Knotless Spline Noise Removal Technique for Improved OHCR

Lajish VL, Vinod Kumar Pandey and Sunil Kumar Kopparapu

TCS Innovation Labs - Mumbai, Tata Consultancy Services Ltd., Yantra Park
Pokharan Rd. 2, Thane, Maharashtra - 400 601, India

{lajish.vl,vinod.pande,sunilkumar.kopparapu}@tcs.com

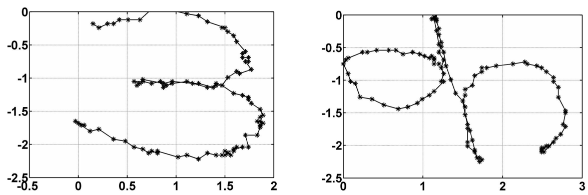
***Abstract*— Noise in on-line hand written characters due to natural shaking of the hand and noise due to the process of digitization is inherent and this can lead to a degraded performance of character recognition system. In this paper, we propose a noise removal technique based on knotless spline. We first show that its noise removal property is independent of the amount of noise unlike the traditional Gaussian smoothing and further show that the noise removal can significantly enhance the performance of the character recognition algorithm. We specifically perform experiments with Devanagari data and show that the noise removal can enhance the performance of character recognition by as much as 10%.**

***Keywords*— On-line Handwritten Character Recognition; Gaussian filters; smoothing; knotless spline**

I. INTRODUCTION

With the percolation of mobile devices into rural areas there is an additionally segment of users who are more adept in using vernacular language to communicate. While this is a positive the unfortunate thing is that entering vernacular languages in general has been hard and in particular using the tiny mobile keypad is painful. The PDAs, palms and handheld PCs are getting cheaper and are more frequently being used for composing short messages and e-mails. These reasons have made Electronic pen (e-pen; essentially non-keyboard mechanism to enter data) as a easy and efficient mode of entering information [1], [2], [3] in ones own hand writing. These devices themselves, sometimes, use on-board handwriting recognition algorithms to transform graphical form of characters into an electronically transferable character string. Generally, information entry is through a series of strokes. A stroke is defined as a uniformly sampled in time, trace of a pen between a pen-down and a pen-up. As a consequence of uniform sampling in time, the number of sampled data points is very sparse especially when the pen movement is fast and dense and more prone to noise when the pen movement is slow. Noise is inherent in any on-line handwritten data and it can be attributed to (a) the inherent and natural shake of the hand of the writer especially at the beginning and at the end of a stroke and (b) the digitization process of the device. Fig.1 (a) shows a sample handwritten Devanagari character *u* and 1(b) show a Devanagari character *ka* collected using Mobile e-Note Taker [1]. These characters are highly contaminated by noise which severely affects the performance of any on-line character recognition algorithm. There are essentially two ways of taking care of the noise, namely, (a) pre processing of raw data to enable noise removal or (b) use of a set of features that can compensate for the inherent noise in one form or another.

In this paper, we propose a noise reduction technique based on knotless spline and compare its noise removal performance with a Gaussian filter. The knotless spline technique is found to be better in suppressing noise as compared to the Gaussian filter. We also show the enhanced performance of a character recognition algorithm due to noise removal on a real on-line Devanagari data set using Fuzzy Directional Features (FDF) [4]. The rest of the paper is organized as follows. We introduce the knotless spline based noise reduction technique in Section II and do a quantitative analysis of the noise reduction in Section III. In Section IV we present experimental results and conclude in Section V.



a) **(b)**

Fig. 1. Devanagari characters *u* and *ka*: raw on-line data

II. KNOTLESS SPLINE FOR NOISE REMOVAL

Let the sequence $\left(x_i^r, y_i^r\right)_{i=0}^m$ represent a raw handwritten stroke made up of *m* points. Invariably the raw on-line data is often found noisy, as seen in Fig. 1, and this can severely affect the performance of even the most effective on-line script recognition algorithms. This necessitates use of noise removal technique prior to character recognition [5], [6], [7], [8], [9]. Splines can be used to construct a smooth curve on these noisy data points and hence remove noise. In general, splines are piecewise polynomials with pieces smoothly connected together. The joining points of the polynomial pieces are called knots. When each segment of a spline is a polynomial of degree *n*, we say that the spline is a spline of degree[1] *n*. Different spline [10], [11] based smoothing

[1] when $n = 3$ it is called a cubic spline

techniques have been successfully applied for denoising noisy signal [12], [13]. However, the degree of smoothness depends on the number and position of the chosen knots. If the knots are close to each other, the smooth curve between the two knots would be linear and if the knots are far apart, a higher order polynomial would be needed for fitting a smooth curve between the two knots. We propose a technique based on cubic spline to smooth the noisy data with both the number of knots and the location of the knots being chosen automatically. Hence, the smoothing technique becomes a cubic polynomial curve fitting with a variable span[2], having continuity at successive knots. For the purpose of noise removal we treat the character $\left(x_i^r, y_i^r\right)_{i=0}^{m}$ as two data sequence $\left(x_i^r\right)_{i=0}^{m}$ and $\left(y_i^r\right)_{i=0}^{m}$ separately and remove noise from each of these independently. The noise removal process is described in Algorithm 1 for the sequence $\left(x_i^r\right)_{i=0}^{m}$ but nevertheless is applicable to the sequence $\left(y_i^r\right)_{i=0}^{m}$.

Algorithm 1: Knotless spline for data smoothing

Step 0 Let $i = 0$ and set a knot at $k_i = 0$

Step 1 Set the span ($s = S$) be $m/2$ (consider only $m/2$ of the original m points, namely, $\{x_i^r\}_{i=0}^{m/2}$

Step 2 Fit a cubic spline in the span and compute the mean squared error (*MSE*) between the spline fit and the actual data points, namely find $\{a_i\}_{i=0}^{3} = 1$ such that MSE is minimized

$$MSE = \sum_{i=1}^{m/2}\left(f\left(x_i^r\right) - x_i^r\right)^2$$

where $f\left(x_i^r\right) = a_0 + a_1 x_i^r + a_2 x_i^{r2} + a_3 x_i^{r3}$

Step 3 Repeat until span is 20% of the initial span ($s = S/5$)

3a Reduce the span by 25% ($s = (7/8)S$), namely, consider $\{x_i^r\}_{i=0}^{m/2-m/8}$

3b Repeat Step 2

Step 4 Select the span with smallest *MSE* (say l) as the optimum span; the starting and end points of the span are the chosen knots (namely, k_{i+1} = l) and a cubic spline is fit in this span.

Step 5 $i = i + 1$; Repeat on the remaining data points until all the data points are covered.

It is to be noted that the knotless based spline technique automatically selects the number (i) and the location of the knots (k_i's). This is unlike other spline denoising techniques which require the user to specify the number of knots. This is one of the contributions of this paper.

III. ANALYSIS OF NOISE REMOVAL PROPERTY

We have quantitatively evaluated the knotless spline based noise removal technique by applying it on simulated noisy (additive Gaussian) data. For the simulation experiments we used a character from the HP Lab's Tamil OHCR database [14]. Fig. 2 shows a sample handwritten Tamil character *u* taken from the database. For quantifying the noise suppression performance, root mean squared (RMS) error between the original noise-free data and processed output data is computed. The proposed technique is also compared with the performance of a 7-tap and a 15-tap Gaussian filter with noise of 30 dB, 20 dB and 10 dB.

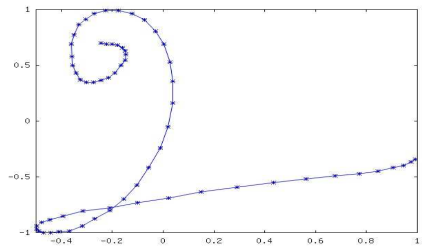


Fig. 2. Tamil character *u*: on-line raw

Fig. 3 shows the with the Tamil character u affected by additive white Gaussian noise of 20 dB. The processed outputs obtained by applying 7 and 15-tap Gaussian filters, and the spline based technique (Algorithm 1), , are shown in Fig. 4 (a), (b), and (c), respectively. Clearly, the 15-tap Gaussian filter and the knotless spline smoothing perform the best (compare Fig. 4(b) and 4(c)).

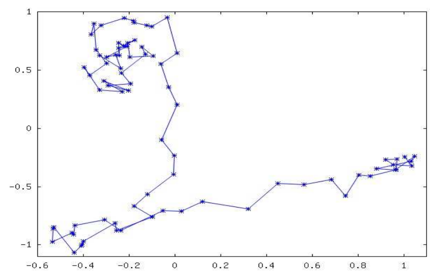


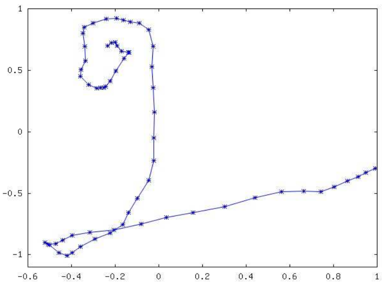
Fig. 3. Tamil character '*u*': with Gaussian noise of 20 dB

Table 1 presents the RMS error between the original and the filtered outputs using 7-tap and 15-tap Gaussian filter and the knotless spline smoothing technique on the noise-free data with input signal-to-noise ratio (SNR) values of 10 dB, 20 dB and 30 dB.

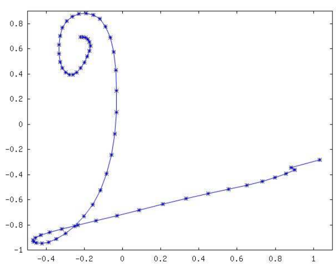
As seen in Table 1, among the Gaussian filters; the 7-tap Gaussian filter performs best at high input SNR values while

[2] span is the distance between the two successive knots

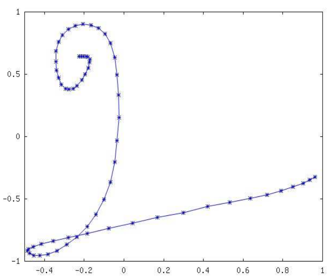
the performance is better for 15-tap Gaussian filter at lower SNR values. Clearly, the performance of the most commonly used Gaussian filter depends on the amount of noise present. i. To get the best noise removal one needs to identify the tap-length of the Gaussian filter which is dependent on the noise in the signal which is usually not known a priori. In this scenario, the knotless spline technique works at par the 7-tap Gaussian at higher input SNR values and as good as the 15-tap when the input SNR values are low; making the noise removal independent of the amount of noise. This is one of the major advantages of the knotless spline for noise removal. For this reason, in the rest of the paper, we use the knotless spline based technique in all our experiments.



(a) 7-tap Gaussian



(b) 15-tap Gaussian



(c) knotless spline

Fig. 4. Processed noisy Tamil character *using*: (a) 7-tap Gaussian filter, (b) 15-tap Gaussian filter and (c) knotless spline technique.

Table 1. Normalized RMS error for the data with the simulated white Gaussian noise (G7: 7-tap Gaussian filter, G15: 15-tap Gaussian filter)

Input SNR (dB)	RMS error		
	G7	G15	Knotless spline
30	**0.02**	0.06	0.04
20	0.04	0.05	0.05
10	0.17	**0.13**	0.11

IV. RECOGNITION EXPERIMENTS

Recognition experiments were carried out for several paragraphs of data (continuous text) where the writers were asked to write a paragraph of text naturally. All the data was captured using Mobile e-Note Taker which allows user to write on an ordinary paper and captures a $\{x^r, y^r\}$ trace of the pen (raw data), sampled at 100 Hz. We used knotless spline noise removal in all our experiments. Fig. 5 shows the Devanagari character *u* and its *x* and *y* data plot. As seen in the figure, the character is contaminated by noise. The corresponding processed outputs after applying the knotless spline technique are shown in Fig. 6. It can be observed that there is significant noise removal which in turn enables robust feature extraction from the character leading to good recognition. Another Devanagari character *ka* and the processed output is shown in Fig. 7. Similar improvement was observed on all characters.

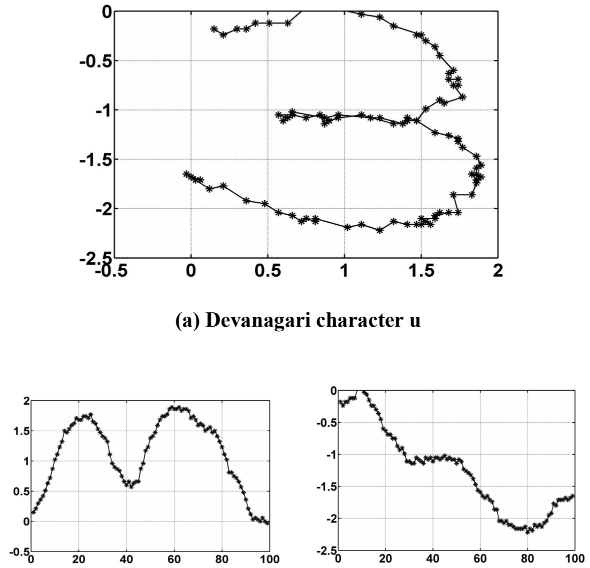
Extraction of critical points is an essential requirement for any character recognition [4]. We extract critical points by using the procedure described in [4]. As seen in Table 2 the second order statistics (mean and variance) of the number of critical points for the same character written by different people for a sample of ten selected Devanagari characters improves (lower variance) when data is smoothened. As seen in the Table 2, $\mu = 22.4$ and $\sigma^2 = 259.4$ for the raw noisy data compared to $\mu = 7.7$ and $\sigma^2 = 1.6$ for the smoothed data. This implies that the knotless spline smoothing can be effectively used for noise removal and it improves the consistency of the feature extraction process. The urgent need for noise removal can be observed in the form of exceptionally high variance in the number of critical points for raw data (Table 2).

We used Fuzzy Directional Features (FDF) [4] to build the reference models for OHCR. We performed a five-fold cross validation to measure the character recognition performance of the OHCR on Devanagari data. We specifically measured the recognition with (a) raw data (b) with knotless spline smoothed data. As a training step, we extracted the FDF for all the training samples corresponding to the same stroke. The average FDF is considered as the cluster center for that stroke.

Table 2 Mean and variance of the number of critical points

Character	Raw data		Knotless Spline smoothed data	
	μ	σ2	μ	σ2
u	22.4	259.4	7.7	1.6
i	19.9	103.3	7.7	0.4
e	13.9	63.7	3.4	0.8
k	29.3	113.2	11.9	1.5
R	12.0	23.8	4.4	0.2
v	18.4	64.6	7.9	1.4
g	14.4	53.4	5.5	0.4
gh	19.6	167.6	9.8	2.4
D	32.7	88.4	5.6	3.2
c	20.3	136.0	9.6	8.4

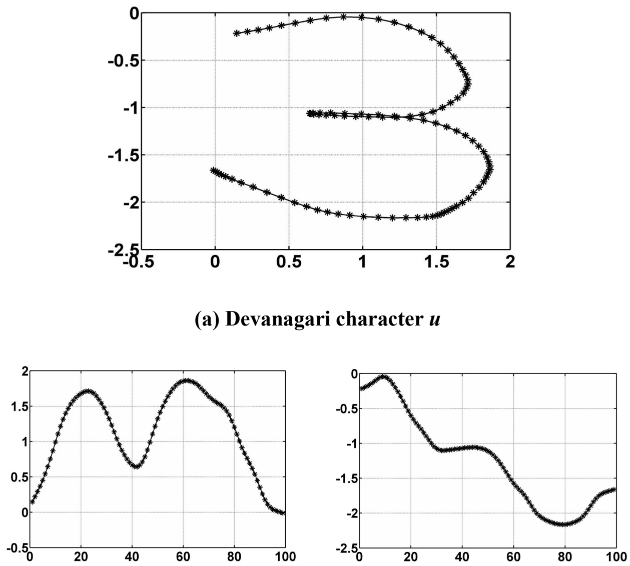
We used a multi-class SVM classifier with a Radial Basis Function (Gaussian) Kernel function [4] for the purpose of recognition. The average recognition accuracy obtained was 89.4% compared to 79% for the raw data without noise removal. Clearly, there is a 10% improvement in the character recognition because of noise removal.



(a) Devanagari character u

b) x of u **(c) y of u**

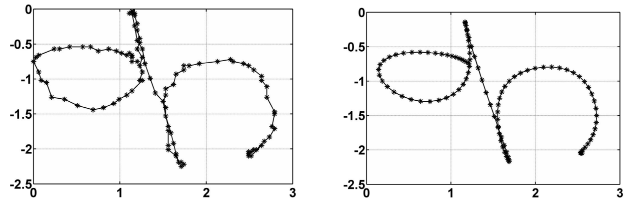
Fig. 5. Devanagari character *u* : raw data and its *x* and *y* data



(a) Devanagari character *u*

(b) *x* of *u* **(c) y of *u***

Fig. 6. Devanagari character *u* and its *x* and *y* data: smoothed using the knotless spline technique



(a) **(b)**

Fig. 7. Devanagari character *ka*: (a) raw and (b) smoothed using the knotless spline technique

V. CONCLUSIONS

In this paper we have introduced a knotless spline based smoothing technique for denoising on-line handwritten data. The proposed method is quantitatively evaluated on Tamil on-line handwritten characters and it was found to be effective in suppressing a range of noise levels unlike the Gaussian filter. Application of the knotless spline technique on the character with wide range of input SNR values showed that the technique is effective in removing the noise irrespective of the input SNR values. Actual recognition experiments were carried on raw and processed (denoised) Devanagari characters. Results show that the performance of the recognition algorithm significantly improves when data is smoothed using the knotless spline based technique.

REFERENCES

[1] SmartTech, “Smart technologies inc. homepage.” in ://www.smarttech.com/, last accessed Dec. 2009.

[2] Hi-tech Solutions, “Mobile e-note taker,” in http://www.hitechin.com/mobile-e-note-taker.htm, last accessed Dec. 2009.

[3] Microsoft, “Windows XP Tablet PC edition homepage,” in http://www.microsoft.com/windowsxp/tabletpc/default.mspx, last accessed Dec. 2009.

[4] V. L. Lajish and Sunil Kumar Kopparapu, “Fuzzy directional features for unconstrained on-line Devanagari handwriting recognition,” in *Proc. 16th National Conf. on Comm.* (*NCC 2010*), IIT Madras, India, 2010.

[5] B. Blesser, “Multistage digital filtering utilizing several criteria,” *U.S. Patent*, No. 4,375,081, 1983.

[6] C. C. Tappert, C. Y. Suen, and T. Wakahara, “The state of the art in on-line handwriting recognition,” *IEEE Trans. PAMI*, pp. 787–808, 1990.

[7] A. M. Namboodiri and A. K. Jain, “On-line handwritten script recognition,” *IEEE Trans. PAMI*, vol. 26, pp. 124–130, 2004.

[8] N. Joshi, G. Sita, A. Ramakrishnan, V. Deepu, and S. Madhvanath, “Machine recognition of on-line handwritten Devanagari characters,” in *Proc. ICDAR05*, 2005, pp. 1156–1160.

[9] S. Malik and S. A. Khan, “Urdu online handwriting recognition,” in *Proc. IEEE Int. Conf. on Emerging Technolgies*, 2005, pp. 27– 31.

[10] D. Solomon, *Computer Graphics and Geometrical Modeling*. Springer: New York, 1999.

[11] C. De Boor, *A Practical Guide to Splines*. Springer Verlag: New York, 1978.

[12] A. M. Namboodiri, “On-line handwritten document understanding,” *PhD Thesis*, Michigan State University, 2004.

[13] B. Q. Huang, Y. B. Zhang, and M. T. Kechadi, “Pre-processing techniques for on-line handwriting recognition,” in *Proc. Int. Conf. on Intelligent Systems Design and Applications*. Los Alamitos, CA, USA, 2007, pp. 793–800.

[14] http://www.hpl.hp.com/india/research/penhw/resources/tamil-iso-char.html, last accessed Oct. 2010.