A Semi-automatic Annotation Scheme for Bangla Online Mixed Cursive Handwriting Samples

U. Bhattacharya[1], R. Banerjee[2], S. Baral[1], R. De[2] and S. K. Parui[1]
[1]*CVPR Unit, Indian Statistical Institute, Kolkata, India,* [2]*Jadavpur University, Kolkata, India*
ujjwal@isical.ac.in, raja24oct@gmail.com, baral_sudarshan@yahoo.com, rohit_supreme00@yahoo.co.in, swapan.parui@gmail.com

Abstract

Requirement of annotated handwriting samples for the development of relevant recognition algorithms is an established fact. Although such annotated databases of unconstrained handwriting exist for several scripts of a few languages, the same is not true for any of the scripts of India. As far as Indian scripts are concerned a few databases of handwritten isolated characters are publicly available. These include samples of both online and offline handwriting. However, no such publicly available database of unconstrained handwriting in any of the Indian scripts exists. On the other hand, unconstrained handwriting in Bangla, the second most popular among Indian scripts, is mixed cursive in nature unlike the other scripts of India. Thus, annotation of Bangla unconstrained handwriting samples needs special consideration. During the last few years our group at the Indian Statistical Institute, Kolkata has been working towards the development of a large annotated database of online Bangla handwriting samples and has developed a GUI-based semi-automatic scheme for their annotation at character boundary levels and a scheme for XML representation of such annotated data. The present system implemented for annotation of unconstrained handwriting of Bangla may easily be customized for other scripts. Currently this system is in use for annotation of a large database of Bangla unconstrained online handwriting.

1. Introduction

Soon after pen computers (computers with pen-based handwriting input facility) appeared in the market in early 90's, a need for online handwriting recognizers was strongly felt. However, building a workable recognizer for online handwriting remained a difficult task and soon researchers realized the importance of developing annotated databases of handwriting samples of various scripts. Initially, such an effort of generating publicly available databases of handwriting samples could not materialize due to non-availability of proper funding. To the best of our knowledge, the first such effort was the UNIPEN project [1] of data exchange and recognizer benchmarks in which a few leading industry players and researchers from several Universities participated. Since then a lot of research works on handwriting recognition of several scripts was carried out [2-8]. However, no such major research activities for handwriting recognition of Indic scripts was observed in the past, though Devanagari and Bangla, the two most popular Indian languages, are the second and the fifth most popular in the world. Moreover, due to the increasing popularity of hand-held devices capable of accepting handwriting inputs, online handwriting recognition for Indian scripts with large character sets has become very important. Additionally, there is an encouraging development in the Indian scenario due to the Government funding to a consortium of several Indian Institutes for the development of online handwriting recognizers of the major Indian scripts. Indian Statistical Institute is one of the members of this consortium and a massive attempt is being undertaken here to develop a large annotated database of online Bangla handwriting with an aim to make it available to the community. The rest of the paper is organized as follows. Section 2 provides brief descriptions of Bangla script, its unconstrained handwriting style and existing recognition studies for its online handwriting. Section 3 provides a systematic description of the proposed annotation and representation schemes for mixed cursive handwriting of Bangla. Section 5 concludes the paper.

2. Bangla handwriting samples

Bangla is the second most popular language of the Indian subcontinent used by more than 220 million people across the two neighboring countries India and Bangladesh. This script is used to write texts in Assamese and Manipuri languages also.

2.1. Bangla character set

The Bangla script has 50 basic characters and their shapes are shown in Fig. 1(a). The first eleven in this list are basic vowels and the remaining 39 are basic consonants. All the vowels barring only the first one and two among the consonants have modified shapes known as modified characters. These modified character shapes are shown in Fig. 1(b). Additionally, there are approximately 200 compound characters of complex shapes formed due to merging of two or more basic characters. Shapes of a few compound characters are shown in Fig. 1(c).

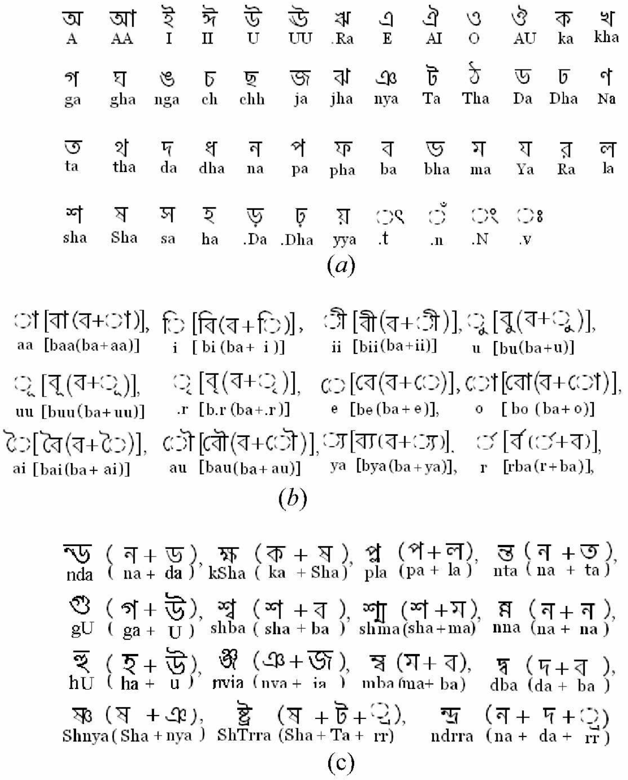

A AA I II U UU .Ra E AI O AU ka kha
ga gha nga ch chh ja jha nya Ta Tha Da Dha Na
ta tha da dha na pa pha ba bha ma Ya Ra la
sha Sha sa ha .Da .Dha yya .t .n .N .v
(a)
aa [baa(ba+aa)] i [bi(ba+i)] ii [bii(ba+ii)] u [bu(ba+u)]
uu [buu(ba+uu)] .r [b.r(ba+.r)] e [be(ba+e)], o [bo(ba+o)]
ai [bai(ba+ai)] au [bau(ba+au)] ya [bya(ba+ya)], r [rba(r+ba)],
(b)
nda (na + da), kSha (ka + Sha), pla (pa + la), nta (na + ta)
gU (ga + U), shba (sha + ba), shma(sha+ma), nna (na + na)
hU (ha + u), nvia (nva + ia), mba (ma+ ba), dba (da + ba)
Shnya(Sha + nya), ShTrra (Sha + Ta + rr), ndrra (na + da + rr)
(c)

Figure 1. Bangla characters; (*a*) basic characters, (*b*) character modifiers, (*c*) a few compound characters.

2.2. Bangla handwriting recognition studies

Bangla online handwriting recognition has been studied in the recent past [9-16]. In [9], a hidden Markov model (HMM) based recognition system was used for Bangla online handwritten numerals. Recently, a database of online handwritten Bangla basic characters and its recognition based on substroke features has been described in [12]. Benchmark recognition results for this database was first presented in [14]. Earlier in [10], direction code histogram feature was studied for recognition of these characters. Later in [11], an HMM based recognition scheme was studied for character recognition. Bangla online cursive handwriting recognition was first studied in [13]. Later Fink et al. [15] studied writer independent online Bangla word recognition task based on HMMs. and observed that an HMM based on context dependent sub-word units achieves promising recognition performance for these word samples. Recently in [16], a strategy combining multilayer perceptron (MLP) and support vector machine (SVM) has been studied for limited vocabulary recognition of unconstrained Bangla handwriting.

2.3. Bangla unconstrained handwriting style

Unlike other Indian scripts, unconstrained handwriting in Bangla is cursive in nature similar to English. However, connectedness between adjacent characters in a cursively written Bangla word occurs in its upper portion (see Fig. 2) in contrast to English where it usually occurs in the lower portion. Another dissimilarity with English is that Bangla like several other Indic scripts, consists of a very large character set many of which are extremely complex in shape.

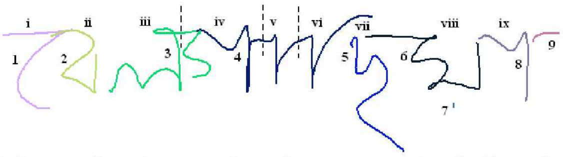


Figure 2: A sample of unconstrained Bangla handwritten word consisting of 9 characters numbered i, ii, …, ix and written using 9 strokes numbered 1, 2, …, 9; stroke number 9 represents a headline (“matra”); dotted vertical line segments show the character boundaries in connected situations. One color represents one stroke.

3. Proposed annotation scheme for unconstrained online Bangla handwriting

The present work deals with annotation of Bangla mixed cursive online handwriting. The nature of handwriting considered here is largely unconstrained in the sense that writing took place on plain surface with equally spaced horizontal ruling. There was no other restriction during data collection. Also, following the usual practice of similar data collection attempts, writers were provided a piece of texts satisfying the requirements of the ultimate goal of this data collection.

3.1. Related works

There exist a few studies of similar annotation and representation of online handwriting samples. The first such attempt was the UNIPEN project [1]. Later

Bhaskarbhatla *et al.* [17] presented an XML-based representation scheme and an annotation tool for online handwritten data. Although this work has some similarity with the proposed scheme, it did not consider segmenting a stroke into sub-strokes to take care of situations where parts of two characters are written using a single stroke.

Moreover, in the scheme described in [17] ground truths are entered by direct keying which is time intensive. In another work [18], Agrawal et al. described a few shortcomings of UNIPEN representation and proposed UPX as an alternative representation scheme for annotated online handwritten data. However, this scheme also did not consider requirement of annotation at the sub-stroke level. Kumar et al. [19] proposed a model-based annotation scheme suitable for character level annotation of documents in Indian scripts. In this work, it was assumed that a character is composed of an integral number of strokes, which is frequently violated in unconstrained handwriting samples of Bangla.

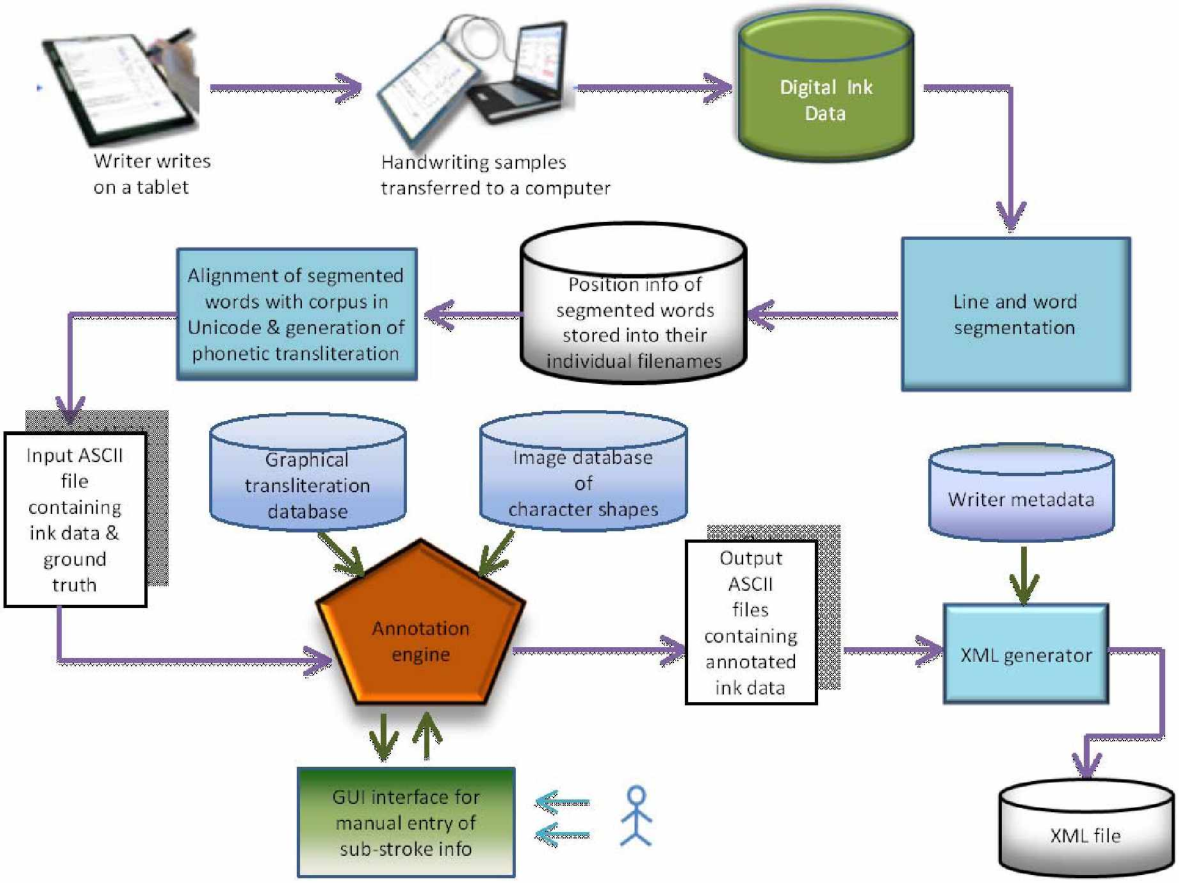

Writer writes on a tablet
Handwriting samples transferred to a computer
Digital Ink Data
Line and word segmentation
Position info of segmented words stored into their individual filenames
Alignment of segmented words with corpus in Unicode & generation of phonetic transliteration
Input ASCII file containing ink data & ground truth
Graphical transliteration database
Image database of character shapes
Annotation engine
Output ASCII files containing annotated ink data
Writer metadata
XML generator
XML file
GUI interface for manual entry of sub-stroke info

Figure 3. Block diagram of the annotation tool

3.2. Architecture of annotation tool

Usually, the creation of an annotated database of handwriting samples consists of several phases, which include (i) data collection, (ii) data annotation and (iii) data representation. For collection of online handwritten data (ink data), we selected a piece of text containing most of the characters of Bangla alphabet. Data collection devices include WACOM Intuous 2 tablet, Genius G-Note 7000 tablet and HP tablet PC. A correspondence of the ink files and file name containing its writer's metadata is maintained. The annotation tool has two separate units one of which takes care of segmentation of a document page into words and the other is used for annotation of individual words at the sub-stroke level. A block diagram of our annotation scheme is shown in Fig. 3.

3.2.1. Line and word segmentation. Our annotation tool consists of a GUI-based unit for segmenting an online handwritten document page into lines and each line into words. Line segmentation module compares two consecutive (temporal order) strokes w.r.t. the distributions of both *x*- and *y*-values of the respective sets of sample points to decide if the next stroke belongs to a new line. Off-line information such as horizontal

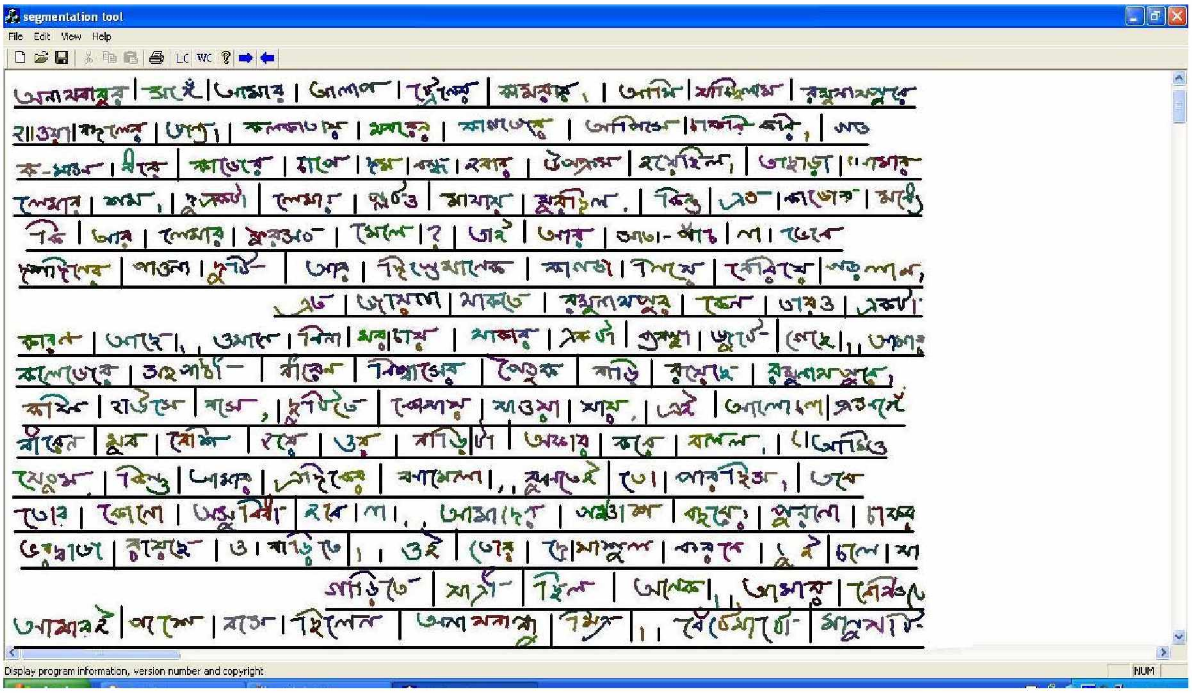


Figure 4. Screenshot of the segmentation output – horizontal and vertical line segments are used to show line and word segmentations respectively

projection profile is considered to settle possible confusions. Although this line segmentation approach is less prone to errors, the GUI-based module provides options for manual correction of any error occurring during this stage.

Automatic segmentation of words is usually more difficult than line segmentation. Often the rectangular box enclosing a word overlaps with a similar box enclosing an adjacent word. We deal with this problem in the following way. For each pair of consecutive strokes in a line, we calculate the absolute difference (*Dstroke*) in *x*-values of the rightmost sample point of the first stroke and the leftmost sample point of the next stroke. A histogram of *Dstroke* corresponding to the set of all pairs of consecutive strokes in a page is obtained to estimate a threshold value (*T*) for segmentation of words. *T* is selected so that 5% of *Dstroke* values are greater than *T*. The above approach is usually efficient in segmenting words barring only a few cases when two characters within a word are so wide apart that the gap between them is more than the minimum gap (*T*) between two words or the gap between two words is less than *T*. In such situations, the GUI-based tool allows manual correction using the mouse. An output of our segmentation approach is shown in Fig. 4. This output is manually checked and then a "save" button is clicked. Each segmented word is stored as a separate ASCII file named such as "RitaChatterjee_2_8_Word_665_siddhAnta.txt". This file stores the ink of the word "siddhAnta" (phonetic transliteration) written by the writer named "Rita Chatterjee" in the 8th position of the 2nd line of the document page and the serial number of the word in our corpus is 665. This file stores the coordinates of the sample points in addition to the Unicode of the word.

3.2.2. Character level annotation. Our annotation tool consists of a GUI-based second unit for semi-automatic annotation of each word (stored as mentioned above) at the character levels. Since a character in a word may consist of non-integral number of strokes and no off-the-shelf recognition engine is available, strokes across multiple characters are manually segmented into sub-strokes by clicking the mouse. The tool displays a cardinal number against each stroke representing their temporal order. The annotator uses the mouse to mark the desired segmentation points (character boundary) on the display of the word. It is not necessary to place the mouse pointer exactly on the trajectory – the toolkit searches for a point on the trajectory nearest to the mouse click and introduces a soft PEN_UP creating two sub-strokes. The "Refresh" button (provided in the second row of the top panel) can be used to undo placement of any unintentional mark. Once all such segmentation points are marked, the "Accept" button is pressed to finalize segmentation of the word into sub-strokes. These are shown in Fig. 5.

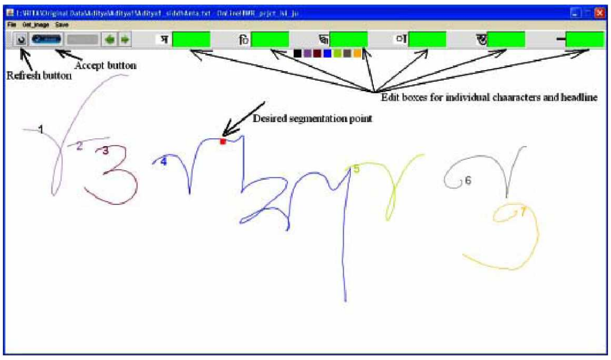

File Get_image Save
Refresh button
Accept button
Edit boxes for individual characters and headline
Desired segmentation point
1
2
3
4
5
6
7

Figure 5. A screenshot of the character level annotation tool. The red mark is given using the mouse to introduce a segmentation point.

Once the tool accepts manually introduced segmentation points, it places new cardinal numbers alongside individual sub-strokes (if a stroke is not segmented by placing a soft PEN_UP over it, the stroke is composed of a single sub-stroke). Now, the annotator manually places these cardinal numbers in the respective edit boxes provided in the second row of the top panel. There is one edit box for each character of the word and an additional edit box for the headline or "matra". Each of these edit boxes is labeled by the printed shape of the corresponding character. The Unicodes of the characters present in the word are obtained from the input inkfile containing the word data and their printed shapes are obtained from a folder of images of each printed character. The file of each character image is named according to the transliteration of the respective character for its easy reference. If a character is composed of multiple sub-strokes, then corresponding cardinal numbers are written in the edit box using

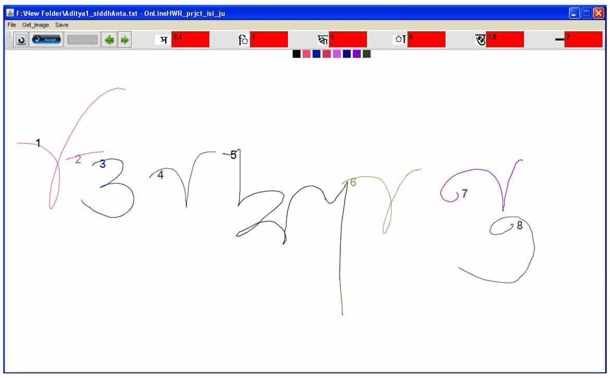

F:\New Folder\Aditya1_siddhAnta.txt - OnLineHWR_prjct_isi_ju
File Get_image Save
1
2
3
4
5
6
7
8

Figure 6. A screenshot of the final stage of annotation. The edit boxes in the top panel turned red denoting the annotation is saved.

"comma", or consecutive strokes may be entered by writing only the first and last sub-stroke numbers along with the symbol ".." between the two numbers. If no headline or "matra" is used in the sample word, then the corresponding edit box is left blank. Once all the entries are incorporated, the "Save" button provided in the first row of the top panel is pressed to store the ink data along with annotation information in an ASCII file. This final stage is shown in Fig. 6 and the annotation information stored for the word in Fig. 5 or Fig. 6 is "sa[3,4] i[1] ddha[5] aa[6] nta[7,8] matra[2]".

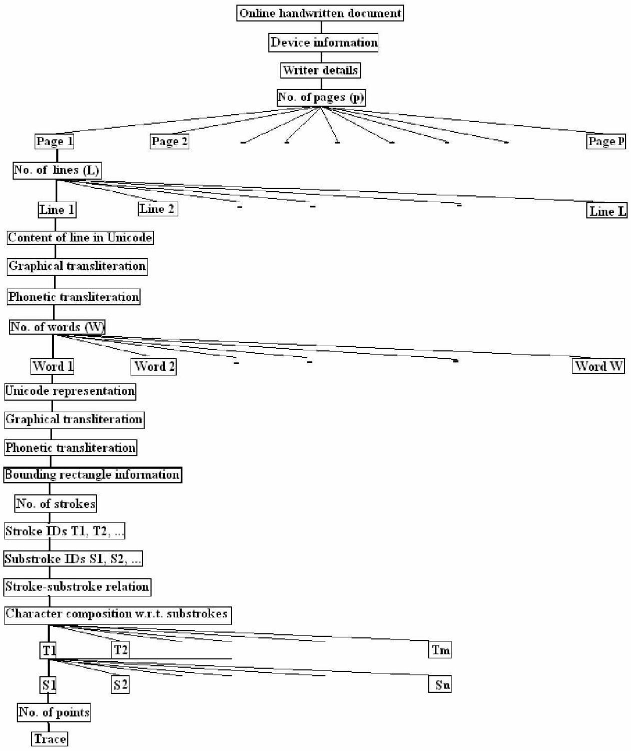
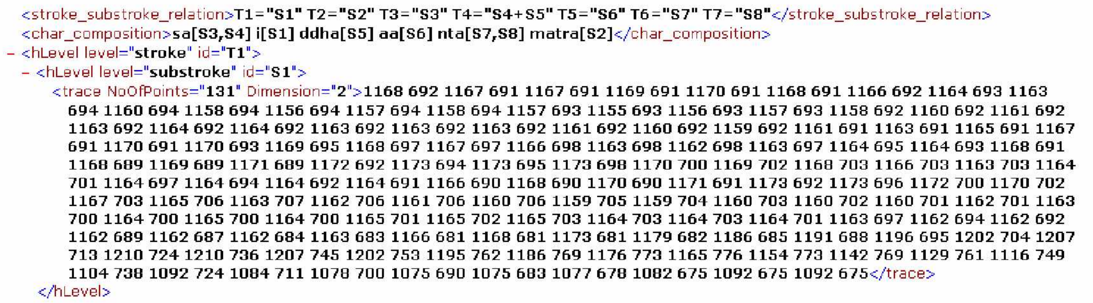


Figure 7. Chart showing the XML schema

3.2.2. XML representation. XML has been the most widely accepted representation format for annotated data due to several factors such as its hierarchical nature and the extensibility. Representation of annotated handwriting samples of Indic scripts was studied earlier [17,19]. However, these studies did not consider the peculiarity of unconstrained Bangla handwriting in which a character may be formed by a non-integral number of strokes. Here, we describe our novel strategy of XML representation of annotated Bangla handwriting samples. The peculiar nature of character-stroke relationship existing in unconstrained Bangla handwriting is tackled by introducing tags called "stroke_substroke_relation" and "char_composition". Due to space limitations we describe the further details of our XML schema with the help of Fig. 7 and Fig. 8.



```
  <stroke_substroke_relation>T1="S1" T2="S2" T3="S3" T4="S4+S5" T5="S6" T6="S7" T7="S8"</stroke_substroke_relation>
  <char_composition>sa[S3,S4] i[S1] ddha[S5] aa[S6] nta[S7,S8] matra[S2]</char_composition>
- <hLevel level="stroke" id="T1">
  - <hLevel level="substroke" id="S1">
      <trace NoOfPoints="131" Dimension="2">1168 692 1167 691 1167 691 1169 691 1170 691 1168 691 1166 692 1164 693 1163
        694 1160 694 1158 694 1156 694 1157 694 1158 694 1157 693 1155 693 1156 693 1157 693 1158 692 1160 692 1161 692
        1163 692 1164 692 1164 692 1163 692 1163 692 1163 692 1161 692 1160 692 1159 692 1161 691 1163 691 1165 691 1167
        691 1170 691 1170 693 1169 695 1168 697 1167 697 1166 698 1163 698 1162 698 1163 697 1164 695 1164 693 1168 691
        1168 689 1169 689 1171 689 1172 692 1173 694 1173 695 1173 698 1170 700 1169 702 1168 703 1166 703 1163 703 1164
        701 1164 697 1164 694 1164 692 1164 691 1166 690 1168 690 1170 690 1171 691 1173 692 1173 696 1172 700 1170 702
        1167 703 1165 706 1163 707 1162 706 1161 706 1160 706 1159 705 1159 704 1160 703 1160 702 1160 701 1162 701 1163
        700 1164 700 1165 700 1164 700 1165 701 1165 702 1165 703 1164 703 1164 703 1164 701 1163 697 1162 694 1162 692
        1162 689 1162 687 1162 684 1163 683 1166 681 1168 681 1173 681 1179 682 1186 685 1191 688 1196 695 1202 704 1207
        713 1210 724 1210 736 1207 745 1202 753 1195 762 1186 769 1176 773 1165 776 1154 773 1142 769 1129 761 1116 749
        1104 738 1092 724 1084 711 1078 700 1075 690 1075 683 1077 678 1082 675 1092 675 1092 675</trace>
    </hLevel>
```

Figure 8. A part of XML file

4. Results

The present annotation tool has been tested on handwriting samples provided by 16 writers. Each of them wrote a pre-selected text consisting of 589 words aligned along 60 lines. Thus, the total number of lines and word samples used for testing are respectively 960 and 9424. The automatic line segmentation module failed for 23 lines and the word segmentation module failed for 838 word samples. Thus, the line segmentation and the word segmentation accuracies obtained on the above test samples are respectively 97.6% and 91.11%.

5. Conclusion and future works

A major contribution of the proposed approach to both annotation and XML representation is the successful handling of situations when a single stroke forms parts of more than one character. The segmentation module of our annotation tool often fails to segment the punctuation marks separately. It also cannot handle delayed strokes across the words either in the same line or in different lines. Moreover, the proposed annotation scheme cannot handle spelling errors. In future, we shall study the above limitations.

6. Acknowledgement

This work has been partially supported by the Dept. of Information Technology, Govt. of India.

7. References

[1] I. Guyon, L. Schomaker, R. Plamondon, M. Liberman and S. Janet. UNIPEN project of online data exchange and recogniser benchmarks. *Proc. of the 12th ICPR*, 29-33, 1994.

[2] R. Plamondon, S. N. Srihari. On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Trans.Pattern Anal. & Machine Intell.*, 22(1), 63-84, 2000.

[3] A. Vinciarelli, A survey on on-line cursive script recognition. *Pattern Recognition*, 35(7), 1433–1446, 2002.

[4] H. Bunke. Recognition of cursive Roman handwriting – past, present and future. *Proc. of 7th ICDAR*, 448-459, 2003.

[5] S. Jaeger, C.-L. Liu, M. Nakagawa. The state of art in Japanese online handwriting recognition compared to techniques in western handwriting recognition. *Int. Journ. of Doc. Anal. and Recog.*, 6(2), 75-88, 2003.

[6] C.-L. Liu, S. Jaeger, M. Nakagawa. Online recognition of Chinese characters: the state-of-the-art. *IEEE Trans. on Patt. Anal. & Mach. Intell.*, 26(2), 198-213, 2004.

[7] J. Morwing, J. Andersson and C. Friberg. On-line Arabic handwriting recognition with templates. *Pattern Recognition*, 42, 3278-3286, 2009.

[8] H. E. Abed, M. Kherallah, V. Märgner, A. M. Alimi. On-line Arabic handwriting recognition competition ADAB database and participating systems. *Int. Journ. Doc. Anal. Recog.*, 14, 15-23, 2011.

[9] S. K. Parui, U. Bhattacharya, B. Shaw and K. Guin, A hidden Markov model for recognition of online handwritten Bangla numerals, *Proc. of the 41st National Annual Convention of CSI*, 27-31, 2006.

[10] U. Bhattacharya, B. K. Gupta and S. K. Parui, Direction code based features for recognition of online handwritten characters of Bangla, *Proc. 9th ICDAR*, vol. 1, 58-62, 2007.

[11] S. K. Parui, K. Guin, U. Bhattacharya, and B. B. Chaudhuri, Online handwritten Bangla character recognition using HMM, *Proc. of 19th Int. Conf. on Patt. Recog*, 2008.

[12] T. Mondal, U. Bhattacharya, S. K. Parui, K. Das and V. Roy, Database generation and recognition of online handwritten Bangla characters, *Proc. of Int. Workshop on Multilingual OCR*, ACM Int. Conf. Proc. Barcelona, 2009.

[13] U. Bhattacharya, A. Nigam, Y. S. Rawat and S. K. Parui, An analytic scheme for online handwritten Bangla cursive word recognition, *Proc. of ICFHR*, 320-325, 2008.

[14] T. Mondal, U. Bhattacharya, S. K. Parui, K. Das and D. Mandalapu, On-line handwriting recognition of Indian scripts - the first benchmark, *Proc. of 12th ICFHR*, 200-205, 2010.

[15] G. A. Fink, S. Vajda, U. Bhattacharya, S. K. Parui and B. B. Chaudhuri, "Online Bangla word recognition using sub-stroke level features and hidden Markov models," *Proc. of 12th. ICFHR*, 393-398, 2010.

[16] S. Mohiuddin, U. Bhattacharya and S. K. Parui, Unconstrained Bangla online handwriting recognition based on MLP and SVM, *Proc. of Int. Joint Workshop on MOCR and AND*, ACM Int. Conf. Proceeding Series, Beijing, 2011.

[17] A. Bhaskarbhatla, S. Madhavanath, M. Pavan Kumar, A. Balasubramanian and C. V. Jawahar, Representation and annotation of online handwritten data, *Proc. of the ICFHR*, 136-141, 2004.

[18] M. Agrawal, K. Bali, S. Madhvanath, and L. Vuurpijl, UPX: a new XML representation for annotated datasets of online handwriting data, *Proc. ICDAR*, 1161-1165, 2005.

[19] A. Kumar, A. Balasubramanian, A. Namboodiri, C. V. Jawahar, Model-based annotation of online handwritten datasets, Proc. of 10th IWFHR, 9-14, 2006.