

多様な曲率を持つストローク生成のための 毛筆調レンダリングの改良

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

本稿では、筆の軌跡を表す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す。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、機械学習の手法を用いてこの問題の解決を試みる。学習に用いる教師データは、実際の毛筆ストロークを撮影し、その動画に対して処理を行うことにより取得する。その取得したデータを用いて、自然な毛筆ストロークのレンダリングを行うためのモデルを獲得する。本研究で用いる教師データは連続値であり、かつ時系列なデータであることから、学習アルゴリズムとして連続HMM(Hidden Markov Model)を採用した。この連続HMMを利用することによって、与えられた曲線の軌跡に対してフットプリントを割り当て、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。

しかし、従来の手法では波線や折れ線などの曲率が大きく変化するストロークの生成には対応していなかった。そこで本稿では、連続HMMの利用方法および教師データを改良し、曲率が大きく変化するようなストロークに対しても、適切な毛筆ストロークを生成することを可能とする。

2 小区間毎のHMMによるモデル化

HMM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。

このような学習上の問題のために、従来は曲線を直線と右曲がりと左曲がりなどの曲線毎に分類し、その曲線毎にHMMを作成した[2]。しかし、この方法では各曲線毎に作成されるHMMの状態数はせいぜい6、7個であるため、波線のようにフットプリントの向きとストロークの軌跡との組合せが数多く存在するような曲線を再現することは難しかった。

そこで、波線や折れ線のような曲率が大きく変化する

ようなストロークに対しても適切な毛筆ストロークを生成するために、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を割り当て学習させる。

3 教師データの取得

小区間毎の連続HMMを学習させるために必要な教師データ取得方法について述べる。教師データを作成するために、直線と右曲がり、左曲がりの曲線をそれぞれ直筆と側筆の2通りで描いたストロークを描き、さらに波線や折れ線などのストロークをそれぞれ8ストロークずつ撮影した。撮影された毛筆ストロークからのフットプリント特徴量の取得方法は、参考文献[2]と同じであるため、ここでは簡単に説明する。まず、描かれたストロークを1秒間に15フレームずつ撮影し、撮影された画像からフットプリントに掃討する影の部分を取り出す。時刻 t におけるフットプリントから、毛先側の両端点の距離 $d(t)$ 、フットプリントの高さ $h(t)$ と横幅 $w(t)$ 、さらにストロークの移動方向の角度 $\theta(t)$ とフットプリントの向き $\phi(t)$ を計測する。しかし、連続HMMを用いてフットプリント系列を推定するときには、一定距離でサンプリングされたデータを使用するのに対し、撮影画像から得られるベクトルは一定の時間間隔で取得されたものであるため、データの整合性が保たれていない。そこで、一定の時間間隔で取得されたフットプリントの特徴ベクトルを一定距離で再サンプリングする。処理の流れを次に示す。

1. ストロークの軌跡をスプラインで近似
2. スプラインを一定距離でサンプリング
3. サンプリングされた点上でベクトルデータを再計算

ただし、ベクトルデータの再計算は、サンプリングされた点をはさむ2点のベクトルデータを距離に応じて内分する。このようにして再サンプリングされた i 番目のフットプリントの特徴量を $v_p(i) = (d_p(i), h_p(i), w_p(i), \theta_p(i), \phi_p(i))$ で表す。

次に、得られたフットプリントのベクトルデータ小区間毎の教師データを作成する。小区間のサンプリング点の数を s で表し、 k 番目の小区間を Σ_k で表す。小区間 Σ_k の j 番目の教師データ $v_k(j)$ は以下のように定義される。

$$\begin{aligned} v_k(j) &= (d_k(j), h_k(j), w_k(j), \theta_k(j), \phi_k(j)) \\ &= (d_p(k+j), h_p(k+j), w_p(k+j), \theta_p(k+j) - \theta_p(k), \\ &\quad \phi_p(k+j) - \phi_p(k)). \quad (1) \\ &\quad i=0, \dots, s-1 \end{aligned}$$

ただし、 $\theta_k(j)$ と $\phi_k(j)$ はそれぞれ小区間の最初と j 番目の筆の軌跡方向の変化量とフットプリントの向きの変化量を表している。

このような小区間毎の教師データをすべてのストロークから取得する。ただし、フットプリントの向きを表す

An Improvement of Oriental Brush Rendering for the Stroke Generation with Various Curvature.

Yuta OKABE[†], Suguru SAITO[†], Hiroki TAKAHASHI[†] and Masayuki NAKAJIMA^{†‡}

[†]Graduate School of Information Science & Engineering,

[‡]National Institute of Informatics

$\phi_k(j)$ は相対的な値しか持たないので、入力された筆の軌跡から筆のストロークを生成するためには、基準となるフットプリントの向きが必要である。そこで、小区間毎にその小区間の最初のフットプリントの向きを保存している。

4 連続 HMM によるフットプリントの推定

前節で得られた小区間の教師データに対して Baum-Welch アルゴリズムを適用し、連続 HMM の学習を行う。この学習を各々の小区間毎に行ない、小区間と同数の連続 HMM を作成する。HMM の学習には、あらかじめ HMM の状態数を定めておく必要がある。各教師データの分散は小区間毎に異なるため、分散が大きい場合には状態数を多く、分散が小さい場合には状態数を少なくすることにより、各小区間の状態数を適切に表現する。そこで、教師データをマハラノビス距離によって正規化し、正規化された教師データを距離に応じてクラスタリングする。そのときのクラスタ数を状態数として使用した。

次に、作成された複数の連続 HMM を使用して、入力された線の角度変化からフットプリント列を推定する方法を説明する。各小区間に対して Viterbi アルゴリズムによって最尤状態列を定め、その状態列からフットプリント列を決定する [2]。各状態には、教師データ 5 次元の値に相当する学習されたパラメータが割り当てられている。これらのパラメータを利用して入力線の角度変化にしたがって滑らかに変化するフットプリント列を表現するために、次の処理を行う。ここでは説明のために、状態が 0、1、2 の順番に遷移している 3 状態からなる状態列を考える。

1. 状態が 1 と推定されたサンプリング点の中から、角度変化 θ_{input} が状態 1 の角度変化に対応する値と最も近くなる点を求め、その点の番号を K_1 で表す。
2. あるサンプリング点の番号が K_1 以下であれば、そのサンプリング点は状態 0 から状態 1 に遷移している途中であると定める。そうでない場合は、状態 1 から状態 2 に遷移している途中であると定める。
3. ある状態 i からある状態 j に遷移しているとき、フットプリントパラメータ v_e は次のように計算される。状態 i と j のパラメータをそれぞれ v_i 、 v_j で表す。このとき、フットプリントのパラメータは $v_e = v_i + r \times (v_j - v_i)$ で計算する。ただし、 $r = (\theta_{input} - \theta_i) / |\theta_j - \theta_i|$ であり、 θ_i 、 θ_j はそれぞれ状態 i 、 j での学習した角度変化を表す。

このようにして小区間毎にフットプリント列を決定する。小区間を 1 点ずつずらしながら、すべてのサンプリング点に対してフットプリントを推定し、最後に各サンプリング点において、割り当てられたフットプリントを平均する [2]。フットプリントの推定結果を図 1 に示す。

黒い線は筆の軌跡を表し、赤い線分はフットプリントの向きと幅を表している。また、図 1 の (a)、(c) はそれぞれ波線と折れ線の教師データを表して、図 1 の (b)、(d) はそれぞれ波線と折れ線の入力線に対するフットプリントの推定結果を表している。この結果から、これまで難しかった曲率の大きく変化する部分において、教師データに現れている紙と筆の摩擦によって起こるフットプリントの拘束条件を学習できていることがわかる。

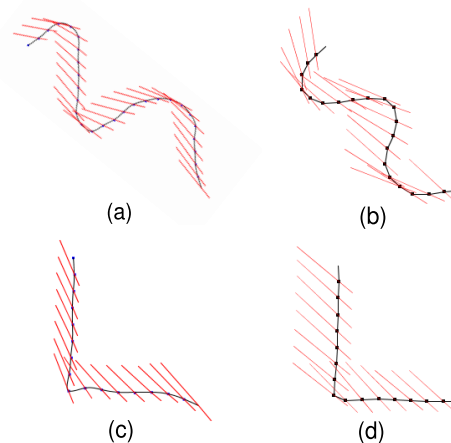


図 1 教師データ (左) とフットプリント推定結果 (右)

5 レンダリング方法

かすれ方の異な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前節での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 [2, 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た。

レンダリング結果は図 2 ようになる。

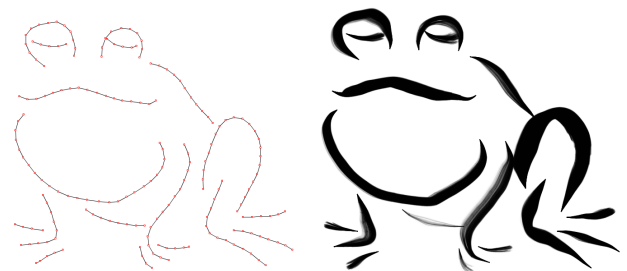


図 2 入力線 (左) と結果画像 (右)

6 おわりに

教師データ及び学習方法を改良することにより、波線や折れ線などの曲率が大きく変化するストロークに対しても紙と筆の摩擦により起こる物理的な拘束条件を満たした毛筆ストロークを生成することが可能になった。

参考文献

- [1] Lawrence Rabiner and Biing-Hwang Juang : “Fundamentals of Speech Recognition ” , PTR Prentice-Hall,1993.
- [2] 岡部雄太, 齋藤豪, 中嶋正之 : “連続 HMM を用いた入力線からの毛筆調レンダリングに関する研究” , 第 115 回グラフィックスと CAD 研究会, pp1-6, 2004.
- [3] 岡部雄太, 前田大介, 齋藤豪, 中嶋正之 : “線の毛筆調レンダリングに関する研究” , 第 20 回 NICO-GRAPH 論文コンテスト, pp47-52, 2004.