社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

HMMを用いたオフライン手書き単語認識における環境クラスタリングとGMMの同時最適化

浜村　倫行[†,††]　入江　文平[†]　西本　卓也[††]　小野　順貴[††]　嵯峨山茂樹[††]

† 株式会社東芝　〒 183–8511 東京都府中市東芝町 1
†† 東京大学大学院情報理工学系研究科　〒 113-8656 東京都文京区本郷 7-3-1
E-mail: †tomoyuki.hamamura@toshiba.co.jp

あらまし　音声認識で広く使われている環境依存 HMM には、環境クラスタリングと tied-mixture の二つのアプローチがあり、環境クラスタリングの方が認識精度が高いことが報告されている。しかし、手書き単語認識の場合、筆記体とブロック体など全く異なる字体が一つの文字カテゴリに混在するため、環境クラスタリングが困難となる。そこで、これを解決する方法として、環境クラスタリングと混合ガウス分布 (GMM) の同時最適化法を提案する。まず環境クラスタリングを EM アルゴリズムで最適化する方法を述べ、更にそれを拡張し GMM の同時最適化を導く。CEDAR データベースを用いた実験により tied-mixture による従来法と比べ最大 24.2%のエラー削減率を確認した。また計算効率同等の条件でも提案法の認識精度が高いことを確認した。

キーワード　手書き単語認識、環境依存 HMM、環境クラスタリング、GMM、EM アルゴリズム

Simultaneous Optimization of Context Clustering and GMM for Offline Handwritten Word Recognition Using HMM

Tomoyuki HAMAMURA[†,††], Bunpei IRIE[†], Takuya NISHIMOTO[††], Nobutaka ONO[††], and Shigeki SAGAYAMA[††]

† TOSHIBA Corp.　1, Toshiba-Cho, Fuchu-Shi, Tokyo, 183–8511, Japan
†† Graduate School of Information Science and Technology, The University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656 Japan
E-mail: †tomoyuki.hamamura@toshiba.co.jp

Abstract　Context-dependent HMM is commonly used in speech recognition. The model can be realized by two ways: context clustering or tied-mixuture. In speech recognition, the former is reported to be more efficient. However, there is some difficulty in applying context clustering to handwritten word recognition, since the distribution of each character is typically a mixture of some different distributions, such as block-printed, cursive, etc. To deal with this problem, a method for concurrent optimization of context clustering and Gaussian Mixture Model (GMM) is proposed in this paper. Optimization of context clustering by EM algorithm is described first, followed by its expansion to concurrent optimization of context clustering and GMM. The recognition rate of the proposed method is higher than the conventional one which exploits tied-mixture with equivalent computational cost. Experimental results showed 24.2% error reduction on CEDAR database, compared with the conventional tied-mixture based method.

Key words　Handwritten word recognition, Context-dependent HMM, Context clustering, GMM, EM algorithm

1. ま え が き

スキャンされた画像から手書き単語を読み取るオフライン手書き単語認識技術は、小切手の金額認識、郵便物の住所認識などで早くから実用化されている重要な技術である [1]。認識手法には様々なものが提案されているが、それらは解析的手法 (analytic approach) と全体的手法 (holistic approach) に大別できる。解析的手法は単語を分割して文字候補を生成し個別の

文字を認識するアプローチであり、全体的手法は単語を分割せず直接認識するアプローチである。両手法は相補的であり、高い性能を達成するには両手法を併用するのがよいとされている [2]。

全体的手法では主に Hidden Markov Model(HMM) が用いられる。HMM は系列の認識全般に用いることができ、音声認識で早くから発展してきている手法である。手書き単語では左から右へ、音声では時刻にそって注目領域をスライドさせながら特徴抽出を行うことで、特徴ベクトルの系列ができる。語彙 (単語辞書、単語リスト) に含まれる各単語のモデルからこの系列が出力される確率 (尤度) を計算することで単語を認識する。単語モデルは文字 (音素) のモデルを連結したものであり、各文字 (音素) モデルのパラメータを学習しておけば任意の単語を認識することができる。

音声認識の音素モデルには、triphone と呼ばれる、先行・後続音素別の音素モデルが広く用いられている。音声波形は先行・後続音素 (これを「環境」と呼ぶ) に応じて異なった歪みを生じるため、環境別のモデル (環境依存モデル) を用いることで性能が向上する。一方、手書き単語認識で環境依存モデルを用いた研究は少なく、主に環境に依存しないモデルが用いられている。

環境依存モデルを用いるとモデル数が大幅に増加するため、モデル一つ当たりの学習データが減少し、過学習に陥る。これを防ぐため、モデル間のパラメータ共有が必要となる。パラメータ共有法には、「環境クラスタリング」と「tied-mixture」の二つのアプローチが存在する。環境クラスタリングでは、環境を分割し、同一クラスタに所属する環境間でパラメータを共有する。クラスタリングは各音素モデルの各状態位置ごとに行われ、状態単位で共有するのが一般的である。また、一般に tied-mixture より精度が高い。一方、tied-mixture は、混合ガウス分布の各ガウス分布 (mixture) を複数の状態間で共有し、混合率のみを変化させる手法である。環境のみ異なる同一音素の同一状態位置に限り共有する方法の性能が高いことが報告されている [3], [4]。少ない mixture 数でも精度の低下が小さいため、一般に処理効率を向上したい場合に用いられる。

環境クラスタリングには、認識対象固有の先見知識を用いるものと、用いないものがある。音声認識で広く用いられている Tree-Based Clustering は前者である。前者は認識対象に対する専門知識を必要とし、また対象が変わると同じ知識を用いることができない問題がある。

上述の通り環境クラスタリングは精度が高く、手書き単語認識への適用が期待される。しかし、これまでのところ、先見知識なしでの環境クラスタリングによる認識精度向上は報告されていない。これは、一つの文字に大きく異なる字形 (ブロック体と筆記体など) が存在するなど、同一カテゴリ内の変動が環境による変動を大きく上回っており、環境クラスタリングが有効に機能しないためと考えられる。

そこで本論文では、同一カテゴリ内の変動が大きい場合にも有効な環境クラスタリング手法を提案する。提案法では、環境クラスタリング時に、同一カテゴリ内の変動の推定に相当する混合ガウス分布モデル (GMM) のパラメータ推定も同時に行う。両者を同じ尤度基準で最適化することで、比較的小さな変動である環境の違いを捉えることができる。以下、2. で関連研究を紹介し、3. で従来法の問題点と提案法のアイデアを説明し、パラメータ更新式を導出する。4. では手書き単語認識実験により提案法の有効性を示す。

2. 関 連 研 究

オフライン手書き単語認識に環境依存モデルを用いた研究は少なく、筆者の知る範囲では以下の 5 文献のみである。Fink らは、アルファベット固有の知識 (アセンダ、ディセンダの有無など) を用いて環境を 5 クラスタに分類し用いることで、環境依存なしでのエラー率 24% を 22.5%に低減した [5]。El-Hajj らはアラビア語にてディセンダ有無によるクラスタ分類を行った [6]。Bianne らは、分割ルールを作成し Tree-Based Clustering を適用することで、環境依存なしのモデルに比べ 18.0%のエラー削減率を達成した [7]。Schussler らは tied-mixture を用いてドイツ語の住所認識を行った [8]。Natarajan らは tied-mixture を用い、mixture 共有の範囲を同一文字内の全ての環境・状態としたもの (CTM) と同一文字内かつ同一状態位置の全ての環境としたもの (STM) を比較し、STM の性能が高いことを示した [4]。

上記 5 文献のうち、はじめの 3 文献は認識対象固有の先見知識を用いて環境をクラスタリングする手法であり、残り 2 文献は先見知識なしで tied-mixture を用いる手法である。Fink らの実験では、先見知識なしで環境クラスタリングを行うとエラー率が 34.0%に増加したと報告されている。

3. 環境クラスタリングと GMM の同時最適化

本節では、音声認識で成功している環境クラスタリングが手書き文字認識で有効に機能しない理由を考察し、それを克服する方法として環境クラスタリングと GMM の同時最適化を提案する。

3.1 従来法の問題点

従来の環境クラスタリングを用いたモデル学習の流れは以下の通りである。

step1　各音素の各状態位置ごとに、何らかのクラスタリング手法を用いて環境を分割

step2　同一クラスタに属する環境は状態単位でパラメータ共有

step3　各状態の混合数を増加

図 1(a) は、ある音素 (r) のある状態位置に対応する特徴ベクトルデータの、先行音素環境別 (a-、i-、u-、e-、o-) の分布例である。状態と特徴ベクトルデータの対応付けは Baum-Welch アルゴリズムや Viterbi アルゴリズムにより行うことができる。図 1(b) が step1 の結果であり、(i-r、e-r) と (a-r、u-r、o-r) の 2 つのクラスタに分割されている。図 1(c) が step3 の結果であり、各クラスタが混合数 2 の GMM で分布推定されている。このように、環境による変動が比較的大きい場合、従来の学習ステップが有効に働くと考えられる。

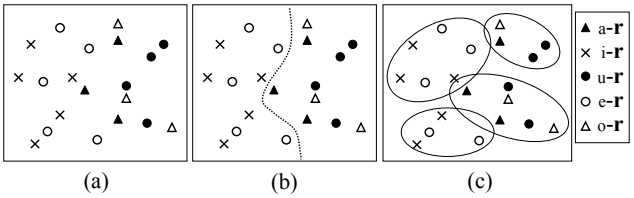

図 1　音素 r の先行音素環境別分布の例

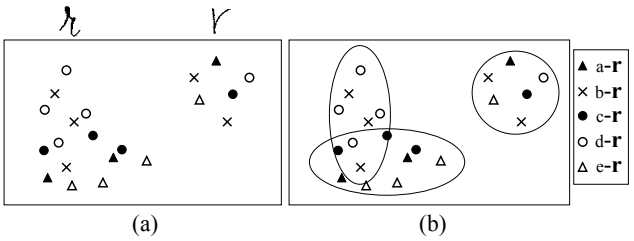

図 2　文字 r の左隣文字環境別分布の例

ところが文字の場合、同一文字に異なる字形が存在し、字形による変動が環境による変動を大きく上回ることがある。図 2(a) は文字 r の特徴ベクトルデータの、左隣の文字環境別 (a-、b-、c-、d-、e-) の分布例である。文字 r は、筆記体とブロック体で字形が大きく異なるため、分布は図のように環境によらず大きく二分され、環境の影響は二分された各々の中での小さな変動になると考えられる。このようなデータに対し上述の step1 を施しても適切なクラスタリングができず、これが精度向上を妨げていると予想される。

3.2　解決のアイデア

3.1 節の通り、従来法は異なる字体が混在したままクラスタリングしていることに問題があった。そこで、step3 まで含めて考え、最終的にどうなるのが望ましいかを考察する。

図 2(a) にて a-r を考える。左に分布している筆記体の r はパターン数が多いため、a-r の近くに分布している c-r、e-r とクラスタを形成すれば過学習が防げるとする。一方、ブロック体の r はパターン数が少ないため、全環境をまとめてようやく過学習を防げるとする。すると、図 2(b) のように、筆記体は 2 つに分割し、ブロック体は 1 つにまとめ、各々から一つずつ選んで a-r〜e-r のモデルを構成するのが望ましいと考えられる。これを少し一般化すると、全部で G 個のガウス分布を使い、そのうちの M 個を選んで一つの環境を表す、と表現できる。これが提案法のアイデアである。上記例は $G=3, M=2$ の場合に相当する。$M=1$ の場合は従来の環境クラスタリングに相当しており、提案法はその拡張と見ることができる。また、$M=G$ の場合は tied-mixture に相当している(注1)。

各環境の選択する M 個の組合せと、各ガウス分布のパラメータは、同一の基準である尤度最大化により決定するのが望ましい。GMM のパラメータは、EM アルゴリズムで最適化できることが広く知られているが [9]、環境クラスタリングを尤度最大化で決定するのは難しい問題である。例えばアルファベットの大文字小文字を想定すると、環境は 52 通り存在する。これを二つのクラスタに分割することを考えると、その組合せは $(2^{52}-2)/2 \quad 2.3\times10^{15}$ 通りにも上るため、これら全ての尤度を計算し最大となる組合せを選択するのは実質的に不可能である。そこで従来は、先見知識に基づいた分割ルールを複数用意し、尤度最大となる分割を選ぶ方法 [7]、全環境のデータで推定した GMM を用い各環境を最も尤度の高いガウス分布に所属させる方法 [10] などが用いられてきた。

我々はこの環境クラスタリングの最適化問題を、EM アルゴリズムを用いて解く方法を提案する。これは上述の通り $M=1$ の場合に相当する。更にそれを $M \geqq 2$ に拡張することで、環境クラスタリング (M 個の組合せの選択) と GMM を同時に最適化する方法を提案する。

3.3　EM アルゴリズムによる環境クラスタリング

クラスタリングの対象が環境ではなく個々のデータであれば、EM アルゴリズムを用いて混合ガウス分布をあてはめ、各データをいずれかのガウス分布に属させることで達成できる。これは、事前確率 a_m で第 m 番目のガウス分布 $\mathcal{N}(\mathbf{x};\mu_m,\Sigma_m)$ (μ_m:平均、Σ_m:共分散行列) を選択し、そのガウス分布からデータが出力される、とするモデルを考え、観測された全データ $\mathbf{X}$ の尤度を最大化するようパラメータ a_m,μ_m,Σ_m を決定することと等価である。図 3(a) にその模式図を示す。

環境のクラスタリングにこの考え方を応用する。今、環境 l ($l=1,...,L$) に属するデータを $\mathbf{x}_{li}(i=1,...,N_l, N_l$:環境 l のデータ数) とする。図 3(a) のモデルを、データごとではなく環境 l ごとに事前確率 a_m でガウス分布 m を選択し、そこから環境 l に属する全データ $\mathbf{x}_{l1},...,\mathbf{x}_{lN_l}$ が出力されるモデルであると考えれば、個々のデータのクラスタリングと同様に環境のクラスタリングを行うことができると考えられる。モデルの各パラメータは、全環境の全データ $\mathbf{X}$ の尤度 $p(\mathbf{X}|\Theta)$(Θ:全パラメータ) を最大化するよう決定する。この最大化は、EM アルゴリズムを用いて行うことができる。パラメータの更新式は、以下の通り、混合ガウス分布あてはめによる個々のデータのクラスタリングの場合と類似した式展開により導出することができる。

隠れ変数 y_l を、環境 l が選択したガウス分布の番号とする。$y_l \in \{1,...,G\}$ である。全隠れ変数を $\mathbf{Y}=(y_1,...,y_L)$ とする。EM アルゴリズムでは、パラメータ Θ を、以下の式で与えられる Q 関数を最大化するパラメータ $\widehat{\Theta}$ に繰り返し更新することで尤度を増加させる。

$$Q=\sum_{\mathbf{Y}} p(\mathbf{Y}|\mathbf{X},\Theta)\log p(\mathbf{X},\mathbf{Y}|\widehat{\Theta}) \tag{1}$$

式 (1) を展開し整理すると、以下の式が導かれる。

$$Q=\sum_{l=1}^{L}\sum_{m=1}^{G} Y_{lm}\log\widehat{a}_m+\sum_{l=1}^{L}\sum_{m=1}^{G}\sum_{i=1}^{N_l} Y_{lm}\log\mathcal{N}(\mathbf{x}_{li};\widehat{\mu}_m,\widehat{\Sigma}_m) \tag{2}$$

ただし、$\widehat{a}_m,\widehat{\mu}_m,\widehat{\Sigma}_m$ は更新後の各パラメータである。また Y_{lm} は環境 l がガウス分布 m を選択する事後確率 $P(y_l=m|\mathbf{X},\Theta)$ であり、

$$Y_{lm}=\frac{a_m\prod_{i=1}^{N_l}\mathcal{N}(\mathbf{x}_{li};\mu_m,\Sigma_m)}{\sum_{m'=1}^{G}a_{m'}\prod_{i=1}^{N_l}\mathcal{N}(\mathbf{x}_{li};\mu_{m'},\Sigma_{m'})} \tag{3}$$

(注1)：モデルは同じだが学習法が違うため、両者の結果は異なる。

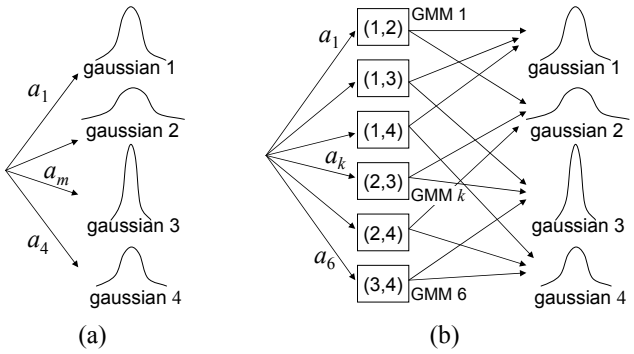

図 3　(a) 混合ガウス分布モデルの模式図 (b) 提案する環境クラスタリングモデルの模式図

として計算される。式 (2) を制約条件 $\sum_{m=1}^{G} \widehat{a}_m = 1$ のもとで最大化するには、ラグランジュの未定乗数法を用い、$J = Q - \lambda(\sum_{m=1}^{G} \widehat{a}_m - 1)$ とし、$\frac{\partial J}{\partial \widehat{\Theta}} = 0$ を解けばよい。その結果、$\widehat{a}_m, \widehat{\mu}_m, \widehat{\Sigma}_m$ は以下の通り導かれる。

$$\widehat{a}_m = \frac{\sum_{l=1}^{L} Y_{lm}}{L}, \quad \widehat{\mu}_m = \frac{\sum_{l=1}^{L}\sum_{i=1}^{N_l} Y_{lm}\mathbf{x}_{li}}{\sum_{l=1}^{L}\sum_{i=1}^{N_l} Y_{lm}} \tag{4}$$

$$\widehat{\Sigma}_m = \frac{\sum_{l=1}^{L}\sum_{i=1}^{N_l} Y_{lm}(\mathbf{x}_{li}-\widehat{\mu}_m)(\mathbf{x}_{li}-\widehat{\mu}_m)^T}{\sum_{l=1}^{L}\sum_{i=1}^{N_l} Y_{lm}} \tag{5}$$

$\widehat{\mu}_m, \widehat{\Sigma}_m$ の分母は $\sum_l N_l Y_{lm}$ とも変形できる。式 (3) が E-step、式 (4)(5) が M-step であり、両 step を交互に繰り返すことで尤度を増加させる。

環境クラスタリングは、学習後のパラメータを用いて、環境 l を事後確率 Y_{lm} が最大となるガウス分布 m に所属させることで達成される。

3.4　GMM の同時最適化

3.3 節を $M \geqq 2$ に拡張する。G 個のガウス分布から M 個を選び GMM を構成する組合せは ${}_GC_M$ 通り存在する。各環境 l はこれらの中から事前確率 a_k で k 番目の GMM を選択すると考える。更に環境 l に所属する各データ $\mathbf{x}_{li}$ が、各々独立に GMM 内のガウス分布を選択すると考える。図 3(b) に $G = 4, M = 2$ の例を示す。${}_4C_2 = 6$ 通りの GMM が構成されている。3.3 節と同様に、全データ $\mathbf{X}$ の尤度 $p(\mathbf{X}|\Theta)$ を最大化するパラメータ Θ を EM アルゴリズムにより推定する。

GMM の総数を K とする。$K = {}_GC_M$ である。各 GMM k を構成するガウス分布の番号の集合を C_k とする。例えば図 3(b) の GMM 6 の場合 $C_6 = \{3, 4\}$ となる。GMM k におけるガウス分布 m の混合率を b_{km} とする。GMM k は $\sum_{m\in C_k} b_{km}\mathcal{N}(\mathbf{x}; \mu_m, \Sigma_m)$ で表されることになる。隠れ変数 z_l を環境 l が選択した GMM 番号とする。$z_l \in \{1, ..., K\}$ である。隠れ変数 v_{li} を、データ $\mathbf{x}_{li}$ が選択したガウス分布番号とする。$v_{li} \in C_{z_l}$ である。全隠れ変数を $\mathbf{Z}$ と表記する。$\mathbf{Z}$ には $z_l\ (\forall l), v_{li}\ (\forall l, i)$ が含まれることになる。その他の記号は 3.3 節と同じものを用いる。

Q 関数を展開し整理すると、以下の通り導かれる。

$$Q = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta)\log p(\mathbf{X}, \mathbf{Z}|\widehat{\Theta}) \tag{6}$$

$$= \sum_{l=1}^{L}\sum_{k=1}^{K} Z_{lk}\log\widehat{a}_k + \sum_{l=1}^{L}\sum_{k=1}^{K}\sum_{i=1}^{N_l}\sum_{m\in C_k} Z_{lk}V_{likm}\log\widehat{b}_{km}$$
$$+ \sum_{l=1}^{L}\sum_{k=1}^{K}\sum_{i=1}^{N_l}\sum_{m\in C_k} Z_{lk}V_{likm}\log\mathcal{N}(\mathbf{x}_{li}; \widehat{\mu}_m, \widehat{\Sigma}_m) \tag{7}$$

ただし、$\widehat{a}_k, \widehat{b}_{km}, \widehat{\mu}_m, \widehat{\Sigma}_m$ は更新後の各パラメータである。また、Z_{lk} は環境 l が GMM k を選択する事後確率 $P(z_l = k|\mathbf{X}, \Theta)$、$V_{likm}$ は環境 l が GMM k を選択した条件下でデータ $\mathbf{x}_{li}$ がガウス分布 m を選択する事後確率 $P(v_{li} = m|z_l = k, \mathbf{X}, \Theta)$ であり、それぞれ

$$Z_{lk} = \frac{a_k\prod_{i=1}^{N_l}\sum_{m\in C_k} b_{km}\mathcal{N}(\mathbf{x}_{li}; \mu_m, \Sigma_m)}{\sum_{k'=1}^{K}\left\{a_{k'}\prod_{i=1}^{N_l}\sum_{m\in C_{k'}} b_{k'm}\mathcal{N}(\mathbf{x}_{li}; \mu_m, \Sigma_m)\right\}} \tag{8}$$

$$V_{likm} = \frac{b_{km}\mathcal{N}(\mathbf{x}_{li}; \mu_m, \Sigma_m)}{\sum_{m'\in C_k} b_{km'}\mathcal{N}(\mathbf{x}_{li}; \mu_{m'}, \Sigma_{m'})} \tag{9}$$

として計算される。

制約条件は $\sum_{k=1}^{K}\widehat{a}_k = 1$ と $\sum_{m\in C_k}\widehat{b}_{km} = 1\ (\forall k)$ であるため、ラグランジュの未定乗数法により

$$J = Q - \lambda_0\left(\sum_{k=1}^{K}\widehat{a}_k - 1\right) - \sum_{k=1}^{K}\lambda_k\left(\sum_{m\in C_k}\widehat{b}_{km} - 1\right) \tag{10}$$

を $\frac{\partial J}{\partial \widehat{\Theta}} = 0$ として解くことでパラメータ $\widehat{\Theta}$ が最大化される。その結果、$\widehat{\mu}_m, \widehat{\Sigma}_m$ は 3.3 節と同様に以下の通り重み付き平均、重み付き共分散行列の形で導かれる。

$$\widehat{\mu}_m = \frac{\sum_{l=1}^{L}\sum_{i=1}^{N_l}\sum_{k=1}^{K} Z_{lk}V_{likm}\mathbf{x}_{li}}{\sum_{l=1}^{L}\sum_{i=1}^{N_l}\sum_{k=1}^{K} Z_{lk}V_{likm}} \tag{11}$$

$$\widehat{\Sigma}_m = \frac{\sum_{l=1}^{L}\sum_{i=1}^{N_l}\sum_{k=1}^{K} Z_{lk}V_{likm}(\mathbf{x}_{li}-\widehat{\mu}_m)(\mathbf{x}_{li}-\widehat{\mu}_m)^T}{\sum_{l=1}^{L}\sum_{i=1}^{N_l}\sum_{k=1}^{K} Z_{lk}V_{likm}} \tag{12}$$

また、$\widehat{a}_k, \widehat{b}_{km}$ は以下の通り導かれる。

$$\widehat{a}_k = \frac{\sum_{l=1}^{L} Z_{lk}}{L}, \quad \widehat{b}_{km} = \frac{\sum_{l=1}^{L}\sum_{i=1}^{N_l} Z_{lk}V_{likm}}{\sum_{l=1}^{L} N_l Z_{lk}} \tag{13}$$

式 (8)(9) が E-step、式 (11)〜(13) が M-step となる。

学習後、各環境 l の出力確率分布 $p_l(\mathbf{x})$ には事後確率 Z_{lk} が最大となる GMM k を選んでもよいが、ここでは全 GMM の事後確率 Z_{lk} が求まっているため、k で期待値を取り予測分布 [9] を計算することで精度を上げる。また、b_{km} は全環境共通の値であり GMM k の混合率に用いるのは不適切であるため、通常の GMM 推定と同様に以下の値を用いる。

$$w_{lkm} = \frac{1}{N_l}\sum_{i=1}^{N_l} V_{likm} \tag{14}$$

以上より $p_l(\mathbf{x})$ は以下の通り表される。

$$p_l(\mathbf{x}) = \sum_{k=1}^{K} Z_{lk}\sum_{m\in C_k} w_{lkm}\mathcal{N}(\mathbf{x}; \mu_m, \Sigma_m) \tag{15}$$

式 (15) は、展開すると環境 l に依存しない同じ G 個のガウス分布の和となり、tied-mixture と全く同じ形となっている。学

習法の違いにより混合率に違いが表れ、性能に差が出ることが期待される。

4. 実験検証

提案法の有効性を検証するため、CEDAR データベース [11] を用いた米国都市名の手書き単語認識実験を行った。データベース中の学習データ 3670 単語、テストデータ 377 単語のうち、上下の行から文字が入り込んでいるもの、下線の引かれているものなどを除き、学習 3213 単語、テスト 352 単語を準備した(注2)。前処理として Ding らのスラント補正 [12] を施した。サイズ正規化、スキュー補正は行っていない。特徴抽出には LGH 特徴 [13] を用いた。大文字と小文字は区別した。各モデルには 10 状態の left-to-right モデルを用いた。認識に用いる語彙は、テストデータのラベルに学習データのラベルを追加し 1000 単語とした。語彙サイズは 10 から 1000 まで変化させ実験した。1000 以外の実験では、語彙を動的に変化させ、必ず正解単語が含まれるようにした(注3)。

比較検証のため、以下の 3 つのモデルを作成した。

CIM　環境に依存しないモデル。混合数を 12 まで 1 ずつ増加させ学習した。

STM　環境に依存した、状態単位の tied-mixture モデル [4]。本実験では環境依存を左右どちらか一方のみとし、10 状態のうち前 5 状態は左隣の文字、後ろ 5 状態は右隣の文字の環境下にあるものとした(注4)。CIM で得られたガウス分布のパラメータを初期値として学習した。

提案法　提案する環境依存モデル。STM と同じ環境依存設定とした。CIM のパラメータを初期値として学習した。

図 4 は語彙サイズ、混合数 (ガウス分布数) を変化させた時の CIM のエラー率である。語彙サイズごとのエラーの最小値を太字で示し、また最下段の “min” と書かれた行にも示した。図 5 は STM の実験結果であり、エラー率、エラー削減率 (ERR)、p 値を示している。ERR は、CIM のエラー率 E' と比較した時のエラー率 E の削減割合で、$ERR = \frac{E'-E}{E'}$ として計算される。エラー率 E', E には “min” の値を用いた。p 値は CIM の各ガウス分布数のモデルに対し全て計算し、最大となる値を示した(注5)。図 4 は提案法にて $M = 2$ とした場合の実験結果である。図 5 と同様にエラー率、ERR、p 値を示した。ERR、p 値は CIM と STM それぞれに対する値を示した。危険率 5%以下の値は太字で示した。

(注2)：全都市名データセットのうち、都市名が偏りなく収集されている BD、BS の全画像を用いた。

(注3)：実際にはサイズ 1000 での実験のみ行い、他のサイズでの認識率は計算で求めた。正解単語の順位を r 位 ($1 \leqq r \leqq 1000$) とすると、語彙サイズを s に減らした時に 1 位になる確率は $\frac{{}_{1000-r}C_{s-1}}{{}_{999}C_{s-1}}$ であり、これを全データで平均した。これは 1000 単語中 s 単語を選ぶ組合せ全てに対し実験したことに相当する。有効数字が高まるため、語彙サイズ 200 以下のエラー率は小数点以下 2 位まで示した。

(注4)：本設定は認識対象固有の先見知識に基づくものではない。どのような対象でも本設定は実行可能である。

(注5)：語彙サイズ 1000 での正解単語の順位を用い、その変化量に対し Wilcoxon の符号順位検定を行った。

		語彙サイズ						
		10	20	50	100	200	500	1000
ガウス分布数	1	3.08	4.75	7.41	10.0	13.4	19.3	24.4
	2	2.14	3.46	5.55	7.58	10.3	15.2	19.6
	3	**1.92**	3.15	5.16	7.05	9.59	14.7	20.2
	4	2.07	3.29	5.16	6.69	**8.47**	**12.0**	**15.9**
	5	2.12	3.33	5.23	6.87	8.79	12.4	**15.9**
	6	**1.92**	2.99	4.83	6.55	8.58	12.2	16.2
	7	1.93	**2.96**	**4.73**	**6.44**	8.51	12.2	16.2
	8	2.01	3.05	4.81	6.50	8.67	12.6	16.8
	9	2.00	3.02	4.86	6.66	8.95	13.0	16.8
	10	2.00	3.00	4.74	6.47	8.75	13.0	17.3
	11	2.09	3.11	4.83	6.50	8.70	12.9	17.0
	12	2.12	3.11	4.85	6.56	8.76	12.6	16.2
min		1.92	2.96	4.73	6.44	8.47	12.0	15.9

図 4　CIM におけるエラー率

“min” の値を比較すると、どの語彙サイズにおいても CIM ＞ STM ＞提案法となっている。環境依存モデルによる効果は STM でも認められるが、提案法の方がより効果が高いことが分かる。CIM と比較したエラー削減率は、STM では 3.0%〜6.0%のところ、提案法は 10.1%〜28.8%と大きく上回っている。提案法と STM の直接比較でも、提案法は 6.7%〜24.2%の削減率であり、優位性が認められる。

p 値で見ると、STM では最小でもガウス分布数 7 での 6.4%であり、5%の有意水準では CIM よりエラー率が低いとは言えない。一方提案法では、ガウス分布数 10 にて、対 CIM で 0.4%、対 STM でも 1.1%であり、CIM、STM のいずれに対しても有意にエラー率が低いと言える。

エラー率が最小となるガウス分布数を見ると、CIM では 4〜7 個、STM では 7 個周辺であるのに対し、提案法では 8〜10 個と増えている。環境情報なしではとらえられなかった分布の小さな構造が環境情報を与えることでとらえられるようになり、ガウス分布数が増えたものと考えられる。

逆に、STM で最大性能となるガウス分布数 7 に固定して比較すると、全語彙サイズで提案法のエラー率の方が低い。上記と同じ理由により、同じガウス分布数でもより効率的に分布構造を表現できているものと考えられる。また、計算効率はガウス分布数で決まるため、提案法は tied-mixture と同等の計算効率の高さを保ちつつ認識精度がより高いと言える。

図 7 は、提案法にてガウス分布数を 8 に固定し、M の値を変化させた時のエラー率である。各語彙サイズにてエラー率の最小値を太字で示した。どの語彙サイズにおいても $M = 2$ でエラー率最小となっている。本実験条件では、各環境別モデルを 2 つのガウス分布でモデル化するのが妥当であることが分かる。これは、3.1 節での考察の通り、字形で大きく二分されているためと考えられる。ただし、学習データが多くなれば、より細かい分布形状が推定できるため、最適値は $M = 3$ 以上にシフトしていくと予想される。

5. むすび

本論文では、環境依存 HMM モデルを用いた手書き単語認識において効果的なモデル学習法を提案した。手書き文字では、同一カテゴリ内に大きく異なる字体が存在するため、従来の学

		語彙サイズ							p値
		10	20	50	100	200	500	1000	CIM
ガウス分布数	5	2.01	3.22	5.12	6.72	8.52	11.9	**15.1**	79.8
	6	1.84	2.94	4.80	6.50	8.36	**11.5**	**15.1**	24.4
	7	**1.80**	2.84	4.59	**6.25**	**8.16**	**11.5**	**15.1**	6.4
	8	1.85	2.91	4.69	6.37	8.47	12.4	16.5	39.6
	9	1.91	2.91	4.65	6.33	8.41	12.0	15.9	52.6
	10	1.83	**2.82**	**4.54**	**6.25**	8.52	12.8	17.0	62.8
	11	1.95	2.95	4.63	6.26	8.42	12.4	16.5	60.8
	12	1.98	2.96	4.67	6.39	8.60	12.5	16.5	83.2
min		1.80	2.82	4.54	6.25	8.16	11.5	15.1	
ERR	CIM	6.0	4.8	4.1	3.0	3.7	3.7	5.4	

図 5　STM におけるエラー率、ERR、p 値

		語彙サイズ							p値	
		10	20	50	100	200	500	1000	CIM	STM
ガウス分布数	5	1.68	2.76	4.55	6.13	8.00	11.5	15.6	9.5	28.1
	6	1.59	2.74	4.72	6.46	8.45	11.9	15.9	13.7	35.8
	7	1.55	2.60	4.42	6.13	8.13	11.4	14.2	**4.9**	16.7
	8	1.41	2.34	4.07	5.77	7.68	**10.3**	**12.8**	**0.9**	**3.4**
	9	1.44	2.33	4.06	5.87	7.96	11.2	14.2	**1.3**	6.5
	10	**1.36**	**2.24**	**3.90**	**5.60**	**7.61**	10.7	13.6	**0.4**	**1.1**
	11	1.45	2.33	4.02	5.78	7.92	11.5	15.1	**1.2**	**2.8**
	12	1.59	2.50	4.21	6.02	8.40	12.5	16.8	25.2	48.9
min		1.36	2.24	3.90	5.60	7.61	10.3	12.8		
ERR	CIM	28.8	24.5	17.6	12.9	10.1	13.4	19.6		
	STM	24.2	20.7	14.1	10.3	6.7	10.1	15.1		

図 6　提案法におけるエラー率、ERR、p 値

	語彙サイズ						
	10	20	50	100	200	500	1000
M 1	1.67	2.94	5.48	8.08	11.2	16.3	21.3
M 2	**1.41**	**2.34**	**4.07**	**5.77**	**7.68**	**10.3**	**12.8**
M 3	1.73	2.73	4.51	6.25	8.34	12.0	15.9
M 4	1.93	2.99	4.76	6.47	8.54	12.2	16.2
M 5	1.96	3.01	4.78	6.49	8.62	12.5	16.8
M 6	1.99	3.05	4.83	6.55	8.74	12.7	16.8
M 7	1.98	3.02	4.78	6.51	8.70	12.7	16.8
M 8	1.97	3.01	4.74	6.42	8.60	12.7	17.0

図 7　$G = 8$ とし M を変化させた時の提案法のエラー率

習法で用いられている環境クラスタリングは有効に機能しないと考えた。そこで、G 個のガウス分布の中から M 個を選ぶモデルとし、これを EM アルゴリズムにより学習する方法を提案した。まず EM アルゴリズムによる環境クラスタリングを提案し、それを拡張することで上記モデルの学習法を導いた。手書き単語認識実験により、従来法である tied-mixture に対しエラーを最大 24.2%削減した。また、計算効率同等の条件で提案法のエラー率の方が低いことを確認した。

今後の課題としては学習計算の高速化が挙げられる。ガウス分布数 G が増大すると学習時間は指数関数的に増加する。学習データを増やすと最適な G, M の値は増えると考えられ、学習が困難となる。${}_GC_M$ 通りを全て計算するのではなく、値の小さなところは枝刈りするようなアプローチにより解決できるのではないかと考えている。

文　献

[1] F. Camastra and A. Vinciarelli, Machine Learning for Audio, Image and Video Analysis, Springer, 2008.

[2] R. Plamondon and S.N. Srihari, “On-line and off-line handwriting recognition: A comprehensive survey,” IEEE Trans. PAMI, vol.22, no.1, pp.63–84, Jan. 2000.

[3] 李晃伸, 河原達也, 武田一哉, 鹿野清宏, “Phonetic tied-mixture モデルを用いた大語彙連続音声認識,” 信学論 (D-II), vol.J83-D-II, no.12, pp.2517–2525, Dec. 2000.

[4] P. Natarajan, S. Saleem, R. Prasad, E. MacRostie, and K. Subramanian, “Multi-lingual offline handwriting recognition using hidden markov models: A script-independent approach,” Springer Book Chapter on Arabic and Chinese Handwriting Recognition, vol.4768, pp.231–250, Mar. 2008.

[5] G.A. Fink and T. Plotz, “On the use of context-dependent modeling units for HMM-based offline handwriting recognition,” Proc. 9th ICDAR, pp.729–733, Sep. 2007.

[6] R. El-Hajj, C. Mokbel, and L.L-Sulem, “Recognition of arabic handwritten words using contextual character models,” Proc. SPIE, vol.6815, pp.681503–681503–9, Jan. 2008.

[7] A.-L. Bianne, C. Kermorvant, and L. L.-Sulem, “Context-dependent HMM modeling using tree-based clustering for the recognition of handwritten words,” Proc. SPIE, vol.7534, p.75340I, Jan. 2010.

[8] M. Schussler and H. Niemann, “A HMM-based system for recognition of handwritten address words,” Proc. 6th IWFHR, pp.505–514, Aug. 1998.

[9] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[10] 鷹見淳一, 嵯峨山茂樹, “逐次状態分割法による隠れマルコフ網の自動生成,” 信学論 (D-II), vol.J76-D-II, no.10, pp.2155–2164, Oct. 1993.

[11] J.J. Hull, “A database for handwritten text recognition research,” IEEE Trans. PAMI, vol.16, no.5, pp.550–554, May 1994.

[12] Y. Ding, F. Kimura, Y. Miyake, and M. Shridhar, “Accuracy improvement of slant estimation for handwritten words,” Proc. 15th ICPR, vol.4, pp.527–530, Sep. 2000.

[13] J.A. Rodriguez and F. Perronnin, “Local gradient histogram features for word spotting in unconstrained handwritten documents,” Proc. 1st ICFHR, pp.7–12, Aug. 2008.