

毛筆で筆記可能な仮想書道システムの開発

島田 英之[†] 島田 恭宏[†] 大倉 充[†]

近年、コンピュータにより高品質の毛筆書体をリアルタイムに筆記する研究例が報告されているが、現実の筆記環境と異なる方法で筆記するために、対話性が十分とはいえない。本論文では、筆者らが構築した仮想書道システムについて述べる。ユーザは、3次元位置センサを内蔵した毛筆でスクリーン上に直接筆記し、実際の毛筆、仮想的な半紙や硯など、現実の書道とほぼ同じ作業環境のもとで、仮想的な書道を体験できる。毛筆が触れた場所に正確に筆跡を生成できるようにシステムをキャリブレーションし、実際の毛筆で筆記した筆跡の実測値に基づいて筆跡を生成する。これにより、毛筆が触れた場所に、実際に毛筆で筆記した場合と同じ太さの筆跡を表示できるので、手本をなぞる初歩的な書写練習も可能となった。また、毛筆の空間的な挙動をモニタし、墨の滴りや飛び散りなどの視覚効果も表現した。この結果、年齢を問わず簡単に操作でき、かつ多様な表現が可能なシステムを構築したので報告する。

A Virtual Calligraphy System Drawable with a Chinese Brush

HIDEYUKI SHIMADA,[†] YASUHIRO SHIMADA[†] and MITSURU OHKURA[†]

In recent years several methods to generate oriental calligraphic characters are proposed. However, because of unnatural way for drawing (e.g., drawing on a tablet with a pen) these methods are not enough to draw interactively. In this paper we propose a virtual calligraphy system with a real Chinese brush. While a user draws at a screen by Chinese brush with build-in six-degree-of-freedom sensor, the calligraphic handwriting is displayed at same place of drawing. Our system is calibrated by using actual survey of real handwriting and can display handwriting on a screen accurately, therefore the user can practice calligraphy. And also we expressed the situation of dripping and scattering effects of India ink from a brush. As a result any people can perform calligraphy easily with various expression by our system.

1. はじめに

毛筆は、他の筆記用具にはない豊かな表現の世界を持つ。たとえば、最近、北京で「地書」と呼ばれるパフォーマンスがみられるという¹⁾。これは、大筆を水に浸し、公園の石畳に漢詩などを書き連ねるものであるが、毛筆の世界が、単なる筆記用具の枠にとどまらず、癒し、娯楽、芸術など、文化として奥深い広がりを持つことを表す一例といえるだろう。

幸い、わが国では、毛筆による書写が義務教育に導入されているので、だれもが書道を経験し、毛筆の基本的な使い方を知っている。しかし、実用に供するにはある程度の熟達を要するうえに、取扱いも煩雑なので、日常生活で毛筆を手にする機会は少ない。

一方、旧来から利用されてきた表現手段に現代の技術を導入し、新たな価値を創成した例は多い。電子

メールやデジタルカメラがその例である。そして、古代より現代に至るまで連綿と使われ続けている毛筆もまた、デジタル技術との融合によって新たな表現力を獲得できるポテンシャルを持つ。

以上のような背景のもと、コンピュータによる毛筆書体生成のための研究が行われてきた。当初は、オフラインでの毛筆書体生成^{2),3)}が主流であったが、近年、筆記にともなう毛筆の挙動を物理的にシミュレートし、リアルタイムに毛筆風の筆跡を得る方法^{4),5)}が提案された。これらの手法ではいずれも、高品質の筆跡を生成することに成功しているが、タブレットに対してペンで筆記する、あるいは、実際の毛筆で筆記するが筆跡は別のディスプレイに表示する方法をとっている。つまり、現実の筆記環境とは異なるために、刻々と状態が変化する柔らかな毛筆と、視覚、力覚の連携を緊密に保てないという問題が存在する。

そこで筆者らは、実際の毛筆で筆記し、筆記した位置に筆跡を投影する形式のインタフェースについて提案してきた^{6),7)}。この手法では、実際の毛筆でスク

[†] 岡山理科大学工学部情報工学科
Department of Information & Computer Engineering,
Okayama University of Science

リーン上に直接筆記でき、筆記した場所にリアルタイムに筆跡が表示される。毛筆側にセンサが内蔵されているので、原理的には、特殊なタブレットなど、筆記面積に応じた大きさの入力デバイスを要しない。

本論文では、市販の毛筆を入力デバイスとして用いた仮想書道システムについて述べる。ユーザは、本システムによって、実際の毛筆、仮想的な半紙と硯など、現実の書道とほぼ同じ作業環境のもとで、仮想的な書道を体験できる。筆記した場所に正確に筆跡を表示するために、毛筆のキャリブレーションを行った。また、実際の毛筆で筆記した筆跡の実測データに基づき、毛筆の角度や筆圧に応じて筆跡の太さを決定している。これにより、毛筆が触れた場所に、実際の毛筆で筆記した場合と同じ太さの筆跡を表示できるので、手本をなぞる初歩的な書写練習も可能となった。また、墨の滴りや飛び散り、硯の波紋や乾きなどの視覚効果を表現し、使用時のリアリティを高めた。なお、本システムでは、乾燥した毛筆で滑らかなスクリーン上に筆記するので、現実に墨を含んだ毛筆の挙動を完全に模倣するわけではない。しかし、乾燥した毛筆であっても、その挙動に見合った筆跡が生成されるならば、ユーザと毛筆の連携は保たれ、毛筆の表現力を大きく損なうことはないと考えた。結果的に、初心者は毛筆に慣れ親しむことができ、経験者は毛筆を扱う楽しさを再発見できるように、ユーザを問わず簡単に書道を仮想体験できるシステムの開発を目指した。

2. システムの構成

図 1 に、本システムの構成を示す。ユーザが毛筆によってスクリーン内の仮想半紙上に筆記すると、スクリーンにリアルタイムに筆跡が表示される。本システムは Java により構築され、PC (Xeon 3.2 GHz, Windows XP) 上で動作している。本章では、本システムの主要な構成要素について述べる。

2.1 毛筆

市販されている毛筆のうち、軸が中空のものを選び、軸の内部に 3 次元位置センサ（以下、単にセンサと呼ぶ）を設置して毛筆の状態をモニタする。センサには、Ascension 社 miniBIRD を用いた。このセンサは、送信器の周囲に形成される直流磁場を検知し、センサの 6 自由度の情報を取得できる。また、センサが 5 mm × 5 mm × 10 mm と小型かつ軽量なので、毛筆の軸内に直接挿入できる。これにより、毛筆の先端にできるだけ近い場所で測定でき、筆先の空間座標を求める際の誤差を抑えられる。また、センサが軸内に内蔵されるので、毛筆の持ち方が制限されず、外観や使

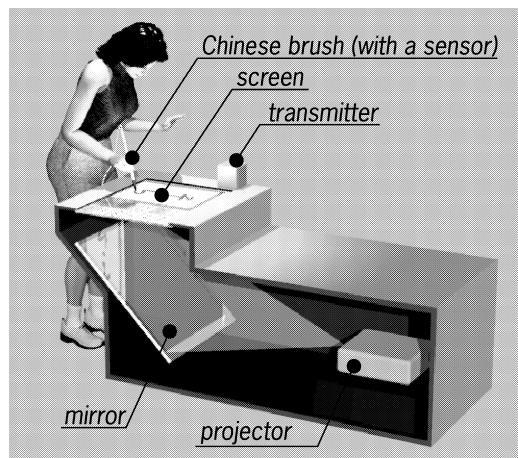


図 1 システムの構成

Fig. 1 Components of proposed system.



図 2 使用した毛筆とセンサ

Fig. 2 A Chinese brush and a sensor.

用感を大きく損ねることがない。センサは、軸内で動揺しないように合成樹脂製のカバーで覆い、軸内に固定した。センサと毛筆の外観を図 2 に示す。

2.2 リア投写ボックス

使用するセンサは磁気式であり、金属、あるいは機器類が発する磁場が近接していると、測定誤差を生じる。そこで、機器類がセンサと干渉しないよう、木製のリア投写ボックスを製作した。この装置の内部に液晶プロジェクタを設置し、25 インチ程度の画面サイズの水平なスクリーンに裏面から映像を投影する。また、投影面と筆記面を兼ねる樹脂製リア投写用スクリーンは、ユーザが手をついても割れない程度の十分な強度を持っている。この装置により、センサと機器類の干渉を防止できるだけでなく、ユーザは、ほぼ原寸大の半紙や硯が投影された水平面に、自らの影を落すことなく筆記できる。スクリーンの外観を図 3 に示す。

2.3 利用手順

本システムのインターフェースは、幅広い年齢層のユーザを想定し、なるべく平易なものにした。システムは、書写練習と自由筆記の状態を持つ。画面中の「手本 1」から「手本 5」のいずれかのボタンに毛筆が触れると、対応する手本が朱色で提示される。この手本をユーザが毛筆でなぞり書きし、「先生」ボタンに毛筆が触れると、評価結果が表示される。「自由」ボタンに毛筆で触

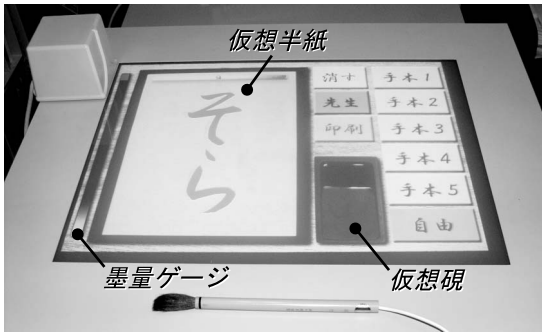


図3 スクリーンの外観

Fig. 3 Appearance of a screen.

れると、「先生」ボタンが暗転して半紙は白紙になり、自由に筆記できる状態になる。「消す」ボタンに毛筆で触れると、ただちに仮想半紙がスクロールアウトし、書写練習中ならば手本が、自由筆記中ならば白紙の半紙が再表示される。「印刷」ボタンにより、任意の時点で、仮想半紙のハードコピーが得られる。

3. 筆跡生成法

本章では、本システムで最も重要な部分を占める、センサから情報を得て、筆記した場所に筆跡を表示するまでのプロセスについて述べる。

3.1 センサからの情報取得

センサからは、空間座標と姿勢行列を最大で毎秒144回得ることができるが、安定性を確保する理由から、実際には毎秒60回に設定してサンプリングしている。センサの空間座標を P 、姿勢行列から得た、センサの向きを示す単位ベクトルを a とする。これらの情報を、一連の筆記動作を通じて連続的に取得したものを、以下では運筆情報と呼ぶ。

3.2 筆のパラメータ推定

毛筆に関する基本的なパラメータを図4に示す。センサから筆先までの距離 L については、センサが軸内部にあって見えないことから、正確な実測が難しい。そこで、筆先を筆記面にわずかに接触させて保った状態で P 、 a を測定し、次式により L を得る。なお、 a の x, y, z 成分をそれぞれ a_x, a_y, a_z とする。

$$L = -\frac{P_z}{a_z} \quad (1)$$

また、毛筆の先端座標 P' 、接地座標 P_0 、筆圧 f 、および毛筆の傾き θ と ϕ を、それぞれ次式のように定義する。なお、式中の $\text{atan2}(a_y, a_x)$ は、 $\tan^{-1}\left(\frac{a_y}{a_x}\right)$ を4象限すべてにおいて角度が求まるように拡張したものである。

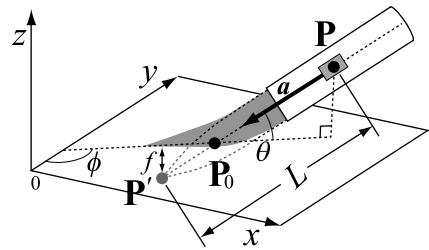


図4 毛筆のパラメータ

Fig. 4 Parameters of a brush.

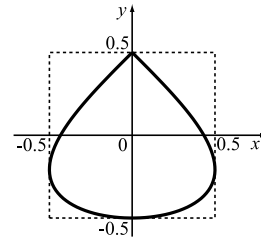


図5 筆触の基本形状

Fig. 5 Basic shape of footprint.

$$P' = P + La \quad (2)$$

$$P_0 = P + \frac{P_z}{P_z - P'_z} La \quad (3)$$

$$f = -P'_z \quad (4)$$

$$\theta = \sin^{-1}(-a_z) \quad (5)$$

$$\phi = \text{atan2}(a_y, a_x) - \frac{\pi}{2} \quad (6)$$

なお、筆の姿勢が垂直に近い場合は ϕ が安定して得られない。そこで、時刻 t における ϕ_t が ϕ_{t-1} からしきい値を超えて変動しようとする場合、 $\phi_t = \phi_{t-1}$ とすることで、現実には起こりえない急激な変動を抑制した。また、本システムでは乾燥した毛筆により摩擦の小さいスクリーンに筆記するので、湿った毛筆と半紙で生じるような、筆記に従って筆先の向きと軸の向きにずれが生じる現象は考慮していない。

3.3 筆触形状の取得

ある瞬間に、毛筆が筆記面に接している部分の形状のことを、筆触形状と呼ぶ。そして、実際の筆触形状の観察より、その基本形状は次の関数で表されると仮定する³⁾。関数の形状を図5に示す。

$$\begin{cases} x = \frac{\sin 2\psi}{2} \\ y = \frac{1}{2} - \cos \psi \end{cases} \quad \left(-\frac{\pi}{2} \leq \psi \leq \frac{\pi}{2}\right) \quad (7)$$

この基本形状に、筆記の状態に応じてスケーリングおよび回転の変換を施し、筆触形状を得る必要がある。そこで、実際に毛筆で筆記した筆跡を実測した結果に基づき、毛筆の状態に応じた筆触形状を得る試み

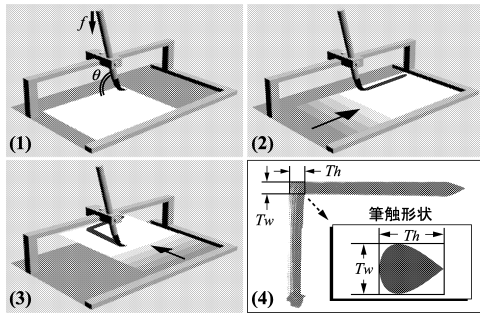


図 6 実測による筆触形状の取得
Fig. 6 Obtaining a footprint from a real stroke.

を行った．ここで，毛筆を傾き θ ，筆圧 f で筆記面に押し付けた場合の筆触形状は，幅 $T_w(\theta, f)$ ，高さ $T_h(\theta, f)$ の大きさの矩形に図 5 の基本形状が内接したものとする．以下， $T_w(\theta, f)$ ， $T_h(\theta, f)$ の組を筆触関数と呼ぶ．

測定手順を図 6 に示す．最初に，スポットで計量した一定量の墨汁を含ませた毛筆を，固定器具によって， θ, f の状態で筆記面に接触させる．次に，毛筆を固定したまま筆記面を縦横にスライドすると，筆記面に貼り付けた紙に L 字形の筆跡が残る．そして，この筆跡の屈曲部に現れる矩形の寸法を実測し， (θ, f) の場合の筆触形状を得る．

θ は 15° から 90° まで 15° 単位， f は 5 mm から 35 mm まで 5 mm 単位で変化させながら筆跡を収集した．1 組の θ と f について各 20 本ずつの筆跡を描画し，実測ののち平均した．

筆触関数より，任意の (θ, f) に対する筆触形状をリアルタイムに得るには，実測した離散的なデータ点しか持たない筆触関数を高速に補間する必要がある．そこで，実測データをスプライン補間して θ 方向が 76， f 方向が 36 の格子点を持つメッシュ状データを生成し，メモリ上に保持しておく．このようにして得た筆触関数を，図 7 に示す．任意の (θ, f) に対する筆触形状が必要な場合は，メッシュを線形補間することにより，高速に取得できる．

求められた筆のパラメータと筆触関数より，その時点の筆が筆記面と接する筆触形状は次式で得られる．

$$\begin{cases} X = x' \cdot \cos \phi - y' \cdot \sin \phi + P_{0x} \\ Y = x' \cdot \sin \phi + y' \cdot \cos \phi + P_{0y} \end{cases} \quad (8)$$

ここで，

$$\begin{cases} x' = T_w(\theta, f) \cdot \frac{\sin 2\psi}{2} R \\ y' = T_h(\theta, f) \cdot \left(\frac{1}{2} - \cos \psi\right) R \end{cases} \quad (9)$$

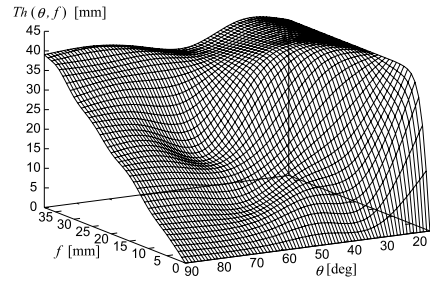
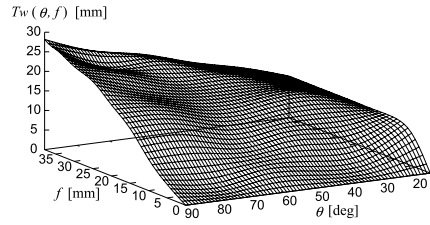


図 7 筆触関数

Fig. 7 Footprint functions.

であり， ψ は $-\frac{\pi}{2}$ から $\frac{\pi}{2}$ まで連続的に変化させる． R はスクリーン上に筆触形状を実寸表示するための定数である．

3.4 筆跡のキャリブレーション

式 (3) により求めた接地座標 P_0 を中心とし，幅と高さがそれぞれ $T_w(\theta, f)$ ， $T_h(\theta, f)$ の筆触形状を筆の向きに応じて回転させ，連続して描画すれば筆跡が生成できる．しかし，実際には，いくつかの要因により，ユーザが意図した筆跡と実際に生成される筆跡との間に，ずれが生じる．次では，ずれの原因について列挙し，補正方法について述べる．

3.4.1 筆触関数の補正

筆跡の実測によって得た筆触関数は，毛筆自体の経時変化，あるいは測定したもとは別の毛筆を使用する場合などのために，スケールを調整する必要がある．そこで， $T_h(\theta, f)$ ， $T_w(\theta, f)$ のそれぞれにパラメータ α を乗じてスケールを補正する．

3.4.2 描画座標の補正

ユーザの筆記座標と，システムが生成する筆跡の表示座標には，主に次の原因によりずれが生じる．

(1) 現実の毛筆は柔軟であり，筆圧によってしなる．ゆえに， P_0 がつねに筆触形状の中心点付近に位置するように，筆圧の増加にともなって P_0 を筆先方向に移動させる必要がある (図 8 (a))．

(2) 筆記面はスクリーンの表面，投影面は裏面であり，スクリーンの厚さは 6 mm あることから，ユーザは手本を正確になぞったつもりでも，実際にはずれが生じている (図 8 (b))．

そこで，このずれを考慮した描画座標 P'_0 を次の

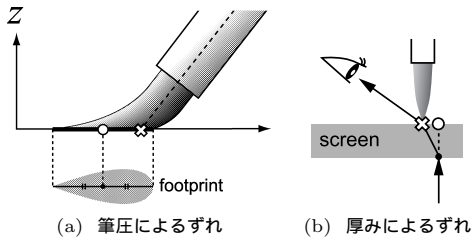


図 8 筆記位置ずれの要因

Fig.8 Errors caused by drawing.

ように定義する． β は筆圧による描画座標のずれの程度を決定するパラメータである．また，スクリーンの厚みによる位置ずれは平行移動成分とし，描画位置に γ_x, γ_y を加算する．

$$\begin{cases} P'_{0x} = P_{0x} + \beta \cdot f \cdot a_x + \gamma_x \\ P'_{0y} = P_{0y} + \beta \cdot f \cdot a_y + \gamma_y \\ P'_{0z} = 0 \end{cases} \quad (10)$$

3.4.3 補正パラメータの決定

前述のパラメータ $\alpha, \beta, \gamma_x, \gamma_y$ を適切に設定するためには，毛筆を実際に墨汁に浸して紙に墨痕を残すと同時に，運筆情報も記録しておき，運筆情報から生成した筆跡と実際の墨痕が画像としてできるだけ一致するように各パラメータを調整するという枠組みが考えられる．しかし，この方法では，スクリーンの厚みに起因するずれを補正できないうえに，調整のごとに現実の書道と同じ手間を要する．そこで，次の方法を用いた．

(1) あらかじめ用意した手本画像をスクリーンに投影する．

(2) ユーザに，センサ内蔵の毛筆で手本画像をなぞらせ，運筆情報を記録する．スクリーン上に筆跡のフィードバックは表示しないが，手本をなるべく忠実になぞるように指示する．

(3) 運筆情報から再生した筆跡が，手本画像と最もよく一致するように，パラメータ $\alpha, \beta, \gamma_x, \gamma_y$ を調整する．

この手順は，手本画像を T とし，生成した筆跡を $L(\alpha, \beta, \gamma_x, \gamma_y)$ として，誤差 $E = |T - L|$ を最小にする $\alpha, \beta, \gamma_x, \gamma_y$ を求めるという 4 次元の最小化問題である．これらのパラメータの最適値を直接に求めることは計算量の点で困難なので，共役勾配法¹⁰⁾ を用いて誤差の最小化を行った．

キャリブレーション前後の，手本画像とそれをなぞった際の運筆情報から生成した筆跡画像とのずれを，図 9 に示す．キャリブレーション後は，より手本画像に近



(a) キャリブレーション前

(b) キャリブレーション後

図 9 手本画像（黒）と生成した筆跡（灰）との差異

Fig.9 Differences between a copybook (black) and generated handwritten (gray).

い筆跡が生成できている．誤差最小化のための計算は，図 9 の例では 25 秒で完了した．

3.5 かすれをとまなう筆跡の描画

実際の毛筆では，墨の残り量や筆記速度に応じてかすれが生じ，そのことが，毛筆らしい勢いを生む．毛筆のかすれを表現する方法はいくつか提案されているが^{8),9)}，本システムでは，毛筆を繊維の束と見なし，毛筆の状態に応じて繊維の半径に変化を与える方法を用いた．繊維の半径を小さくすると，描画時に繊維間に隙間が生じ，かすれが表現できる．

毛筆断面には，筆触形状に内接するように 100 個の小円を配置し，繊維を表現した．かすれがあまり均質にならないよう，各小円の半径には，乱数により若干のばらつきを与えた．各小円の半径は毛筆の状態により変動するが，どのような場合でも各小円は必ず筆触形状に内接し，筆触形状からはみ出すことはない．以下，上述の方法で表現された毛筆断面を毛筆断面パターンと呼ぶ．

繊維の半径は，ある時点で筆が含んでいる仮想的な墨量と，その時点での筆記速度によって決まる．まず，繊維の半径を r ，繊維の最大半径を r_{max} ，ある時点で筆が含む仮想墨量を u ，かすれが生じ始める墨量を U_L とし，墨量により変化を受けた繊維の半径 r_u を次式で表す．

$$r_u = \frac{u}{U_L} \cdot r_{max} \quad (11)$$

次に，ある時点の筆記速度を v ，かすれ始める筆記速度を V_L ，最大のかすれ量に達する筆記速度を V_H とし，筆記速度により変化を受けた繊維の半径 r_v を次式で表す．

$$r_v = \frac{V_H - v}{V_H - V_L} \cdot r_{max} \quad (12)$$

ここで， r_u および r_v は，最小値を 0.5，最大値を

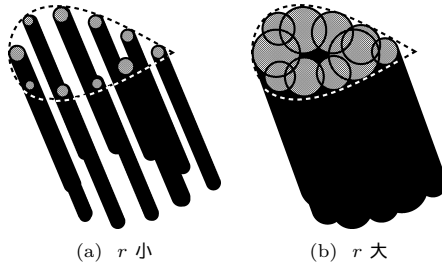


図 10 毛筆断面パターンと筆跡の概念図

Fig. 10 Concept of cross sections of a brush and swept patterns.



図 11 かすれをともなう筆跡の生成例
Fig. 11 An example of scratched stroke.

r_{max} に制限する．最終的に， $r = \min(r_u, r_v)$ として，より強いかすれる側を採用する．この r を用い，筆触形状の位置と形状を更新しつつ，毛筆断面パターンを連続して描画すれば，かすれをともなう筆跡が得られる．なお，ユーザが表現手段として意図的にかすれを生成できるように，現実の毛筆よりもかすれやすく設定してある．また， r が最小であっても，繊維の数が多いと十分にかすれないので， r の減少にともなう繊維をランダムに間引くようにし，より強いかすれを表現する．これらの手順は，両端が円弧で丸められた 100 本の直線を描画して筆触形状間を補間することにより高速に処理できる．本システムで用いた毛筆断面パターンと r による筆跡の違いの概念図を，図 10 に示す．また，生成されたかすれの例を図 11 に示す．

4. システムの周辺要素

本システムでは，筆跡を生成する機能を軸に，書道の動作にともなう視覚効果や評価の要素を統合している．本章では，各要素について述べる．

4.1 毛筆が含む墨量の表現

ユーザは，スクリーンの左端にある墨量ゲージ(図 3 参照)により，毛筆が含んでいる墨の量をリアルタイムに把握できる．筆記した面積にほぼ比例して墨が減少し，墨量ゲージが短くなる．

4.2 硯について

硯は，墨汁が溜まっている部分である「墨池」と、墨を擦ったり，毛筆の余分な墨を落して筆先を整えたりする「硯面」とに大きく分けられる．それぞれの

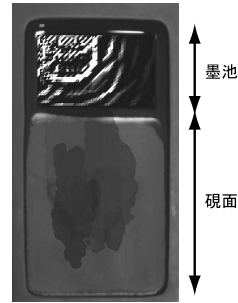


図 12 仮想硯
Fig. 12 A virtual inkstone.

領域の機能と表現について次項に述べる．

4.2.1 墨池の機能と表現

毛筆が含む墨が少なくなると前述のようにかすれが生じるので，ユーザは，墨池の部分に毛筆で触れて，毛筆に墨を補充できる．補充の速度や量は，墨池に触れる筆圧や時間によって自由に調整できる．

また，視覚効果として，筆先が触れた部分に波紋を発生させる．まず，墨池の部分を 46×28 の格子に分割し，2次元の波動方程式によって格子点の波高を求める．次に，格子点を連結して液面のサーフェスマデルを構成し，シェーディングを行うことで，リアルタイムに波紋のアニメーションを表示する．

4.2.2 硯面の機能と表現

墨量ゲージの上端にある赤い領域に達するまで墨を補充すると，毛筆が含める墨量の上限を超えたことを意味する．ユーザはこのとき，硯面で余分な墨を落としてから筆記する必要がある．そこで，ユーザが硯面で毛筆の先を整える動作をすると，筆の墨量が減少するようにした．

視覚効果として，筆先が硯面上を移動した面積に応じて墨量ゲージを短くすると同時に，硯面に触れた部分に筆跡を描き，余分な墨が落ちたことを表現している．また，硯面上の墨が徐々に乾燥していく様子表現するため，1秒ごとに硯面の原画像を透過率 95%で上書きする．これにより，時間経過とともに硯面上の筆跡は薄れていく．硯面の画像更新を一定回数繰り返したのちは更新を停止し，若干の跡を残すようにした．仮想硯の様子を，図 12 に示す．

4.3 墨の落下の表現

現実の墨を含んだ毛筆では，半紙の上に墨が滴り落ちることがある．このような滴りや飛び散りは，荒々しさの演出効果も持つ．そこで，仮想硯の使用に必然性を持たせる意味も含め，これらの現象を表現した．

現実の滴りを観察したうえで，滴りの本体である大円の周囲に微細なとげがあるモデルを用いた．大円の

半径は、筆先から落下した滴りが仮想半紙に衝突する際のおおよその速度より決定する。また、乱数の要素を含む微小な2個の三角形を組み合わせてとげを表現し、滴りの衝突速度が一定以上の場合にのみ描画する。

仮想半紙の上空において、次のいずれかの条件が満たされた場合、その時点での筆先座標の直下に滴りを描画する。

- (1) 筆先が一定以上の加速度で振り下ろされた場合。
- (2) 毛筆が過剰に墨を含んでいる場合。

両者の場合とも、滴りを描画するごとに一定量を墨量ゲージから減ずる。

(1) の場合は、加速度がしきい値を超えている限り、仮想半紙上に連続して滴りを描画する。現実には、振り下ろされた筆が急停止するという一連の動作を検知して滴りを生じさせるべきだが、センサの性能の理由から、振り下ろし動作のみで滴りを生じさせた。ただし、意図的に振り下ろさない限りは滴りが生じないようにしきい値を設定した。

(2) の場合は、滴りの描画ごとに 500ms 前後のランダムな時間待ちを行い、まだ墨量が過剰ならば滴りの描画を続けるので、墨量が適正な量に減少するまで、数滴の滴りが描画される。滴りのとげの表現と実際の

描画例を図 13 に示す。

4.4 書写練習と評価

ユーザが、スクリーン上の「手本 1」から「手本 5」までのボタンに毛筆で触れると、対応した手本画像が朱色で表示される。この手本を上からなぞり、初歩的な書写練習を行う。

まず、手本と筆跡の重なりに対して加点、はみ出しに対して減点を行って素点を求める。次に、ユーザが重ね塗りして文字を整えても高得点が出ないように、手本ごとに定められた画数との差をペナルティとして減点する。そして、点数に応じて、丸なし、丸 1 個、2 個、3 個、花丸の 5 段階評価を朱色で表示する。手本には、市販の書道教本¹¹⁾ をスキャンし、2 値化したものを用いた。

5. 結果と議論

本章では、本システムによる筆跡の生成結果を示し、結果についての議論を行う。

5.1 結果

図 14 に、筆記例を示す。(a), (b) は書写練習の例、(c), (d) はそれぞれに筆遣いを変えて自由に筆記した例である。(d) では、かすれ、墨の飛び散りを意図的に生じさせ、毛筆らしいダイナミックな表現を試みた。

5.2 システムの評価

本システムを評価するために、まず、本システムで用いている筆跡の生成方法が妥当であるかを検証し、次に、ユーザの運筆情報から、本人の意図に沿った筆跡を生成できるかを検証する。各種の検証は、なぞり書きによってシステムをキャリブレーションした状態で、生成した筆跡と手本との誤差を比較する方法で行う。キャリブレーションの手順は、3.4 節で述べた手順に準じる。

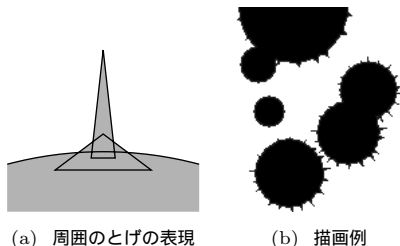


図 13 滴りの表現
Fig. 13 Expression of drops.

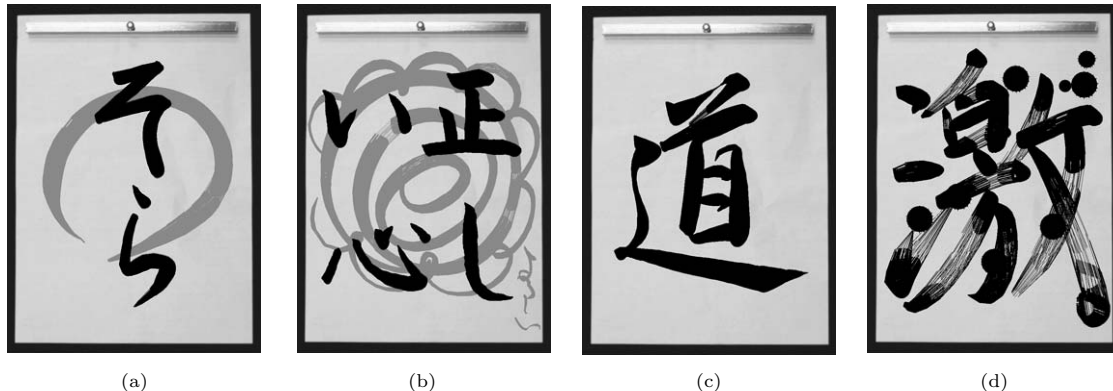


図 14 筆記例
Fig. 14 Generated handwritings.

表 1 筆触形状による誤差の比較

Table 1 Comparison of errors between two footprint functions.

	楕円	滴形
合計誤差	620105	606401

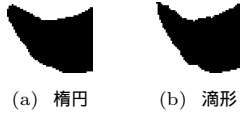


図 15 各筆触関数による書き始めの様子

Fig. 15 Appearances of handwriting generated by two footprint functions.

5.2.1 筆跡生成法の検証

本システムでは、図 5 に示した滴形の筆触形状を連続して描く方法により筆跡を生成している。この筆跡生成法は従来より用いられているが、その妥当性については十分に検証されていない。そこで、他の筆触形状を用いた場合と比較して検証実験を行った。

まず、同一ユーザが同一の手本を 10 回なぞり書きし、10 通りの運筆情報を得る。次に、これらの運筆情報から、筆触形状が楕円形の場合と滴形の場合それぞれの筆跡を生成し、手本との誤差を最小化する。この手順により、楕円、滴形それぞれの筆跡と手本との誤差（画像間のユークリッド距離）を 10 回合計したものを表 1 に示す。この結果より、滴形の筆触形状は、楕円の場合の 97.8%の誤差で手本を再現できることが判明した。両者の差は大きくはないが、生成された筆跡を観察すると、滴形の筆触形状は、特に書き始めなどの鋭い部分の表現力に優れていることが分かる（図 15 参照）。

5.2.2 運筆情報からの筆跡再現能力の検証

次に、システムが、ユーザの運筆情報から本人の意図どおりの筆跡を生成できるかを検証する。検証する点は、(1) ある手本によるキャリブレーション結果が他の手本に対しても有効か、(2) あるユーザによるキャリブレーション結果が他人に対しても有効か、の 2 点である。検証実験の手順を次に示す。

(1) ユーザ 5 名に毛筆で手本 L をなぞらせ、運筆情報を得る。これを $L_A \sim L_E$ とする。なお、ユーザ A, B, C はこの動作を行うのは初めてであり、ユーザ D, E はこの動作に習熟している。

(2) 同様に手本 U をなぞらせ、運筆情報 $U_A \sim U_E$ を得る。

(3) キャリブレーションにより、 $L_A \sim L_E$ のそれぞれについて手本 L との誤差が最小となる筆跡を生成できるパラメータ $P_A \sim P_E$ を得る。

表 2 キャリブレーション後の各手本と筆跡との誤差

Table 2 Errors between copybooks and handwritings after calibration.

(a) 手本 L に関する誤差

	P_A	P_B	P_C	P_D	P_E	Raw
L_A	<u>64890</u>	91633	88388	81446	74741	77870
L_B	85667	<u>59037</u>	<u>64006</u>	65762	68217	92117
L_C	89637	74444	67387	71409	74769	97630
L_D	88865	73400	69141	65379	67662	100563
L_E	77499	69634	64968	<u>61063</u>	<u>56712</u>	89697
合計	406558	368148	353890	345059	342101	457877
割合	88.8%	80.4%	77.3%	75.4%	74.7%	100%
平均	79.3%					

(b) 手本 U に関する誤差

	P_A	P_B	P_C	P_D	P_E	Raw
U_A	<u>46925</u>	69587	68634	61890	57508	51692
U_B	62350	<u>52957</u>	<u>54672</u>	53553	54224	72513
U_C	77996	67590	65857	66548	70834	85901
U_D	67961	62283	61476	57522	58999	80812
U_E	56504	56919	59671	<u>50461</u>	<u>50898</u>	66857
合計	311736	309336	310310	289974	292463	357775
割合	87.1%	86.5%	86.7%	81.0%	81.7%	100%
平均	84.6%					

(4) $P_A \sim P_E$ を用いて、 $L_A \sim L_E, U_A \sim U_E$ についての筆跡を生成し、各手本との誤差（画像間のユークリッド距離）を調べる。

表 2 (a) に手本 L に関する結果を示す。表の右端の列 Raw は、キャリブレーションをいっさい行わなかった場合の結果である ($\alpha = 1, \beta = \gamma_x = \gamma_y = 0$)。また、Raw の合計を 100% とし、各パラメータごとに誤差の割合を求めたものを最後の行に示した。

同表について、まず、同一ユーザの運筆情報に対し、全ユーザのパラメータを順次適用した結果に注目する（表では横方向に相当）。全ユーザについて、自分自身のパラメータを用いた際に誤差が最小となっていることから、最小化は問題なく行われているといえる。

次に、同一ユーザのパラメータを、全ユーザの運筆情報に順次適用した結果に注目する（表では縦方向に相当。アンダーラインは同一パラメータ内での最小値）。ユーザ A, B, E の運筆情報については、ユーザ自身のパラメータを用いた際に誤差が最小となり、ユーザ C, D の運筆情報については、自身のパラメータがそれぞれ 3 番目、2 番目に良好な結果となっている。また、どのユーザのパラメータを用いた場合でも、キャリブレーションを行わない場合と比べて誤差は平均的に小さくなっており、特に、なぞり動作に習熟しているユーザのパラメータの場合が低い平均値を示した。

次に、キャリブレーションに用いていない手本 U に対して、手本 L で得たパラメータを適用した結果を表 2 (b) に示す。システムにとって未知の文字が筆記

されたゆえに、平均的に誤差は大きくなっているが、全体的な傾向は同表 (a) と変わらない。

これらの実験結果より、ユーザ自身がキャリブレーションを行えば、最も良好な筆跡が得られることが判明した。さらに、習熟したユーザが一度だけキャリブレーションを行っておけば、だれがどのような文字を筆記しても、キャリブレーション前と比べて、良好な筆跡が得られることも分かった。これらの結果は、本システムが、個人の筆跡の記録や、不特定多数のユーザを対象とするイベントなどの幅広い用途に対応できることを示唆している。

5.3 システムの構成に関して

今回製作したりア投写ボックスは大型であり、移動や設置が容易ではない。これは、センサの性質と、使用したプロジェクタの焦点距離により仕様を決定したことによる。ゆえに、周囲の環境に影響されずに6自由度の情報が得られるセンサが使用できれば、水平に設置した大型ディスプレイ上にも直接筆記できる。また、現状のセンサであっても、短焦点プロジェクタの使用や光路の改良により、装置を小型化できる。

通常、乾燥した毛筆の先端は、1点に収束せず、ある程度の広がりを持っているので、特に細い線を描こうとする際に書き始め位置を定めにくい。また、室内の環境によっては、樹脂製のスクリーンとの摩擦で静電気が生じ、毛先が乱れる。現在は市販の毛筆をそのまま流用しているが、理想的には、乾燥していても先端が1点に収束する素材で毛筆を製作すれば、キャリブレーションの精度が高まり、使用感が向上する。

5.4 筆跡生成法に関して

現実の、多種多様な毛筆に柔軟に対応するには、さらなる工夫が求められる。本システムでは、筆触形状として一定の形状の滴形を用いたが、実際には筆遣いによって楕円や三角形など多様な変化が見られる。筆触形状は、特に書き始めや書き終わりの表現力を左右するので、今後検討する余地がある。また、現状では、毛筆の筆跡データの実測に多大な手間を要するので、既存の実測データを大きさや形状の異なる毛筆に適応させる方法の検討が必要である。

本システムで用いたキャリブレーションの枠組みは、筆跡生成に影響を与える関数の入れ替えや増減が容易であり、パラメトリックな筆モデルであれば同じ枠組みでキャリブレーションが可能という利点がある。ただし、次元数が増えると、収束に時間を要するうえに、不適な極小値への収束などの弊害も生まれることが考えられる。また、現状では、書き始めや書き終わりの部分など、毛筆の状態が大きくかつ不規則に変化する

部分は、キャリブレーションが十分ではない。実際には、画像の面積としては小さい部分だが、全体の印象を決定することも多いので、単純に全体の重なりを評価するだけでなく、手本画像の重要な個所に重みを付けるなどの方法により、主観を重視したキャリブレーションを検討する必要がある。

5.5 評価機能に関して

現状での書写練習の評価機能は、システムとの対話性を高めるために補助的な機能として追加したものであり、手本文字画像との重なりを評価するというごく単純な基準しか用いていない。現実の書写教育の現場で利用するには、文字の構造、筆順、作法、審美的な見地からの指導や評価などについても考慮する必要がある。機能の大幅な追加や見直しが必要となる。しかし、現状の評価機能程度であっても、ユーザは一種のゲーム性を見出し、高い評価を得ようと繰り返し練習するので、システムを使用させる動機付けとしての意味は大きい。また、特別なフォーマットのデータを必要としないので、市販の書道教本などの手本を簡単に利用できるという利点もある。

5.6 ユーザの反応

学内外の複数のイベントにおいて、本システムを展示する機会があった。その際、幼稚園児から老人まで、のべ1,000名以上がシステムを操作したが、ごく簡単な説明だけで自由に操作できることが確認できた。全体的に反応は好意的であり、高い評価を得ようと書写練習に熱中する来場者や、筆文字や絵の自由な筆記を楽しむ来場者が多く見られた。特に、毛筆の扱いには比較的慣れているものの、コンピュータの操作経験に乏しい多くの年輩者が、熱心にシステムを操作したことは興味深い。そのほか、肯定的な意見として、次のようなものがあった。

- 幼少時の書道教育は、墨汁の扱いの不馴れによる衣服や周囲の汚れが大きな障害となるが、それを回避できる。
- 近年、学校で書道教育を行える教師が減少し、ビデオ教材などに頼らざるをえなくなってきたため、このような新たな指導の枠組みが求められている。

このようなユーザの反応から、本システムが毛筆の魅力に触れ、あるいは再発見する契機となり、ひいては従来の書道教育を補助し、補完しようという感触を得た。しかし、意見の中には、書道教室の主宰者による「毛筆で画面に触れて操作するという行為自体、容認しかねる」という否定的な意見もあった。用途によっては、機器や画面のデザインの工夫などで違和感を低減し、文化的背景に対して十分に配慮する必要も生じ

るだろう。

6. ま と め

本論文では、実際の毛筆でスクリーン上に筆跡を生成し、仮想的な書道を体験できるシステムについて述べた。出展を通じて、本システムは、初心者が毛筆に慣れ親しむだけでなく、経験者が毛筆の表現力を再発見する機会にもなりうるということが分かった。映像さえ投影できれば筆記面に制約はないので、この枠組みは、書写教育、芸術書道、ペイントシステム、さらには冒頭に述べた「地書」のようなパフォーマンスなど、幅広い応用が見込める。現状では、墨を含んだ現実の毛筆と完全に一致する筆跡を生成することは原理的に難しいが、将来的には、e-Learningの一環としてネットワーク経由での書道教育に応用したり¹²⁾、書道家の運筆情報を記録し、自由な形態で再現できる高次の毛筆書体データベースを構築したりすることも可能と考える。

参 考 文 献

- 1) ANA グループ機内誌：翼の王国, 1月号 (2005).
- 2) 戸倉 毅, 鈴木隆子, 中村浩子, 牧野優子, 高倉 穂: つづけ字を可能とする毛筆体文字生成システム, 情報処理学会論文誌, Vol.29, No.1, pp.20-28 (1988).
- 3) 塩野 充, 吉村 治, 真田英彦, 手塚慶一: 筆軸の3次元運動に制御された筆触間数をういた毛筆漢字パターンの生成, 電子情報通信学会論文誌, Vol.J72-D-II, No.1, pp.76-84 (1989).
- 4) 斎藤 豪, 中嶋正之: インタラクティブペインティングのための力学的三次元筆モデル, 情報処理学会論文誌, Vol.41, No.3, pp.608-615 (2000).
- 5) Chu, N. and Tai, C.: Real-Time Painting with an Expressive Virtual Chinese Brush, *IEEE Computer Graphics and Applications*, Vol.24, No.5, pp.76-85 (2004).
- 6) 島田英之, 島田恭宏, 大倉 充, 宮垣嘉也: 3次元位置センサを用いた筆文字描画システムの提案, 平成14年度電気・情報関連学会中国支部連合大会, 192708, p.529 (2002).
- 7) 島田英之, 濱本高志, 島田恭宏, 大倉 充: パーチャル書道システムの開発, 情報科学技術フォーラム (FIT2003) 情報技術レターズ, LK-011, pp.291-292 (2003).
- 8) 真野淳治, 中村剛士, 世木博久, 伊藤英則: 毛筆書体におけるくりこみ群を用いたかすれ・にじみ表現, 情報処理学会論文誌, Vol.38, No.4, pp.806-814 (1997).

- 9) Yu, Y.J., Lee, D.H., Lee, Y.B. and Cho H.G.: Interactive Rendering Technique for Realistic Oriental Painting, *Journal of WSCG*, Vol.11, No.1, pp.538-545 (2003).
- 10) Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T. (著), 丹慶勝市, 奥村晴彦, 佐藤俊郎, 小林 誠(訳): Numerical Recipes in C 日本語版, 技術評論社 (1993).
- 11) 大貫思水: 小学生の習字, 金園社 (2000).
- 12) 富田浩介, 河合亮児, 原田哲也: Mixed Reality を利用した書道教示システム—仮想空間と現実空間の一致と力覚提示—, 日本バーチャルリアリティ学会第7回大会論文集 (2002).

(平成17年10月31日受付)

(平成18年9月14日採録)



島田 英之 (正会員)

1989年岡山理科大学理学部電子理学科卒業。1994年同大学大学院博士課程修了。同年同大学助手。2001年同講師、現在に至る。画像処理、コンピュータグラフィクス、HCI等の研究に従事。博士(工学)。電子情報通信学会、映像情報メディア学会、日本バーチャルリアリティ学会各会員。



島田 恭宏 (正会員)

1986年岡山理科大学理学部電子理学科卒業。1991年同大学大学院博士課程修了。同年同大学助手。1997年同講師。2006年同助教授、現在に至る。HCI等の研究に従事。理学博士。電子情報通信学会、映像情報メディア学会、日本バーチャルリアリティ学会各会員。



大倉 充 (正会員)

1983年愛媛大学工学部機械工学科卒業。1985年同大学大学院修士課程修了。1985~1986年日立製作所勤務。1990年岡山理科大学大学院博士課程修了。同年同大学助手。1992年同講師。2005年同助教授、現在に至る。パターン認識、画像処理等の研究に従事。理学博士。電子情報通信学会、映像情報メディア学会、日本知能ファジィ学会各会員。