修 士 論 文

隠れマルコフモデルに基づく
オンライン手書き文字列認識に関する研究

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

須藤 隆

2002 年 3 月

修 士 論 文

隠れマルコフモデルに基づく
オンライン手書き文字列認識に関する研究

指導教官　下平 博 助教授

審査委員主査　下平 博 助教授
審査委員　嵯峨山 茂樹 教授
審査委員　阿部 亨 助教授

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

010061 須藤 隆

提出年月: 2002 年 2 月

概 要

本論文では，ストローク HMM に基づくオンライン手書き文字認識手法に，連続音声認識で用いられている１パスビーム探索や統計的言語モデルを用い，筆記領域の切り出しによる文字境界検出を不要とするオンライン文字列認識システムを構築する．

まず初めに，基本性能の向上の為に筆圧情報の特徴量としての新たな利用法を検討し，認識実験により走り書き文字の画数変動に対する頑健性の向上を確認する．

次に，入力画面の小さい携帯端末への実装や視覚障害者の入力装置を想定した重ね書き文字列入力法を提案し，位置情報に依存しない特徴量（速度・方向・筆圧情報）を用いることで，重ね書き文字列認識を実現する．

最後に，文字境界での隣接文字への移動方向に注目し，重ね書きも含めた筆記方向自由文字列に対する検討，認識実験を行い，文字列の筆記方向に依存しない認識システムを実現する．

目 次

図 目 次

表 目 次

第1章　序論

1.1　研究の背景と目的

携帯情報端末等のモバイル機器の小型化により，文字入力インタフェースとしてのオンライン手書き文字認識技術への期待が高まっている．また，手で文字を書くということは日常的に身近な行為であり，高齢者などのキーボード 操作に不慣れな人々のための情報化社会へのアクセシビリティという視点からも認識技術の向上が望まれている．このような活字に変換する手段としての手書きは，紙に文字を書いて筆跡を残す場合とは異なり，非常に素早くメモ書きのような感覚で入力できることが望まれる．その為，字形の崩れや画の連結など，筆跡からでは判読が難しい手書き文字も頻繁に生じる．

このような手書き文字に対して，音声認識とオンライン手書き文字認識の同型性に着目したストローク HMM （Hidden Markov Model）に基づくオンライン手書き文字認識手法 [1, 2, 3, 4] では，基本特徴量に連続した 2 点間の差分である速度ベクトルを用い，文字を書こうとするペンの動きを観測する．絶対座標を用いないので，画の重なりに頑健であり，非目視手書き文字でも認識可能であることが実証された [5, 6, 7]．また，筆記速度が速いなど，前後の画の影響を受けて湾曲した手書き文字についても，環境依存型モデルにより認識率が向上した [8, 9]．ストローク HMM は，僅か 25 種類の HMM によってあらゆる漢字を表現することで，字種毎に異なる HMM で 漢字を表現する手法 [10, 11] と比較して，小規模の辞書による高速な文字認識が可能であり，モバイル環境の文字認識手法として有望である．

しかし，実用化されている手書き文字認識手法の多くと同様に，ストローク HMM に基づくオンライン手書き文字認識手法は，１文字毎に筆記の始端・終端を与えて認識している孤立文字認識法となっており，筆記終端を入力する負担や思考が中断するという不快感を与えている．そこで，任意個の連続した文字を認識するオンライン文字列認識技術が必要となっている．

本研究では，ストローク HMM に基づくオンライン手書き文字認識手法に連続音声認識の手法を応用し，任意個の連続した文字が認識可能なオンライン手書き文字列認識システムを構築することを目的とする．また，入力画面の小さい携帯情報端末への実装を想定した１文字ずつ重ねて書く重ね書き文字列入力法を提案する．従来の多くのオンライン手書き文字列認識手法 [12, 13] では，１文字毎に筆記範囲の高さや幅を求め筆記領域を切り出して文字境界を検出する処理 [13, 14, 15, 16] を必要とし，重ねて書かれた文字に対処できない．また，切り出しを行わずに音声認識の手法を応用する方法 [17] があるが，少数

データによる実験であり，重ね書き文字列に対する認識は見られない．本研究では，文字境界での隣接文字への移動方向に注目し，重ね書きを含め，筆記方向の自由な文字列に対する認識システムを構築する．

しかし，複数文字単位の入力は手軽で素早い入力が可能となる分，字形の崩れや画の連結が多い走り書き文字となる恐れがある．その為，走り書き文字に対する基本性能の向上が必要となる．そこで，筆圧情報を特徴量に併用することで基本性能の向上を行う．また，連続音声認識やオフライン文字認識などでも用いられている統計的言語モデルを加え，高性能なオンライン手書き文字列認識システムを実現する．

1.2 本論文の構成

本論文では、第 2 章にてストローク HMM に基づくオンライン手書き文字認識手法についての説明を行う．第 3 章にて認識実験に用いるために収集した手書き文字列データについて述べる．第 4 章にて孤立手書き文字認識から連続手書き文字認識への拡張について説明し，ストローク HMM に基づくオンライン手書き文字列認識についての説明行う．また，統計的言語モデルの適応，平仮名を含めたオンライン手書き文字列認識システムの構築，重ね書きを含め筆記方向を一般化する文字境界モデルの構築手法について述べる．第 5 章では，筆圧情報特徴量を検討し，基本性能の向上と走り書き文字における画数変動に対する頑健性向上を図る．第 6 章にてオンライン手書き文字列認識システムの性能評価実験を行う．最後に，第 7 章にて結論を述べる。

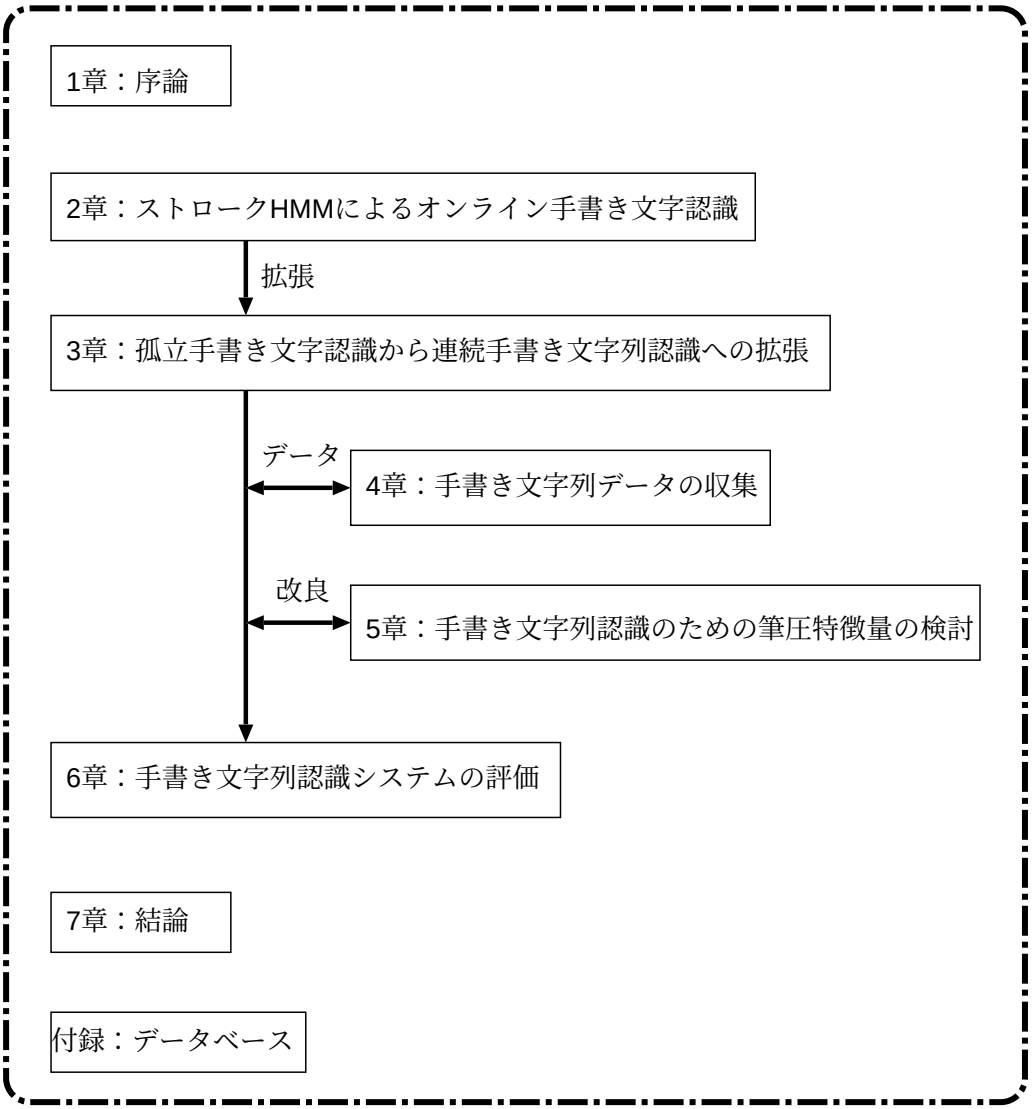

1章：序論
2章：ストロークHMMによるオンライン手書き文字認識
拡張
3章：孤立手書き文字認識から連続手書き文字列認識への拡張
データ
4章：手書き文字列データの収集
改良
5章：手書き文字列認識のための筆圧特徴量の検討
6章：手書き文字列認識システムの評価
7章：結論
付録：データベース

図 1.1: 本論文の構成

第2章　ストロークHMMによるオンライン手書き文字認識

2.1　隠れマルコフモデルを用いたオンライン手書き文字認識

2.1.1　認識

オンライン文字認識においては，認識される筆跡は時間 t ごとに特徴ベクトル $\boldsymbol{O}_t$ に変換され，その結果得られる特徴ベクトル時系列 $\boldsymbol{O} = \boldsymbol{O}_1, \boldsymbol{O}_2, \cdots, \boldsymbol{O}_T$ に基づいて認識が行われる．画，文字，単語，文章などの認識単位に相当する認識カテゴリ $W = \{w_1, w_2, \cdots, w_n\}$ について，観測された特徴ベクトル時系列 $\boldsymbol{O}$ に対する認識語彙 $w \in W$ である確率 $P(w|\boldsymbol{O})$ を計算し，$P(w|\boldsymbol{O})$ が最大となる認識語彙 $\hat{w}$ を求め，認識結果とする．しかし，$P(w|\boldsymbol{O})$ を直接求めるのは通常困難であるので，ベイズ (Bayes) の定理により次式を満たすように推定する．

$$\hat{w} = \arg\max_{w \in W} P(w|\boldsymbol{O}) = \arg\max_{w \in W} \frac{P(\boldsymbol{O}|w)P(w)}{P(\boldsymbol{O})} \tag{2.1}$$

$P(\boldsymbol{O})$ は，特徴ベクトル時系列の事前確率であり，認識語彙 w に依存しないので，無視することができる．$P(w)$ は，認識語彙 w の事前生起確率であり，言語モデルにより与えられる．言語モデルを用いない場合，全ての認識語彙の生起確率を等確率として扱う為に，無視することができる．その為，以下のように式を単純化することが可能となる．

$$\hat{w} = \arg\max_{w \in W} P(w|\boldsymbol{O}) = \arg\max_{w \in W} P(\boldsymbol{O}|w) \tag{2.2}$$

隠れマルコフモデル (Hidden Markov Model: HMM) とは，確率的手法を用いて，非定常信号源を定常信号源の連結で表す定常信号源の切替えモデルである．音声認識の分野で長い間研究されており，音声信号などの非定常信号をモデル化するのに非常に有効な手法である．HMM を用いるオンライン手書き文字認識手法では，得られた特徴ベクトル時系列 $\boldsymbol{O}$ を HMM により定常信号源の連結としてモデル化する．すなわち，文字等の認識語彙 w に対応する HMM を λ^w と定義し，前記特徴ベクトル系列 $\boldsymbol{O}$ が発生する確率 $P(\boldsymbol{O}|\lambda^w)$ が最も高い $\lambda^{\hat{w}}$ 対する認識語彙 $\hat{w}$ を認識結果とする（式 2.3）．

$$\hat{w} = \arg\max_{w \in W} P(w|\boldsymbol{O}) = \arg\max_{w \in W} P(\boldsymbol{O}|w) = \arg\max_{w \in W} P(\boldsymbol{O}|\lambda^w) \tag{2.3}$$

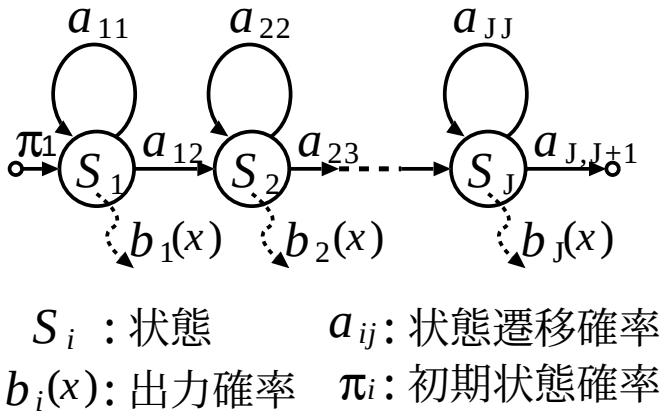


図 2.1: HMM λ^w の例

認識語彙 w の HMM λ^w は，状態 $S_1^w, S_2^w, \cdots, S_J^w, S_{J+1}^w$ で構成される連続分布型 HMM モデルとした場合，以下のパラメータ $\lambda^w = (A^w, B^w, \pi^w)$ によって表される．また，図 2.1 に HMM λ^w の例を挙げる．

$$
\begin{aligned}
A^w &= \{a_{ij}^w\} : \text{状態 } S_i\text{から } S_j\text{に遷移する確率の集合,} \\
B^w &= \{b_i^w(\boldsymbol{O}_t)\} : \text{状態 } S_i\text{が}\boldsymbol{O}_t\text{を出力する確率密度の集合,} \\
\pi^w &= \{\pi_i^w\} : \text{状態 } S_i\text{の初期状態確率の集合}
\end{aligned}
$$

ここで，特徴ベクトル $\boldsymbol{O}_t$ に対する連続分布型 HMM の出力確率 $b_i(\boldsymbol{O}_t)$ は，

$$
b_i(\boldsymbol{O}_t) = \sum_{m=1}^{M} c_{im} \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma}_{im}|}} e^{-\frac{1}{2}(\boldsymbol{O}_t - \boldsymbol{\mu}_{im})^t \boldsymbol{\Sigma}_{im}^{-1} (\boldsymbol{O}_t - \boldsymbol{\mu}_{im})}
$$

で表される n 次元 M 混合正規分布で与える．c_{im} は状態 i の m 番目の分布に対する混合分布重み，$\boldsymbol{\mu}_{im}$ は平均ベクトル，$\boldsymbol{\Sigma}_{im}$ は共分散行列を表す．

時間 t における状態を q_t ，状態系列を $\boldsymbol{q} = q_1, q_2, \cdots, q_{T+1}$ とすれば，HMM λ^w から特徴ベクトル時系列 $\boldsymbol{O}$ が発生する確率 $P(\boldsymbol{O}|\lambda^w)$ は，

$$
P(\boldsymbol{O}|\lambda^w) = \sum_{all\boldsymbol{q}} P(\boldsymbol{O}, \boldsymbol{q}|\lambda^w) \tag{2.4}
$$

$$
P(\boldsymbol{O}, \boldsymbol{q}|\lambda^w) = \pi_{q_1}^w \prod_{t=1}^{T} a_{q_t, q_{t+1}}^w b_{q_t}^w(\boldsymbol{O}_t) \tag{2.5}
$$

となる．但し，本論文で用いる認識システムでは認識速度の高速化の為に，$P(\boldsymbol{O}|\lambda^w)$ を計算せず，$P(\boldsymbol{O}, \boldsymbol{q}|\lambda^w)$ が最大となる認識語彙 w についての最適状態系列 $\hat{\boldsymbol{q}}$ を探索する Viterbi アルゴリズム を用いる．そうした Viterbi 探索による最も尤度の高い字種を認識結果とする．また，実際の尤度計算では，桁落ち防止及び計算高速化の為，対数尤度計算を行う．

2.1.2 学習

HMM のモデル学習では，モデル λ に対して学習資料セットにより与えれた特徴ベクトル時系列 $\boldsymbol{O}$ が発生する確率 $P(\boldsymbol{O}|\lambda)$ を最大にするモデルパラメータの推定を，Viterbi 学習により行う．

2.2 HMMのモデル単位

HMM を用いた時系列パターン認識においては認識単位をどのレベルのモデル単位 k の連結で表現するかという HMM のモデル単位の問題がある．音声認識においても，多くの場合単位モデル単位 k として音素モデルが用いられているが， HMM のモデル単位の問題を扱っている文献 [18, 19] もある．

HMM を用いたオンライン手書き文字認識手法では，認識単位を 1 文字とした上で，モデル単位 k も 1 字種とする研究 [10, 11] がある．すなわち，文字毎に 1 つの HMM モデルを用意する（字種 HMM）．一方で，モデルの単位を 1 字種とするのではなく，線分などのように文字を構成している一部分をモデルとする手法（ストローク HMM [1, 2, 3]）がある．この 2 つの手法ついて，以下で述べる

表 2.1: HMM のモデル単位による分類

	モデル単位	認識単位
字種 HMM	1 字種	1 文字
ストローク HMM	方向線分	1 文字

2.3 字種 HMM (Whole Character Model)

モデル単位 k を文字とした隠れマルコフモデルを用いたオンライン手書き文字認識手法 [10, 11] では，文字毎に 1 つのモデルを用意するため，モデル数，記憶容量が大きくなる問題点がある．例えば，高橋ら [10] の例では，1448 モデルで約 2 MB である．辞書内語彙の増加などに伴い，更に多くの記憶容量が必要となるために，記憶容量の比較的小さい携帯情報端末などに搭載する際には，深刻な問題となる．

また，字種専用の HMM を持つ為に，図 2.2 のような字種の HMM 状態系列を線形に並べたリニアネットワーク [20] により探索を行うことになる．これにより，探索空間が巨大になり，高速な認識が期待できない．これは辞書内語彙が増加する程顕著なものとなる．

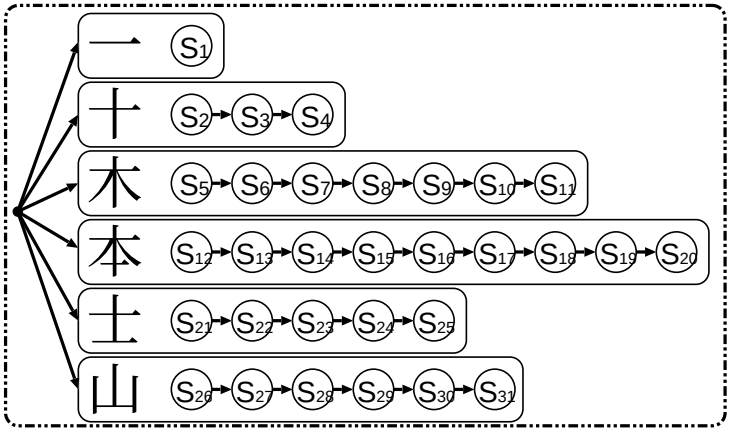


図 2.2: リニアネットワーク

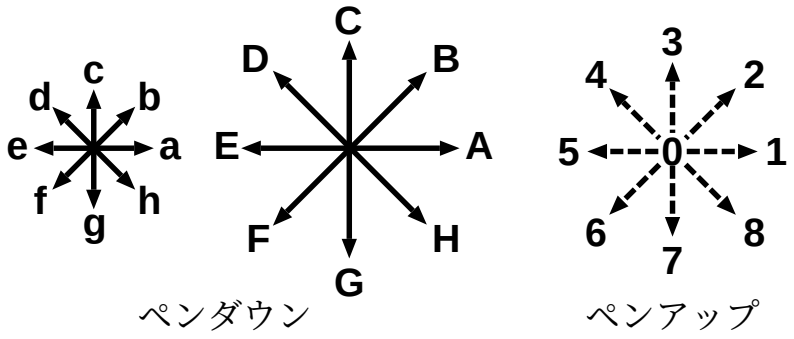


図 2.3: ストロークの種類と対応するモデルのラベル

2.4 ストローク HMM

ストローク HMM に基づくオンライン手書き文字認識手法 [1, 2, 3] では，モデル単位 k を 1 字種とするのではなく，線分程度の小さな単位 (ストローク) にすることで，字種 HMM の問題点に対処する．

図 2.3 に示すように，8 方向の長短 2 種類の線分（A ～ H, a ～ h），ペンアップ時に生じる 8 方向の移動ベクトル（1 ～ 8），移動の生じないペンアップ（0）の計 25 種類のストロークを定義し，それぞれのストロークを連続分布出力型 HMM でモデル化する．ペンダウンは 3 状態の left-to-right モデルとし，ペンアップは 1 状態のモデルとする (図 2.4)．但し，従来のペンアップモデル [1, 3, 4, 5, 6, 7] では自己遷移確率は 0.0 であったが，本論文ではペンアップ区間も等時間サンプリングで観測しているので，自己遷移を付加する．

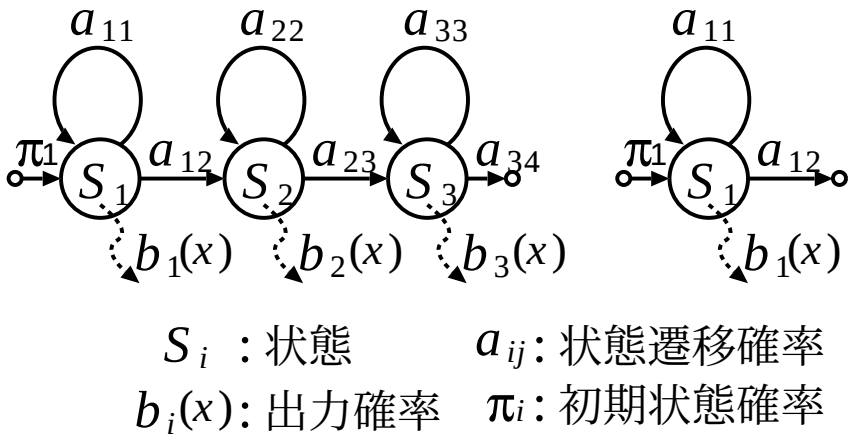


図 2.4: ストローク HMM：(左) ペンダウンモデル，(右) ペンアップモデル

2.4.1 辞書

学習・認識時には，階層的な構造で記述されている辞書 [3] を，最も下の階層である方向線分まで展開し，

「二」 ··· a 6 A
「十」 ··· A 4 G
「子」 ··· A f 0 G d 4 A
「文」 ··· g 5 A 5 F 3 H
「字」 ··· g 5 g 3 A f 6 A f 0 G d 4 A
「田」 ··· G 3 A G 4 G 4 A 6 A
「由」 ··· g 3 A g 4 G 4 A 6 A

のように記述される辞書を用いる．実験では，文献 [4] で用いた辞書に若干の修正を加えたものを用いる (Ver.30)．

第 2.3 節で述べた字種毎にモデルの作成を行う字種 HMM[10, 11] に対するストローク HMM の利点としては，

- モデル数，記憶容量の削減が可能
- 少量データでの効率の良い学習が可能
- 未学習字種でも辞書登録により認識が可能
- 簡単な辞書登録で筆順違いに対応可能
- ネットワーク探索等により高速な認識が可能

などが挙げられる．

ストローク HMM に基づくオンライン手書き文字認識のシステムの全体の流れを図 2.5 に示す．

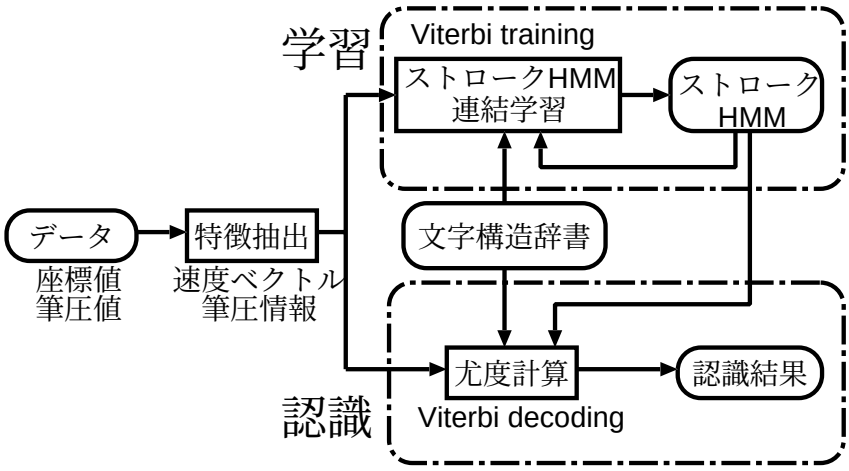


図 2.5: ストローク HMM に基づくオンライン手書き文字認識システム

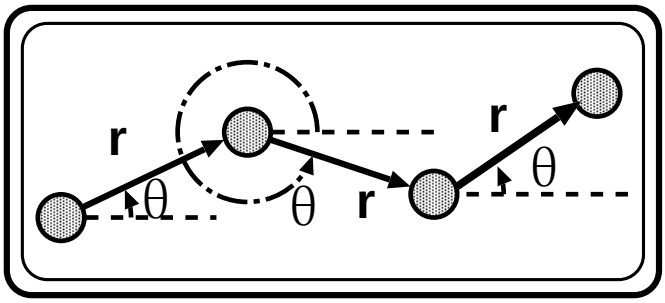


図 2.6: 速度・方向特徴量 (r, θ)

2.4.2 特徴量（速度・方向特徴量の抽出）

入力デバイス（タブレット）から得られる筆跡情報は，一定時間間隔でサンプリングされた座標値 (x_t, y_t)，筆圧値 (z_t)，ペンの上げ下げ情報などの時系列データである．これらをストローク HMM で用いる特徴量に変換する．

ストローク HMM に基づく手法では，異なる位置に筆記されるストローク（線分）を同じモデルとして扱う為に，絶対座標値は用いずに連続した 2 点間の座標差分より，

- 速度: $r_t = \sqrt{(x_t - x_{t-1})^2 + (y_t - y_{t-1})^2}$
- 方向: $\theta_t =$ 水平右方向と $(x_t - x_{t-1}, y_t - y_{t-1})$ の成す角

を用いる [4, 6]．これらは前述したストロークモデルの長短と方向を特徴付ける為の基本特徴量である．但し，特徴量の内の方向特徴量 (θ) に対しては， 2π 周期の連続確率分布となるように平均値操作をする [4, 6].

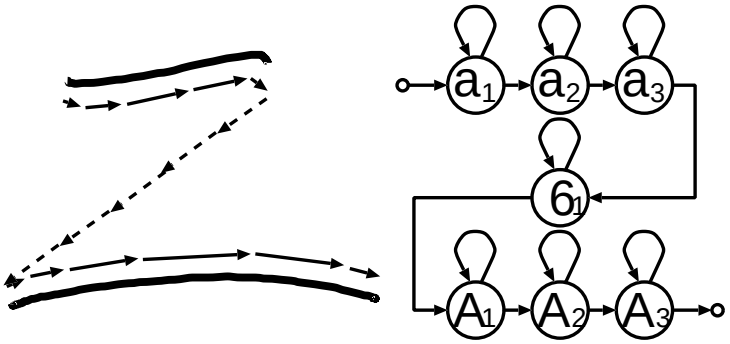


図 2.7: 漢字「二」の HMM

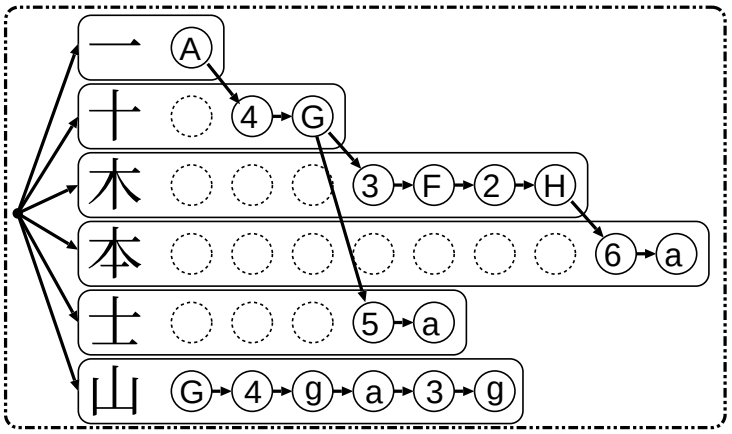


図 2.8: 木構造ネットワーク

2.4.3 認識

認識時には，モデルと辞書により各字種の連結モデルを作成し（図 2.7），観測される特徴ベクトル時系列 $\boldsymbol{O}$ の尤度計算を行う．

より効率的に探索を行う為に，木構造ネットワークによるビーム探索を用いる [6]．木構造ネットワークは，図 2.8 のように，筆記初めのストロークを複数の字種で共有する．前述のリニアネットワーク（図 2.2）に比べて，探索空間が大幅に削減される．これは辞書内語彙が増加する程顕著になる．またビーム探索は，計算対象の HMM 状態を尤度を用いて一定数（ビーム幅）に絞りながら，探索を行うものである．

2.4.4 学習

学習時には，モデル λ に対して学習資料セットにより与えれた特徴ベクトル時系列 $\boldsymbol{O}$ が発生する確率 $P(\boldsymbol{O}|\lambda)$ を最大にするモデルパラメータの推定を，Viterbi 学習により行う．学習データには入力文字の字種のみがラベルとして与えられており，ストローク単位のラベルは与えられていない．そこで，辞書を用いて字種に相当する HMM をストローク

HMM を連結して作成し，連結 Viterbi 学習により，各ストロークモデルの学習を行う.

第3章　手書き文字列データの収集

本章では，収集した手書き文字列データについて，データの特徴，及び認識を行う際の問題点について述べる．

3.1　手書き文字列データの収集

3.1.1　手書き文字列データの収集方法

収集環境には，Linux の X Window System とペンタブレット (Wacom intuos i-400) を使用し，図 3.1，図 3.2 のように，画面上には筆記指示用の見本文字列（横書き）を示し，Gtk+/Gdk で構築したキャンバス上に筆記して，ペンの絶対座標値 (x, y) ，ペンのアップダウン情報，筆圧値 (1,024 レベル)，ペンの傾き (θ_x, θ_y)，時刻を収集した．文字の大きさや筆順は自由とした．

3.1.2　手書き文字列データの筆記方向について

以下の 2 つの種類の文字列データを収集した．

- 筆記方向任意文字列セット（ζ_1 セット）
- 重ね書き文字列データセット（ζ_2 セット）

本論文における筆記方向とは，文字列において前の文字に対する次の文字への移動方向と定義する．すなわち，文字単位での移動方向である．非目視文字 [5, 6, 7] とは異なり，本論文では基本的に画単位での移動方向が崩れない文字を対象としている．

筆記方向任意文字列セット（ζ_1 セット）

データ収集被験者には,「１つ前の文字に重ならないように次の文字を書くこと」と指示した．次の文字を書く方向については特に定めなかったが，見本文字列が横書きの為，横書き文字列が目立った．一部，縦書きの被験者もいる．

図 3.1: 筆記方向任意文字列収集風景

図 3.2: 重ね書き文字列収集風景

図 3.3: 重ね書き文字列データ収集画面

重ね書き文字列データセット（ζ_2 セット）

これは，筆記形態として入力画面の小さい携帯情報端末への文字列入力を想定したものである．本論文では，重ね書き文字列を１文字ずつ上書きして筆記した文字列と定義する．

データ収集被験者には,「１つ前の文字の書き始めの位置あたりから，次の文字を上書きすること」と指示した．しかしながら，筆記する文字列が長い程，視覚的なフィードバックが無くなっていく．その為，図 3.3 のように，上書きしたストローク周辺の過去の筆跡が消えていくように配慮し，筆記負担を軽減した．

3.1.3　手書き文字列データの内容

以下の内容の手書き文字列に対して，前述の２種類のデータを収集した．これら全てについて，60 人の筆者が筆記した．

- （新旧教育漢字）二字熟語 $\cdots$ 343 語
- 短い語句（４文字以上サ変動詞を除く） $\cdots$ 25 語
- 長い語句（７〜８文字） $\cdots$ 95 語
- 挨拶文例語句 $\cdots$ 218 語

第 4.2.2 節で後述するが，本論文での認識実験で用いる辞書内語彙の内訳は，新旧教育漢字 1016 字種，平仮名 71 字種（小文字を除く）の計 1087 字種である．これらの辞書内語彙で構成されている文字列データ 578 語（1,714 文字）を本論文におけるオンライン手書き文字列認識の評価資料とする．

3.2　手書き文字列データの整備

収集した手書き文字列データには，被験者が指示通りに筆記しなかった等の筆記ミスによる異常データが含まれる．本手法の認識性能を評価する上で，こうした筆記ミスによる異常データを誤認識要因から削除する為に，手書き文字列データの整備を行った．

手書き文字列データの整備の基本的な基準は，以下の通りである．但し，筆順違いの見られるデータは除外対象にしていない．

- 指定された文字を書いていない $\cdots$ 不可
- 文字以外の余分な点の付加 $\cdots$ 不可
- 画の過不足 $\cdots$ 可（他の文字と混同しないもののみ）
- 続け字 $\cdots$ 可
- 位置関係がおかしい文字 $\cdots$ 可
- 傾いている文字列 $\cdots$ 可
- 視覚的に認識不可能なもの $\cdots$ 不可

- 略字や旧字 $\cdots$ 不可

これらの基準以外に手書き文字列の各種類毎に以下の通りである．

3.2.1 筆記方向任意文字列データセット（ζ_1 セット）

- 重ね書き文字列になっているもの $\cdots$ 不可

3.2.2 重ね書き文字列データセット（ζ_2 セット）

重ね書き文字列は，視覚的にどのような文字列であるか認識不可能である為，一画ずつストローク単位にチェックした．そのうち「同一文字列内の過去の文字に半分以上文字が重なっているもの」を重ね書き文字列と定義し，明らかに重ねて筆記した意図の見られない文字列を不可とした．

- 重ね書き文字列になっていないもの $\cdots$ 不可

3.3 手書き文字列データの特徴

収集した筆記方向任意文字列の一例を，図 3.4 に示す．筆記方向任意文字列の特徴としては，

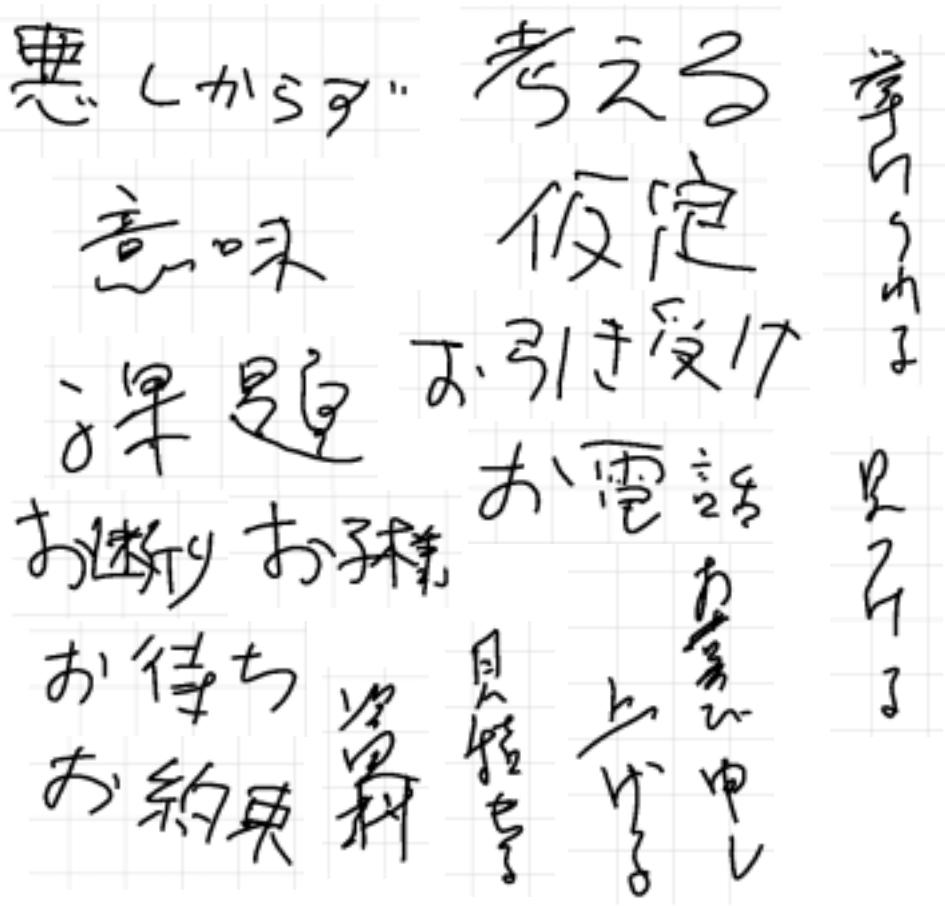
- 筆記方向の個人差（縦書き，横書き）
- 走り書き（画の連結，字形の崩れ）
- 文字列の傾斜（筆記方向の変動）
- 文字間での画の重なり
- 文字間での画の連結
- 筆順違い
- 画の過不足
- 文字以外の余計な点

が挙げられる．被験者の筆記ミスによる異常データについては，第 3.2 節による基準で除去したが，手作業による為，画の過不足や文字以外の余計な点については多少残っている．

筆記方向任意文字列の特徴のうち，字形の崩れはストローク HMM の特徴により対応が可能であり，文字間での画の重なりについても筆記位置に依存しない特徴量を用い文字領域切り出しをしない本手法で対応可能である．しかしながら，画の連結が認識性能低下の原因として挙げられる．また，筆記方向の個人差や変動についてもより一般的な処理をする必要がある．そこで，第 5 章にて，筆圧情報を特徴量に併用することで走り書き文字における画の連結に対する頑健性を向上させる手法について述べる．

また，漢字仮名混じり文字列を認識対象とする為，第 4.2.2 節にて，ストローク HMM の平仮名に対する認識性能について述べ，第 6.1 節にて，認識時の脱字や挿入ミス等に対処する為，統計的言語モデルの作成について述べる．

さらに，第 4.2.5 節にて，文字境界のペンアップモデルに注目し，重ね書きを含め筆記方向に依存しない手法について述べる．



筆記方向任意文字列（ζ1セット）の例

重ね書き文字列（ζ2セット）の例

図 3.4: 文字列データの例

第4章　孤立手書き文字認識から連続手書き文字列認識への拡張

4.1　従来のオンライン手書き文字列認識

オンライン手書き文字認識における入力方式を大別すると，孤立文字入力と文字列入力がある．前者は，文字間の区切りが明示的に与えられる場合を意味し，例えば，予め定められた 2 つ以上の枠の中に文字を順次書いていく方式や，個々の文字の筆記終了情報を筆記者が明示的に与える方式が相当する．一方，後者は，文字間の区切りが明示的に与えられない場合で,「枠無し文字認識」あるいは「文字列認識」が相当する．文字境界を明示的に与える必要のない文字列認識は，筆記者への負担が少なく，思考の邪魔にもなり難いので，孤立文字入力方式よりもユーザインタフェースとして好ましい．しかし，文字列認識は認識システム側で文字列を文字単位に区切る処理（セグメンテーション）が必要となるため，実現が技術的には難しい．

従来提案されてきたのオンライン手書き文字列認識では，１）文字領域切り出し（セグメンテーション），２）個別文字認識，という 2 段階方式が非常に多い [12, 13, 21, 22]．

セグメンテーションの方式としては，文字の連接付近における空間的あるいは時間的な情報を用いた手法が多く，例えば，複数のストローク特徴を利用する方法 [14, 15, 16]，ヒストグラム等により文字サイズを推定する方法 [13, 21] 等がある．前者の方法では，漢字・仮名等の複数のタイプの異なる文字が混在した日本語を扱う場合，高精度の処理が難しい．後者の方法では，隣接文字間での画の重なりに弱い．

認識手法としては，切り出しによる文字境界のみに基づいて個別文字認識を行う方法と複数の文字境界候補を求め総合的な判断をする方法 [12, 13, 21] 等がある．前者の方法では，非常に高精度な切り出しを必要とし，切り出しの精度に認識精度が依存する．後者の方法では，隣接文字間で画の重なった文字や筆記方向が途中で変化した文字等を認識できない．

2 段階方式は，処理が単純で計算量が比較的少なくて済むが，セグメンテーション誤りが文字認識の誤りを引き起こすため，セグメンテーションの精度が全体の性能を大きく左右する．しかし，文字の知識無しには高いセグメンテーション精度を達成するには限界があるため，2 段階方式にはジレンマが存在する．

そこで，セグメンテーションと認識を同時に行って総合的に最適な解（文字列）を求める手法，あるいはセグメンテーション自体を明示的には行わずに最適な文字列を求める手法が有望である．このような最適化の枠組みでオンライン手書き文字列認識を行う手法と

表 4.1: 音声認識とオンライン文字認識の同型性

音声認識		オンライン文字認識	
音素	音素認識	画（ストローク）	
音節		扁旁冠脚	
単語音声	単語認識	一字種	孤立文字認識
文音声	連続音声認識	文字列	文字列認識

して，明示的な切り出しを行わずに DP マッチングを用いて少数語彙（10 数字）の文字列を認識する手法も提案されている [17].

4.2 ストローク HMM に基づくオンライン手書き文字列認識

本論文では，文字切り出し等のセグメンテーションを行わず最適な文字列を求める手法として，ストローク HMM に基づくオンライン手書き文字認識手法をオンライン文字列認識手法へと拡張する．

オンライン文字認識と音声認識とは，表 4.1 のように基本構成要素において同型性が見られる [2]．第 2 章での ストローク HMM に基づくオンライン手書き文字認識手法は，この同型性に着目した手法となっている．本論文ではこの手法を拡張し，ストローク HMM に基づくオンライン手書き文字列認識システムを構築する．すなわち，認識単位を拡張し，連続音声認識との対応を目指しオンライン手書き文字列認識システムを実現する．

連続音声認識では単語境界の切り出しを必要としないのと同様に，ストローク HMM に基づくオンライン手書き文字列認識では文字境界の切り出しが一切不要となる．また，第 2.4.2 節のような筆記位置に依存しない相対的な特徴量を用いることで，文字間の画の重なりに頑健となる．第 4.1 節で述べた従来のオンライン手書き文字列認識に対する本方式の利点としては，

- 切り出しによる高精度な文字境界の検出が不要
- 画の重なった文字列の認識が可能
- 筆記方向を自由に変えた文字列の認識が可能

などが挙げられる．

4.2.1 認識単位

本論文では，認識単位を文字の連結である語句（単語・文節）などの文字列に拡張する．これと区別する為に，認識単位が 1 文字である認識手法を孤立文字認識と呼ぶ．

孤立文字認識手法では，１文字毎に筆記終端を入力する負担やその負担により思考が中断するという不快感を与えている．その為，認識単位を１文字よりも大きい単位にする必要性がある．一方で，認識単位を文章とすると思考の中断は軽減されると考えられるが，筆記入力ミスによる訂正等が大きな問題となる．認識性能の面からも，脱字や挿入ミス等が増えることが予想される．

そこで，これらの中間の認識単位である語句（単語・文節）単位であれば，適切な認識単位であると考えられる．キーボード入力後の漢字仮名変換をする場合を考慮すると，語句単位での入力であれば思考の流れを妨げないタイミングであると予想できる．

4.2.2 辞書内語彙

本論文で構築するオンライン手書き文字列認識システムの認識対象語彙の設定を行う．認識対象語彙はあらかじめストローク列で構成しておき，辞書内語彙とする．

日本語の文字列文章を取り扱う場合，漢字，平仮名，片仮名，英数字，記号等の複数に分離可能なカテゴリが存在する．本論文では，認識対象として漢字と平仮名混じりの文章を取り扱うとし，辞書内語彙を漢字と平仮名に限定する．具体的には，新旧教育漢字 1016 字種，平仮名 71 字種（小文字は除く）の計 1087 字種を辞書内語彙とする．

平仮名ストローク HMM モデルの導入

本論文では，環境依存型モデル [8] を考慮しない 25 モデルによる環境独立型モデルを使用する．その為，漢字によるストローク HMM モデルを用いて平仮名モデルを構築しても，画の湾曲が多い平仮名に対する認識性能が良くないことが想定される．

そこで，漢字によるストローク HMM のペンダウンモデル 16 モデルと別に，平仮名によるペンダウンモデル 16 モデルを併用し，ペンアップモデルについては共通とする平仮名ストローク HMM モデルを導入する．

4.2.3 認識

孤立文字認識と同様に，筆跡情報の特徴ベクトル時系列 $\boldsymbol{O}$ が観測されたとき，文字列 $W = W_1^n = \{W_1 W_2 \cdots W_n\}$ である確率 $P(W|\boldsymbol{O})$ を計算し，$P(W|\boldsymbol{O})$ が最大となる文字列 $\hat{W}$ を求め，認識結果とする．ここで，文字列 $\{W_i, W_{i+1}, \cdots, W_j\}$ を W_i^j と表記する．

同様に，式 (2.1) より，

$$\hat{W} = \arg\max_{W} P(\boldsymbol{O}|W)P(W) \tag{4.1}$$

$P(W)$ は，文字列 W の事前生起確率であり，言語モデルにより与えられる．（言語モデルについては，第 4.3 節で述べる．）

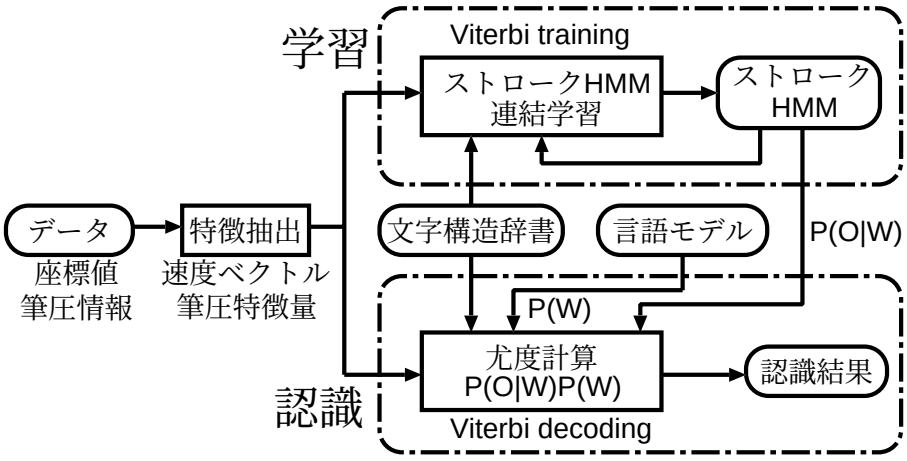


図 4.1: HMM に基づくオンライン文字列認識システム

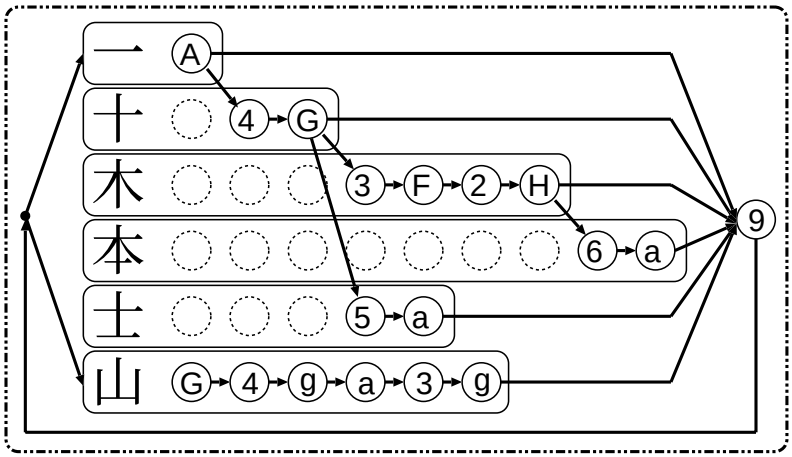


図 4.2: 探索ネットワーク

すなわち，言語モデルを用いたオンライン文字列認識とは，HMM による尤度 $P(\boldsymbol{O}|W)$ と言語モデルによる尤度 $P(W)$ の総積（対数尤度の総和）が最大になるような経路を Viterbi 探索し，その経路に対応する文字列を認識結果とする． HMM に基づくオンライン文字列認識システムの全体の流れを図 4.1 に示す．

探索ネットワーク

まず探索ネットワークに関して，第 2.4.3 節における孤立文字認識の木構造化ネットワークについて拡張する．第 4.2.2 節の辞書内語彙に対して，漢字辞書 (Ver.30) と平仮名辞書 (Ver.4) を使用して探索ネットワークを構成する．図 4.2 のように，文字境界に相当する字種末尾に新たにペンアップモデル（ラベル名 9）を付加し，そこから字種先頭へ戻るループを加える．文字終端に相当する状態にある文字列履歴を認識結果とする．

探索アルゴリズム

次に，探索アルゴリズムについて述べる．探索アルゴリズムについては，孤立文字認識で用いていた Viterbi アルゴリズムの単純な拡張であり，連続音声認識において最も良く用いられる１パスフレーム同期ビームサーチ [23] を用いる．

表 4.2: １パスフレーム同期ビームサーチの内容

探索アルゴリズム	:	Viterbi 探索
入力走査回数	:	１パス
入力走査単位	:	時間（フレーム）同期
仮説展開順序	:	ビームサーチ（枝刈り基準：仮説数）
仮説マージ	:	単語対近似（N-Best 探索）

遅延言語処理

図 4.2 の探索ネットワークにおいて，文字終端に相当する状態に到達し認識した後に，言語モデルによる尤度 $P(W)$ を与える方法を遅延言語処理と言う．HMM による尤度 $P(\boldsymbol{O}|W)$ により文字が認識されてから，言語モデルが駆動される為，言語モデルを１文字遅らせて計算するのと同等である．本システムでは，計算量削減の為に遅延言語処理を用いる．

探索パラメータ

ベイズ (Bayes) の定理に従うと式 (4.1) より，HMM による尤度 $P(\boldsymbol{O}|W)$ と言語モデルによる尤度 $P(W)$ との積を評価値とする．しかし実際には，連続分布型 HMM の確率分布に比べて，言語モデルの確率分布の分散が小さい為，言語モデルによる確率値 $P(W)$ に 1 より大きい重み，言語モデル重み（Language Model Weight）を乗じる方が認識精度が高いことが一般に知られている [23]．また，局所的なマッチングの連続により，低画数の文字による挿入誤りが生じる場合がある．この挿入誤りを回避する為に，文字履歴毎に定数，文字挿入ペナルティ（Insertion Penalty）を尤度に課すことが効果的であることが知られている [23]．また，本システムでは，全ての文字列について，初頭文字の生起確率 $P(w_1)$ は一定と仮定する．

以上から，本システムにおいて言語モデルを適用し，言語モデル重みと文字挿入ペナルティの 2 つの探索パラメータを用いて，評価値が最大になる経路に対応する文字列を探索する過程を以下に定式化する．

特徴ベクトル時系列 $\boldsymbol{O} = \{\boldsymbol{O}_1, \boldsymbol{O}_2, \cdots, \boldsymbol{O}_t\}$ に対する認識結果は，式 (4.1) に対数スケー

ルをとって，

$$\arg\max_{W} P(W|\boldsymbol{O}) = \arg\max_{W\in J^n} \{\log P(\boldsymbol{O}|W) + \log P(W)\} \tag{4.2}$$

と表せる．ここで，n 文字の仮説 $W = w_1 w_2 \cdots w_n$ の評価値を言語モデル重み L_W と文字挿入ペナルティ I_P を用いて表し，認識結果 $\hat{W}$ は以下のように表される．

$$\hat{W} = \arg\max_{W\in J^n} \left[\log P(\boldsymbol{O}|W) + L_W \times \log \left\{ \sum_{i=1}^{n-1} P(w_{i+1}|w_i) \right\} + I_P \times n \right] \tag{4.3}$$

漢字と平仮名の混在文字列認識時において，平仮名は漢字よりもデータ長が短い為，局所的なマッチングを起しやすく，文字挿入ペナルティや言語モデル重みの設定が問題となる．

4.2.4 学習

文字列データを用いずに，字種ラベルのみが与えられた孤立文字データを用い，辞書を用いた連結 Viterbi 学習（第 2.4.4 節）により，各ストロークモデルの学習を行う．音声認識においても，単語データを用いて連結学習した音素 HMM を大語彙連続音声認識に用いている．

4.2.5 文字境界ペンアップモデルと筆記方向

第 3.3 節において収集した手書き文字列データに対し，文字境界のペンアップモデルに注目することで，重ね書きを含めた筆記方向自由文字列に対する認識手法について述べる．

筆記方向とは，文字列において前の文字に対する次の文字への移動方向である文字単位での移動方向であるとしている．

この筆記方向により，オンライン手書き文字列データを表 4.3 のように分類する．この分類を文字列に対応する HMM の状態系列を作成する視点からみると，図 4.2 における文字境界に相当するペンアップモデルを使い分けることに相当する．

表 4.3: オンライン文字列データの筆記方向による分類

文字列の種類	筆記方向	モデル
横書き文字列	右上方向へ	2
縦書き文字列	左下方向へ	7
重ね書き文字列	左上方向へ	4

筆記方向固定文字列認識

筆記される文字列の筆記方向が，固定的であり事前に既知である認識システムを筆記方向固定文字列認識と呼ぶ．この認識システムでは，あらかじめ縦書きなのか，横書きなのか分かっている．従って，筆記方向が固定的な学習データを用いて筆記方向専用の文字境界ペンアップモデルを構築する．重ね書き文字列認識もこの一部であり，同様にして文字境界ペンアップモデルを重ね書き文字列から構築する．

筆記方向自由文字列認識

筆記される文字列の筆記方向が，全く自由である認識システムを筆記方向自由文字列認識と呼ぶ．この認識システムでは，筆記方向が定まっていない分，前者よりもユーザの自由度が高い．例えば，筆記領域の拡大に伴う縦書き・横書き・斜め書き等の混在，また円滑な文字列入力の必要性という視点から，実現されることが望ましい [21]．

筆記方向自由文字列認識を達成する為に，筆記方向が固定的でない学習データを用いて文字境界ペンアップモデルを構築する．つまり，文字境界ペンアップモデルの方向特徴量 θ の分散値を大きくするようにモデル構築をする．

4.2.6 システムの応用

辞書内語彙を漢字・平仮名に加えて，片仮名・記号・英数字と増やすことで，本論文で実現する重ね書きを含めた筆記方向自由文字列認識システムの応用性は益々広がると考える．用途としては，

- 携帯情報端末や携帯電話での手書き連続文字入力インターフェース
- 重ね書きに頑健な電子ノート
- 電子メモ帳
- 手書き電卓
- 視覚障害者のための連続手書き文字入力装置
- 筆記方向自由な電子黒板

などが挙げられる．

4.3 言語モデル

言語モデルとは，与えられた文字列 $w_1^n = w_1 w_2 \cdots w_n$ に対して，その出現確率 $P(w_1 w_2 \cdots w_n)$ を与えるモデルである．言語モデルとしては様々なものが考えられている．サンプルデータから統計的な手法によって確率推定を行う，統計的言語モデルを用いるのが現在の主流となっている．

統計的言語モデルには確率文脈自由文法など様々なものがあるが，その中でも最も単純でかつ最も広く用いられているのが N グラムモデルである．N グラムモデルは，音声認識やオフライン文字認識 [24, 25] の分野でも用いられており，その有効性が示されている．

4.3.1 N グラムモデル

文字列 $w_1^n = w_1 w_2 \cdots w_n$ に対して，その出現確率 $P(w_1^n)$ は，乗法定理を用いると，

$$P(w_1^n) = P(w_1 w_2 \cdots w_n) = P(w_1)P(w_2|w_1)\cdots P(w_n|w_1^{n-1}) \tag{4.4}$$

となる．

N グラムモデルとは，$P(w_1^n)$ の推定をする場合に，

$$P(w_1^n) = P(w_1 w_2 \cdots w_n) = \prod_{i=1}^{N} P(w_i|w_{i-N+1}\cdots w_{i-1}) = \prod_{i=1}^{N} P(w_i|w_{i-N+1}^{i-1}) \tag{4.5}$$

のように，文字の生起を $N-1$ 重マルコフ過程で近似したモデルである．つまり，N グラムモデルでは，i 番目の文字 w_i の出現確率が，直前の $N-1$ 個の文字列 $w_{i-N+1}\cdots w_{i-1}$ だけに依存すると考える．特に，$N=1$ のときをユニグラム (unigram)，$N=2$ のときをバイグラム (bigram)，$N=3$ のときをトライグラム (trigram) と言う．ユニグラムは，文字が以前の文字に依存せずに生起するので，文字の生起確率に等しい．また，全ての文字が等確率で生起すると考えたモデルのことをゼログラムと呼ぶ [26].

4.3.2 N グラム確率の算出

N グラム確率の算出は，基本的には最尤推定法を用いる．すなわち N グラム確率は，学習データ中に出現する文字の N 組と N_1 組の相対頻度から推定する．ここで，文字列 w_1^n が学習データ中に出現する回数を $C(w_1^n)$ で表すと，$P(w_n|w_1^{n-1}) = P(w_n|w_{n-N+1}^{n-1})$ は，

$$P(w_n|w_{n-N+1}^{n-1}) = \frac{C(w_{n-N+1}^{n})}{C(w_{n-N+1}^{n-1})} \tag{4.6}$$

と推定される．

4.3.3 N グラム確率のスムージング

統計元となった学習データにたまたま出現しなかった N グラムに対する出現確率が 0 となってしまう（ゼロ頻出問題）．適切な推定値を得るためには，確率値のスムージング（平滑化）を行う必要がある．

確率値のスムージングとは，大きい確率値を小さく，小さい確率値を大きくすることで確率が 0 になることを回避する手法である．代表的なスムージングとして，加算スムージング，バックオフ・スムージング，線形補間法などがある．本論文では最も単純であり容易に実現できる加算スムージングを用いており，本節ではこれについて説明する．

加算スムージング (Additive Smoothing)

加算スムージングは，N グラム確率の算出において，単純に文字列の出現回数を用いるのではなく，出現回数に一律に一定数を加えた値を用いる．出現回数に加える定数を δ $(0 < \delta \leq 1)$，文字列の異なり総数を V とすると，加算スムージングでは N グラム確率を以下のように推定する．

$$P(w_n|w_{n-N+1}^{n-1}) = \frac{C(w_{n-N+1}^{n-1}) + \delta}{C(w_{n-N+1}^{n}) + \delta V} \tag{4.7}$$

4.3.4 言語モデルの評価

作成した言語モデルの良さは，認識システムにどの程度貢献し，認識精度がどの程度良くなったかという尺度によって測られる．しかし，認識システムの性能には様々な要素が影響する為，認識精度の良し悪しが言語モデルの良さを反映したかどうかを検証するのは難しい．そこで言語モデルの評価を，手軽に使われている尺度であるパープレキシティによって行うことが多い．

パープレキシティ(perplexity)

パープレキシティ PP は，ある文字 1 個が出現しうる確率の相乗平均の逆数で表現される．

$$PP = \left\{ \prod_{i=1}^{n} P(w_i) \right\}^{-\frac{1}{n}} \tag{4.8}$$

実際には，以下のように対数確率の相加平均を取って計算されることが多い．

$$\log_2 PP = -\frac{1}{n} \sum_{i=1}^{n} \log_2 P(w_i) \tag{4.9}$$

テストセット・パープレキシティ(test-set perplexity)

連続音声認識システムでは，認識性能はタスクやテキストなどの処理対象に依存する．すなわち，同じ言語モデルを用いる場合でも，タスクが異なれば，異なった認識性能を

示す．従って，言語モデルの性能評価のためのテキスト集合を別に定めて，そのテキスト集合に対するパープレキシティを調べることが多い．これをテストセット・パープレキシティと言い，式 (4.9) における $w_1 w_2 \cdots w_n$ として，学習に使ったテキストとは別に言語モデルの性能評価のためのテキストを用いて算出したものとなる．

パープレキシティが低いならば，実際に出現する文（テストセット）の出現確率が高く，認識したい文と他の文を識別する能力が高いことを表す．但し，パープレキシティによる言語モデルの性能評価には「文字自体の間違いやすさ」という指標が入っていない為，パープレキシティによる性能評価は認識率に直結しないこともある．

第5章　手書き文字列認識のための筆圧特徴量の検討

5.1　高次元特徴量の検討

第2章で述べたストローク HMM に基づくオンライン文字認識手法の認識性能を更に向上させるための方策としては，入力デバイスから得られるあらゆる情報を用いた高次元特徴量の検討が挙げられる．先行研究 [3, 4, 5, 6, 8, 9] では，第 2.4.2 節で述べた速度・方向特徴量しか用いていない．本論文では，基本性能向上を目指し特徴量の検討を行う．

5.2　筆圧情報の特徴量併用

第 3.3 節において文字列データに見られた画が連結した文字は，筆記画数と辞書パターンとの画数変動がある為，認識性能低下の要因となっている．このような画数変動に対し，画連結パターンを生成する手法 [27, 28]，画数に幅を持たせ探索する手法 [29, 30]，ペンアップ区間に重みをつける手法 [31, 32, 33] などが提案されている．本章では，高次元特徴量の検討の１つとして，画の連結に対する頑健性を高めるという視点で，筆圧情報を特徴量に併用することを検討する．

現在普及している多くのペンタブレットでは筆圧が検知できる．しかし，筆圧情報は個人性を多く含む為，筆者認証 [34, 35] では用いられるものの，手書き文字認識に用いた研究例は少ない．筆圧を用いた研究例としては，字形と筆圧に基づく DP マッチング法 [31, 32] があるが，筆圧時系列を正規化して個人性を除去する処理が必要であり，筆記終了を待たないと認識処理が開始できない．文献 [31] に関しては，認識対象が特定筆者であった．

これに対し，本章では不特定多数筆者のためのオンライン的な筆圧情報の利用法について提案する [36]．すなわち，ある時刻の筆圧特徴量はその前後数十ミリ秒の情報のみで抽出し，オフライン的な正規化処理を必要としない．この為，従来のストローク HMM に基づくオンライン手書き文字認識手法のアルゴリズムを拡張することなく，その特徴量として利用することができる．

5.3 筆圧特徴量

筆圧情報の特徴量化については，以下に述べる筆圧値と筆圧変化量の 2 種類を考え，いずれかを基本特徴量と併用する．

5.3.1 筆圧値

筆圧値 (z_t) をペンのアップダウンを特徴付ける連続量として捉え，(r_t, θ_t, z_t) を特徴量とする．

先行研究 [3, 4, 5, 6, 8, 9] では，ペンがタブレット上で離れている，あるいは接地しているという情報は用いなかった．何故なら，これを用いると確率的に筆記画数の一致しない文字は候補から除外されるからである．この為，ペンアップ時にはペンが離れた点から次にペンが接地する点までの比較的大きな移動ベクトルを観測することによって，ペンのアップダウンを確率的に検出できるようにしていた．しかし，問題点として，移動量の小さいペンアップが検出されにくい事，ペンアップモデルの分散が大きい為に尤度が小さい事が判明した．

そこで，ペンアップ区間の空中の筆跡（タブレットの表面から 8mm 程度離れた空中のペンの軌跡も観測できるデバイスを使用）も等時間サンプリングすることによって，ペンアップモデルの平均と分散をペンダウンモデルと同じくらいに小さくし，代わりに筆圧値 $(0.0 \sim 1.0)$ を用いることによってペンのアップダウンを確率的にモデル化する．

筆圧値は筆者認証に用いられるくらいに個人差の大きい情報ではあるが，ペンアップの状態とペンダウンの状態を識別する上ではあまり筆者依存性が無く，丁寧にリズム良く書かれた文字に対し有効な情報であると考える．

5.3.2 筆圧変化量

筆圧の強弱には個人差があるが，筆圧の増減のパターンは多くの筆者に共通するであろうという考えから，筆圧を第 3 の座標軸とした 3 次元空間内の軌跡としてペンの動きを捉える．すなわち $v_t = (x_t - x_{t-1}, y_t - y_{t-1}, z_t - z_{t-1})$ を観測し，$(r_t, \theta_t, dz_t (= z_t - z_{t-1}))$ を特徴量とする． dz_t を筆圧差分値と呼ぶ．また，ペンダウン時における筆圧値の急変動に対し，より滑らかな筆圧変化量を得る為に，前後 L 点を分析窓幅とする筆圧回帰係数 $\Delta z_t = \sum_{i=-L}^{L} i z_{t+i} / \sum_{i=-L}^{L} i^2$ を dz_t の代わりに用いる事もできる．

この他に $(||v_t||$, X-Y 平面上の角度, X-Y 平面からの仰角$)$ を特徴量とする方法もあるが，例えば $||v_t||$ はモデルのストロークの長短を特徴付けるのに適してなく，新たなモデルと辞書の再定義を要するので，本論文では用いない．

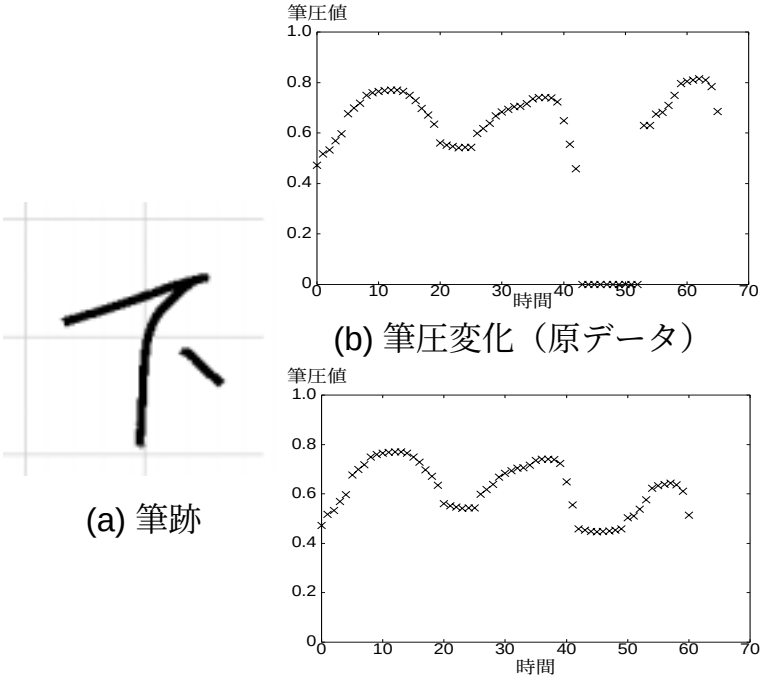


図 5.1: 走り書き文字 “下” の筆圧値の擬似一筆書き補間

5.4 走り書き文字のための速度・方向・筆圧特徴抽出の前処理

5.4.1 部分的な一筆書きによる問題

従来の速度・方向特徴量の場合も，新たに筆圧特徴量を併用した場合も，ペンのアップダウンを確率的に捉えているので，実際の筆記画数が辞書の画数と異なっていても認識候補から除外されることはない．例えば，2 画の “了” を 1 画で書こうが，3 画の “⺁” を 2 画で書こうが，恐らく正しく認識できる．しかしながら，筆圧特徴量はペンのアップダウンを特徴付けるので，本来のペンアップと続け画のペンアップでは尤度が大きく異なり，極端な走り書き文字では筆圧情報が悪影響を及ぼす．このような続け画に対する方法として，

- 本来のペンアップモデルと続け画のペンアップモデルを異なるモデルで表し，マルチテンプレート方式で認識する．
- 本来のペンアップ区間を続け画のペンアップ区間に変形して一つのモデルで吸収する．(その逆は困難である．)

が考えられる．前者の方法では，HMM モデルの学習において，実際に書かれた個々のデータについて，何画目と何画目が続け書きされているかという情報が必要となり，ラベル付けの労力を要する．従って，本論文では後者の方法を用いる．

5.4.2 前処理：擬似一筆書き処理

図 5.1 は “下”（3 画文字）を続け書きした例で 1 ～ 2 画目が続け画であり，連結箇所における筆圧値が，滑らかに変化する傾向がある（図 5.1(b)）．この筆圧変化に類似するように，本来のペンアップ区間の筆圧値を底上げし，擬似的な続け書き文字に変形する（図 5.1(c)）．

また，ペンアップしている状態での筆跡 (x_t, y_t) は筆が迷っている事もあり得るので，空中で実際に観測されている軌跡は用いずに，補間によって擬似的な軌跡を生成する．尚，本論文ではストローク環境独立型の HMM を用いているので，前後のストロークの筆跡に依存せずに，線形に補間する．

［アルゴリズム］

観測された時系列を (x_t, y_t, z_t)，補間変形後の時系列を (x'_t, y'_t, z'_t) とし，先行画の終点時刻を t_1，後続画の始点時刻を t_2 とする．

1. ペンアップ区間の前後の筆圧値を等しくし，筆圧変化を 0.0 にする．すなわち，$z'_{t_2} = z_{t_1}$ とし，時刻 t_2 以降の筆圧値には $z_{t_1} - z_{t_2}$ を加算する．
2. ペンアップ区間の補間
   (a) 筆圧 (z'_t) $\cdots$ ペンアップ区間の前後の筆圧値 z'_{t_1-1}, z'_{t_1}, z'_{t_2}, z'_{t_2+1} を用いて $(t_2 - t_1)/(S+1)$ 間隔の等時間サンプリングとなる S 点で 3 次スプライン補間をする．
   (b) X-Y 座標 (x'_t, y'_t) $\cdots$ ペンアップ区間の前後の座標値 (x_{t_1}, y_{t_1}), (x_{t_2}, y_{t_2}) より，この区間を等速・等方向に移動するように $(x_{t_2} - x_{t_1}, y_{t_2} - y_{t_2})/(S+1)$ 間隔で補間する．

この補間アルゴリズムは，ペンアップ区間に関しては次のペンダウンが観測されるまで特徴量系列が得られないが，ペンダウン区間に関しては時間遅れが無く，オンライン的な特徴量であると言える．図 5.1(c) は $S = 6$ でペンアップ区間の筆圧値を補間した例である．

5.5 オンライン手書き文字認識実験

5.5.1 実験 1: 丁寧な手書き文字による筆圧特徴量の評価

従来の速度・方向特徴量 (r, θ) と筆圧値，筆圧差分値，筆圧回帰係数を併用した特徴量 (r, θ, z)，(r, θ, dz)，$(r, \theta, \Delta z)$ との認識性能の比較を行った．また，ペンアップ区間で実際に観測される速度・方向特徴量を使用した場合と従来通りに画間の移動ベクトルのみを使用した場合の比較も行った．筆圧回帰係数 (Δx) を求める為の分析窓幅は $L = 2$ とした．

表 5.1: 丁寧な手書き文字に対する特徴量別認識率（%）

特徴量	N 位累積認識率 [%]				
	1 位	2 位	3 位	5 位	10 位
ペンアップ区間の軌跡を利用した場合					
r, θ	93.59	96.88	97.87	98.45	98.99
r, θ, z	**97.62**	**99.41**	**99.67**	**99.75**	**99.82**
r, θ, dz	93.13	96.61	97.60	98.42	99.00
$r, \theta, \Delta z$	95.67	98.23	98.88	99.26	99.55
ペンアップ区間の軌跡を利用しない場合					
r, θ	96.90	98.92	99.47	99.69	99.81
r, θ, z	**97.24**	**99.27**	**99.62**	**99.76**	**99.83**
r, θ, dz	96.77	98.87	99.41	99.63	99.80
$r, \theta, \Delta z$	96.80	98.96	99.47	99.66	99.83

実験条件

データベース	:	丁寧な手書き文字セット（γ_2）
辞書内語彙	:	新旧教育漢字 1,016 字種
学習資料	:	奇数番目の 30 筆者
評価資料	:	偶数番目の 30 筆者
モデル	:	全共分散型 2 混合正規分布

ペンの軌跡を観測しない場合，すなわち，画間の移動ベクトルのみを用いる場合は 1 フレームしか特徴量系列が観測されないが，ペンアップ中のペンの軌跡を特徴量として用いる場合と同様に自己遷移確率があるモデルとして学習した．

実験結果

表 5.1 に特徴量別の認識率を示す．全般に筆圧情報を併用した方が認識率が高くなった．また，筆圧値 (z) を併用した場合に限り，ペンアップ区間の軌跡を用いた方が認識率が高いという結果が得られた．

表 5.2 に従来特徴量と比較して筆圧値 (z) 及びペンアップ区間の軌跡を用いたことにより改善された例と誤認識に転じた例を上位 6 字種の例を示す．表中の “(有)” はペンアップの軌跡を用いた場合，“(無)” は用いない場合を意味する．誤認識に転じた例のうち右肩に＊の付いている字種は辞書におけるペンアップ方向のラベルが間違っていたものである．例えば “布” は “G f 3 A 2 (正しくは 6) g 3 A g d 4 G” のように定義されていた為に，ペンアップ系列の類似した “市” に誤認識した．他の 3 字種も同様であり，これらは

表 5.2: 筆圧特徴量併用により改善された例・誤認識に転じた例（上位 6 字種，評価資料は 30 文字/字種）

改善された例

字種	正解認識数		誤認識例 (r, θ)
	r, θ (無)	r, θ, z (有)	
力	5	29	才，刀
考	6	20	老
必	16	29	州，冷，冬
売	12	24	壱
圧	14	26	正，左
五	17	28	正

誤認識に転じた例

字種	正解認識数		誤認識例 (r, θ, z)
	r, θ (無)	r, θ, z (有)	
布*	24	3	市，幼，年
希*	30	11	命，谷
有*	27	14	角
天	11	6	夫
失	21	16	矢
座*	20	15	産

辞書の修正で解決するので，実質的に認識率が低下した字種はかなり少なくなる．

このように従来特徴量に比べて，良くも悪くもペンアップの違いが大きく認識結果に影響した．これは文字全体の尤度に対し，ペンアップ区間のサンプル点による尤度の占める割合が大きくなっていることで説明できる．すなわち，表 5.3 に示すように，様々な移動ベクトルを観測して分散の大きかった従来特徴量に比べて，ペンアップの軌跡を用いた方が分散が小さい事，また複数フレーム観測している事に起因している．従って，丁寧に書かれた文字に対しては有効な手法ではあるが，既報 [5, 7] の非目視手書き文字のように画間のペンアップ方向が崩れる場合には不適となる恐れがある．

尚，筆圧情報なしにペンアップ区間の軌跡を用いると認識率は低く，誤認識例としては “片” → “用”，“天” → “尺” などが見られた．

筆圧回帰係数特徴量 (Δz) は窓幅の設定という問題があるものの，筆圧差分値 (dz) に比べて良好な結果が得られた．

表 5.3: ペンアップモデル 6 の速度特徴量 (r) の混合正規分布パラメータ

特徴量	混合重み	平均	分散
r, θ (無)	0.53	88.54	1743.35
	0.47	56.15	571.29
r, θ, z (有)	0.58	7.43	20.45
	0.42	3.16	2.53

5.5.2 実験 2: 筆圧特徴抽出の前処理の評価

実験条件

データベース	:	走り書き文字セット（ϵ_1）中の 筆順の正しいサブセット
辞書内語彙	:	新旧教育漢字 1,016 字種
学習資料	:	奇数番目の 34 筆者
評価資料	:	偶数番目の 34 筆者
モデル	:	全共分散型 2 混合正規分布

3 節に述べた走り書き文字における前処理の有用性について検証した．

続け画の筆圧値 (z) の最大値・最小値には個人差があるので，筆圧特徴量には筆圧変化量 (dz)，(Δz) を用いた．ペンアップ区間の補間点数 S については $1 \sim 8$ まで変化させ，認識率の変化を調べた．

実験結果

認識結果について，図 5.2 にペンアップ区間の補間点数 S による 1 位認識率の変化を示す．このうちの筆圧回帰係数特徴量 (Δz) を併用した場合の認識率が最大となる補間点数 $S = 6$ を基準に，筆圧情報を用いたことにより改善された文字例を図 5.3 に，誤認識に転じた文字例を図 5.4 に示す．また，筆者別の認識率を図 5.5 に，文字の続け度別の認識率を図 5.6 に示す．

ペンアップ区間の補間点数について（図 5.2）

丁寧な手書き文字ではペンアップ区間の軌跡を用いない場合でも，筆圧特徴量による認識率の向上が見られたが，走り書き文字では $S \leq 2$ の場合は筆圧特徴量を用いない方が良いという結果となった．前処理において補間点数が少ない為に，ペンアップ前後で比較的大きな筆圧変化量が観測された為と考えられる．

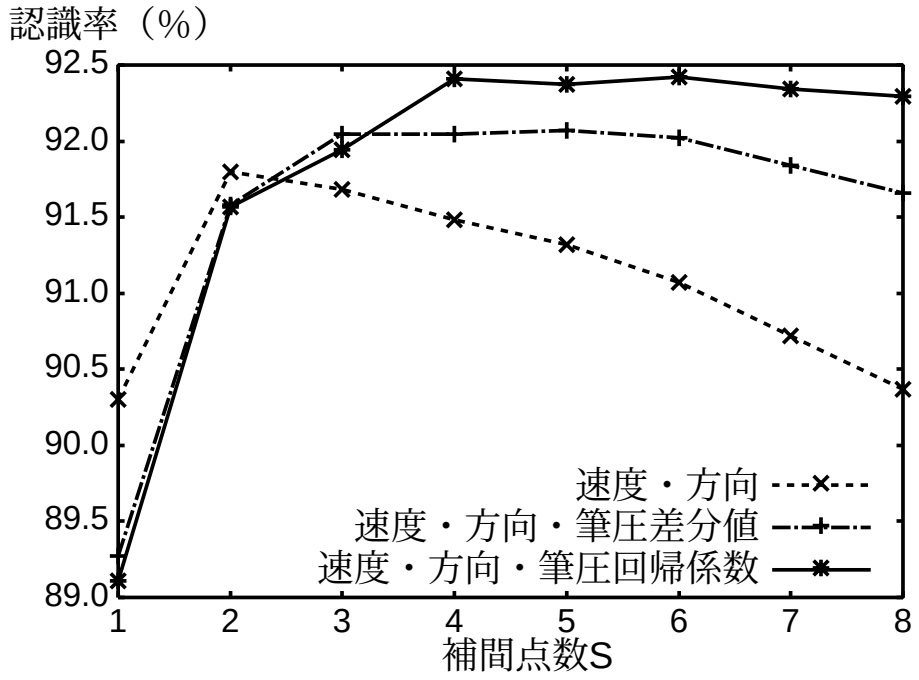


図 5.2: 走り書き文字（筆順の正しいサブセット）に対する補間点数別認識率

一筆で書かれた文字のみを利用し HMM 統計モデルを作成した結果，ペンアップモデルの自己遷移確率は $0.75 \sim 0.85$ となり，続け画のペンアップ区間の観測点数は期待値として約 $3 \sim 6$ となった．筆圧差分値特徴量 (dz) を併用した場合は $S = 5$ の時に，筆圧回帰係数 (Δz) を併用した場合は $S = 6$ の時に認識率が最大となり，続け画のペンアップ区間の観測点数にほぼ一致した．

参考までに，ペンアップ区間の前処理を行わずに，観測されるペンの軌跡をそのまま認識に用いた場合は，筆圧差分値併用時は 89.63 %，筆圧回帰係数併用時は 90.55 %と前処理を行ったときの最大認識率よりも認識率が低下した．これは筆の迷いと，ペンアップダウン時の急激な筆圧変化が原因である．

誤認識文字について（図 5.3，図 5.4）

図中の「在→左」は速度・方向特徴量では “在” に認識されていたものが，筆圧特徴量を併用することによって “左” に認識されるようになったことを意味する．筆圧特徴量を併用し改善された文字数は 1,039 文字，誤認識に転じた文字数は 96 文字であった．改善された理由としては，文字全体の尤度に占めるペンアップ区間の尤度が大きくなったことが挙げられる．また，誤認識に転じた原因としては，前述の辞書誤定義の他に，続け画の筆圧変化と擬似一筆書き処理をした筆圧変化とが合わなかったことが挙げられる．

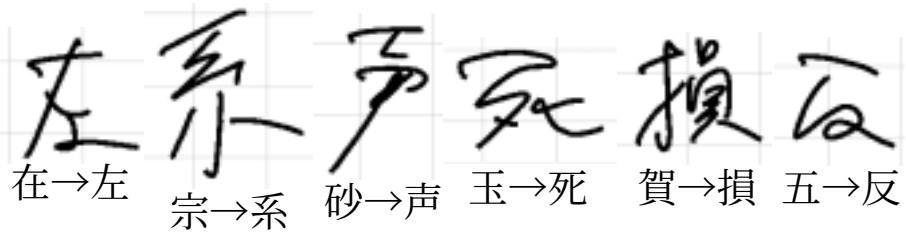


図 5.3: 走り書き文字認識に筆圧情報を用いることにより改善された例

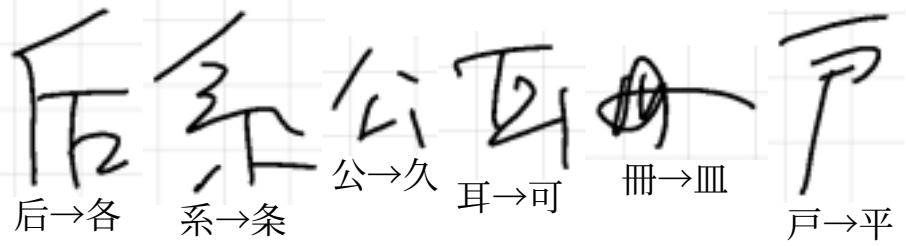


図 5.4: 走り書き文字認識に筆圧情報を用いることにより誤認識に転じた例

筆者別の認識性能について（図 5.5）

特徴量として筆圧回帰係数（$S = 6$）を用いた場合，認識率の向上した筆者は 25 名，逆に認識率の低下した筆者は 8 名であり，比較的大勢の筆者に対して有効であるという結果が得られた．

ストローク HMM による 3 状態の HMM により，ストロークの開始から終端への筆跡がよく表現できたと考える．すなわち，速度：(低速）—（高速）—（低速)，方向：(分散大)–(分散小)–(分散大)，筆圧変化量：(増加)–(一定)–(減少)，とそれぞれの特徴量変化が 3 状態の HMM と非常に相性が良かったと考える [6]．その為，ペンアップ時の筆圧変化量を画境界と捉えつつ，ペンダウン時の筆圧変化量に含まれる個人的な変動が，HMM の統計モデルで吸収されていると考える．こうした上で，筆圧の増減のパターンは筆者依存性が無いと言える．

一般に筆圧値変動が起きやすいのは，弱い筆圧かつ遅い筆記速度で筆記するペンダウン時である．走り書き文字では，素早く筆記する為に，ペンダウン時における急な筆圧値変動は見られにくい傾向がある．その為，走り書き文字に対しては筆圧差分値と筆圧値変動に頑健な筆圧回帰係数の間にはそれ程差はなく見えるが，筆圧回帰係数は，窓幅の最適化の問題があるにせよ，不特定筆者の手書き文字認識において有効な特徴量であると言える．

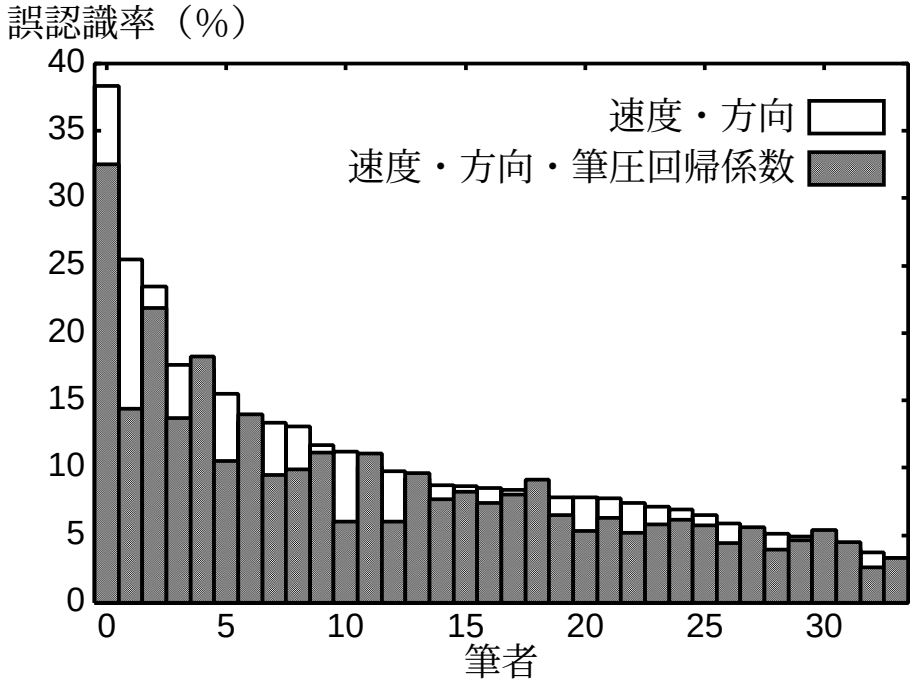


図 5.5: 筆者別走り書き文字認識率

続け字の対処について（図 5.6）

図 5.6 は，辞書画数 N，筆記画数 n であるとき文字のペンアップ数当りの続け画数を表す量として，続け度を $\frac{N-n}{N-1}$ （但し $N=1$ のときは 0）と定義し，これを基準として 8 段階のクラスに分類した時の各々の認識率を示したものである．括弧内は各クラスに相当する続け度の範囲である．クラス 0 は続け画の無い文字を表し，クラス 7 は一筆書きされた文字を表す．特徴量は筆圧回帰係数 (Δz) を併用し，ペンアップ区間の軌跡を利用した場合と，利用しない場合の補間点数 $S=6$ で前処理した場合を示す．疑似一筆書き処理の有効性が見られた．

丁寧な手書き文字を用いた補足実験

丁寧な手書き文字（γ_2 セット）認識における特徴抽出の前処理が及ぼす影響について，補足実験を行った．各々の特徴量について用いた補間点数 S は走り書き文字データセットにおいて最大の認識率を与えた値を用いた．その他の実験条件は実験 1 と同じにした．表 5.4 より，ペンアップ区間の補間処理による悪影響は無く，走り書き文字と同様に認識率の向上が確認できた．

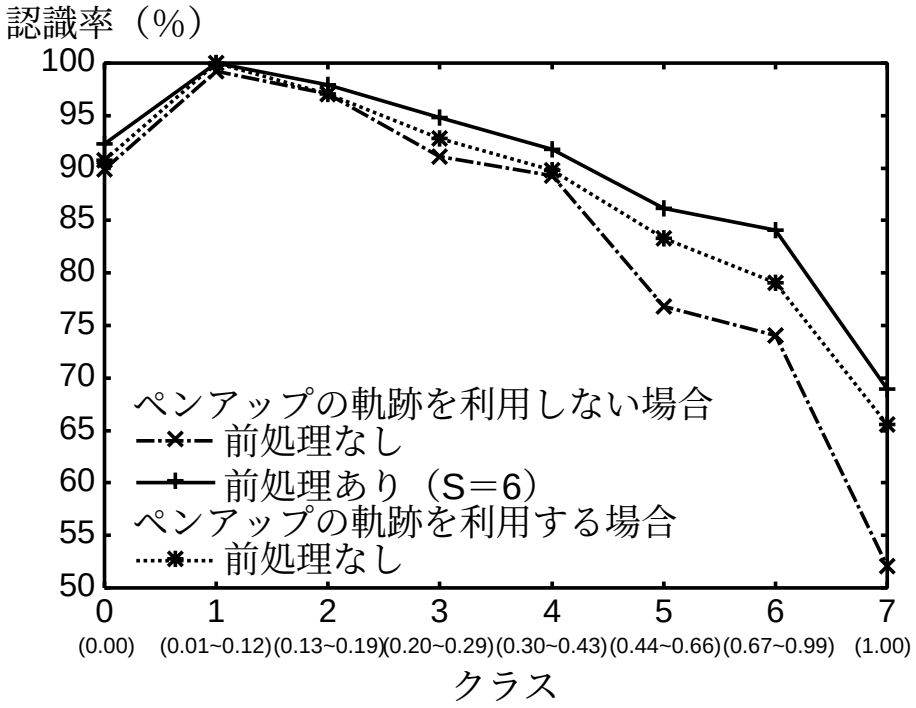


図 5.6: 文字の続け度別認識率

表 5.4: 丁寧な手書き文字に対する特徴抽出の前処理の有無による認識率（%）

特徴量 (補間点数 S)	N 位累積認識率 [%]				
	1 位	2 位	3 位	5 位	10 位
r, θ	96.90	98.92	99.47	99.69	99.81
r, θ, dz (5)	**97.77**	**99.42**	**99.71**	**99.83**	**99.91**
$r, \theta, \Delta z$ (6)	97.42	99.34	99.69	**99.83**	**99.91**

表 5.5: 字種 HMM 手書き文字認識方式における特徴量別認識率（%）

特徴量	N 位累積認識率 [%]				
	1 位	2 位	3 位	5 位	10 位
r, θ	98.58	99.60	99.76	99.83	99.90
r, θ, dz	98.85	99.70	**99.85**	**99.91**	**99.94**
$r, \theta, \Delta z$	**98.98**	**99.74**	99.83	99.89	**99.94**

5.5.3 実験 3: 字種 HMM 手書き文字認識方式における筆圧特徴量の評価

ストローク HMM を用いる手法は，辞書定義の正誤により，認識性能が大きく変わる．そこで，字種毎にモデルを用意する字種 HMM 文字認識方式 [10, 11] において，特徴量の比較を行った．

実験条件

データベース	:	丁寧な手書き文字セット（γ_2）
辞書内語彙	:	新旧教育漢字 1,016 字種
学習資料	:	奇数番目の 30 筆者
評価資料	:	偶数番目の 30 筆者
モデル	:	全共分散型 1 混合正規分布

字種 HMM の学習に用いる初期モデルは，ストローク HMM を連結したものを与えた．従って，学習されたモデルは，どの字種の何画目であるかを考慮した極端な環境依存モデルとみなすこともできる．尚，ペンアップ区間の補間点数は，実験 2 で最大認識率を与えた値（dz の場合は $S = 5$，Δz の場合は $S = 6$）を用いた．

実験結果

結果を表 5.5 に示す．特徴量に筆圧差分値 (dz) を併用，あるいは筆圧回帰係数 (Δz) を併用 した方が認識率が高いことから，字種 HMM においても筆圧情報が有効であることが分かった．

5.6 まとめ

不特定筆者の手書き文字認識におけるオンライン的な筆圧特徴量の利用法を提案し，その有効性を示した．また，部分的な続け書き文字に対処した特徴量の前処理（一筆書き変形）手法を提案し，走り書き文字と丁寧な手書き文字の両方において認識率の向上を確認した．

また，走り書き文字に対しては，前後ストロークの環境依存型モデル [8] との組み合わせにより，更なる認識率の向上が期待できる．

第6章　手書き文字列認識システムの評価

6.1　統計的言語モデルの作成

毎日新聞社の新聞記事データ「ＣＤ- 毎日新聞 97 年版」（12ヶ月分）を学習プレーンテキストとして，統計的言語モデル（バイグラムモデル）を作成した．具体的には，第 4.2.2 節での辞書内語彙（新旧教育漢字 1016 字種，平仮名 71 字種）の 1087 字種について，前述した加算スムージング（$\delta = 0.01$）を用いて文字間 bigram 確率を算出した．

6.1.1　文字間 bigram 確率の算出

前述のバイグラムモデルを用いる為，n 文字からなる文字列 $w_1^n = w_1 w_2 \cdots w_n$ が与えられたとき，文字列 w_1^n の生成確率は，式 (4.5) を用いて，

$$P(w_1^n) = \prod_{i=1}^{n} P(w_i|w_{i-1}) \tag{6.1}$$

とする手法もある．但し，$i = 1$ の場合，$P(w_1|w_0)$ となるが，w_0 は文頭を表す特別な記号 $\langle s \rangle$ であるとする．本システムでは，文頭記号 $\langle s \rangle$ を一切考慮せずに，式 (4.4) を用いて，

$$\begin{aligned} P(w) = P(w_1 w_2 \cdots w_n) &= P(w_1)P(w_2|w_1)P(w_3|w_2)\cdots P(w_n|w_{n-1}) && (6.2) \\ &= P(w_1)\prod_{i=1}^{n-1} P(w_{i+1}|w_i) && (6.3) \end{aligned}$$

とした．また，前述した加算スムージング式 (4.7) より，

$$P(w_{i+1}|w_i) = \frac{C(w_{i+1}) + 0.01}{C(w_i) + 0.01 \times V}$$

とした．

6.1.2　言語モデルの評価

作成したバイグラムモデルに対して，第 4.3.4 節で述べたパープレキシティによる評価を行う．

文字間バイグラムモデルに対してパープレキシティは，式 (4.9) より，

$$\log_2 PP = -\frac{1}{N}\sum_{i=1}^{N}\log_2 P(w_{i2}|w_{i1}) \tag{6.4}$$

となる．

第 3.1.3 節で述べた本論文の認識実験で用いる辞書内語彙 1087 文字で構成される語句 578 語（1,714 文字）をテストセットとみなし，バイグラムモデルを用いてテストセット・パープレキシティを計算し，以下の結果を得た．

- $\log_2 PP = 7.067$
- テストセット・パープレキシティ $PP = 134.07$

また，認識文字列中に未知な文字が存在しない設定になっているので，カバー率（カバレージ）は 100% である．

6.2 予備実験１：ストローク HMM を用いた平仮名認識実験

ストローク HMM に基づくオンライン手書き文字認識における先行研究 [1, 2, 3, 4, 6, 7, 8, 9, 36] では，漢字以外のカテゴリへの認識性能評価が行われていない．漢字平仮名混じりの手書き文字列認識にあたり，平仮名に対する認識性能評価を行う必要がある．そこで，ストローク HMM に基づくオンライン手書き文字認識における平仮名に対する孤立文字認識の性能評価を行う．

6.2.1 平仮名ストロークモデルによる平仮名認識実験

平仮名のみに対してストローク HMM モデルを用いることで，ストローク HMM に基づくオンライン手書き文字認識における平仮名に対する性能評価を行う．また 5 章で述べた，筆圧差分値特徴量を併用した場合の性能評価も行う．

実験条件

データベース	:	丁寧な手書き文字セット（γ_2）
辞書内語彙	:	平仮名 71 字種
学習資料	:	平仮名データ（奇数番目の 30 筆者）
評価資料	:	平仮名データ（偶数番目の 30 筆者）
モデル	:	全共分散型 2 混合正規分布

学習資料は平仮名データのみを使用し，平仮名ストローク HMM モデルを構築した．また，筆圧差分値特徴量 dz を併用したときのペンアップ区間の補間点数は，第 5.5 節の実験 2 で最大認識率を与えた値 $S = 5$ を用いた．

表 6.1: ストローク HMM を用いた平仮名認識における特徴量別認識率（%）

特徴量	N 位累積認識率 [%]				
	1 位	2 位	3 位	5 位	10 位
r, θ	88.45	95.68	97.42	98.50	99.20
r, θ, dz	**90.85**	**97.32**	**98.12**	**98.83**	**99.62**

実験結果

結果を表 6.1 に示す．特徴量に筆圧差分値 (dz) を併用した方が認識率が高いことから，平仮名に対しても筆圧情報が有効であることが分かった．

6.2.2 漢字平仮名併用ストロークモデルによる平仮名認識実験

第 4.2.2 節で述べた，漢字と平仮名のそれぞれのペンダウンモデルを持つストローク HMM モデルを用いた場合，平仮名と漢字間で誤認識について評価した．

実験条件

データベース	:	丁寧な手書き文字セット（γ_2）
辞書内語彙	:	新旧教育漢字 1,016 字種 ・ 平仮名 71 字種
学習資料	:	漢字データ・平仮名データ（奇数番目の 30 筆者）
評価資料	:	平仮名データ（偶数番目の 30 筆者）
モデル	:	全共分散型 2 混合正規分布

学習資料には，新旧教育漢字データと平仮名データを使用し，漢字と平仮名のそれぞれのペンダウンモデルを持つストローク HMM モデルを構築した．筆圧差分値特徴量 dz を併用したときのペンアップ区間の補間点数は，第 5.5 節の実験 2 で最大認識率を与えた値 $S = 5$ を用いた．

実験結果

結果を表 6.2 に示す．表 6.1 と比べると，漢字と平仮名間での誤認識が多少確認された．また，漢字と平仮名のそれぞれのペンダウンモデルを持つストローク HMM モデルは，41 モデル（105 状態）である為，従来の 25 モデル（57 状態）よりも認識率が高いことがわかる．

表 6.2: 平仮名認識における特徴量・モデル別認識率（%）

モデル	特徴量	N 位累積認識率 [%]				
		1 位	2 位	3 位	5 位	10 位
25 モデル	r, θ	74.27	86.10	89.72	93.29	96.43
	r, θ, dz	74.88	86.53	90.42	94.23	97.09
41 モデル	r, θ	84.88	91.55	93.99	96.06	97.32
	r, θ, dz	**87.93**	**95.59**	**97.28**	**98.54**	**99.44**

6.2.3 考察

漢字と平仮名のそれぞれのペンダウンモデルを持つストローク HMM モデルは，認識率の低下を防いではいるが，平仮名については環境独立型 HMM では十分な対応が難しいと言える．

また，漢字と平仮名間での誤認識は孤立文字認識の場合でも多少確認されたが，平仮名はデータ長が短い為，文字列認識では複数平仮名と漢字 1 文字の誤認識などが懸念される．

さらに，筆圧特徴量抽出の前処理であるペンアップ区間を複数サンプル観測することで，筆圧特徴量を併用した文字列認識では，より文字境界が不鮮明になり漢字と平仮名間での誤認識が増加することが考えられる．

6.3 予備実験２：二字熟語データに対する筆記方向固定文字列認識

第 4.2.5 節で述べた事前に筆記方向が既知である筆記方向固定文字列に対する文字列認識の予備実験を行う．ここで用いるデータは，ζ_1, ζ_2 セット内の新旧教育漢字で構成されている二字熟語 343 語のみを用いる．

以下の予備実験において，学習は，あらかじめ二字熟語を文字境界ペンアップモデルを 9 としたストロークラベルの連結で構成した二字熟語辞書を用意しておき，Viterbi 連結学習を行った．認識は，言語モデル重み $L_W = 5.0$，挿入ペナルティ $I_P = -0.1$ とし，ビームサーチ（幅 1000）を用いた．

表 6.3: 筆記方向任意二字熟語文字列認識率（%）

特徴量	(r,θ)		(r,θ,dz)	
N-best	1-best	10-best	1-best	10-best
PenUpModel 2 代用	52.71	66.70	63.93	73.57
+文字境界モデル	47.46	62.92	64.55	74.46
+言語モデル	77.86	80.70	80.83	83.22
+文字境界モデル+言語モデル	77.31	80.09	**81.12**	**83.60**

6.3.1 筆記方向任意文字列

実験条件

辞書内語彙	:	新旧教育漢字 1,016 字種
学習資料	:	ζ_1 セット（熟語 343 語），奇数番 30 筆者
評価資料	:	ζ_1 セット（熟語 343 語），偶数番 30 筆者
モデル	:	全共分散行列 2 混合正規分布型

文字境界ペンアップモデルをモデル 2 で代用したものと連結学習によって作成したモデル 9 を使用したものに対し，それぞれ言語モデルを併用したものとしないものの比較実験を行った．特徴量はそれぞれ (r,θ)，(r,θ,dz)（$S=5$）を用いた．

実験結果

筆記方向任意文字列（ζ_1 セット）に対する文字列単位の認識率を図 6.3 に示す．言語モデルを併用した場合に飛躍的に認識率が向上することが分かる．また，筆圧差分値特徴量を併用した場合にも，良好な結果を得た．

6.3.2 重ね書き文字列

実験条件

辞書内語彙	:	新旧教育漢字 1,016 字種
学習資料	:	ζ_2 セット（熟語 343 語），奇数番 30 筆者
評価資料	:	ζ_2 セット（熟語 343 語），偶数番 30 筆者
モデル	:	全共分散行列 2 混合正規分布型

文字境界ペンアップモデルをモデル 4 で代用したものと連結学習によって作成したモデル 9 を使用したものに対し，それぞれ言語モデルを併用したものとしないものの比較実験を行った．特徴量はそれぞれ (r,θ)，(r,θ,dz)（$S=5$）を用いた．

表 6.4: 重ね書き二字熟語文字列認識率（%）

特徴量	(r, θ)		(r, θ, dz)	
N-best	1-best	10-best	1-best	10-best
PenUpModel 4 代用	39.77	56.03	57.40	69.94
+文字境界モデル	43.68	59.29	61.00	72.34
+言語モデル	74.11	77.57	78.56	81.46
+文字境界モデル+言語モデル	74.45	77.94	**79.25**	**82.10**

実験結果

重ね書き文字列（ζ_2 セット）に対する文字列単位の認識率を図 6.4 に示す．言語モデルを併用した場合に飛躍的に認識率が向上することが分かる．また，筆圧差分値特徴量を併用した場合にも，良好な結果を得た．

6.3.3 考察

新旧教育漢字のみで構成される二字熟語に対して，筆圧差分値特徴量を併用，言語モデル併用による認識率向上が確認できた．特徴量 (r, θ, dz) において，連結学習により作成した文字境界ペンアップを用いる効果が見られた．これは，筆圧特徴量抽出の前処理によるペンアップモデルの尤度が大きくなった為であると考える．

また，重ね書き文字列についても，文字が重なっていない筆記方向任意文字列と同様に認識可能であることが確認できた．

6.4 文字列認識の性能評価

本システムにおけるオンライン文字列認識の性能評価には，隠れマルコフモデルを用いた連続音声認識ツールキット HTK[37] の認識性能評価ツール HResults を使用した．正解ラベルと認識結果とのシンボルによる DP マッチングを行い，文字列単位での正解率と，文字単位での正解率と認識精度を算出する．全文字数を N，正解数を H，挿入誤り数を I としたとき，正解率 $Correct$ と認識精度 $Accuracy$ は以下のように計算される．

$$Correct = \frac{H}{N} \times 100 \quad [\%] \tag{6.5}$$

$$Accuracy = \frac{H - I}{N} \times 100 \quad [\%] \tag{6.6}$$

また，置換誤り数 S，削除誤り数 D についても算出する．

6.5 評価実験１：文字境界ペンアップモデルに関する実験

筆記方向に依存しない文字境界ペンアップモデルを用い，筆記方向任意文字列（ζ_1）セットと重ね書き文字列（ζ_2）セットの両方に対して認識評価を行った．特徴量 (r, θ) と (r, θ, dz) について認識評価をそれぞれ行う．但し，特徴量 (r, θ, dz) についてはペンアップ区間の補間点数 $S = 3$ とした．

6.5.1 実験条件

データベース	:	筆記方向任意文字列セット（ζ_1）
		重ね書き文字列セット（ζ_2）
学習資料	:	奇数番目の 30 筆者
評価資料	:	偶数番目の 30 筆者
モデル	:	全共分散型 2 混合正規分布

筆記方向任意文字列および重ね書き文字列の両方のデータを併わせて，共通の文字境界ペンアップモデル，すなわち筆記方向に依存しないペンアップモデルを作成した．学習資料は文字列長が 4 字以下の語句 497 語であり，文字間をモデル ‘9’ で連結する Viterbi 連結学習により，42 種類（仮名/漢字用ペンダウンモデル各 16 種，共通ペンアップモデル 10 種）のストローク HMM を学習した．HMM は全共分散型の正規分布であり，その混合数は 2 とした．

言語モデルの重み係数を $L_W = 4.0$，文字挿入ペナルティは特徴量 (r, θ) のとき $I_P = -0.5$，特徴量 (r, θ, dz) のとき $I_P = 3.0$ とし，ビーム幅 1,000 の枝刈り探索を行った．筆圧情報を特徴量併用する場合には，第 5 章で述べたペンアップ区間における特徴量抽出の前処理により，文字境界が検出されにくい為，文字挿入ペナルティを大きくした．

6.5.2 実験結果

筆記形態別（データセット別）の正解率と認識精度について，表 6.5 に，文字境界ペンアップモデルを共有して学習した場合の認識性能詳細を表 6.7 に示す．また，表 6.6 に，特徴量 (r, θ) について 10 筆者分のデータに対する文字境界ペンアップモデルの組み合わせを変えたときの正解率と認識精度を示す．

また，特徴量 (r, θ, dz) について文字境界ペンアップモデルを共有して学習した場合の重ね書き文字列データセットに対する筆者別正解率を図 6.1 に示す．

文字境界ペンアップモデルにおける方向特徴量 θ の分布が，筆記形態別に学習した場合，筆記方向任意文字列では主に横書きが多いためにペンアップモデル ‘2’（右上）の分布に類似し，重ね書き文字列ではペンアップモデル ‘4’（左上）の分布に類似する事が確認できた．これに対して，共有して学習した場合では方向特徴量 θ の分散が，筆記形態別に学習した場合に比べてかなり大きくなる．表 6.5 中の文字境界ペンアップの違いによ

表 6.5: 文字境界ペンアップモデルによる筆記形態別 1 位認識率 [%]（括弧内は 10 位までの累積認識率）

筆記形態	文字境界ペンアップ	特徴量	文字列単位	文字単位	
			正解率	正解率	認識精度
筆記方向任意 (ζ_1)	共有モデル	(r, θ)	69.15 (73.40)	81.93 (85.82)	75.43 (81.89)
		(r, θ, dz)	69.87 (73.59)	80.59 (84.28)	74.91 (80.85)
	筆記方向任意専用モデル	(r, θ)	69.82 (74.24)	83.22 (87.06)	76.37 (82.95)
		(r, θ, dz)	70.16 (74.16)	81.60 (85.46)	75.60 (81.79)
重ね書き (ζ_2)	共有モデル	(r, θ)	67.95 (72.54)	82.74 (86.77)	74.49 (81.86)
		(r, θ, dz)	69.34 (73.46)	80.52 (84.07)	73.67 (80.00)
	重ね書き専用モデル	(r, θ)	69.10 (73.61)	83.30 (87.24)	76.37 (83.12)
		(r, θ, dz)	70.59 (74.71)	81.72 (85.25)	76.01 (81.87)

表 6.6: 文字境界ペンアップモデル別 1 位認識率 [%]（括弧内は 10 位までの累積認識率）

データセット	文字境界ペンアップ	文字列単位	文字単位	
		正解率	正解率	認識精度
筆記方向任意 (ζ_1)	重ね書き専用	67.56 (72.04)	77.50 (82.13)	72.93 (79.52)
	共有モデル	68.96 (73.53)	81.59 (85.48)	75.96 (82.04)
	筆記方向任意専用	70.01 (74.63)	83.36 (87.13)	77.37 (83.45)
重ね書き (ζ_2)	筆記方向任意専用	68.34 (72.74)	81.49 (85.83)	73.69 (81.19)
	共有モデル	68.87 (73.13)	82.15 (86.15)	74.41 (81.52)
	重ね書き専用	70.10 (74.34)	82.79 (86.61)	76.39 (82.81)

表 6.7: 文字境界ペンアップモデル共有による文字列認識性能評価

データセット	特徴量	N 位累積	文字列単位			文字単位				
			H	S	N	H	D	S	I	N
筆記方向任意 (ζ_1)	(r,θ)	1 位	11866	5295	17161	41630	1341	7840	3304	50811
		10 位	12596	4565	17161	43604	910	6297	1997	50811
	(r,θ,dz)	1 位	11992	5172	17164	40951	1605	8261	2882	50817
		10 位	12631	4533	17164	42830	1137	6850	1742	50817
重ね書き (ζ_2)	(r,θ)	1 位	11533	5439	16972	41538	639	8027	4141	50204
		10 位	12312	4660	16972	43561	373	6270	2464	50204
	(r,θ,dz)	1 位	11774	5205	16979	40451	1042	8742	3444	50235
		10 位	12473	4506	16979	42232	745	7258	2044	50235

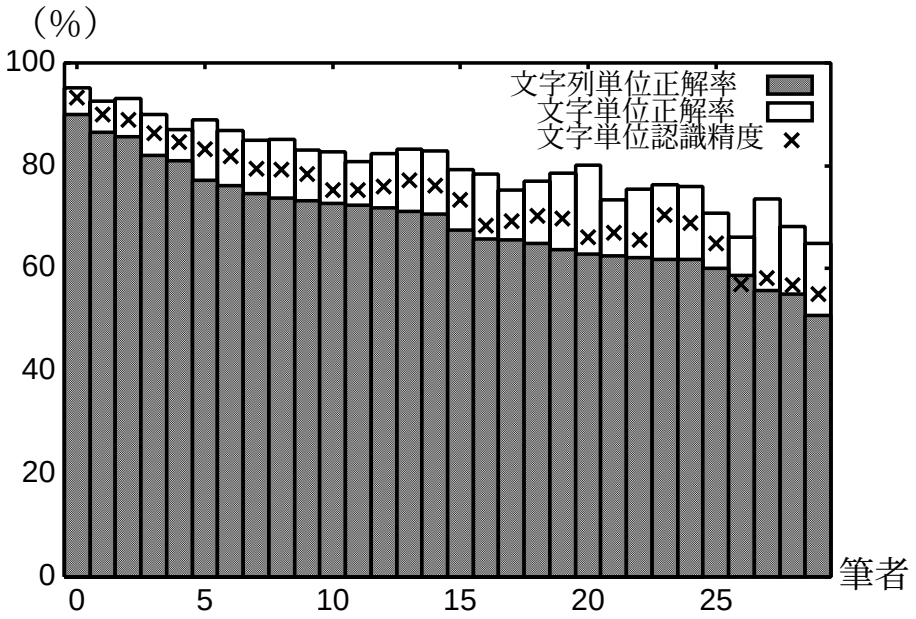


図 6.1: 重ね書き文字列に対する筆者別正解率

る正解率の差が特徴量に関わらず小さい事から，共有して学習した文字境界ペンアップモデルの筆記方向への依存性は小さいことが分かる．また，表 6.6 に於いても正解率の差が小さいことが分かる．本認識手法は，方向について頑健であることから，一文字毎に筆記方向が変動するような手書き文字列でも認識が可能である．

筆圧特徴量について

特徴量 (r, θ, dz) については，第 6.2.1 節の平仮名孤立文字認識では効果が見られ，また第 6.3 節の漢字のみを用いた予備実験においても効果が見られている．

文字単位での正解率では，特徴量 (r, θ) に比べ，特徴量 (r, θ, dz) を用いる場合の正解率が低い．原因の 1 つとして，第 5 章で述べた辞書定義の間違いによる影響により誤認識数が増えたとが挙げられる．他には，表 6.7 から，特徴量 (r, θ) に比べて削除誤り数 D が大きく挿入誤り数 I が小さいことから，筆圧特徴量抽出の前処理による文字境界検出の精度の低下と，評価資料に平仮名データが含まれることによる局所的なマッチングの影響が考えられる．辞書定義の修正の他に，特徴量抽出の前処理における補間点数 S や探索パラメータ（言語モデル重み L_W，文字挿入ペナルティ I_P）の適切な設定と，文字境界ペンアップモデルを前後の字種やストロークによる環境依存型モデルの導入が検討課題に挙げられる．

文字列単位での正解率では，文字境界ペンアップモデルを共有して学習したときで，特徴量 (r, θ) に比べ，特徴量 (r, θ, dz) を用いた場合，筆記方向任意文字列に対して 69.82% から 70.16% へ，重ね書き文字列に対して 69.10% から 70.59% へ正解率が向上した．筆記形態別に学習した場合についても，同様の傾向が見られ，文字列単位での正解率は向上した．

誤認識要因の考察

表 6.8 に，特徴量 (r, θ, dz) を用いて文字境界ペンアップを共有して学習したとき頻繁に見られた誤認識の順（両データセットを合わせて）に，誤認識例を示す．

「有」,「成」,「右」を含む筆記文字列のうち右肩に＊印の付いている文字は辞書と筆順が異なる．また,「表わす」⇒「表れす」は，文字列境界が正しく得られているにも関わらず，誤認識した例である．この他，ストロークの崩れ，文字やストロークの傾きにより誤認識した例も見られた．これらの誤認識傾向は，孤立手書き文字認識と共通の問題である．

一方，文字列認識特有の誤認識の要因として，探索によって得られる一文字の区間の不一致が挙げられる．例えば,「じ」が分離して「しい」となったり（挿入誤り）,「明日」が結合して「間」となったり（脱落誤り）する誤認識例が見られた．これらは，言語モデル重み係数と挿入ペナルティの調整，あるいは，trigram などのより高精度な言語モデルの導入によってある程度は改善すると考えている.「一人」を「大」に誤認識する例は，重ね

表 6.8: 文字境界ペンアップモデル共有による文字列認識の誤認識例（各文字列データ 60 サンプルに対する誤り例の多い順）

筆記文字列		認識結果	誤り数
人手	⇒	入手	51
共有*	⇒	共存	36
会計士	⇒	会証	35
反りが合わない	⇒	列が合わない	34
表わす	⇒	表れす	32
一人	⇒	大	31
決して	⇒	決に	30
右*上	⇒	石上	28
念じる	⇒	念しいる	27
差しつ差されつ	⇒	差して差されつ	27
成*分	⇒	友人分	26
案じる	⇒	案しいる	26
長*短	⇒	取短	24
向かい合わせる	⇒	向が合わせる	24
有*効	⇒	存効	22
有*望	⇒	否望	21
長*い間	⇒	取り間	21
右*から左へ	⇒	石から左へ	21
明日はどうですか	⇒	間はどうですか	20
感謝	⇒	一感謝	20

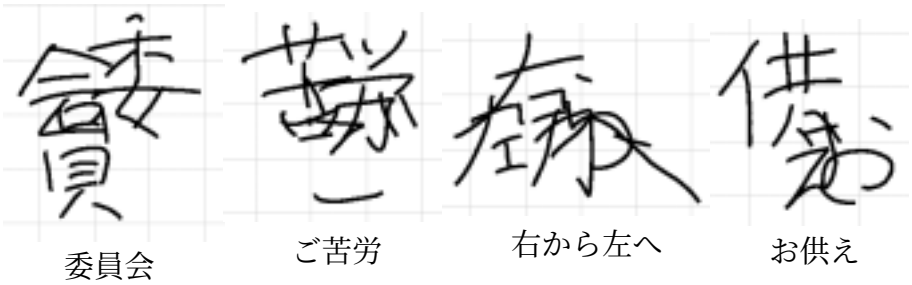


図 6.2: 筆順の正しい孤立手書き文字を連結して作成した擬似手書き文字列の例

書き文字列特有の問題であり，漢字構造辞書を用いる本認識方式では，ストロークによる辞書定義が（文字間ペンアップを除いて）同一となっているため両者の区別は困難である．

6.6 評価実験２：孤立手書き文字認識との比較実験

収集した文字列について，事前に文字境界を視察で与えた場合との比較認識実験を行った．また，筆順の正しい孤立手書き文字を連結した擬似文字列データを使用して，筆順違い以外の誤認識要因を調査した．

6.6.1 実験条件

重ね書き文字列（ζ_2）	
学習資料	： 奇数番目の 30 筆者
評価資料	： 偶数番目の 5 筆者
筆順の正しい手書き文字（γ_2）による疑似文字列	
学習資料	： 奇数番目の 30 筆者
評価資料	： 偶数番目の 30 筆者

重ね書き文字列データセット中の 5 名の筆者（0008, 0013, 0028, 0055, 0058）の 2,865 語（8,504 文字）に対し，手作業によって文字境界ラベルを付与した．また，筆順の正しい丁寧な手書き文字データセット（γ_2）[36] の個々の文字データを筆跡座標を変えずに連結し，重ね書き文字列データセットと同じ 578 語を作成した．この擬似文字列は筆記位置に連続性が無いため，図 6.2 に示すように筆記方向に一貫性のない文字列となる．

文字境界を付与した場合，必ず文字境界となる時刻に探索ネットワークのモデル ‘9’ に滞在し，それ以外の状態にある仮説は棄却するように実装した．他の実験条件は実験 1 と同様にした．

表 6.9: 文字列の正解率 [%]（重ね書き文字列については 5 筆者分の平均，括弧内は 10 位までの累積認識率）

筆記形態	言語モデル	文字境界	特徴量	文字列単位	文字単位	
				正解率	正解率	認識精度
重ね書き文字列	無	無	(r,θ)	27.03 (43.46)	60.88 (70.39)	3.57 (30.20)
			(r,θ,dz)	11.68 (22.97)	42.76 (52.83)	-86.13 (-46.17)
		付与	(r,θ)	62.51 (78.74)	83.20 (91.10)	83.20 (91.10)
			(r,θ,dz)	65.22 (80.64)	83.77 (91.42)	83.77 (91.42)
	有	無	(r,θ)	69.19 (73.38)	83.14 (87.01)	74.93 (82.01)
			(r,θ,dz)	70.14 (74.47)	80.89 (84.19)	74.51 (80.24)
		付与	(r,θ)	78.81 (83.56)	90.51 (93.36)	90.51 (93.36)
			(r,θ,dz)	80.26 (85.31)	90.99 (93.84)	90.91 (93.80)
筆順の正しい擬似文字列	有	無	(r,θ)	87.99 (91.95)	93.84 (96.18)	91.11 (94.78)
			(r,θ,dz)	88.79 (92.98)	94.52 (96.69)	92.11 (95.22)
		付与	(r,θ)	93.44 (96.18)	97.32 (98.22)	97.32 (98.22)
			(r,θ,dz)	94.09 (98.47)	97.70 (99.40)	97.70 (99.40)

6.6.2 実験結果

重ね書き文字列データセットについては，言語モデルの有無についての実験も行った．それぞれの認識結果を表 6.9 に示す．また，筆順の正しい疑似文字列データセットに対する認識結果も併わせて示す．

特徴量については，(r,θ) に比べて筆圧特徴量併用 (r,θ,dz) の方が，文字単位での正解率，文字列単位での正解率共に正解率が向上した．

孤立手書き文字認識との比較

文字境界を付与して言語モデルを使用しない場合の認識は従来の孤立手書き文字認識に相当し，文字単位の正解率は 83.20% であった．これは第 5 章での 96.90% に比べて低く，探索の枝刈りによる影響もあるが，筆順誤りを含む文字の品質の低さを表す一つの指標と考えられる．一文字毎の文字の筆順誤りや粗雑さの他に，文字を上に重ねて筆記する為にどのようなストロークを筆記したか視認できなくなる影響もあると考える．

文字列と孤立文字（文字境界付与）の認識精度の差は言語モデルの使用によって大幅に縮められる事が確認できたが，全般的に孤立文字の方が認識性能が高い傾向にある．これは，文字列を認識する場合ではあらゆる時刻において文字境界である仮説を展開しなけれ

ばならないので，枝刈りによって残す仮説数が同じ場合には，文字列認識は不利である．一方，文字列の方が有利な場合として次の例がある．例えば，筆順が重要となる本手書き文字認識手法では「博」の右上の点を付け忘れた程度であれば正しく認識できるが，筆順を誤って最後に付けると全然違う文字として認識される．しかし，文字列認識で「博学」を認識すると，筆順を誤った最後の点は「学」が吸収して，正しく「博学」として認識する事ができる．このように，文字境界が分からない事によって，逆に改善された例が 26 個あり，例えば「重要」⇒「重無」の誤認識（文字間に左下方向の余分な線が付加したため）も改善された．

尚，特徴量 (r, θ) を用いた重ね書き文字列認識によって事後的に得られる文字境界の正解検出率は 88.00%，誤挿入率は 20.39%であった．

筆順の正しい疑似文字列データセットとの比較

筆順を正しく丁寧に書いた文字列であれば，文字単位正解率 94.52%，文字列単位正解率 88.79% を達成できることが分かる．また，特徴量 (r, θ) を用いた手書き文字列認識によって事後的に得られる文字境界の正解検出率は 94.60%，誤挿入率は 6.59 % と高い検出精度であり，手書き文字列認識の基本性能の良さ，および今回の実験で設定したパラメータの妥当性を示すものと考えている．

表 6.10 に，特徴量 (r, θ, dz) のとき，認識率順に 下位の 19 種の文字列とその誤認識例を示す．第 5 章や従来のにおける孤立手書き文字認識 [5] と同様に，平仮名や低画数漢字を含む文字列の認識率が低く，重要な課題である事が確認できた．

6.7 まとめ

筆圧情報特徴量の併用，言語モデルの導入により，認識性能の向上を達成した．また，重ね書き文字列を含め，筆記方向の自由な文字列に対するオンライン手書き文字列認識システムを実現した．評価実験では，新旧教育漢字 1,016 字種と平仮名 71 字種からなる文字 bigram 確率を言語モデルとして用い，収集した 578 種類の重ね書き文字列に対して 69.34%，擬似的に生成した筆順の正しい文字列に対して 88.79 % の文字列単位正解率を達成した．

今後，先行研究である異筆順の辞書登録 [5, 6] と前後ストロークの環境依存型モデル [8, 9] を併用することで，筆順違いへの対応と HMM モデルの精度向上が図れ，更なる認識率の向上が期待できる．また，高精度な言語モデル適応（[22] など）をすることでも認識率向上が期待できる．

表 6.10: 筆順の正しい疑似文字列認識の誤認識例（低認識率順に 19 種の文字列）

文字列 (認識率%)	⇒	誤認識例
身じろぎもしない (3)	⇒	身しいろぎもしない，息つつぎもしない
差しつ差されつ (3)	⇒	差して差されつ，差して差孝
急きたてられる (7)	⇒	急きたこられる，急きた一人られる
何を言うてんねん (7)	⇒	何を言うとんねん，何を言うこんねん
善かれ悪しかれ (10)	⇒	善かれ悪いかれ，善が悪しかれ
人手 (10)	⇒	入手，八人へ，八つま
有利 (13)	⇒	八日利，八月利，角利，右利
有望 (13)	⇒	八日望，角望，八月望，価望
有効 (13)	⇒	八月効，角効，八日効，価効，右効
有限 (13)	⇒	八日限，八月限，角限
分かちあたえる (17)	⇒	分からあたえる，分かおめたえる
念じる (17)	⇒	念しいる，念ぶる
座標 (17)	⇒	私も標，米生標，広生標
金がものを言う (20)	⇒	金がものちと言う，金がものすと言う
穴だらけにする (23)	⇒	穴だのけにする，穴だわけにする
大まかに言えば (27)	⇒	大きかに言えば，大なかに言えば
後述の場合を除き (27)	⇒	後述への場合を除き，後述の場合すく除き
一人 (30)	⇒	大，一八
悪しからず (30)	⇒	悪いからず，悪くからず

第7章　結論

7.1　本研究の成果

ストローク HMM に基づくオンライン文字認識手法に，言語モデルを併用し，オンライン文字列認識システムを構築した．まず，筆圧情報の特徴量としての新たな利用法を提案し，基本性能の向上を行った．次に，入力画面の小さい携帯端末への実装や視覚障害者の入力装置を想定した重ね書き文字列入力方式を提案し，重ね書きも含める筆記方向自由文字列認識を実現した．

筆圧特徴量の利用法としては，ペンの上げ下げを表わす連続量としての筆圧値，筆圧の増減のパターンを表わす特徴量としての筆圧変化量の二種類の筆圧特徴量のいずれかを従来特徴量（速度・方向）と併用し，3 次元特徴量を用いた認識システムについて検討した．また，特徴抽出の前処理として，ペンアップ区間の筆圧情報の補間手法についても提案した．新旧教育漢字 1,016 字種を用いた不特定筆者認識評価実験の結果，速度・方向特徴量と比較して，筆圧情報を併用した特徴量では，丁寧な手書き文字に対しては 96.90% から 97.77% へ，走り書き文字に対しては 90.30% から 92.37% へと認識率が向上した．

オンライン文字列認識システムの構築では，オンライン文字認識と音声認識との同型性に着目することで，筆記領域の切り出しや文字境界検出を不要とし．また位置情報に依存しない特徴量（速度・方向・筆圧情報）を用いることで，重ね書き文字列認識を実現した．さらに，文字境界での隣接文字 への移動方向に注目し，重ね書きを含め，筆記方向の自由な文字列に対する認識システムを実現した．新旧教育漢字 1,016 字種と平仮名 71 字種を辞書内語彙とした認識評価実験の結果，擬似的に生成した筆順の正しい文字列に対して，文字列単位で 88.79%，文字単位で 94.52% の認識率を達成した．

7.2　今後の課題

本論文で構築したオンライン文字列認識手法について，今後の課題である拡張について述べる．

7.2.1 更なる認識性能の向上

前後ストロークによる環境依存型モデル [8, 9] を導入し， HMM モデルを改善することで，認識性能の向上が図れる．

孤立文字認識に比べて文字列認識では，筆順違いによる誤認識が顕著になる為，先行研究 [5, 6] による異筆順の辞書追加法などを導入することが挙げられる．

本論文で用いた言語モデルは，加算スムージング法による文字間バイグラムモデルであるが，線形補間（削除空間法）やバックオフ平滑化による文字間トライグラムモデルを導入し，統計的言語モデルの性能向上が望まれる．出現頻度の高い単語による形態素辞書を用いて，誤認識訂正する [22] 利用法が考えられる．

本論文では遅延言語処理を用いているが，言語的制約の適用が遅延すると最適な解が探索途中でビーム幅から漏れる可能性がある為，言語モデルを分解 (factoring) して探索ネットワーク中に振り分ける必要がある．また，膨大な組み合わせが存在するトライグラムモデルなどの導入を考慮して，探索法を１パス方式から２パス方式などのマルチパス探索方式にすることが考えられる．

7.2.2 大語彙オンライン手書き文章認識に向けて

本論文で提案した手法は，認識単位を１文章に拡張した大語彙オンライン手書き文章認識へと拡張できる．文字列の文字数増加による誤認識を防ぐ為に，単語間 N グラムモデルや文節間 N グラム等の導入が考えられる．

また実用性を考慮し，使用頻度の高い漢字 JIS 第一水準 2965 字種や片仮名，英数字，記号「、。々」などを追加し辞書語彙数を増加させることが望まれる．

一方で入力インターフェースの視点で捉えると，孤立文字認識手法とは異なり，文字列中の誤認識を訂正するヒューマンインターフェース [38] が必要となる．

謝辞

本研究を行うにあたり，大変有益な御指導と御助言を頂きました北陸先端科学技術大学院大学の下平 博 助教授に心から感謝致します．本研究をまとめるにあたり，有益な御助言を頂きました東京大学大学院情報理工学系研究科の嵯峨山 茂樹 教授に大変感謝致します．北陸先端科学技術大学院大学の中井 満 助手には，研究を進めるにあたり数多くの御助言を頂き，深く感謝致します．北陸先端科学技術大学院大学博士前期課程修了生の秋良直人氏，井波暢人氏には，研究の立ち上げに際して多くの御助言を頂き，感謝致します．また，文字データ収集にあたり，多くの方にご協力頂きましたことに深く感謝いたします．

そして，日々の研究生活を支えて下さった，嵯峨山・下平研究室の博士前期課程２年生をはじめとする研究室の皆様と，大学院生活を大変有意義なものとして頂いた親友である長谷川信氏，長谷川勝巳氏をはじめとする北陸先端科学技術大学院大学の多くの友人に深く感謝の意を表します．

最後に，今までの長きにわたる学生生活を陰ながら支えて頂いた両親に大変深く感謝の意を表し，本論文の結びと致します．

関連図書

[1] 川口 弘光, “HMM を用いたオンライン手書き文字認識の研究 ,” 北陸先端科学技術大学院大学 修士論文 (2000).

[2] 嵯峨山茂樹, 中井満, 下平博, “ストローク HMM に基づくオンライン手書き文字認識方式,” 信学技報 PRMU2000-35, pp. 1–pp.8 (2000-06).

[3] 中井満, 嵯峨山茂樹, 秋良直人, 小場久雄, 下平博, “ストローク HMM によるオンライン手書き文字認識の性能評価,” 信学技報 PRMU2000-36, pp. 9–pp.16 (2000-06).

[4] 秋良直人, 中井満, 下平博, 嵯峨山茂樹, “ストローク HMM に基づくオンライン手書き文字認識の特徴量の検討,” 信学技報, PRMU2000-134, pp. 31-38, (2000-12).

[5] 秋良直人, 中井満, 下平博, 嵯峨山茂樹, “ストローク HMM を用いたオンライン非目視手書き文字認識の性能評価,” 信学技報 PRMU2000-206, pp. 39–46 (2001-03).

[6] 秋良直人, “隠れマルコフモデルを用いたオンライン非目視手書き文字認識の研究 ,” 北陸先端科学技術大学院大学 修士論文 (2001).

[7] 徳野淳子, 中井満, 下平博, 嵯峨山茂樹, 細川啓子, “視覚障害者を対象としたストローク HMM オンライン文字認識方式の性能評価,” 信学技報 WIT2001-13, pp. 19–24 (2001-08).

[8] 井波暢人, 松田繁樹, 中井満, 下平博, 嵯峨山茂樹, “環境依存型ストローク HMM を用いたオンライン手書き文字認識,” 信学技報 PRMU2000-135, pp. 39–46 (2000-12).

[9] 井波暢人, “隠れマルコフモデルを用いたオンライン走り書き文字認識の研究 ,” 北陸先端科学技術大学院大学 修士論文 (2001).

[10] 高橋賢一朗, 安田英史, 松本隆, “Hidden Markov Model を用いたオンライン手書き文字認識,” 信学技報 PRMU96-211, pp. 143–150 (1997-03).

[11] 伊藤等, 中川正樹, “Hidden Markov Model に基づくオンライン手書き文字認識,” 信学技報 PRMU97-85, pp. 95–100 (1997-07).

[12] 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 PRMU98-138, pp. 17–24 (1998-12).

[13] 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 PRMU98-139, pp. 25–30 (1998-12).

[14] 岡本正義, 山本英人, 吉川隆敏, 堀井洋, “物理的特徴量を用いたオンライン文字自動切り出し手法,” 信学技報 PRU95-13, pp. 93–100 (1995-05).

[15] 相澤博, 若原徹, 小高和己, “複数のストローク特徴を用いた手書き文字列からの実時間切り出し,”’ 信学論 Vol. J80-D-II, No. 5, pp. 1178–1185 (1997-05).

[16] 仙田修司, 濱中雅彦, 山田敬嗣, “切り出しパラメータが学習可能なオンライン手書き文字切り出し手法,” 信学技報 PRMU97-219, pp. 17–24 (1998-02).

[17] 迫江博昭, 平尾浩一郎, 片山喜規, “２次元 - １次元ＤＰマッチングと t 同期ＤＰマッチングに基づく筆順フリーな枠無し文字列のオンライン認識,” 信学技報 PRMU94-50, pp. 23–30 (1994-10).

[18] 中川聖一, 斎藤稔, “エルゴディック HMM に基づく音声の自動獲得単位を用いた音声認識,” 信学技報 SP97-9, pp.55–62 (1997-05).

[19] 林貴文, 森大毅, 鈴木基之, 牧野正三, 阿曽弘具, “音響類似性に基づく認識単位を用いた音声認識,” 信学論 Vol. J83-D-II, No. 11, pp. 2137–2145 (2000-11).

[20] 緒方淳, 有木康雄, “大語彙連続音声認識における最ゆう単語 back-off 接続を用いた効率的な N-best 探索法,” 信学論 Vol. J84-D-II, No. 12, pp. 2489–2500 (2001-12).

[21] 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,” 信学技報 PRMU2000-37, pp. 17–24 (2000-06).

[22] 岡野祐一, 川又武典, 依田文夫, “オンライン文字列認識精度向上に関する検討,” 信学技報 PRMU2000-223, pp. 9–14 (2001-03).

[23] 鹿野 清宏 , 伊藤 克亘 , 河原 達也 , 武田 一哉 , 山本 幹雄 : 音声認識システム , オーム社 , 2001.

[24] 西野文人, “文字認識における自然言語処理,” 情処学誌 Vol.34, No. 10, pp. 1274–1280 (1993-10).

[25] 永田昌明, “文字類似度と統計的言語モデルを用いた日本語文字認識誤り訂正法,” 信学論 Vol. J81-D-II, No. 11, pp. 2624–2634 (1998-11).

[26] 北 研二 : 確率的言語モデル , 東京大学出版会 , 1999.

[27] 若原徹, 梅田三千雄, “ストローク結合規則を用いたオンラインくずし字分類,” 信学論 Vol. J67-D, No. 11, pp. 1285–1292 (1984-11).

[28] 大森健児, “続け字と崩し字に対応したヒューリスティックなストローク合わせ法によるオンライン手書き漢字認識,” 情処学論 Vol.31, No. 5，pp. 710–720 (1990-05).

[29] 沢井良一, 中川正樹, 高橋延匡, “オンライン手書き日本語文字認識におけるつづけ字の検討,” 信学論 Vol. J70-D, No. 5, pp. 946–957 (1987-05).

[30] 慎重弼, 迫江博昭, “筆順・画数自由オンライン文字認識のための画対応決定法,” 信学論 Vol. J82-D-II, No. 2, pp. 230–239 (1999-02).

[31] 佐藤幸男, 足立秀鋼, “走り書き文字のオンライン認識,” 信学論 Vol. J68-D, No. 12, pp. 2116–2122 (1985-12).

[32] 趙鵬，佐藤幸男, 吉村ミツ, “オンライン走り書き文字認識における汎用辞書の作成,” 情処学論 Vol. 34, No. 3, pp. 418–425, 1993.

[33] 小林充, 真崎晋哉, 宮本修, 中川洋一, 小宮義光, 松本隆, “オンライン手書き文字認識アルゴリズム RAV (Reparameterized Angle Varitions),” 情処学論 Vol.41, No. 9, pp. 2536–2544 (2000-09).

[34] 田口英郎, 桐山公一, 田中永二, 藤井克彦, “ペンの動きに着目したオンライン署名識別法,” 信学論 Vol. J71-D, No. 5, pp. 830–840 (1998-05).

[35] 菊地真美, 赤松則男, “高速筆記者のための高感度筆圧ペンの試作と筆者認証実験,” 信学論 Vol. J83-D-II, No. 8, pp. 1763–1772 (2000-08).

[36] 須藤隆, 中井満, 下平博, 嵯峨山茂樹, “筆圧情報を併用したストローク HMM に基づくオンライン文字認識,” 信学技報 PRMU2001-189, pp. 93–100 (2001-12).

[37] “The Hidden Markov Model Toolkit (HTK),” version 3.1 (2002).
http://htk.eng.cam.ac.uk/

[38] 坂東宏和, 福島貴弘, 加藤直樹, 中川正樹, “枠なし手書き文字列認識における誤認識訂正インターフェースの研究,” 情処研報 2000-HI-89-12, pp. 81–88 (2000-07).

[39] 須藤隆, 中井満, 下平博, 嵯峨山茂樹, “ストローク HMM を用いたオンライン重ね書き文字列認識,” 信学技報 PRMU2001-265, pp. 163–170 (2002-03).

研究業績一覧

- 須藤 隆, 中井 満, 下平 博, 嵯峨山 茂樹, “筆圧を利用したストローク HMM に基づくオンライン走り書き文字認識,” 平成 13 年電気関係学会北陸支部大会講演論文集, F-67, p.407, Oct 2001.
- 須藤 隆, 中井 満, 下平 博, 嵯峨山 茂樹, “筆圧情報を併用したストローク HMM に基づくオンライン文字認識,” 電子情報通信学会技術研究報告, PRMU2001-189, pp.93-100, Dec 2001.
- 須藤 隆, 中井 満, 下平 博, 嵯峨山 茂樹, “ストローク HMM を用いたオンライン重ね書き文字列認識,” 電子情報通信学会技術研究報告, PRMU2001-265, pp. 163–170, Mar 2002.

付 録A　JAIST-IIPL （北陸先端科学技術大学院大学・知能情報処理学研究室）オンライン手書き文字データベース

本論文で用いた文字データベース (JAIST IIPL オンライン手書き文字データベース [3, 36, 39]) に関して，認識実験に使用したデータセットの説明を行う．

A.1　各データセットの特徴

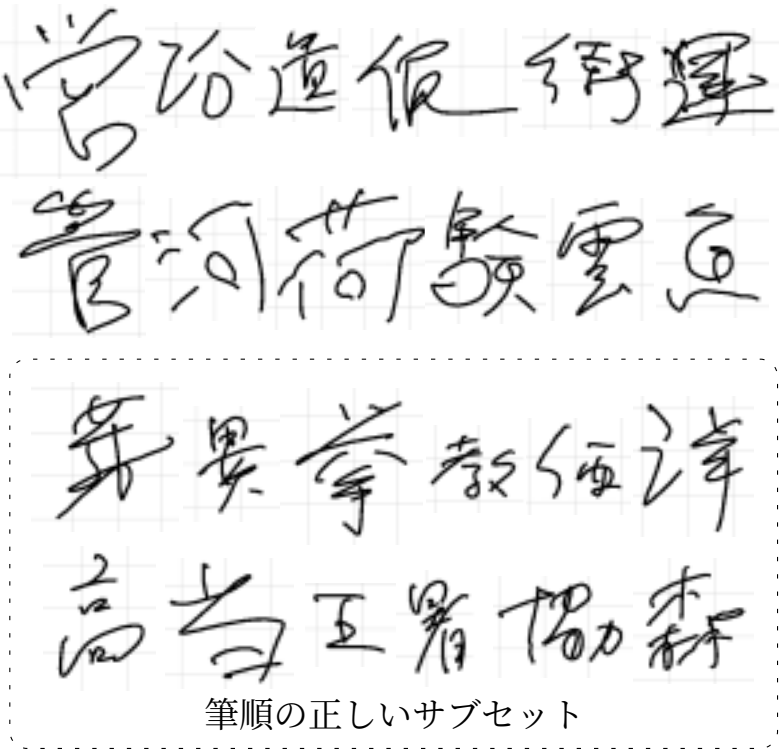
認識実験に用いたデータセットは，$\gamma_2, \epsilon_1, \epsilon_2, \zeta_1, \zeta_2$ セットの 5 セットである．以下に，その特徴を述べる．

- γ_2 セット $\cdots$ 筆順 · 画数 が正しく丁寧に筆記されたデータセットである． 60 筆者分のデータがある．
- ϵ_1, ϵ_2 セット $\cdots$ 筆順 · 画数違いを含み走り書き文字データセットである．68 筆者分のデータがある．
- ζ_1 セット $\cdots$ 筆順 · 画数違いを含む筆記方向任意文字列データセットである．60 筆者分のデータがある．
- ζ_2 セット $\cdots$ 筆順 · 画数違いを含む重ね書き文字列データセットである．60 筆者分のデータがある．

このうち，$\zeta_1, \zeta 2$ セットについては第 3 章において説明しているので，他のセットと ϵ セットの筆順の正しいサブセットについて以下に説明する．

A.2　筆圧情報付き・筆順の正しい手書き文字データセットの収集（γ_2 セット）

既報 [3] の正しい筆順で収集した新旧教育漢字手書き文字データセット（α セット）は筆圧情報が欠けているので，新たに筆圧情報付きで，丁寧に書かれた筆順の正しい手書き



筆順の正しいサブセット

図 A.1: 走り書き文字データセット文字例

文字データを収集した [36]．すなわち収集時にリアルタイムで筆順チェック [3] を行い、画数・筆順が誤っている文字については，その場で書き直すよう要求した．

収集環境には，Linux の X Window System とペンタブレット (Wacom intuos i-400) を使用し，Gtk+/Gdk で構築したキャンバス上に筆記して，ペンの絶対座標値 (x, y) ，ペンのアップダウン情報，筆圧値 (1,024 レベル)，ペンの傾き (θ_x, θ_y)，時刻を収集した．字種の内訳は，以下の通りであり，それら全てについて，60 人の筆者が筆記した．

- 平仮名 $\cdots$ 83 字種
- 片仮名 $\cdots$ 86 字種
- アルファベット $\cdots$ 62 字種
- 新旧教育漢字 $\cdots$ 1016 字種

筆順チェックを除く，その他の収集条件は既報 [5] の英数・仮名・漢字手書き文字データセット（γ セット）と同じであるので，これらを区別する為に，新たに収集したデータセットを γ_2 セットと呼ぶ．

A.3 走り書き文字データセットの収集（ϵ セット）

自由筆記データとして，走り書き文字データを 68 人分収集した [36]．但し，収集字種は以下の通りである．

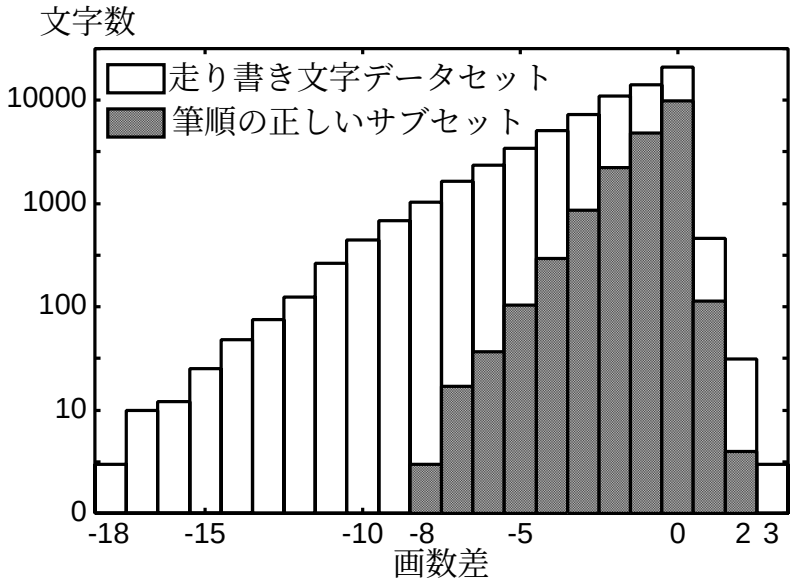


図 A.2: 走り書き文字データセットに占める筆記画数と辞書画数の画数差による頻度

- 平仮名 $\cdots$ 83 字種
- 片仮名 $\cdots$ 86 字種
- アルファベット $\cdots$ 62 字種
- 記号 $\cdots$ 131 字種
- 新旧教育漢字 $\cdots$ 1016 字種

収集環境・条件は γ_2 セットの収集条件に準じているが,

- 普段よりも速い筆記速度
- 画の連結や略字も可
- 筆順は自由

である事を意識して筆記して貰った．また，このデータセットに限り，同一筆者の走り書きに依る字形の変形を調査する目的で 1 字種について 2 文字づつ収集した．但し，同じ文字を続けて 2 回筆記するのではなく，全字種を 1 回ずつ書き終えた後に 2 回目を筆記するようにした．1 回目と 2 回目のデータを区別する為に，それぞれ ϵ_1 セット，ϵ_2 セットと呼ぶ．

データセット中の文字の例を図 A.1 に示す．走り書き文字の特徴である画の連結や前後画方向への湾曲が見られる．また，この図では見れないが，画の連結箇所で特徴的な筆圧の変化が観測できる．この他，極端な略字（“口” を一筆で “〇” と書くなど）も見られる．筆記画数と正しい画数との差のヒストグラムを図 A.2 に示す．筆順違いによる筆記画数の増加も見られ，画数差は $-18 \sim 3$ 画まで変動し，画数の正しいものは全体の約 3 割程度である．

A.3.1　走り書き文字データの筆順の正しいサブセット

走り書き文字に対する学習と認識性能評価の上で筆順違いの要因を考慮しなくても済むように筆順の正しいサブセットを構築した [36]．既報 [3] の筆順チェックは手本データと筆記画数が異なる場合には筆順判定ができない．そこで，辞書に登録されている当該文字のストロークラベルから生成し得る全ての組み合わせのストローク列の中から文字データの出力尤度を最大にするストローク列を Viterbi 探索する．例えば “三 : A 6 a 6 A” という定義であれば，“A u A u a”, “A u a u A”, “a u A u A” の 3 通りである．尚，ペンアップの方向は筆順によって変わるので，u は任意のペンアップモデルである．

速度・方向特徴量 (r, θ)，連続分布モデルに α セット [3] の 96 筆者によって学習した 2 混合正規分布型を用い，ビーム幅（枝刈り候補数） 1,000 で探索した結果，ϵ_1 セットの 26.61% にあたるデータ数となるサブセットが得られ，正しい画数からの変動は $-8 \sim 2$ 画の範囲となった（図 A.2）．サブセット中の文字の例を図 A.1 に示す．