4T-8

HMM を用いた毛筆調レンダリングに関する研究 第二報 -HMM クラスタリングによる実行時間の短縮-

岡部 雄太 † 齋藤 豪 † 高橋 裕樹 † 中嶋 正之 †‡

† 東京工業大学 大学院 情報理工学研究科 ‡ 国立情報学研究所

1 はじめに

我々は、ある入力曲線を自動的に毛筆ストロークとしてレンダリングすることを目指している。レンダリングするために必要なパラメータは多く、それらを入力線のみから決定することは単純ではない。そこで本研究では、機械学習の手法の一つである連続 HMM(Hidden Markov Model) を用い、与えられた曲線の軌跡にフットプリントの系列を推定し、それらを滑らかにつなぐことによって、毛筆ストロークを作成する。

以前の手法 [3] では一つのフットプリント系列の推定に 1300 個以上の HMM を用いているが、それらの中には類似した HMM が多く存在している。そこで本稿では、HMM クラスタリングを行ない、類似した HMM をまとめ、HMM 数を削減し、実行時間を短縮する。

2 小区間毎の HMM の学習

HMM とは、状態遷移と各状態での観測ベクトルが確率的に選択される有限状態オートマトンである。HMM は、観測ベクトルの型によって離散 HMM と連続 HMM にわけることができる [1]。我々は連続値の観測ベクトルを扱うため、連続 HMM を用いる。

一つの HMM で曲線全体をモデル化する場合、様々なタイプの曲線をモデル化するためには、多くの状態が必要になる。HMM の中に多くの状態が存在するとさらに多くの状態遷移が起こるため、十分な精度の状態遷移確率を計算するためには、非常に長い計算時間が必要となり、さらには計算が収束しないという問題が生じる。そのため、波線や折れ線のような曲率が大きく変化するような曲線に対しても適切な毛筆ストロークが生成されるために、毛筆ストロークを小区間毎に区切り、その小区間毎に連続 HMM による学習が必要である。

連続 HMM の学習に用いる訓練データは、実際の毛筆で描かれるストロークから取得する。直線と右曲がり、左曲がり、波線、折れ線の曲線をそれぞれ直筆と側筆の 2 通りのそれぞれ 8 ストロークずつ撮影し、得られた画像からフットプリントの特徴量を参考文献 [3] の方法で測定する。それぞれの画像からフットプリントの高さと横幅、さらにストロークの変位方向の角度とフットプリントの向きの 4 次元ベクトルが測定され、それらを 10 点毎の小区間にまとめて訓練データとする。訓練データに対して Baum-Welch アルゴリズム [1] を適用し、連続 HMM を構築する。それぞれのストロークに対して、小区間を一点ずつずらしながら訓練データを作成し、連続 HMM を学習させることにより、合計 1314 個の連続 HMM が作成される。

しかし、隣り合う小区間の訓練データは非常に似ているため、その訓練データから学習すると非常に類似した HMM が出来上がる。そこで、このような類似した HMM をクラスタリングにより一つにまとめることができれば、フットプリントの推定能力を保ちつつフットプリント列の推定に用いる HMM を削減でき、実行時間を短縮することができる。

3 HMM のクラスタリング

HMM をクラスタリングするためには、HMM 同士の近さを定義する必要がある。本稿では、参考文献 [2] に従い、HMMΛ_i と Λ_j の近さ $s(\Lambda_i, \Lambda_j)$ を以下のように定義する。

$$s(\Lambda_i, \Lambda_j) = \sum_{\boldsymbol{o} \in \boldsymbol{O}} P_i(\boldsymbol{o}) P_j(\boldsymbol{o}) \tag{1}$$

ただし、$\boldsymbol{o}$ は 10 点の連続した訓練データを表し、$\boldsymbol{O}$ は HMM の構築に用いた訓練データの集合を表す。また、$P_i(\boldsymbol{o})$,$P_j(\boldsymbol{o})$ はそれぞれ HMMΛ_i、Λ_j において $\boldsymbol{o}$ を観測する確率を表している。つまり、学習に用いたすべての観測ベクトル列に対して、ある観測ベクトル列を両方の HMM で観測する確率を計算し、その総和を求めている。HMM がある観測ベクトル列 $\boldsymbol{o}$ を受理する確率 $P_i(\boldsymbol{o})$ は forward アルゴリズム [1] により計算することができる。

しかし、本手法における訓練データの角度は角度変化で定義されているので、HMM をクラスタリングするときには、フットプリントの初期方向の近さも考慮する必要がある。そこで、HMM の近さを以下のように定義し直す。

$$s(\Lambda_i, \Lambda_j) = \exp\left(\frac{-(\phi_i^0 - \phi_j^0)^2}{2\sigma^2}\right) \sum_{\boldsymbol{o} \in \boldsymbol{O}} P_i(\boldsymbol{o}) P_j(\boldsymbol{o}) \tag{2}$$

ただし、ϕ_i^0、ϕ_j^0 はそれぞれ HMMΛ_i、Λ_j のフットプリントの初期方向を示しており、σ は初期角度の差をどの程度許容するかを表すパラメータである。

この近さ $s(\Lambda_i, \Lambda_j)$ により HMM をクラスタリングする。クラスタ間の近さは最長距離法により計算する。すなわち、クラスタ C_1 と C_2 の近さ $S(C_1, C_2)$ は、

$$S(C_1, C_2) = \min_{\boldsymbol{\Lambda}_i \in C_1, \boldsymbol{\Lambda}_j \in C_2} s(\Lambda_i, \Lambda_j) \tag{3}$$

A Study on Oriental Brush Rendering using HMMs 2nd Report - Speeding-up by HMM Clustering-

Yuta OKABE †, Suguru SAITO †, Hiroki TAKAHASHI †and Masayuki NAKAJIMA †‡

†Graduate School of Information Science & Engineering, ‡National Institute of Informatics

により計算される。つまり、s は近さであるので最長距離を求めるために最小の s を求めている。

クラスタリングの処理は以下のように行なわれる。まずそれぞれの HMM が一つずつのクラスタに含まれている状態から開始し、S が最も大きい組のクラスタを統合する。この統合の操作をクラスタ数が閾値以下になるまで繰り返す。クラスタの重心に最も近い HMM でそのクラスタ内部に存在する HMM を代表させ、HMM を削減する。クラスタの重心に最も近い HMM には、そのクラスタに含まれる他の HMM との近さの総和が最大となる HMM が選ばれる。

4 HMM の個数の決定

削減された HMM を用いて推定されるフットプリント列と削減前の HMM を用いて推定されるフットプリント列との差を計算し、削減後の HMM のフットプリント推定能力を評価する。フットプリントの差は、推定されたフットプリントの高さと横幅、角度から求まる長方形の四隅の点において、HMM 削減前と削減後で対応する点のユークリッド距離の総和により定義する。

直線や右曲がり、左曲がり、波線、角などの基本ストロークとそれらを組み合わせた複合型ストロークを直筆、側筆で描いた合計 110 本の評価用のストロークを用意し、これらの評価用ストロークにより、HMM 削減前と削減後のフットプリントの差を計算する。様々な HMM の個数に対するフットプリントの差の計算結果を図 1 左に示す。

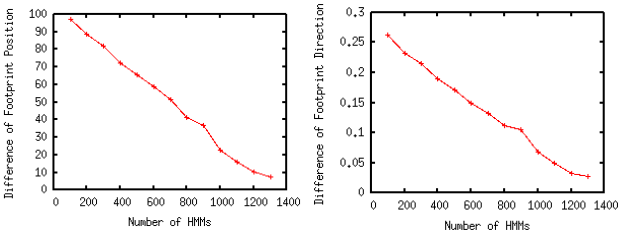

図 1　HMM の個数 (横軸) に対する削減後の HMM の性能 (縦軸)

図 1 左の横軸は削減後の HMM の数を示し、縦軸は HMM 削減前と削減後との推定されるフットプリントの差の総和をフットプリントの総数で割った値である。この結果から、フットプリントの推定能力は使用する HMM の減少と共に線形に低下していくことがわかる。しかしながら、フットプリントの差が 4 隅の点の距離では許容できる大きさがわかりにくいので、適切な HMM の個数を求めることが難しい。そこで、我々はフットプリントの四隅の点に差が生じる本質的な原因はフットプリントの向きが異なっているためであるという仮説を立て、それを検証するために、フットプリントの差をフットプリント方向の角度差と定義し直して同様の計算を行なった。図 1 右に、1 つのフットプリント当りのフットプリント方向の角度差をプロットしたグラフを示す。この結果には図 1 左と非常に似た傾向が表れているため、フットプリントの向きがフットプリントの推定性能に大きく関わっていることが考えられる。そこで、適切な HMM の個数をフットプリントの向きの差により決定することにする。本稿では HMM 削減前と削減後でフットプリントの向きの差が 10 度以下に収まるように HMM の個数を設定する。図 1 右のグラフから HMM が 400 個のときにフットプリントの向きの差が 10 度 ($0.175rad$) 以下に収まっていることがわかる。また実際に、HMM400 個でストロークを描いてみると正しい種類のストロークが選択されていた。これより、HMM の推定能力を保ちつつ HMM を 400 個ぐらいまで削減できることがわかった。

5 レンダリング結果

入力線を小区間に分割し、小区間毎の角度変化に対して、forward アルゴリズム [1] を利用してその角度変化を観測する可能性が最も大きい HMM を削減された HMM の中から選択し、その HMM に対して Viterbi アルゴリズム [1] を適用し最尤状態列を求め、その状態列からフットプリント列を決定する。次に、実際の毛筆によるフットプリント画像を用意し、それらの画像をアフィン変換してフットプリント推定結果に配置し、それらをスプラインで結ぶことによってレンダリングを行なう [3]。また、筆の入りや抜きはストロークの始めと終りでフットプリントの大きさを強制的に変化させることによって実現している [3]。

レンダリング結果は図 2 ようになる。このとき HMM は 1314 個から 400 個に削減されている。

図 2　HMM 削減前 (左) と HMM 削減後 (右)

一部の葉を除いては削減前と削減後ではストロークはほぼ一致しており、一致していない葉のストロークにおいても、描かれるストロークの形状は異なるが描かれるストロークの種類、つまり角のストロークを描く部分は一致している。この結果から、HMM 削減前のストロークの特徴が削減後もそれほど失われていないことがわかる。また、このときの実行時間は、*Opteron 244 processor* を用いて実行した時に、HMM 削減前が 250 秒であったのに対して、削減後は 105 秒に短縮された。

6 おわりに

本稿で提案した HMM クラスタリングにより、削減前の HMM のフットプリント推定性能を保ちつつ実行時間を短縮することができた。

参考文献

[1] Lawrence Rabiner and Biing-Hwang Juang : “Fundamentals of Speech Recognition”, *PTR Prentice-Hall*,1993.

[2] Rune B. Lyngsø; and Christian N. S. Pedersen and Henrik Nielsen：“Metrics and Similarity Measures for Hidden Markov Models”, *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*,pp178–186, 1999.

[3] 岡部雄太, 齋藤豪, 中嶋正之：“HMM を用いた線の毛筆調レンダリングに関する研究”, *VC/GCAD* 合同シンポジウム *2005*,pp81–86, 2005.