるストロークを多く含むためである。このことから、方向別に変位をヒストグラムに取ると、文字方向とその90度($\pi/2$)程度離れたところに二つのピークが現れる(図10)。

したがって、文字列全体で見た場合、これら二つのピークを検出することで、文字方向を推定することが可能である。文字方向が推定できれば、文字方向を下向きになるように文字列全体を回転させることによって、通常どおりの認識処理を行うことができる。

具体的には角度 θ 方向へのヒストグラムの強度を $f(\theta)$ として、 $f(\theta)*f(\theta+\pi/2)$ のピークを探し、 θ の方向を文字方向とすることで実現する。さらにガウス関数 $g(\delta) = \exp(-\delta^2/\sigma^2)$ をこれに畳み込んで、右上がりの文字のように文字の下方向と右方向の角度差が $\pi/2$ から多少ずれている場合でも、二つのピークを安定に抽出することができる(図11)。

もし、文字方向が下向きか上向きしかないと分かっている場合、Y座標、つまり縦方向の筆点座標変動から文字方向の推定を行う。ここでは上下方向に記入方向を限定して考えているので、記入方向が画面上方向か画面下方向かどちらかに判定できればよい。この場合、横書き・縦書きに関わらず筆点間のY方向の差を加えていくことで文字方向を推定できる。画面の座標系を左上に原点をとり、筆点列のY座標変動の和が「正」の値であれば文字方向が下向き、「負」の値であれば上向きと判断できる。

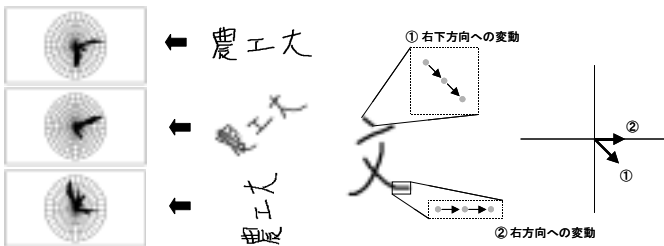


図10. 文字に存在する二つのピーク

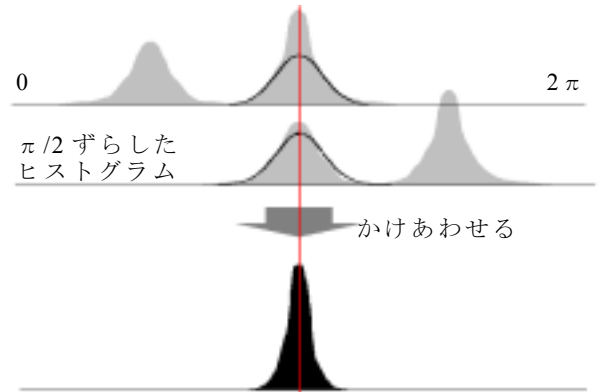


図11. 推定処理

(2) 文字方向仮定処理

本処理では、図12に示すように、文字列構成要素の行方向から良く現れる文字方向を仮定する。この文字方向としては、行方向の正負方向、それに垂直な2方向、入力装置の上下方向、左右方向を仮定している。しかし、これらの文字方向だけに限定するわけではない。さらに、文字方向の推定によって絞り込める場合は、これらの仮定を少なくしたり省いたりすることができる。

文字方向が表示画面に対し上下方向に限られるときは、この処理は省略できる。

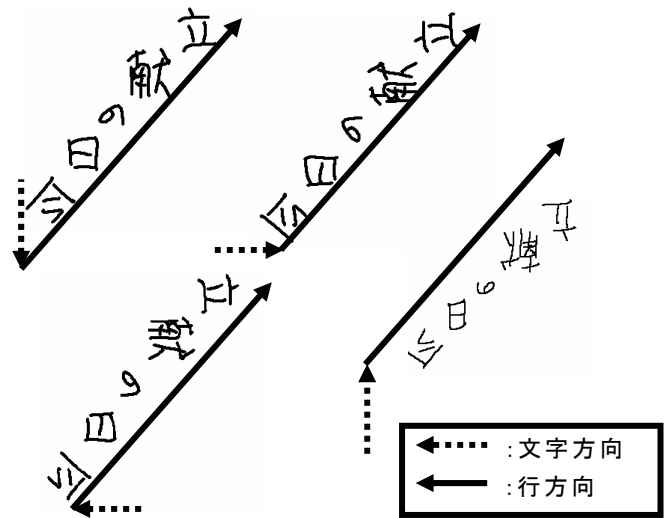


図12. 文字方向仮定処理

3.3. 文字方向・行方向自由枠なし文字列認識

前節の文字方向推定および仮定処理によって推定または仮定された文字方向ごとに、オンライン手書き枠なし文字列認識を図 13 の処理により行う。

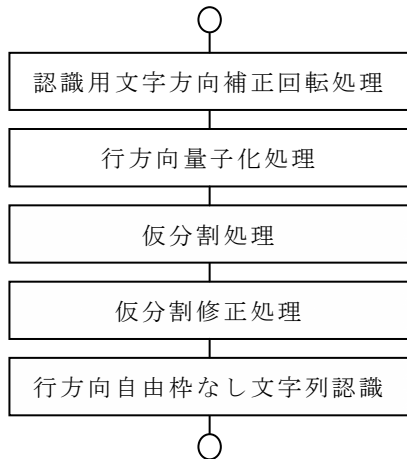


図 13. 処理の流れ(3.3)

(1) 認識用文字方向補正回転処理

文字列構成要素を内部データとしてコピーし、その文字方向が下になるように回転する(図 14)。文字列構成要素は、筆点座標列なので任意角度の回転処理は容易かつ高速に実現できる。それぞれの回転後のパターンを文字方向補正文字列構成要素と呼ぶ。

このとき、後の認識結果整形表示処理で利用するので、文字方向補正文字列構成要素ごとに回転中心、および、回転角を保持しておく。

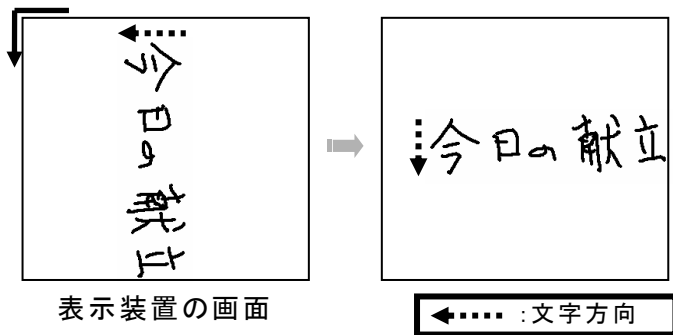


図 14. 回転処理

(2) 行方向量子化処理

算出しておいた行方向に文字方向補正のための回転を適用して、それを改訂された行方向とする。これを一定の量子化区分に分割する。ここでは最低でも4方向(上下左右)に量子化する。本処理のフローチャートを図 15 に示す。

行方向量子化を少なくとも4方向で行うことは、不自然な過分割を防ぐためにも有効である。たとえば今、横書きで、さらに、行方向が右向きの場合、そのなかでの左向きの運筆は文字の区切りでないことが多いためにそれ以前のストロークをまとめるという処理が可能となる。その結果、細かな過分割が少なくなり認識率が向上する。

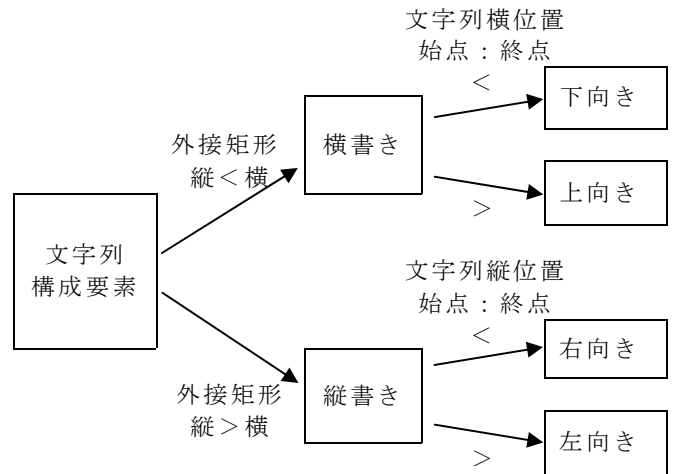


図 15. 行方向量子化処理概要

(3) 仮分割処理

改行や文字列間隔などで区切られる文字列構成要素ごとに、文字サイズ、ストローク間の重心間距離(重心間射影距離)、ストロークの重なり具合などを考慮し、文字としての分割候補の決定やストローク同士の結合などの判定を行う(図 16)。

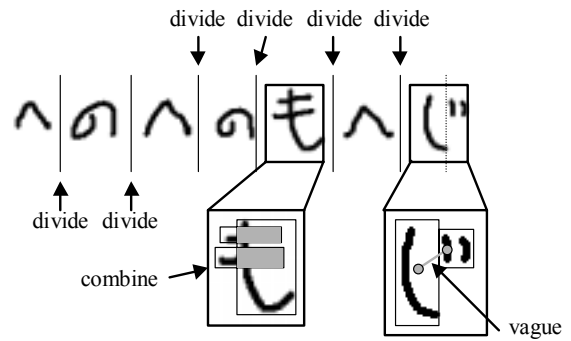


図 16. 仮分割処理

(4) 仮分割修正処理

仮分割処理によって生成された文字候補同士を、文字サイズ、文字要素間の重心間距離(重心間射影距離)、文字要素同士の重なり具合などを考慮し、仮分割によって生成された文字要素をさらに評価し、要素同士の結合を行う(図 17)。

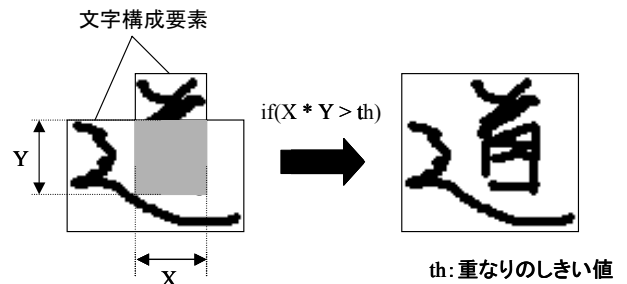


図 17. 仮分割修正

(5) 行方向自由枠なし文字列認識

ここに到達するまえに、文字列構成要素は、文字方向を下向きに補正して、行方向はももとの方向にさらに文字方向の補正を加えた方向を持つことになる、一般にこの方法は任意であるため、ここで適用する枠なし文字列認識は、行方向自由枠なし文字列認識[3]である。この枠なし認識のためには、文字の確からしさ、文字分割の確からしさ、文字接続（文脈）の確からしさ、そして、文字サイズの確からしさを加味して評価する[1].

3.4. 最適文字方向選択処理

認識用文字方向補正回転処理から文字方向・行方向自由枠なし手書き認識処理までは、複数の文字方向補正文字列構成要素に対して適用される。最適文字方向選択処理は、これら複数の文字方向補正文字列構成要素のそれぞれに対する認識処理の結果としての認識スコアのなかで最も高い得点とそのときの文字方向補正文字列構成要素を選択する。

3.5. 認識結果の整形表示処理

本処理は、これまでの認識で得られた結果を利用して、最初の手書き文字列に近い認識結果表示を行う。つまり、文字列構成要素ごとに、その行方向、文字方向、平均文字サイズ、平均文字間隔に従って認識結果文字フォント列を配置して表示する。

4. 認識結果

認識結果の例を図 18 に示す。

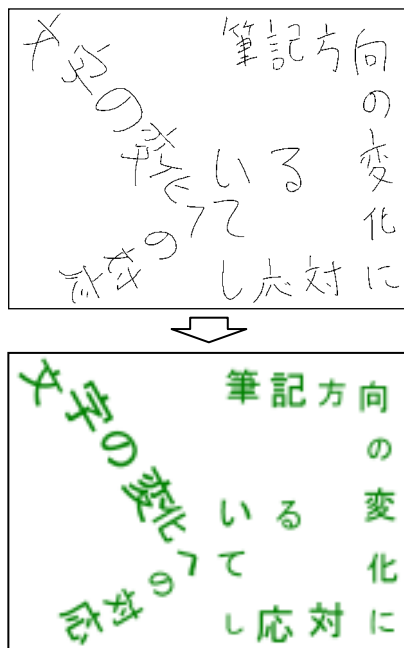


図 18. 認識結果例：“文字の変化への対応
筆記方向の変化に対応している”

5. おわりに

本報告では、筆記制限の緩和を目的に、文字方向および行方向に依存しない文字列認識手法を提案した。本手法では、ペン先座標の時系列情報を利用して文字の方向と行の方向を推定し、その仮定に基づいて、文脈処理を統合した既存の枠なし手書き文字認識手法を適応することにより、文字方向と行方向に依存しない枠なし手書き文字列認識を実現した。現段階では、定性的な評価を行っている。

今後の課題として、以下の項目を挙げる。

- ・ データベースを用いた定量的評価
- ・ 方向変化点検出の高精度化
- ・ 行方向・文字方向推定の高精度化
- ・ 他言語において文字方向の有用性

文 献

- [1] 福島貴弘, 中川正樹, "確率モデルに基づくオンライン枠なし手書き文字列認識," 信学技報, PRMU, vol.98-139, no.489, pp.25-30, 1997.
- [2] 仙田修司, 濱中雅彦, 山田敬嗣, "切り出し・認識・言語の確信度を統合した枠なしオンライン文字列認識手法," 信学技報, PRMU, vol.98-138, no.489, pp.17-24, 1997.
- [3] 稲村祐一, 福島貴弘, 中川正樹, "筆記方向に依存しないオンライン枠なし文字認識システム," 信学技法, PRMU, vol.100-37, no.135, pp.17-24, 2000
- [4] 秋山勝彦, 中川正樹, "オンライン手書き日本語文字認識のための線形処理時間伸縮マッチングアルゴリズム," 電子情報通信学会論文誌, vol.J81-D-□, no.4, pp.651-659, 1998.
- [5] <http://www.anoto.com>.
- [6] <http://www.e-pen.com>.
- [7] <http://www.otmtech.com>.
- [8] S.Jeager, S.Manaka, J.Reichert, and A.Waibel, "Online handwriting Recognition The Npen++ Recognizer," International Journal on Document Analysis and Recognition, 3, pp.169-180, 2001
- [9] Petr Slavik and Venu Govindaraju, "Equivalence of Different Methods for Slant and Skew Corrections in Word Recognition Applications," IEEE Transactions on Pattern analysis and Machine Intelligence Vol.23, No.3, pp.323-326, March 2001.
- [10] 中川正樹, "オンライン手書き文字認識," ヒューマンインタフェース, オーム社, pp.189-193, 1998.
- [11] 秋山勝彦, 中川正樹, "ストロークのつながりに寛容なオンライン手書き文字認識," 画像の認識・理解シンポジウム, MIRU94-□, pp.145-152, 1994.
- [12] M.Nakagawa and K.Akiyama, "A Linear-time Elastic MAtching for Stroke Number Free Recognition of On-line Handwritten Characters," Proc.4th IWFHR, pp.423-430, 1994.
- [13] F. Kimura, M. Shridhar and Z. Chen, "Improvements of a Lexicon Directed Algorithm For Recognition of unconstrained Handwritten Words," International Conference on Document Analysis and Recognition Tsukuba Science City, Japan, pp.18-22, Oct 20 1993.