Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

A Novel and Robust Algorithm to Model Handwriting Skill for Haptic Applications

Li X, Srimathveeravalli G, Singla P and Kesavadas T
Dept. of Mechanical and Aerospace Engineering, University at Buffalo
USA

Abstract— **A necessary step to building effective writing skill training system requires developing good methods for modeling human skill adequately. A number of groups have represented character information in various languages for writing skill based on trajectory information. These methods are often data intensive, tedious to implement and do not encode force information. To overcome these restrictions a novel present a novel modeling methodology that has good accuracy, robustness, flexibility and encodes force information involved in writing characters. This modeling methodology, based on Global-Local Approximation technique, has the capability to provide temporal force or position information independent of time, decoupling velocity information of the sample data used to capture skilled tasks. This modeling approach can be extended many human skilled tasks such as surgery, art and sports**

Keywords—Multiresolution modeling, haptics, virtual reality training, handwriting, calligraphy, human motor skill

I. Introduction

Writing and scribing have closely defined trajectories with minimal variation with well defined acceleration, velocity and force profiles. This information also represents the observable data that can be used to the corresponding model human skill in writing. The simplest means of storing such information is to record the subject's movement and temporally store position, velocity and force data. However, this is not suitable for training purposes because the data obtained is often time dependent. This may curtail the student's ability to independently move and perform an action during the learning process, which is often slower and more deliberate. Hence, there is a need for a mathematical model that can encode force information in a skilled task independent of time. This combined model could then be used to assist the trainee in learning handwriting skill, to devise control laws and calculate required force feedback for assistance.

Common modeling techniques use Spline or Bezier curves to represent written characters exploiting the information to represent and transfer skill. In this paper a new methodology based on Global-Local Approximation technique is proposed for time independent parametric modeling of writing skill using written character information, combining position and force information in a simple mathematical form. A comparison between the new methodology and some existing techniques was performed. The results show that the new technique implemented here provides significant benefits. In the following sections we will describe this in more detail.

Haptically enabled writing, calligraphy simulators and trainers have been developed by many research groups [1][2][3]. Efficient modeling techniques for capturing skill for training and evaluation are required to develop such applications. It is important that the modeling technique be (i) lightweight, (ii) encode haptic (force-pressure) information, (iii) provide a metric for comparison and evaluation, and (iv) be able to provide data for training based on user intent. While researchers have proposed and implemented different approaches towards writing haptic applications, none of the currently available modeling methodologies satisfy listed requirements.

Considering the field of graphonomics or biometrics, extensive work has been performed in modeling writing. Hidden Markov Models, Neural Networks, Direct Template Matching, and Classification by Minimum Distance, to name a few most widely used techniques [4]. However, the primary aim of such work was towards recognition, evaluation or verification of writing. These methods do not cater to haptic based training applications because none of them take force/pressure information into consideration, a crucial aspect for developing haptic applications.

Considering haptics research, Wang D et al[1]modeled Chinese characters based on combination of sequential strokes. Chinese characters were classified into combination of basic and complex strokes; basic strokes were described by third order Bezier curves, complex strokes were modeled as a series of basic strokes. Wang Jue et al[5] modeled handwriting by a tri-unit handwriting model, which focused on characters and the interconnecting strokes using generative models based on control points and B-spline curves. Hennion et al [6] built a hand-writing font based on control points, elliptic arcs and straight lines. They took the beginning of the strokes, the end of the strokes, the vertical tangent points, the horizontal tangent points, the inflexion points and the turn back points as their control point. A loop based model was implemented by Steinherz et al [7]. A loop defined a handwritten pattern, made of several strokes formed when the writing instrument returns to a previous location while touching the pad continuously, giving a closed outline with a "hole" in the center. But this could only be used in modeling cursive letter where a 'loop' appeared. Solis' et al, [2] implemented a modeling system based on discrete Hidden Markov Models. While all of the above techniques are relevant for the presented work, they fail to include forces data captured from subject, into the model. Bezier curves are not an efficient or convenient model for

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

capturing intricate curves. Also, unlike the Kanji script Tamil letters (a widely practiced language from South India, used here for our research) cannot be divided into simple basic strokes. Loop based methods can only be applied when a certain pattern appears in handwriting letters.

II. DATA COLLECTION

To obtain good experimental data for modeling, sample handwriting data was collected from subjects (N=12, All male, Right handed) proficient in writing Tamil and English. Both Tamil letters and English alphabets were segregated into groups based upon number of segments, curvature of characters and direction of movement. The sample letters chosen are shown in Table I .

Table I Sample letters chosen from English and Tamil

English Letter	Tamil Letter
S	அ
P	இ
Q	ண
X	ந
	ய

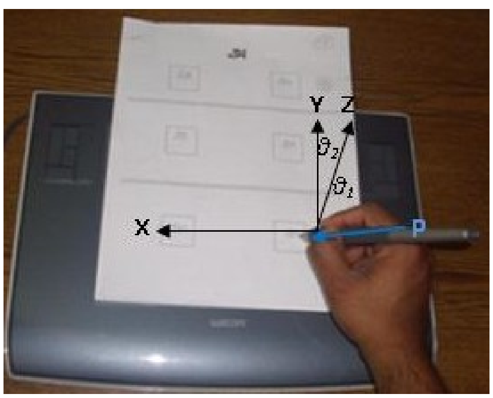

Fig. 1 Subject writing on the tablet, axis marked for force and position reference.

A pressure sensitive tablet (Intuous by Wacom) (Fig 1) was used to collect experimental data. A C++ API's provided by Wacom was used to collect the experimental data. Using the writing pad and stylus, trajectory $[X(t),Y(t)]$, pressure applied against the pen tip *P (t)* and orientations of the pen $\theta \quad \theta$ were obtained with respect to time. For successfully integrating the haptic feedback, pressure intensity value had to read from the tablet were converted into actual force exerted on the pad.

Prior to data collection all subjects were allowed to use the writing pad to familiarize themselves with system, and become comfortable in using it. The sample characters were not ordered in any particular sequence and the subjects progressed from one to the next until all 9 sets of data was collected. Six samples were collected for each character and data collection lasted less than ten minutes. An A4 sized sheet with six 1 inch square boxes was used for data collection. The paper was attached to the tablet and the subject was asked to fill the characters in the boxes in their natural writing motion. The samples were cleaned, aggregated and averaged into a single representative dataset for each subject, which was then used for testing the algorithms.

III. INTRODUCTION OF GLOBAL-LOCAL APPROXIMATION

An adaptation of the Global Local Approximation (GLA) method by Singla et al [8] is presented here. This adaptation allows creation of robust handwriting models for haptic training applications, incorporating both position and force information. GLA allows generation of piecewise globally continuous functions by means of blending locally approximate functions using some predefined weight function. The global function is given by the expression:

$$f(t)=\sum_{i=} \qquad \in -\infty +\infty \qquad \in - \tag{1}$$

where $f_i(t)$ is the parameterized local approximation of $f(t)$ and $\omega \ \tau$ is the weighting function worked on a non-dimensional local coordinate $\tau \in -$.

The prior approximations $f_i(t)$ can be arbitrarily chosen as long as they are smooth and can represent the behavior of $f(t)$ locally. Weighting function was chosen to be:

$$\omega \ \tau = - \quad \left\{ \qquad \sum \qquad \right\} \tag{2}$$

where m indicates the desired order of global continuity.

This arrangement provides for accurate modeling by providing piecewise global continuity while allowing for a variety of choices for the local functions. Two pieces of information: position and pressure are available to create the models. Considering position information:

$X = \quad ., x_n]$ be the raw (sampled) x ordinate data and $Y = \quad ., y_n]$ the corresponding y ordinate values, expressed as an implicit continuous function in *x*. If *X* can then be divided into N portions, where for the i^{th} *x* value, the state of y is defined by local n^{th} order polynomial function $f_{y_i}(x)$, given by:

$$f_{y_i}(x)=\sum_{j=} \tag{3}$$

where a_{yj} is a constant parameter coefficient. A global function can then be constructed by blending the local functions. This was done by taking the sum of the products of the local functions and their corresponding weighting functions.

$$y(x)=\sum_{i=} \qquad a_y) \tag{4}$$

The weighting function ω for a given local function can be defined such that it achieves a maximum value of 1 within the interval of $[x_i, x_{i+}$ ⌉ and has zero value elsewhere. Therefore, the local function is most dominant at the point about which it is centered and least dominant at the edges of boundary for which it is defined. Sample weighting functions are shown in Fig.2. In Fig.3 we can see three local trajectories shown in red, blue and green. They can be merged into a global trajectory as shown by black solid line using weighting functions.

To generate the pressure – force model; for a given *X* value there always exists a corresponding pressure value *P* which could also be represented as a continuous function implicit function in *X*. Hence, the pressure model, similar to *Y* can be written as follows:

$$p(x)=\sum_{i=} \qquad a_p) \tag{5}$$

where ω is the weighting functions and $g_i(x, a_p)$ is the parameterized local approximations of $p(x)$.

IV. ADAPTATION OF GLA FOR MODELING WRITING

A stroke was defined as a continuous curve which could be drawn without lifting the pen. Characters such as 'S' and 'C' fall into this category while other characters like 'X' and 'P' are made of multiple strokes. Tamil characters, despite their complexity are written as a single stroke. Collected data was processed on this basis. To achieve this, pressure data was traced to find out null force condition which corresponding to a 'pen up' motion that could be defined as the end of a single stroke. After a character was divided into strokes, they were then separated into sequential segments. This was done because interpolation or approximation could only be done with respect to distinct values, which means the curve should either be monotonically increasing or decreasing. This was determined by examining the point's value where the monotonous nature of abscissa of the curve is altered, and those points together with starting point and end point were defined as separating points.

$$X_k = \quad \ldots \quad =$$
$$Y_k = \quad \ldots \quad =$$
$$P_k = \quad \ldots \quad =$$

let be the corresponding x, y and pressure values between two separating points. According to GLA algorithm introduced in previous section, trajectory $y(x)$ and pressure $p(x)$ were approximated by the (4) and (5).

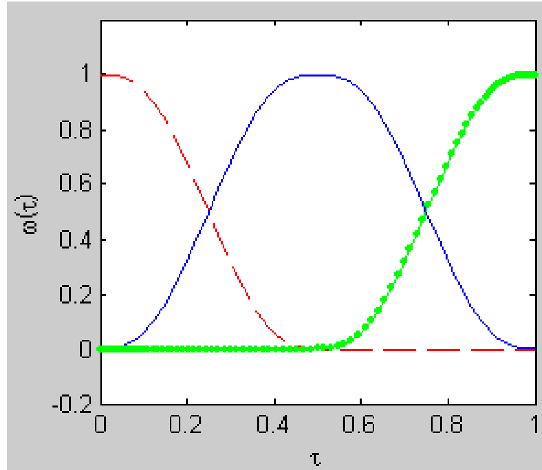

Fig. 2 Weighting functions for second order continuity

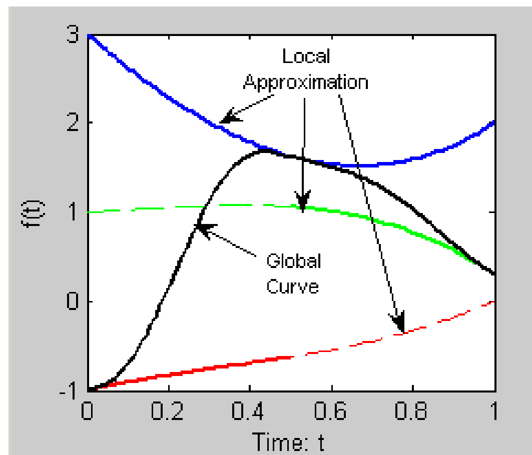

Fig 3 Weighting function approximation of $f(t)$

The set of positive integers:

$$a = \left\{ \quad \right\} \tag{6}$$

represent the parameters need to be solved. Rewrite equation (4) as following:

$$f_i(x, a_y) = \quad \Phi \qquad =$$
$$a_{y,i} = \quad a_{2,i} \quad \ldots \quad a_{n,i}] \tag{7}$$
$$\Phi \quad = \quad \ldots \quad x^n]$$

Now, if A_x and P matrixes are given as below and ω correspond to the value of the weight function corresponding to the i^{th} approximation at position x, then (4) can be implemented as follows:

$$\psi_{y,x} = \begin{bmatrix} \\ \vdots \\ \\ \end{bmatrix} = \begin{bmatrix} \\ \vdots \\ \\ \end{bmatrix} \qquad (8)$$

$$y_x = \qquad (9)$$

Here A_y is the fitting coefficient that needs to be solved. Given the existing handwriting trajectory X and Y, the coefficient A_y can be easily found by applying a Least Square Solution, which leads:

$$A_y = \quad {}_{y,x} \quad {}_{y,x}{}^{-} \psi_{y,x} \qquad (10)$$

Likewise,

$$A_p = \quad {}_{p,x} \quad {}_{p,x}{}^{-} \psi_{p,x} \qquad (11)$$

V. RESULTS

In the current implementation of the algorithm, three n^{th} order polynomials were chosen for local approximation, where n varied as number of data point changed between separating points. m = 2 was chosen as the blending function weight to enforce the 3^{rd} order smoothness. For testing GLA, implementations of Bezier Curve and Cubic Spline based approximation was made. Root Mean Square Error (RMSE), machine time for computation, number of parameters required was used as comparison metrics. Multiple characters from both language sets were used to compare the performance of the algorithms for different sized of raw datasets. Fig. 4-6 shows the comparative performance of Bezier, Spline and GLA. Considering RMSE, GLA provided best performance. Results of comparison can be seen in Table II. Minimal RMSE error over the entire domain for each method ensured that GLA provided best accuracy guaranteeing closest representation of original model. GLA consistently used more number of parameters for representing a model. It was found that GLA used more machine time for computing models, however at the same time exhibited asymptotic behavior with increasing size in datasets (Fig. 7). Though the machine time of GLA was longer than other two methods, it can be seen as an accuracy trade off. It may also be noted that all three modeling methods had a computational time of $O(n^3)$, GLA takes a larger time to execute because of additional parameters. GLA is more powerful than spline based methods due to following reasons: (1) there are fewer restrictions on the choice of local functions. (2) The form of the weighing function is different from NURBS and can guarantee global and local continuity. (3) There is no concept of control points in the GLA. For handwriting task, it is possible that due to non-uniform speed, data points may be denser in certain regions. This can also be true if there were pauses during writing of script. Bezier based methods handle such cases very poorly, the approximation is significantly altered due to the varying distribution of control points. GLA is better adapted to handle this problem. The contrast in modeling can be seeing Fig 8-9. Fig. 10 shows the assistance forces (based on a simple spring – dashpot model) generated based on error feedback between subject's pen (solid line) and GLA model (dotted line). Though the subject wrote at a significantly different speed compared to reference model, GLA was able to generate the appropriate force values based on percentage of character completed by the subject.

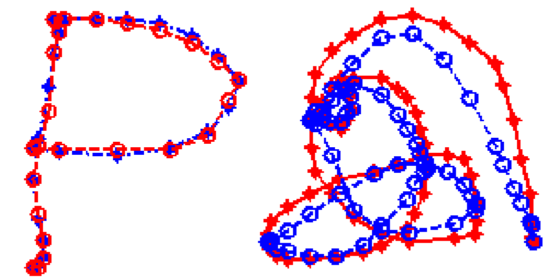

Fig. 4 Bezier Curve

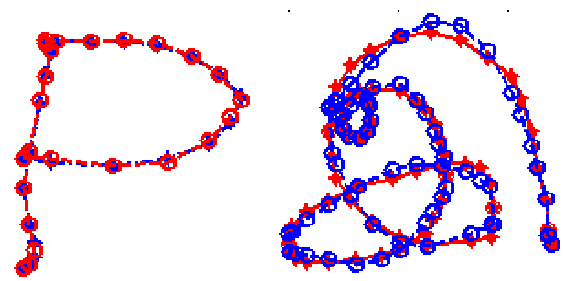

Fig.5 Cubic Spline

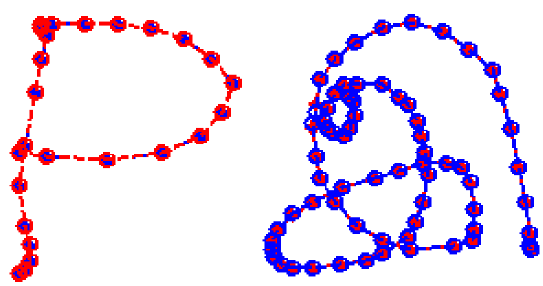

Fig.6 GLA

Table II Performance comparison for metrics between Bezier Curve, Cubic Spline and GLA

	English Letter: S			English Letter: P		
	Bezier	Cubic	GLA	Bezier	Cubic	GLA
No. Data	22	22	22	26	26	26
Running Time(s)	0.016	0.032	0.094	0.016	0.047	0.125
No. Para	21	12	33	25	24	39
RMS Error	2.191	0.738	1.74⁻	1.476	0.807	0
	Tamil Letter: இ			Tamil Letter: ய		

	Bezier	Cubic	GLA	Bezier	Cubic	GLA
No. Data	68	68	68	50	50	50
Running Time	0.047	0.078	0.157	0.031	0.016	0.156
No. Para	49	36	78	67	32	105
RMS Error	4.748	1.048	0	6.651	2.241	0.273

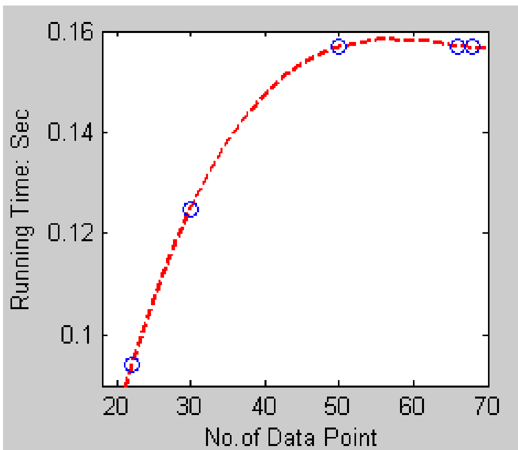

Fig.7 Execution Time of GLA w.r.t number of Data Point

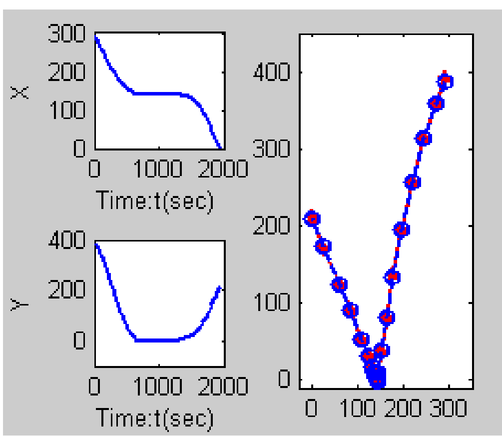

Fig. 8 Special Case solved by GLA

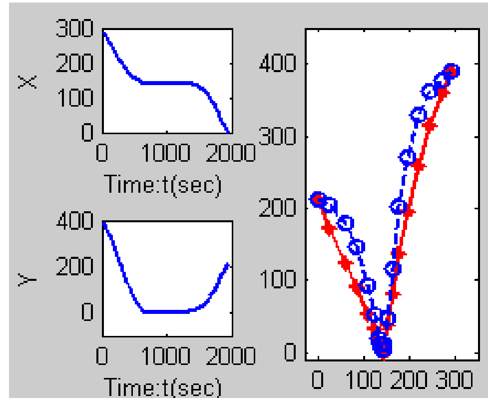

Fig. 9 Special Case solved by Bezier Curve

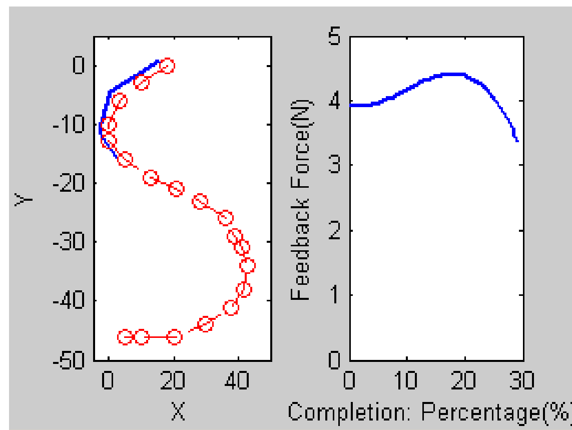

Fig.10 Sample force feedback generated by GLA model as a trainee writes at speeds different from the reference model.

VI. DISCUSSION

Compared to available techniques such as Bezier and Splines, the implementation of the GLA method was seen to be more accurate and robust. The method can also be easily adopted to include force information, and made time independent. Currently research is being undertaken to identify suitable formats for archival of GLA parameter information. Also, the choice of the number of local approximations and the corresponding order of polynomial, which were currently determined through trial and error have to be addressed. While GLA can be adapted for modeling other skills, this needs too be evaluated further. An interesting possibility that arises is that the sequence of weights can also act as a shape descriptor for the character being modeled. This can then be developed into an evaluation method as well.

ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under Grant No. 0632618.

REFERENCES

[1] Wang D, Zhang Y and Yao C "Stroke Based Modeling and Haptic Skill Display for Chinese Calligraphy Simulation System" Virtual Reality, Vol 9. Issue 2,2006

[2] Solis J, Avizzano C A, Bergamasco M "Teaching to Write Japanese Characters using a Haptic Interface" Proceedings of the 10th Symposium On Haptic Interfaces For Virtual Envir. & Teleoperator Systs. (HAPTICS'02) IEEE 2002

[3] Srimathveeravalli G, Gourishankar V, Kumar A and Kesavadas T "Experimental Evaluation of Shared Control for Rehabilitation of Fine Motor Skills", ASME Journal of Computing and Information Science in Engineering, Special Edition on Virtual Reality and Haptics,Vol9,1(March 2009)

[4] Scott Connell. Master Thesis: A comparison of Hidden Markov Model Features for Recognition of Cursive Handwriting. Michigan State University, Computer Science Department. 1996.

[5] Jue Wang; Chenyu Wu; Ying-Qing Xu; Heung-Yeung Shum and Liang Ji "Learning-based cursive handwriting synthesis" Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition, p 157-62, 2002

[6] Hennion, B.; Gentaz, E.; Gouagout, P.; Bara, F. "Telemaque, a new visuo-haptic interface for remediation of dysgraphic children" Eurohaptics Conference, 2005 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2005. World Haptics 2005. First Joint

[7] Steinherz, T.; Doermann, D.; Rivlin, E.and Intrator, N. " Offline loop investigation for handwriting analysis" EEE Transactions on Pattern Analysis and Machine Intelligence, v 31, n 2, p 193-209, Feb. 2009

[8] Bhattacharya, R.; Singla, P. "Nonlinear trajectory generation using global local approximations" Proceedings of the 45th IEEE Conference on Decision and Control (IEEE Cat. No. 06CH37770), p 6 pp., 2006