

## 速さベクトルに基づく自然な空間手書き平仮名認識システムの提案

興水駿<sup>†</sup>      鈴木雅人<sup>†</sup>      北越大輔<sup>†</sup>  
 東京工業高等専門学校情報工学科<sup>†</sup>

### 1 はじめに

近年、直感的・自然な操作法である、いわゆる NUI( Natural User Interface )の重要性が高まっている。その NUI の1つに空間手書き文字がある。

これまでも空間手書き文字認識の研究は盛んに行われており(例えば[1]), 個々の単一文字を高精度に認識する手法は確立されている。しかし、より自然な動作で文字を描けるシステムを実現するためには、文字を描いている部分と拳が移動している部分とを自動判別する方法が必要である。

そこで本研究では、速さベクトルに基づく自然な空間手書き文字認識手法を提案する。Kinect を用いた実験では直感的・自然な動作で文字を描くことが可能で、従来の提案手法よりも高い認識精度を得ることができた。

### 2 提案手法概要

本稿では、Microsoft 社が開発したモーションキャプチャセンサである Kinect を用いた空間手書き平仮名認識の新たな手法を提案する。

より自然な動作で文字を描くためには、文字を描いている部分と拳を移動している部分とを付加情報で明示するのではなく、入力データだけで自動判別する方法が必要である。一般に、日本語を構成する文字は、記号などのように一定の速さで描くことは少なく、「とめ」・「はらい」などを意識して抑揚をつけて描くことが多い。例えば図1は「う」と「ろ」を描いたときの拳の動く速さの違いを示したものである。このように拳が動く速さに着目すれば、拳が移動している部分や描こうとしている文

字を推定できるため、より自然な動作で文字を描くことが可能になると考える。そこで本研究では、速さベクトル(拳を動かす速さ)を特徴として追加することで、自然な動作による空間手書き文字認識手法を実現する。

空間手書き動作から速さ・向きベクトルなどの特徴を取得するために Kinect を使用する。また認識処理は通常のパソコンで行う。なお本研究では、様々な場面で手軽に情報を伝達する手段として本手法を活用することを想定しているため、認識対象は平仮名のみとする。以下で各工程について詳細を説明する。

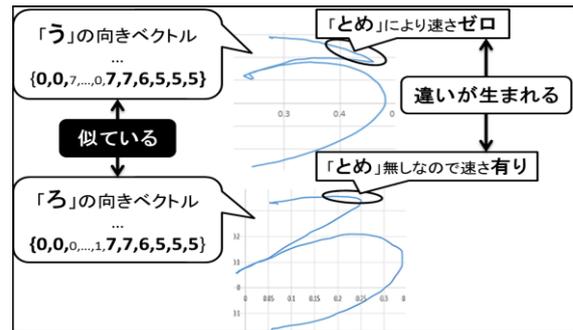


図1 字種「う」「ろ」の特徴の比較

#### 2.1 特徴取得

図2のように Kinect に向かってユーザが文字を描くと、Kinect は 0.1 秒間隔で拳の  $x$  座標、 $y$  座標を時系列で取得し、コンピュータに送る。以下では取得した座標ベクトルを  $(\vec{a}_1, \vec{a}_2, \dots, \vec{a}_n)$  (ただし  $\vec{a}_k = (x_k, y_k)$ ,  $n$  はデータ数,  $1 \leq k \leq n$ ) とおく。

次に式(1)を用いて、座標ベクトルから速さベクトル  $(\vec{v}_1, \vec{v}_2, \dots, \vec{v}_n)$  を求める。



図2 空中手書きの様子

A Study of Natural Space Handwriting Hiragana Recognition System Based on the Speed Vector

<sup>†</sup>Shun Koshimizu <sup>†</sup>Masato Suzuki <sup>†</sup>Daisuke Kitakoshi

<sup>†</sup>National Institute of Technology, Tokyo College.

$$\vec{v}_k = \begin{cases} 0 & (k=1) \\ \frac{\sqrt{(x_k - x_{k-1})^2 + (y_k - y_{k-1})^2}}{0.1} & (k=2,3,\dots,n) \end{cases} \quad (1)$$

また式(2)を用いて、座標ベクトルから角度ベクトル  $(\vec{z}_1, \vec{z}_2, \dots, \vec{z}_{n-1})$  求める。

$$\vec{z}_k = \frac{180}{\pi} \arctan 2 \left( \frac{y_k - y_{k-1}}{x_k - x_{k-1}} \right) \quad (k=2,3,\dots,n) \quad (2)$$

求めた角度ベクトルを図3のように8方向に量子化したものを向きベクトルとする。本研究では速さベクトルと向きベクトルを特徴とする。

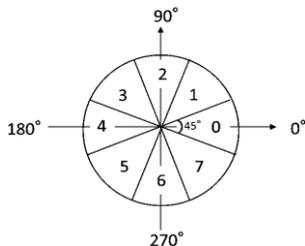


図3 向きベクトルへの変換

## 2.2 認識

認識を行うに先立ち、空間に文字を描くことで辞書データを用意する。辞書データは平仮名の“あ”から“ん”までの46字種であり、1字種ごとに5文字分のサンプル特徴を保存する。この辞書データの特徴と、認識対象文字の特徴とをDPマッチングにより比較し、結果を出力する。DPマッチングは、データ長の異なるパターン間のマッチングを行う手法の一つである。ここでは、辞書データおよび入力データをそれぞれ約500次元に正規化した後、パターン間のユークリッド距離を求める。基本的には、未知入力文字と辞書データに対して向きベクトルのユークリッド距離を計算し、最小距離の字種が認識候補となる。しかし向きベクトルのみでは識別が困難な字種(表1に示す)も存在する。そこでこのような字種が描かれた場合に限り、速さベクトルにも注目する。向きベクトルのみでは識別が難しい字種は「とめ」の有無により速さベクトルに大きな違いが出るため、識別は容易である。

## 3 実験および考察

向きベクトルのみを特徴として用いる従

表1 速さベクトルを使用する字種

対象字種	識別方法
い	認識候補が「い」「り」かつ「とめ」有り
り	認識候補が「い」「り」かつ「とめ」無し
う	認識候補が「う」「ろ」かつ「とめ」有り
ろ	認識候補が「う」「ろ」かつ「とめ」無し

来手法と、速さ・向きベクトルを特徴として用いる提案手法の性能を比較するため、認識実験を行った。それぞれの手法に対して46字種を丁寧に3回ずつ描き、認識精度の比較を行った。

実験の結果、従来の手法の全体の認識率は94.2%となった。それに対して提案手法の全体の認識率は97.8%となり、認識精度は3.6ポイント向上した。精度が3.6ポイント向上した要因は、向きベクトルのみでは認識が困難な表1のような字種が識別できるようになったからであると考えられる。またデータの品質を落とした場合、表1のような字種は向きベクトルがより似通い、識別が難しくなることが予想される。しかし「とめ」は無意識に行うため、速さベクトルまで似通ってしまうことは少ない。よって速さベクトルに注目することで、識別が可能になり、認識率の低下を防ぐことができると考えられる。また誤認識の原因は、特徴データに背景が干渉しているためであり、これはカメラの性能によるものと推察される。

以上の実験によって、速さ・向きベクトルを用いることで、全ての字種を高い精度で認識することが可能となることが分かった。

## 4 まとめ

本稿では、速さベクトルに基づく自然な空間手書き平仮名認識手法を提案し、認識実験によりその性能を評価した。実験の結果、速さ・向きベクトルを用いることで、全ての平仮名を高い精度で認識することが可能となることが分かった。今後は文字品質を落とした場合の効果および改善策について検討する予定である。

## 参考文献

- [1] 西田, 吉田, “Kinect Interaction を利用した空中手書き文字入力の研究”, 福井工業大学研究紀要 第44号, pp.25-30, 2014.